Uživatelská příručka Calibre

Vydání 8.7.0

Kovid Goyal

července 18, 2025

Obsah

1 Grafické uživatelské rozhraní 3

2 Přidávání oblíbených webových stránek se zprávami 33

3 Prohlížeč e-knih 53

4 Převod e-knihy 61

5 Úprava e-knih 79

6 Server s obsahem Calibre 113

7 Porovnání e-knih 121

8 Úprava metadat e-knihy 125

9 Nejčastější dotazy 129

10 Kurzy 153

11 calibre:// URL schéma 251

12 Přizpůsobení Calibre 255

13 Rozhraní příkazového řádku 303

14 Nastavení vývojového prostředí Calibre 343

15 Správa digitálních práv (DRM) 373

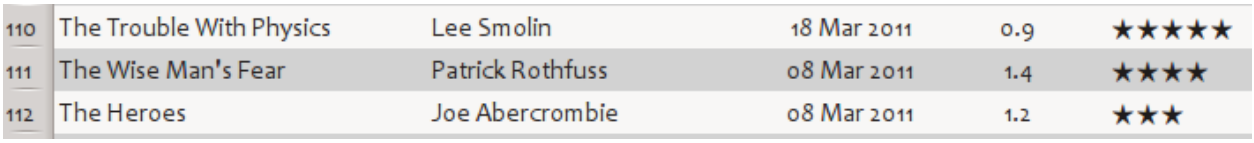
16 Slovník pojmů 375

Rejstřík modulů Pythonu 377

Rejstřík 379

Calibre je správce knihovny e-knih. Může zobrazovat, převádět a katalogizovat e-knihy ve většině hlavních formátů e-knih. Může také komunikovat s mnoha čtecími zařízeními e-knih. Může přejít na internet a načíst metadata pro vaše knihy. Může stahovat noviny a převést je na e-knihy pro pohodlné čtení. Je víceplatformové, funguje na systémech Windows, Linux a macOS.

Právě jste spustili Calibre. Co uděláte nyní? Než může Calibre udělat cokoliv s vašimi e-knihami, musí o nich nejdříve vědět. Přetáhněte pár souborů e-knih do Calibre nebo klikněte na tlačítko „Přidat knihy“ a najděte knihy, se kterými chcete pracovat. Poté, co jste přidali knihy, zobrazí se v hlavním zobrazení, které vypadá nějak takto:



110	The Trouble With Physics	Lee Smolin	18 Mar 2011	0.9	★★★★★
111	The Wise Man's Fear	Patrick Rothfuss	08 Mar 2011	1.4	★★★★
112	The Heroes	Joe Abercrombie	08 Mar 2011	1.2	★★★

Poté, co jste spokojeně obdivovali seznam knih, které jste právě přidali, nejspíš si budete chtít nějakou přečíst. Abyste to mohli udělat, budete muset převést knihu do formátu, kterému vaše čtečka rozumí. Při prvním spuštění Calibre se spustí *Uvítací průvodce* a nastaví Calibre pro vaše čtecí zařízení. Převod je snadná věc. Stačí vybrat knihu, kterou chcete převést, a pak klikněte na tlačítko „Převést knihy“. Prozatím ignorujte všechny volby a klikněte na „OK“. V pravém dolním rohu se začne otáčet malá ikona. Jakmile se přestane otáčet, vaše převedená kniha je připravena. Klikněte na tlačítko „Zobrazit“ pro čtení knihy.

Pokud chcete číst knihu ve své čtečce, připojte čtečku k počítači, počkejte, dokud ji Calibre nerozpozná (10-20 sekund), a pak klikněte na tlačítko „Odeslat do zařízení“. Jakmile se ikona přestane znovu otáčet, odpojte svou čtečku a dejte se do čtení! Pokud jste v předchozím kroku nepřevedli knihu, Calibre ji převede automaticky do formátu, kterému vaše čtecí zařízení rozumí.

Abyste začali s pokročilejším používáním, měli byste si přečíst o *grafickém uživatelském rozhraní* (stránka 3). Pro ještě větší výkonnost a všestrannost si projděte *Rozhraní příkazového řádku* (stránka 303). Také shledáte užitečným seznam *nejčastějších dotazů* (stránka 129).

Pokud máte další otázky nebo chcete probrat Calibre s ostatními uživateli nebo požádat o pomoc s konkrétními věcmi, jsou dostupná fóra a další zdroje nápovědy[1].

Oddíly

[1] https://calibre-ebook.com/help

KAPITOLA 1

Grafické uživatelské rozhraní

Grafické uživatelské rozhraní *(GUI)* poskytuje přístup k celé správě knihovny a funkcím převodů formátů e-knih. Základní pracovní postup pro používání Calibre je nejdříve přidat knihy do knihovny z vašeho pevného disku. Calibre se automaticky pokusí přečíst metadata z knih a přidat je do své vnitřní databáze. Jakmile jsou v databázi, můžete na nich provádět různé *Akce* (stránka 4), což zahrnuje převod z jednoho formátu do druhého, přenos do čtecího zařízení, zobrazení v počítači a úpravu metadat. Posledně zmíněné zahrnuje úpravu obálky, popisu a štítků společně s ostatními podrobnostmi. Pamatujte, že Calibre vytváří kopie souborů, které do něj přidáváte. Vaše původní soubory zůstanou nedotčené.

Rozhraní je rozděleno do různých oddílů:

- *Akce* (stránka 4)
- *Předvolby* (stránka 10)
- *Katalogy* (stránka 11)
- *Hledání a řazení* (stránka 11)
- *Rozhraní hledání* (stránka 12)
- *Ukládání hledání* (stránka 17)
- *Searching the full text of all books* (stránka 18)
- *Virtuální knihovny* (stránka 18)
- *Dočasné označování knih* (stránka 18)
- *Odhadování metadat z názvů souborů* (stránka 18)
- *Podrobnosti o knize* (stránka 20)
- *Prohlížeč štítků* (stránka 22)
- *Mřížka obálek* (stránka 25)
- *Prohlížeč obálek* (stránka 26)

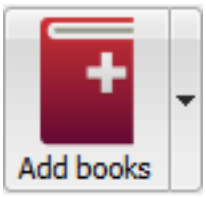
- *Adding notes for authors, series, etc.* (stránka 27)
- *Rychlé zobrazení* (stránka 28)
- *Úlohy* (stránka 29)
- *Klávesové zkratky* (stránka 29)

1.1 Akce

Panel nástrojů akcí poskytuje vhodné zkratky k běžně používaným akcím. Pokud kliknete pravým tlačítkem myši na tlačítka, můžete provést varianty výchozí akce. Pamatujte, prosím, že panel nástrojů akcí bude vypadat mírně odlišně v závislosti na tom, zda máte k počítači připojenou čtečku e-knih.

- *Přidat knihy* (stránka 4)
- *Upravit metadata* (stránka 5)
- *Převést knihy* (stránka 6)
- *Zobrazit* (stránka 6)
- *Odeslat do zařízení* (stránka 6)
- *Načíst zprávy* (stránka 7)
- *Knihovna* (stránka 7)
- *Zařízení* (stránka 8)
- *Uložit na disk* (stránka 8)
- *Připojit nebo sdílet* (stránka 9)
- *Odebrat knihy* (stránka 10)

1.1.1 Přidat knihy



Akce *Přidat knihy* má šedm variant přístupných kliknutím prvým tlačítkem myši na tlačítko.

1. **Přidat knihy z jedné složky**: Otevře dialogové okno výběru souborů a umožní zadat, které knihy ve složce by měly být přidány. Tato akce je *kontextově závislá*, t.j. že závisí na tom, který *katalog* (stránka 11) jste vybrali. Pokud jste vybrali *Knihovnu*, budou knihy přidány do knihovny. Pokud jste vybrali čtečku e-knih, budou knihy nahrány do zařízení, a tak dále.
2. **Přidat knihy ze složek a podsložek**: Umožňuje vám zvolit složku. Složka a všechny její podsložky jsou prohledány rekurzivně a všechny nalezené e-knihy jsou přidány do knihovny. Můžete zvolit, zda má Calibre přidat všechny

soubory v jedné složce do záznamu jedné knihy nebo do záznamů více knih. Všechny soubory e-knih ve složce jsou považovány za stejnou knihu v jiných formátech. Tato akce je opak akce *Uložit na disk* (stránka 8), t.j. můžete *Uložit na disk*, odstranit knihy a znovu je přidat v režimu jedné knihy na složku bez ztráty informací kromě data (toto předpokládá, že jste nezměnili žádné nastavení pro akci Uložit na disk).

3. **Přidat více knih z archivu (ZIP nebo RAR)**: Umožňuje vám přidat více e-knih, které jsou uloženy ve vybraných souborech ZIP nebo RAR. Je to vhodná zkratka, která zabraňuje tomu, abyste museli nejdříve rozbalit archiv, a potom přidat knihy prostřednictvím jedné ze dvou výše uvedených voleb.
4. **Přidat prázdnou knihu (Položka knihy bez formátů)**: Umožňuje vám vytvořit prázdný záznam knihy. To může být použito pro následné ruční vyplnění informací o knize, kterou ještě nemusíte mít ve své sbírce.
5. **Přidat z ISBN**: Umožňuje vám přidat jednu nebo více e-knih zadáním jejich ISBN.
6. **Přidat soubory do vybraných záznamů knih**: Umožňuje vám přidat nebo aktualizovat soubory přidružené k existující knize ve vaší knihovně.
7. **Add data files to selected book records**: Allows you to add any number of extra files that will be stored in a `data` sub-directory in the book directory. See *Přidání dodatečných souborů dat do knihy* (stránka 127) for details.
8. **Přidat prázdný soubor do vybraných záznamů knih**: Umožňuje vám přidat prázdný soubor zadaného formátu do vybraných záznamů knih.

Akce *Přidat knihy* může číst metadata z široké nabídky formátů e-knih. Navíc se pokusí odhadnout metadat z názvu souboru. Podívejte se na oddíl *Odhadování metadat z názvů souborů* (stránka 18), abyste zjistili, jak toto konfigurovat.

Pro přidání dodatečného formátu pro existující knihu můžete udělat kteroukoliv ze tří věcí:

1. Přetáhnout soubor na panel Podrobnosti o knize na pravé straně hlavního okna
2. Kliknout pravým tlačítkem myši na tlačítko *Přidat knihy* a zvolit *Přidat soubory k vybraným záznamům knih*.
3. Kliknout na tlačítko *Přidat knihy* v pravé horní oblasti dialogového okna *Upravit metadata* otevřeného akcí *Upravit metadata* (stránka 5).

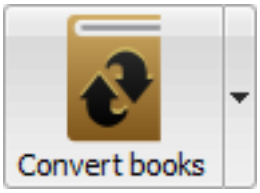
1.1.2 Upravit metadata

Akce *Upravit metadata* má čtyři varianty, které mohou být otevřeny kliknutím pravým tlačítkem myši na tlačítko.

1. **Upravit metadata jednotlivě**: Umožňuje vám upravit metadata knih postupně s možností načtení metadat včetně obálek z internetu. Také vám umožňuje přidat nebo odebrat konkrétní formáty e-knih z knihy.
2. **Upravit metadata hromadně**: Umožňuje vám upravit běžná pole metadat pro velké počty knih souběžně. Toto pracuje na všech knihách, které jste vybrali v *Zobrazení knihovny* (stránka 11).
3. **Stáhnout metadata a obálky**: Stáhne metadata a obálky (pokud jsou dostupné) pro knihy, které jste vybrali v seznamu knih.
4. **Sloučit záznamy knih**: Poskytne vám možnost sloučení metadat a formátů dvou nebo více záznamů knih. Můžete zvolit buď odstranit nebo ponechat záznamy, na které nebylo kliknuto nejdříve.
5. **Manage data files**: Manage the extra data files associated with the selected books.

Pro více podrobností se podívejte na *Úprava metadat e-knihy* (stránka 125).

1.1.3 Převést knihy



E-knihy mohou být převedeny z několika formátů do jakéhokoliv formátu, který vaše čtečka e-knih upřednostňuje. Mnoho e-knih dostupných k zakoupení bude chráněno technologií **:doc:`Správa digitálních práv <drm>`_** technologie *(DRM)*. Celibre tyto knihy nepřevede. Je snadné odebrat DRM z mnoha formátů, ale může to být nezákonné, budete si muset najít nástroje pro osvobození svých knih sami, a pak použít Calibre k jejich převodu.

Pro většinu lidí by převod měla být jednoduchá záležitost jednoho kliknutí. Pokud se chcete dozvědět více o procesu převodu, podívejte se na *Převod e-knihy* (stránka 61).

Akce *Převést knihy* má tři varianty přístupné kliknutím pravým tlačítkem myši na tlačítko.

1. **Převést jednotlivě**: Umožňuje vám zadat volby převodu pro přizpůsobení převodu každé vybrané e-knihy.
2. **Hromadný převod**: Umožňuje vám zadat volby pouze jednou pro převod několika e-knih hromadně.
3. **Vytvořit katalog knih ve vaší knihovně Calibre**: Umožňuje vám generovat kompletní výpis knih ve vaší knihovně, včetně všech metadat, v několika formátech, jako jsou XML, CSV, BiBTeX, EPUB a MOBI. Katalog bude obsahovat všechny knihy aktuálně zobrazené v zobrazení knihovny. To vám umožní použít funkce hledání pro omezení knihy, které mají být uloženy do katalogu. Navíc pokud vyberete více knih pomocí myši, budou do katalogu přidány pouze tyto knihy. Pokud generujete katalog ve formátu e-knihy, jako je EPUB, MOBI nebo AZW3, příště když připojíte svou čtečku e-knih, bude katalog automaticky odeslán do zařízení. Pro více informací o tom, jak katalogy fungují, si přečtěte *Vytváření katalogů AZW3 • EPUB • MOBI* (stránka 243).

1.1.4 Zobrazit

Akce *Zobrazit* zobrazí knihu v programu prohlížeče e-knih. Calibre má vestavěný prohlížeč pro mnoho formátů e-knih. Pro ostatní formáty používá výchozí aplikace operačního systému. Které formáty by se měly otevírat vnitřním prohlížečem, můžete nastavit pomocí *Předvolby → Rozhraní → Chování*. Pokud má kniha více než jeden formát, můžete zobrazit určitý formát kliknutím pravým tlačítkem myši na tlačítko.

1.1.5 Odeslat do zařízení

Akce *Odeslat do zařízení* má osm variant, které jsou přístupné kliknutím pravým tlačítkem myši na tlačítko.

1. **Odeslat do hlavní paměti**: Vybrané knihy jsou přeneseny do hlavní paměti čtečky e-knih.
2. **Odeslat na paměťovou kartu A**: Vybrané knihy jsou přeneseny na paměťovou kartu A ve čtečce e-knih.
3. **Odeslat na paměťovou kartu B**: Vybrané knihy jsou přeneseny na paměťovou kartu B ve čtečce e-knih.

4. **Odeslat konkrétní formát do**: Vybrané knihy jsou přeneseny do vybraného úložiště v zařízení ve formátu, který zadáte.
5. **Vysunout zařízení**: Odpojí zařízení od Calibre.
6. **Nastavit výchozí akci odeslání do zařízení**: Umožňuje vám zadat, která z voleb, 1 až 5 nad nebo 7 pod, bude výchozí akcí, když kliknete na hlavní tlačítko.
7. **Odeslat a odstranit z knihovny**: Vybrané knihy jsou přeneseny do zvoleného úložiště v zařízení a pak **odstraněny** z knihovny.
8. **Načíst poznámky (experimentální)**: Přenese poznámky, které jste si udělali k e-knize ve vašem zařízení, k metadatům komentářů knihy v knihovně Calibre.

Název souboru a strukturu složek souborů odeslaných do zařízení můžete ovládat nastavením šablony v *Předvolby → Import a export → Odesílání knih do zařízení*. Také se podívejte na *Jazyk šablon v programu Calibre* (stránka 160).

1.1.6 Načíst zprávy

Akce *Načíst zprávy* stahuje zprávy z různých webových stránek a převádí je na e-knihu, kterou můžete číst ve své čtečce e-knih. Obyčejně je nově vytvořená e-kniha přidána do vaší knihovny e-knih, ale pokud je v době dokončení stahování připojena čtečka e-knih, zprávy jsou také automaticky nahrány do čtečky.

Akce *Načíst zprávy* používá jednoduché předpisy (10-15 řádků kódu) pro každou stránku se zprávami. Abyste se naučili, jak vytvořit předpisy pro vlastní zdroje zpráv, podívejte se na *Přidávání oblíbených webových stránek se zprávami* (stránka 33).

Akce *Načíst zprávy* má tři varianty, přístupné kliknutím pravým tlačítkem myši na tlačítko.

1. **Naplánovat stahování zpráv**: Umožňuje vám naplánovat stahování z vámi vybraných zdrojů zpráv ze seznamu stovek dostupných. Plánování může být nastaveno jednotlivě pro každý zdroj zpráv, který vyberete, a plánování je flexibilní, což vám umožňuje vybrat konkrétní dny v týdnu nebo počet dnů mezi stahováním.
2. **Přidat vlastní zdroj zpráv**: Umožňuje vám vytvořit jednoduchý předpis pro stahování zpráv z vlastní stránky se zprávami, ke kterým chcete mít přístup. Vytváření předpisu může být tak jednoduché, jako zadání adresy URL informačního kanálu RSS, nebo můžete být více předpisující vytvořením kódu založeném na Pythonu pro tuto úlohu. Více informací naleznete na *Přidávání oblíbených webových stránek se zprávami* (stránka 33).
3. **Stáhnout všechny naplánované zdroje zpráv**: Způsobí, že Calibre okamžitě začne stahovat všechny zdroje zpráv, které jste naplánovali.

1.1.7 Knihovna

Akce *Library* vám umožňuje vytvářet, přepínat mezi, přejmenovávat nebo odebírat knihovny. Calibre vám umožňuje vytvořit tolik knihoven, kolik budete chtít. Mohli byste například vytvořit knihovnu pro sci-fi, knihovnu pro naučnou literaturu, knihovnu pro cizí jazyk, knihovnu pro projekt nebo jakoukoliv strukturu, která vyhovuje vašim potřebám. Knihovny jsou nejvyšší organizační struktura v Calibre. Každá knihovna má svou vlastní sadu knih, štítků, kategorií a základní umístění úložiště.

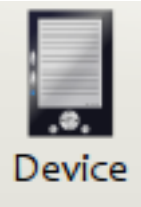
1. **Přepnout nebo vytvořit knihovnu… **: Umožňuje vám a) připojit se k již existující knihovně Calibre v jiném umístění, b) vytvořit prázdnou knihovnu v novém umístění nebo c) přesunout aktuální knihovnu do nově zadaného umístění.
2. **Rychlé přepnutí**: Umožňuje vám přepínat mezi knihovnami, které byly registrovány nebo vytvořeny v Calibre.
3. **Přejmenovat knihovnu**: Umožňuje vám přejmenovat knihovnu.
4. **Vybrat náhodnou knihu**: Zvolí náhodnou knihu z knihovny
5. **Odebrat knihovnu**: Umožňuje vám odregistrovat knihovnu z Calibre.
6. **Exportovat nebo importovat všechna data Calibre**: Umožní vám buď exportovat data Calibre kvůli přechodu na nový počítač nebo importovat dříve exportovaná data.
7. **<library name>**: Akce 7, 8 atd… vám poskytuje okamžitý přístup k přepnutí mezi více knihovnami, které jste vytvořili nebo připojili. Tento seznam obsahuje pouze 5 nejčastěji používaných knihoven. Pro kompletní seznam použijte nabídku Rychlé přepnutí.
8. **Údržba knihovny**: Umožňuje vám zkontrolovat v aktuální knihovně problémy s konzistencí dat a obnovit databázi aktuální knihovny ze záloh.

Poznámka

Metadata o vašich e-knihách, např. název, autor a štítky, jsou uloženy v jednom souboru ve složce knihovny Calibre nazvané metadata.db. Pokud se tento soubor poškodí (velice vzácná událost), můžete ztratit metadata. Naštěstí Calibre automaticky zálohuje metadata pro každou jednotlivou knihu ve složce knihy jako soubor OPF. Použitím akce ‚Obnovit databázi' pod výše popsanou ‚Údržbou knihovny' pro vás může Calibre znovu sestavit soubor metadata.db z jednotlivých souborů OPF.

Můžete kopírovat nebo přesouvat knihy mezi různými knihovnami (jakmile budete mít nastavenou více než jednu knihovnu) kliknutím pravým tlačítkem myši na knihu a výběrem akce *Kopírovat do knihovny*.

1.1.8 Zařízení



Akce *Zařízení* vám umožňuje zobrazit knihy v hlavní paměti nebo na paměťových kartách vašeho zařízení, nebo vysunout zařízení (odpojit ho od Calibre). Tato ikona se automaticky zobrazí na hlavním panelu nástrojů Calibre, když připojíte podporované zařízení. Můžete na ni kliknout, abyste viděli knihy ve svém zařízení. Můžete také přetáhnout knihy ze své knihovny Calibre na ikonu, abyste je přenesli do svého zařízení. A naopak můžete přetáhnout knihy ze svého zařízení na ikonu knihovny na panelu nástrojů, abyste přenesli knihy ze svého zařízení do knihovny Calibre.

1.1.9 Uložit na disk

Akce *Uložit na disk* má pět variant, které jsou přístupné kliknutím pravým tlačítkem myši na tlačítko.

1. **Uložit na disk**: Uloží vybrané knihy na disk roztříděné do složek. Struktura složek vypadá takto:

```
Author_(sort)
    Title
        Book Files
```

Název souboru a strukturu složek souborů uložených na disk můžete ovládat nastavením šablony v *Předvolby → Import a export → Ukládání knih na disk*. Také se podívejte na *Jazyk šablon v programu Calibre* (stránka 160).

2. **Uložit na disk do jedné složky**: Uloží vybrané knihy na disk do jedné složky.

Pro 1. a 2. jsou všechny dostupné formáty, stejně jako metadata, uloženy na disk pro každou vybranou knihu. Metadata jsou uložena do souboru OPF. Uložené knihy mohou být znovu importovány do knihovny bez jakékoliv ztráty informací pomocí akce *Přidat knihy* (stránka 4).

3. **Uložit na disk pouze <vámi upřednostňovaný> formát**: Uloží vybrané knihy na disk ve struktuře složek, jak je zobrazeno v (1.), ale pouze ve vámi upřednostňovaném formátu e-knihy. Svůj upřednostňovaný formát můžete nastavit v *Předvolby → Rozhraní → Chování → Upřednostňovaný výstupní formát*
4. **Uložit na disk do jedné složky pouze <vámi upřednostňovaný> formát**: Uloží vybrané knihy na disk do jedné složky, ale pouze ve vámi upřednostňovaném formátu e-knihy. Svůj upřednostňovaný formát můžete nastavit v *Předvolby → Rozhraní → Chování → Upřednostňovaný výstupní formát*
5. **Uložit na disk jeden formát…**: Uloží vybrané knihy na disk ve struktuře složek, jak je zobrazeno v (1.), ale pouze ve formátu, který jste vybrali z automaticky otevíraného seznamu.

1.1.10 Připojit nebo sdílet

Akce *Připojit nebo sdílet* vám umožňuje ručně se připojit k zařízení nebo složce ve vašem počítači. Také vám to umožňuje nastavit vaši knihovnu Calibre pro přístup prostřednictvím webového prohlížeče nebo e-mailu.

Akce *Připojit nebo sdílet* má čtyři varianty, přístupné kliknutím pravým tlačítkem myši na tlačítko.

1. **Připojit ke složce**: Umožňuje vám připojit se ke kterékoliv složce ve vašem počítači, jako by to bylo zařízení a využívat s touto složkou všechno vybavení, které má Calibre pro zařízení. To je užitečné, pokud vaše zařízení nemůže být podporováno v Calibre, ale je dostupné jako disk USB.
2. **Spustit Server s obsahem**: Spustí vestavěný webový server Calibre. Když je spuštěný, bude vaše knihovna Calibre přístupná prostřednictvím webového prohlížeče z internetu (pokud si to zvolíte). Jak je přistupováno k webovému serveru můžete konfigurovat nastavením předvoleb v *Předvolby → Sdílení → Sdílení po síti*
3. **Nastavit sdílení knih na základě e-mailu**: Umožňuje sdílení knih a kanálů zpráv e-mailem. Po nastavení e-mailové adresy pro tuto volbu bude Calibre odesílat aktualizace zpráv a knih na zadané e--mailové adresy. Jak Calibre odesílá e-maily můžete konfigurovat nastavením předvoleb v *Předvolby → Sdílení → Sdílení knih e-mailem*. Jakmile nastavíte jednu nebo více e-mailových adres, bude tato položka nabídky nahrazena položkami nabídky pro odesílání knih na nastavené e-mailové adresy.

1.1.11 Odebrat knihy

Akce *Odebrat knihy* **trvale odstraní knihy**, takže ji používejte opatrně. Je *kontextově závislá*, t.j. závisí na tom, který *katalog* (stránka 11) jste vybrali. Pokud jste vybrali *Knihovnu*, knihy budou odebrány z knihovny. Pokud jste zvolili zařízení čtečky e-knih, budou knihy odebrány ze zařízení. Abyste odebrali pouze určitý formát pro danou knihu, použijte akci *Upravit metadata* (stránka 5). Odebrání knih má také pět variant, ke kterým můžete získat přístup kliknutím pravým tlačítkem myši na tlačítko.

1. **Odebrat vybrané knihy**: Umožňuje vám **trvale** odebrat všechny knihy, které jsou vybrány v seznamu knih.
2. **Odebrat soubory konkrétního formátu z vybraných knih**: Umožňuje vám **trvale** odebrat soubory e-knih konkrétního formátu z knih, které jsou vybrány v seznamu knih.
3. **Odebrat všechny formáty z vybraných knih, kromě…**: Umožňuje vám **trvale** odebrat soubory e-knih všech formátů kromě zadaného formátu z knih, které jsou vybrány v seznamu knih.
4. **Odebrat všechny formáty z vybraných knih**: Umožňuje vám **trvale** odebrat všechny soubory e-knih z knih, které jsou vybrány v seznamu knih. Zůstanou pouze metadata.
5. **Odebrat obálky z vybraných knih**: Umožňuje vám **trvale** odebrat všechny soubory obrázků obálky z knih, které jsou vybrány v seznamu knih.
6. **Odebrat odpovídající knihy ze zařízení**: Umožňuje vám odebrat soubory e-knih z připojeného zařízení, které odpovídají knihám, které jsou vybrány v seznamu knih.
7. **Restore recently deleted**: Allows you to undo the removal of books or formats.

Poznámka

Note that when you use *Remove books* to delete books from your calibre library, the book record is deleted, but the books are temporarily stored, for a few days, in a trash folder. You can undo the delete by right clicking the *Remove books* button and choosing to *Restore recently deleted* books.

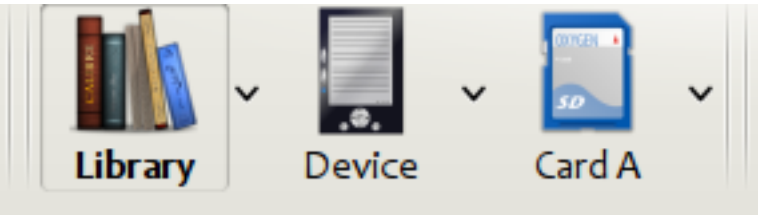
1.2 Předvolby

Akce *Předvolby* vám umožňuje změnit způsob, jakým různé aspekty Calibre fungují. Má čtyři varianty přístupné kliknutím pravým tlačítkem myši na tlačítko.

1. **Předvolby**: Umožňuje vám změnit způsob, jakým různé aspekty Calibre fungují. Kliknutí na tlačítko také provede tuto akci.
2. **Spustit Uvítacího průvodce**: Umožňuje vám spustit *Uvítacího průvodce*, který se objevil, když jste poprvé spustili Calibre.
3. **Získat moduly pro rozšíření Calibre**: Otevře nové okno, které zobrazí moduly pro Calibre. Tyto moduly jsou vyvíjeny třetími stranami pro rozšíření funkčnosti Calibre.

4. **Restartovat v režimu ladění**: Umožňuje vám povolit režim ladění, který může pomáhat vývojářům Calibre při řešení problémů, na které narazíte v programu. Pro většinu uživatelů by toto mělo zůstat zakázáno, pokud jim vývojáři neudělí pokyn, aby to povolili.

1.3 Katalogy



Katalog je kolekce knih. Calibre může spravovat dva typy různých katalogů:

1. **Knihovna**: Toto je kolekce knih uložených ve vaší knihovně Calibre ve vašem počítači.
2. **Zařízení**: Toto je kolekce knih uložených ve vaší čtečce e-knih. Bude dostupná, když připojíte čtečku k počítači.

Mnoho operací, jako je přidávání knih, odstraňování, zobrazování atd., je kontextově závislých. Takže například pokud kliknete na tlačítko *Zobrazit*, když máte vybraný katalog **Zařízení**, Calibre otevře pro zobrazení soubory v zařízení. Pokud máte vybraný katalog **Knihovna**, budou namísto toho otevřeny soubory ve vaší knihovně Calibre.

1.4 Hledání a řazení

Oddíl Hledání a řazení vám umožňuje provést několik výkonných akcí na vaší kolekci knih.

- Můžete je seřadit podle názvu, autora, data, hodnocení atd. kliknutím na názvy sloupců. Můžete také „podřazovat“, t.j. řadit podle více sloupců. Například pokud kliknete na sloupec název a pak na sloupec autor, knihy budou seřazeny podle autora a všechny položky pro stejného autora budou seřazeny podle názvu.
- Pomocí Panelu hledání můžete hledat konkrétní knihu nebo sadu knih. Více o tom níže.
- Můžete rychle a pohodlně upravovat metadata výběrem položky v seznamu, kterou chcete změnit, a stisknutím klávesy `E`.

- Můžete provádět akce na sadách knih. Pro výběr více knih můžete:
  - Ponechat stisknutou klávesu `Ctrl` a kliknout na knihy, které chcete vybrat.
  - Ponechat stisknutou klávesu `Shift` a kliknout na první a poslední knihu z rozsahu knih, které chcete vybrat.
- Můžete konfigurovat, která pole chcete zobrazit pomocí dialogového okna *Předvolby* (stránka 10).
- To perform complex multiple column based sub-sorting add the *Sort by* tool to a toolbar via *Preferences → Toolbars & menus*.

1.5 Rozhraní hledání

Všechna metadata knihy můžete prohledávat zadáním hledaných výrazů do Panelu hledání. Například:

```
Asimov Foundation format:lrf
```

Toto najde všechny knihy ve vaší knihovně, které mají ve svých metadatech `Asimov` a `Foundation` a jsou dostupné ve formátu LRF. Některé další příklady:

```
author:Asimov and not series:Foundation
title:"The Ring" or "This book is about a ring"
format:epub publisher:feedbooks.com
```

Druhy hledání

Existují čtyři druhy hledání: *obsahuje*, *rovnost*, *regulární výraz* (viz regulární výrazy[2]), a *varianta znaků*. Druh hledání volíte pomocí znaku předvolby.

‚Contains' searches

Searches with no prefix character are *contains* and are by default case insensitive. An item matches if the search string appears anywhere in the indicated metadata. You can make *contains* searches case sensitive by checking the option *Case sensitive searching* in *Preferences / Searching*. If the search option *Unaccented characters match accented characters and punctuation is ignored* is checked then a character will match all its variants (e.g., *e* matches *é*, *è*, *ê*, and *ë*) and all punctuation and whitespace are ignored. For example, if the *Unaccented characters match …* option is checked then given the two book titles:

1. Big, Bothéred, and Bad
2. Big Bummer

then these searches find:

- `title:"er"` matches both (‚e' matches both ‚é' and ‚e').
- `title:"g "` matches both because spaces are ignored.
- `title:"g,"` matches both because the comma is ignored.
- `title:"gb"` matches both because „ , is ignored in book 1 and spaces are ignored in book 2.
- `title:"g b"` matches both because comma and space are ignored.
- `title:"db"` matches #1 because the space in ‚and Bad' is ignored.
- `title:","` matches both (it actually matches all books) because commas are ignored.

If the *Unaccented characters match …* option is not checked then character variants, punctuation, and spacing are all significant.

[2] https://cs.wikipedia.org/wiki/Regul%C3%A1rn%C3%AD_v%C3%BDraz

You can set only one of the search options *Case sensitive searching* and *Unaccented characters match accented characters and punctuation is ignored*.

‚Equality' searches

Equality searches are indicated by prefixing the search string with an equals sign (=). For example, the query `tag:"=science"` will match *science*, but not *science fiction* or *hard science*. Character variants are significant: *é* doesn't match *e*.

Two variants of equality searches are used for hierarchical items (e.g., A.B.C): hierarchical prefix searches and hierarchical component searches. The first, indicated by a single period after the equals (`=.`) matches the initial parts of a hierarchical item. The second, indicated by two periods after the equals (`=..`) matches an internal name in the hierarchical item. Examples, using the tag `History.Military.WWII` as the value:

- `tags:"=.History"` : True. `History` is a prefix of the tag.
- `tags:"=.History.Military"` : True. `History.Military` is a prefix of the tag.
- `tags:"=.History.Military.WWII"` : True. `History.Military.WWII` is a prefix of the tag, albeit an improper one.
- `tags:"=.Military"` : False. `Military` is not a prefix of the tag.
- `tags:"=.WWII"` : False. `WWII` is not a prefix of the tag.
- `tags:"=..History"` : True. The hierarchy contains the value `History`.
- `tags:"=..Military"` : True. The hierarchy contains the value `Military`.
- `tags:"=..WWII"` : True. The hierarchy contains the value `WWII`.
- `tags:"=..Military.WWII"` : False. The `..` search looks for single values.

‚Regular expression' searches

Regular expression searches are indicated by prefixing the search string with a tilde (~). Any Python-compatible regular expression[3] can be used. Backslashes used to escape special characters in regular expressions must be doubled because single backslashes will be removed during query parsing. For example, to match a literal parenthesis you must enter `\\(` or alternatively use *super-quotes* (see below). Regular expression searches are ‚contains' searches unless the expression is anchored. Character variants are significant: `~e` doesn't match `é`.

‚Character variant' searches

Character variant searches are indicated by prefixing the search string with a caret (^). This search is similar to the *contains* search (above) except that:

- letter case is always ignored.
- character variants always match each other.
- punctuation and whitespace are always significant.

The search options *Unaccented characters match accented characters and punctuation is ignored* and *Case sensitive searching* are ignored. They have no effect on this search's behavior.

The following compares this search to a contains search assuming the *Unaccented characters match…* option is checked (see above) given the same two book titles:

1. Big, Bothéred, and Bad
2. Big Bummer

then these character variant searches find:

- `title:"^er"` matches both (‚e' matches both ‚é' and ‚e')

[3] https://docs.python.org/library/re.html

- `title:"^g"` matches both
- `title:"^g "` matches #2 because the space is significant
- `title:"^g,"` matches #1 because the comma is significant
- `title:"^gb"` matches nothing because space and comma are significant
- `title:"^g b"` matches #2 because the comma is significant
- `title:"^db"` matches nothing
- `title:"^,"` matches #1 (instead of all books) because the comma is significant

Search Expression Syntax

A *search expression* is a sequence of *search terms* optionally separated by the operators `and` and `or`. If two search terms occur without a separating operator, `and` is assumed. The `and` operator has priority over the `or` operator; for example the expression `a or b and c` is the same as `a or (b and c)`. You can use parenthesis to change the priority; for example `(a or b) and c` to make the `or` evaluate before the `and`. You can use the operator `not` to negate (invert) the result of evaluating a search expression. Examples:

- `not tag:foo` finds all books that don't contain the tag `foo`
- `not (author:Asimov or author:Weber)` finds all books not written by either Asimov or Weber.

The above examples show examples of *search terms*. A basic *search term* is a sequence of characters not including spaces, quotes (`"`), backslashes (`\`), or parentheses (`( )`). It can be optionally preceded by a column name specifier: the *lookup name* of a column followed by a colon (`:`), for example `author:Asimov`. If a search term must contain a space then the entire term must be enclosed in quotes, as in `title:"The Ring"`. If the search term must contain quotes then they must be *escaped* with backslashes. For example, to search for a series named *The „Ball" and The „Chain"*, use:

```
series:"The \"Ball\" and The \"Chain\"
```

If you need an actual backslash, something that happens frequently in *regular expression* searches, use two of them (`\\`).

It is sometimes hard to get all the escapes right so the result is what you want, especially in *regular expression* and *template* searches. In these cases use the *super-quote*: `"""sequence of characters"""`. Super-quoted characters are used unchanged: no escape processing is done.

More information

To search for a string that begins with an equals, tilde, or caret; prefix the string with a backslash.

Enclose search strings with quotes (") if the string contains parenthesis or spaces. For example, to find books with the tag `Science Fiction` you must search for `tag:"=science fiction"`. If you search for `tag:=science fiction` you will find all books with the tag `science` and the word `fiction` in any metadata.

You can build advanced search queries easily using the *Advanced search dialog* accessed by clicking the button on the left of the search box.

Pole dostupná pro vyhledávání jsou: `tag, title, author, publisher, series, series_index, rating, cover, comments, format, identifiers, date, pubdate, search, size, vl` a vlastní sloupce. Pokud je připojeno zařízení, bude při hledání v zobrazení knihovny Calibre dostupné pole `ondevice`. Abyste našli název hledání (ve skutečnosti nazvaný *název vyhledávání*) pro vlastní sloupec, přejeďte myší nad záhlavím sloupce v zobrazení knihovny.

Dates

Syntaxe pro hledání dat je:

```
pubdate:>2000-1 Will find all books published after Jan, 2000
date:<=2000-1-3 Will find all books added to calibre before 3 Jan, 2000
pubdate:=2009 Will find all books published in 2009
```

If the date is ambiguous then the current locale is used for date comparison. For example, in an mm/dd/yyyy locale 2/1/2009 is interpreted as 1 Feb 2009. In a dd/mm/yyyy locale it is interpreted as 2 Jan 2009. Some special date strings are available. The string `today` translates to today's date, whatever it is. The strings `yesterday` and `thismonth` (or the translated equivalent in the current language) also work. In addition, the string `daysago` (also translated) can be used to compare to a date some number of days ago. For example:

```
date:>10daysago
date:<=45daysago
```

Aby se zabránilo případným problémům s přeloženými řetězci při používání neanglické verze Calibre, jsou vždy dostupné řetězce `_today`, `_yesterday`, `_thismonth` a `_daysago`. Ty nejsou přeložené.

Searching dates and numeric values with relational comparisons

Data a číselná pole podporují relační operátory = (rovná se), > (větší než), >= (větší než nebo rovno), < (menší než), <= (menší než nebo rovno) a != (není rovno). Pole hodnocení jsou považována za číselná. Například hledání `rating:>=3` nalezne všechny knihy s hodnocením 3 nebo vyšším.

You can search for books that have a format of a certain size like this:

- `size:>1.1M` will find books with a format larger than 1.1MB
- `size:<=1K` will find books with a format smaller than or equal to 1KB

You can search for the number of items in multiple-valued fields such as tags using the character `#` then using the same syntax as numeric fields. For example, to find all books with more than 4 tags use `tags:#>4`. To find all books with exactly 10 tags use `tags:#=10`.

Series indices

Series indices are searchable. For the standard series, the search name is `series_index`. For custom series columns, use the column search name followed by _index. For example, to search the indices for a custom series column named `#my_series`, you would use the search name `#my_series_index`. Series indices are numbers, so you can use the relational operators described above.

Saved searches

The special field `search` is used for *saved searches* (stránka 17). If you save a search with the name „My spouse's books“ you can enter `search:"My spouse's books"` in the Search bar to reuse the saved search. More about saving searches below.

Virtual libraries

The special field `vl` is used to search for books in a Virtual library. For example, `vl:Read` will find all the books in the *Read* Virtual library. The search `vl:Read and vl:"Science Fiction"` will find all the books that are in both the *Read* and *Science Fiction* virtual libraries. The value following `vl:` must be the name of a Virtual library. If the Virtual library name contains spaces then surround it with quotes.

Whether a field has a value

You can search for the absence or presence of a value for a field using „true“ and „false“. For example:

- `cover:false` finds all books without a cover
- `series:true` finds all books that are in a series
- `series:false` finds all books that are not in a series
- `comments:false` finds all books with an empty comment

- `formats:false` finds all books with no book files (empty records)

Yes/no custom columns

Searching Yes/no custom columns for `false`, `empty`, or `blank` will find all books with undefined values in the column. Searching for `true` will find all books that do not have undefined values in the column. Searching for `yes` or `checked` will find all books with `Yes` in the column. Searching for `no` or `unchecked` will find all books with `No` in the column. Note that the words `yes`, `no`, `blank`, `empty`, `checked` and `unchecked` are translated; you can use either the current language's equivalent word or the English word. The words `true` and `false` and the special values `_yes`, `_no`, and `_empty` are not translated.

Identifiers

Identifiers (e.g., ISBN, DOI, LCCN, etc.) use an extended syntax. An identifier has the form `type:value`, as in `isbn:123456789`. The extended syntax permits you to specify independently the type and value to search for. Both the type and the value parts of the query can use any of the *search kinds* (stránka 12). Examples:

- `identifiers:true` najde knihy s jakýmkoliv identifikátorem.
- `identifiers:false` najde knihy bez identifikátoru.
- `identifiers:123` bude hledat knihy s jakýmkoliv typem majícím hodnotu obsahující *123*.
- `identifiers:=123456789` bude hledat knihy s jakýmkoliv typem majícím hodnotu rovnající se *123456789*.
- `identifiers:=isbn:` a `identifiers:isbn:true` najde knihy s typem rovnajícím se ISBN majícím jakoukoliv hodnotu.
- `identifiers:=isbn:false` najde knihy s žádným typem rovnajícím se ISBN.
- `identifiers:=isbn:123` najde knihy s typem rovnajícím se ISBN majícím hodnotu obsahující *123*.
- `identifiers:=isbn:=123456789` najde knihy s typem rovnajícím se ISBN majícím hodnotu rovnající se *123456789*.
- `identifiers:i:1` najde knihy s typem obsahujícím *i* majícím hodnotu obsahující *1*.

Categories visible in the Tag browser

The search `in_tag_browser:true` finds all books that are in categories (tags, authors, etc.) currently shown in the *Tag browser*. This is useful if you set the two preferences *Preferences → Look & feel → Tag browser → Hide empty categories* and *Find shows all items that match*. With those two preferences set, doing a *Find* in the *Tag browser* shows only categories containing items matched by the *Find*. Then, the search `in_tag_browser:true` additionally finds books with these categories / items. You can easily run this search by pressing the key `Ctrl+Alt+Shift+F` or clicking the configure button in the *Tag browser* and choosing the *Show only books that have visible categories* entry.

Search using templates

You can search using a template in *Jazyk šablon v programu Calibre* (stránka 160) instead of a metadata field. To do so you enter a template, a search type, and the value to search for. The syntax is:

```
template: (the template) #@#: (search type) : (the value)
```

The `template` is any valid calibre template language template. The `search type` must be one of `t` (text search), `d` (date search), `n` (numeric search), or `b` (set/not set (boolean)). The `value` is whatever you want, and can use the *search kinds* (stránka 12) described above for the various search types. You must quote the entire search string if there are spaces anywhere in it.

Příklady:

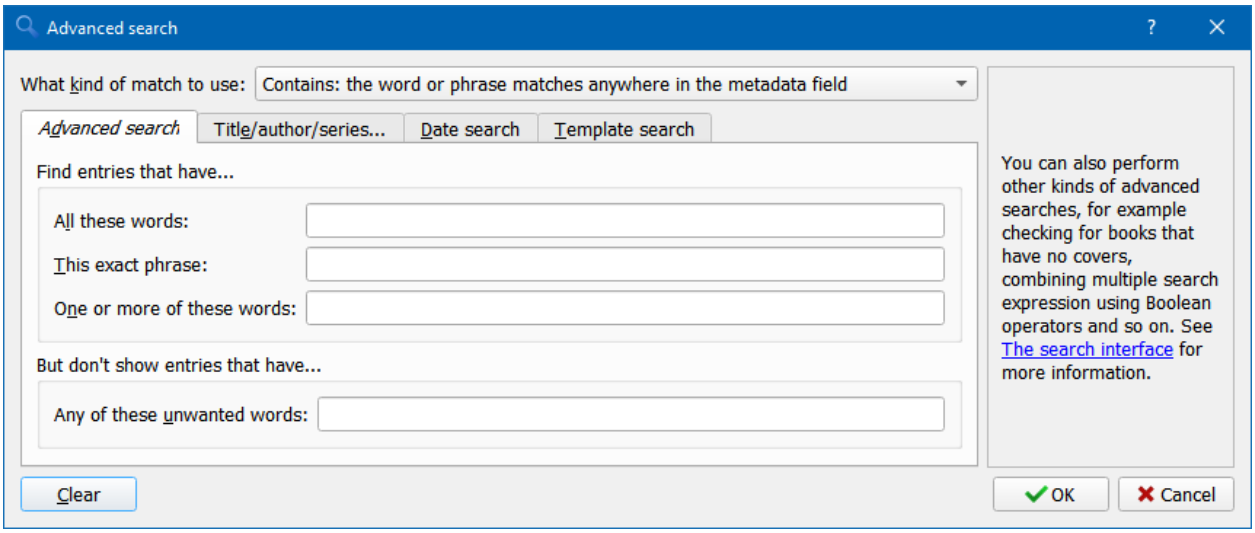
- `template:"program: connected_device_name('main')#@#:t:kindle"` – is true when the `kindle` device is connected.

- `template:"program: select(formats_sizes(), 'EPUB')#@#:n:>1000000"` – finds books with EPUB files larger than 1 MB.
- `template:"program: select(formats_modtimes('iso'), 'EPUB')#@#:d:>10daysago"` – finds books with EPUB files newer than 10 days ago.
- `template:"""program: book_count('tags:^"' & $series & '"', 0) != 0#@#:n:1"""` – finds all books containing the series name in the tags. This example uses super-quoting because the template uses both single quotes (') and double quotes (") when constructing the search expression.

You can build template search queries easily using the *Advanced search dialog* accessed by clicking the button .

You can test templates on specific books using the calibre *Template tester*, which can be added to the toolbars or menus via *Preferences → Toolbars & menus*. It can also be assigned a keyboard shortcut via *Preferences → Shortcuts*.

Advanced search dialog



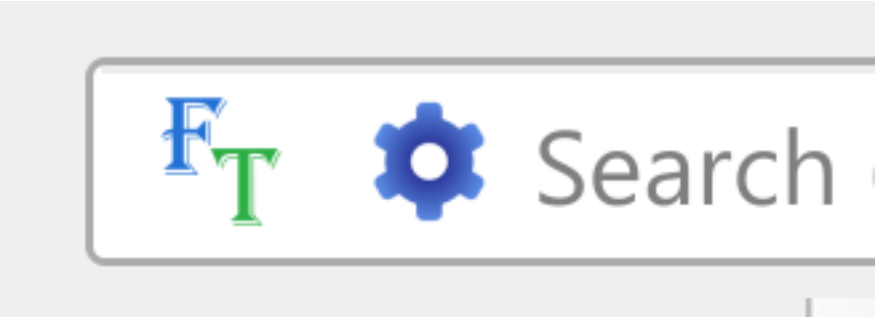
Obr. 1: *Dialogové okno Rozšířené hledání*

1.6 Ukládání hledání

Calibre vám umožňuje uložit často používané hledání pod speciálním názvem a pak znovu použít toto hledání pomocí jediného kliknutí. Abyste to udělali, vytvořte hledání buď zadáním do Panelu hledání, nebo pomocí Prohlížeče štítků. Potom zadejte název, který chcete dát hledání, do pole Uložená hledání vedle Panelu hledání. Klikněte na ikonu plus vedle pole uložených hledání pro uložení hledání.

Nyní můžete přistupovat ke svému uloženému hledání v Prohlížeči štítků pod *Uložená hledání*. Jedno kliknutím vám umožní snadno znovu použít jakékoliv libovolně složité hledání, aniž byste ho museli znovu vytvářet.

1.7 Searching the full text of all books



calibre can *optionally* index the full text of books in the library to allow for instant searching of words inside any book. To use this functionality click the *FT* button at the left edge of the search bar.

Then, enable indexing for the current library. Once indexing is complete you can search all the text in the full library. When you add new books to the library, they will be automatically indexed in the background. This search allows you to quickly find all books that contain a word or combination of words. You can even search for words that occur near other words, as shown in the examples in the search popup window. Note that this search tool will find only one occurrence of the search query in a particular book, not list every occurrence, for that it is best to search inside the book using the calibre *E-book viewer*.

You can re-index an individual book by right clicking on the *Book details panel* in calibre and choosing *Re-index this book for full text searching*.

1.8 Virtuální knihovny

Virtuální knihovna je způsob, jak předstírat, že vaše knihovna Calibre má pouze několik knih, namísto celé její kolekce. Je to vynikající způsob, jak rozdělit velkou kolekci knih na menší, snadno ovladatelné bloky. Abyste zjistili, jak vytvořit a používat Virtuální knihovny, podívejte se na kurz: *Virtuální knihovny* (stránka 247).

1.9 Dočasné označování knih

You can temporarily mark arbitrary sets of books. Marked books will have a pin on them and can be found with the search `marked:true`. To mark a book press `Ctrl+M` or go to *Preferences → Toolbars & menus* and add the *Mark books* button to the main toolbar.

You can mark books with a specific text label by right clicking the *Mark books* button and choosing *Mark books with text label*. Books marked with text labels can later be found using the search `marked:"=the-text-you-entered"`.

1.10 Odhadování metadat z názvů souborů

Obyčejně Calibre čte metadata zevnitř souboru knihy. Může však být nakonfigurováno tak, aby nemísto toho četlo metadata z názvu souboru, a to pomocí *Předvolby → Import a export → Přidávání knih → Číst metadata z obsahu souboru*.

Můžete také určit, jak jsou čtena metadata z názvu souboru pomocí regulárních výrazů (viz *Vše o používání regulárních výrazů v Calibre* (stránka 214)). V oddílu konfiguračního dialogového okna *Přidávání knih* můžete zadat regulární výraz, který Calibre použije, aby se pokusilo odhadnout metadata z názvů souborů e-knih, které přidáte do knihovny. Výchozí regulární výraz je:

```
title - author
```

to znamená, že předpokládá, že všechny znaky až do první – jsou název knihy a následující znaky jsou autor knihy. Například název souboru:

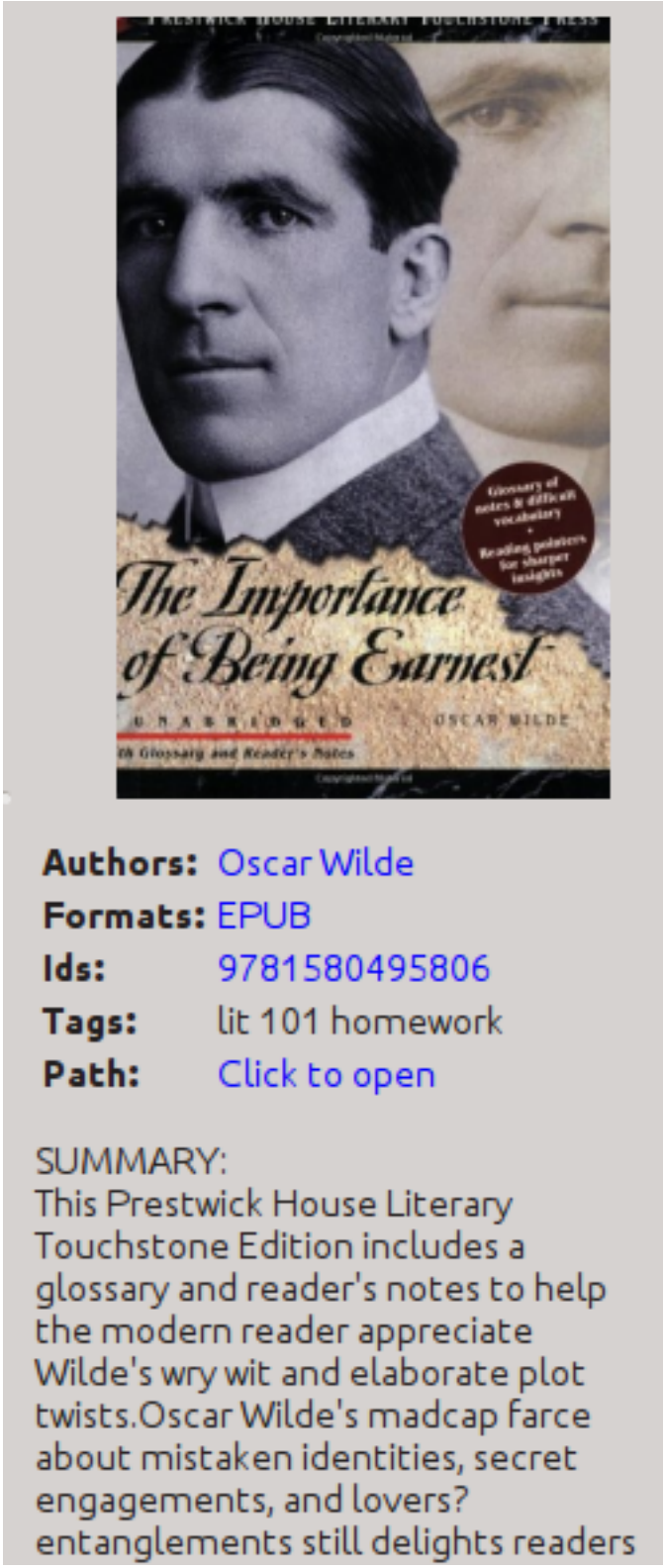
```
Foundation and Earth - Isaac Asimov.txt
```

bude interpretován, že má název: Foundation and Earth a autora: Isaac Asimov

Tip

Pokud název souboru neobsahuje spojovník, výše uvedený regulární výraz selže.

1.11 Podrobnosti o knize



Podrobnosti o knize zobrazují obálku a všechna metadata pro aktuálně vybranou knihu. Mohou být skryty prostřednictvím

tlačítka *Rozložení* v pravém dolním rohu hlavního okna Calibre. Na jména autorů zobrazená v panelu Podrobnosti o knize lze kliknout, ve výchozím nastavení vás přenesou na stránku autora na Wikipedii. Toto může být upraveno kliknutím pravým tlačítkem myši na jméno autora a výběrem Spravovat tohoto autora.

Podobně, pokud stáhnete metadata pro knihu, panel Podrobností o knize vám automaticky zobrazí odkazy směřující na webové stránky pro knihu na Amazonu, WorldCatu atd., odkud byla metadata stažena.

Můžete kliknout pravým tlačítkem myši na jednotlivé formáty e-knih v panelu Podrobnosti o knize k jejich odstranění, porovnání s jejich původními verzemi, uložení na disk, otevření v externím programu atd.

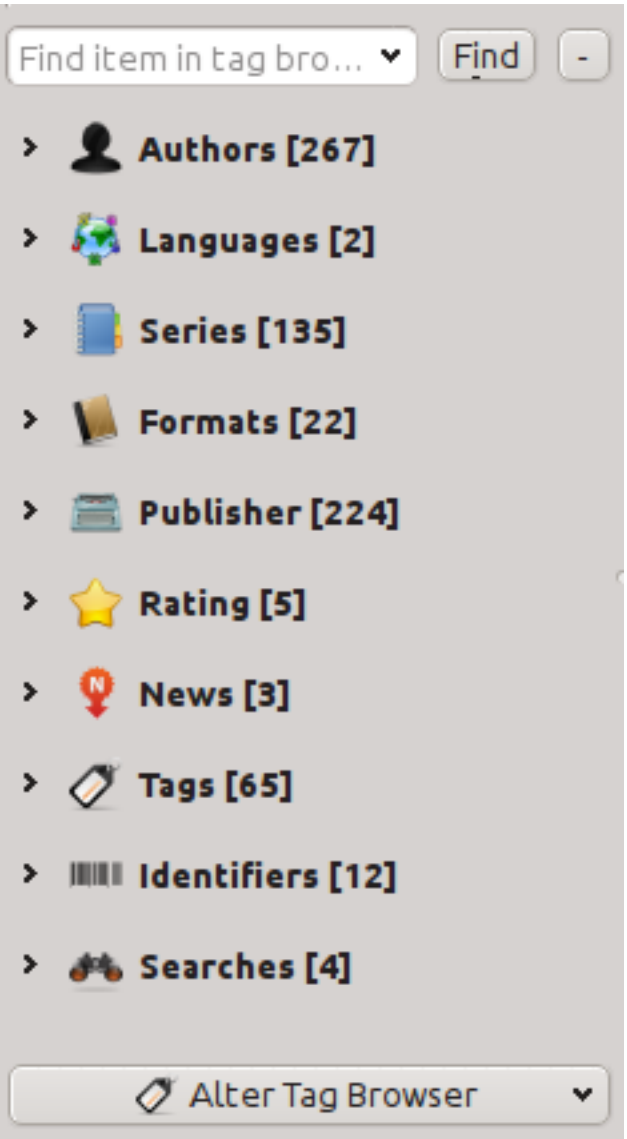
Obálku knihy můžete změnit jednoduše přetažením obrázku na panel Podrobnosti o knize. Pokud si přejete upravit obrázek obálky v externím programu, jednoduše na něj klikněte pravým tlačítkem myši a zvolte *Otevřít v programu*.

Můžete také přidat soubory e-knihy do aktuální knihy přetažením souborů na panel Podrobnosti o knize.

Dvojité kliknutí na panel Podrobnosti o knize ji otevře v samostatném automaticky otevřeném okně.

A nakonec můžete přesně přizpůsobit, jaké informace jsou zobrazeny v panelu Podrobnosti o knize pomocí *Předvolby → Rozhraní → Vzhled a chování → Podrobnosti o knize*.

1.12 Prohlížeč štítků



Prohlížeč štítků vám umožňuje snadno procházet vaši kolekci podle autora, štítků, série atd. Pokud kliknete na kteroukoliv položku v Prohlížeči štítků, například jméno autora Isaac Asimov, pak je seznam knih na pravé straně je omezen, aby zobrazoval knihy od tohoto autora. Můžete kliknout rovněž na názvy kategorií. Například kliknutí na „Série“ zobrazí všechny knihy v kterékoliv sérii.

První kliknutí na položku omezí seznam knih na ty, které obsahují nebo odpovídají položce. Jako pokračování výše uvedeného příkladu kliknutí na Isaaca Asimova zobrazí knihy od tohoto autora. Nové kliknutí na položku změní to, co je zobrazeno, v závislosti na tom, zda má položka podřízené položky (viz podkategorie a hierarchické položky níže). Jako pokračování příkladu s Isaacem Asimovem nové kliknutí na Isaaca Asimova omezí seznam knih na ty, které nejsou od Isaaca Asimova. Třetí kliknutí odebere omezení a zobrazí všechny knihy. Pokud podržíte klávesu Ctrl nebo `Shift` a kliknete na více položek, pak jsou vytvořena omezení na základě více položek. Například byste mohli držet klávesu Ctrl a kliknout na štítky Historická a Evropa pro nalezení knih o dějinách Evropy. Prohlížeč štítků funguje sestavováním vyhledávacích výrazů, které jsou automaticky zadávány do panelu hledání. Dívání se na to, co Prohlížeč štítků generuje, je dobrý způsob zjištění, jak vytvořit základní vyhledávací výrazy.

Položky v Prohlížeči štítků mají částečně barevné ikony. Množství barev závisí na průměrném hodnocení knih v dané

kategorii. Takže například pokud knihy od Isaaca Asimova mají průměrně čtyři hvězdičky, bude ikona pro Isaaca Asimova v Prohlížeči štítků barevná ze 4/5. Můžete přejet myší nad ikonou, abyste viděli průměrné hodnocení.

Položky na vnější úrovni v *Prohlížeči štítků*, jako jsou Autoři a Série, se nazývají kategorie. Můžete si vytvořit své vlastní kategorie, nazvané *Uživatelské kategorie*, které jsou užitečné pro organizování položek. Například můžete použít *Editor uživatelských kategorií* (vlevo pod *Prohlížečem štítků* klikněte na tlačítko *Konfigurovat* a zvolte *Spravovat autory, štítky atd.* → *Uživatelské kategorie*) pro vytvoření Uživatelské kategorie nazvané Oblíbení autoři, a pak umístit své oblíbené položky do kategorie. Uživatelské kategorie mohou mít podkategorie. Například Uživatelská kategorie `Oblíbené.Autoři` je podkategorie `Oblíbené`. Možná také máte `Oblíbené.Série`, v takovém případě budou pod `Oblíbené` dvě podkategorie. Podkategorie mohou být vytvořeny kliknutím pravým tlačítkem myši na Uživatelskou kategorii, výběrem *Přidat podkategorii do…* a zadáním názvu podkategorie; nebo pomocí *Editoru uživatelských kategorií* zadáním názvů, jako výše uvedený příklad Oblíbené.

V Uživatelských kategoriích můžete vyhledávat stejným způsobem jako ve vestavěných kategoriích kliknutím na ně. Při kliknutí kolují čtyři různá vyhledávání:

1. „všechno odpovídající položce v kategorii“ označené jednoduchým zeleným znaménkem plus.
2. „všechno odpovídající položce v kategorii nebo jejím podkategoriím“ označené dvěma zelenými znaménky plus.
3. „všechno neodpovídající položce v kategorii“ označené jednoduchým červeným znaménkem minus.
4. „všechno neodpovídající položce v kategorii nebo jejím podkategoriím“ označené dvěma červenými znaménky minus.

Je také možné vytvářet hierarchie uvnitř některých textových kategorií, jako jsou štítky, série a vlastní sloupce. Tyto hierarchie se zobrazují s malým trojúhelníkem umožňujícím skrytí podpoložek. Pro použití hierarchií položek v kategorii musíte nejprve přejít do *Předvolby* → *Rozhraní* → *Vzhled a chování* a zadat názvy kategorií do pole „Kategorie s hierarchickými položkami“. Jakmile to uděláte, položky v této kategorii, které obsahují tečky, se zobrazí pomocí malého trojúhelníku. Například předpokládejme, že vytvoříte vlastní sloupec nazvaný „Žánr“ a označíte ho, že obsahuje hierarchické položky. Jakmile to uděláte, položky jako Mysteriózní.Thriller a Mysteriózní.Česky se zobrazí jako Mysteriózní s malým trojúhelníkem. Kliknutí na trojúhelník zobrazí Thriller a Česky jako podpoložky. Pro více informací se podívejte na *Správa podskupiny knih, například „žánr“* (stránka 153).

Hierarchické položky (položky s podřízenými položkami) používají stejná čtyři ‚kliknutelná‘ vyhledávání jako Uživatelské kategorie. Položky, které nemají podřízené položky, používají dvě z vyhledávání: „všechno odpovídající“ a „všechno neodpovídající“.

Položky v Prohlížeči štítků můžete přetahovat na Uživatelské kategorie pro jejich přidání do této kategorie. Pokud je zdrojem Uživatelská kategorie, podržení klávesy `Shift` při přetažení přesune položku do nové kategorie. Můžete také přetahovat knihy ze seznamu knih na položky v Prohlížeči štítků; přetažení knihy na položku způsobí, že položka bude automaticky použita na přetaženou knihu. Například přetažení knihy na Isaaca Asimova nastaví autora této knihy na Isaaca Asimova. Její přetažení na štítek Historická přidá štítek Historická do štítků knihy.

You can easily find any item in the Tag browser by clicking the search button at the lower-right corner. In addition, you can right click on any item and choose one of several operations. Some examples are to hide it, rename it, or open a „Manage x“ dialog that allows you to manage items of that kind. For example, the *Manage authors* dialog allows you to rename authors and control how their names are sorted.

Jak jsou položky v Prohlížeči štítků řazeny, můžete určit prostřednictvím tlačítka *Konfigurovat* vlevo pod Prohlížečem štítků. Můžete si zvolit seřazení podle názvu, průměrného hodnocení nebo oblíbenosti (oblíbenost je počet knih u položky ve vaší knihovně; například oblíbenost Isaaca Asimova je počet knih ve vaší knihovně od Isaaca Asimova).

You can use your own icons for categories and values in categories. To change the icon for a category, right-click on the category (the outer-level item) and choose *Change (category name) category icon*. A dialog will open where you can pick an image to be used as the icon. To restore the icon to its default choose *Restore (category name) default icon*.

To choose icons for values in categories, right-click on a value then choose *Manage icon for (value name)*. You will see a list of choices:

- *Choose an icon for this value but not its children*. A dialog will open where you choose an icon for the value. Children of that value will not inherit that icon.
- *Choose an icon for this value and its children*. A dialog will open where you choose an icon for the value. Any children that don't have their own specified icon will inherit this icon.
- *Use the existing icon for this value but not its children*. This option is offered if the value already has an icon that is inherited by the value's children. Selecting it will make the icon apply to the value but not its children.
- *Use the existing icon for this value and its children*. This option is offered if the value already has an icon that is not inherited by the value's children. Selecting it will make the icon apply to the value and its children.
- *Use the default icon for this value*. This option is offered if the item has an icon. It removes the icon from the value and any children inheriting the icon. The default icon is what is specified below.
- *Reset all value icons to the default icon*. This option removes all item value icons for the category. It does not remove a template if one exists. There is no undo.
- *Use/edit a template to choose the default value icon*. This option permits you to provide a calibre template that returns the name of an icon file to be used as a default icon. The template can use two variables:
  - `category`: the lookup name of the category, for example `authors`, `series`, `#mycolumn`.
  - `value`: the value of the item within the category.
  - `count`: the number of books with this value. If the value is part of a hierarchy then the count includes the children.
  - `avg_rating`: the average rating for books with this value. If the value is part of a hierarchy then the average includes the children.

  Book metadata such as title is not available. Template database functions such as *book_count* (stránka 186) and *book_values* (stránka 187) will work, but the performance might not be acceptable. The following template functions will work in the GUI but won't work in the content server: *connected_device_name* (stránka 194), *connected_device_uuid* (stránka 194), *current_virtual_library_name* (stránka 195), *is_marked* (stránka 196), and *virtual_libraries* (stránka 197).

  In the GUI, Python templates have full access to the calibre database. In the content server, Python templates have access to new API (see API documentation for the database interface[4]) but not the old API (LibraryDatabase).

  For example, this template specifies that any value in the clicked-on category beginning with *History* will have an icon named `flower.png`:

```
program:
  if substr($value, 0, 7) == 'History' then 'flower.png' fi
```

  If a template returns the empty string (`''`) then the category icon will be used. If the template returns a file name that doesn't exist then no icon is displayed.
- *Use the category icon as the default*. This option specifies that the icon used for the category should be used for any value that doesn't otherwise have an icon. Selecting this option removes any template icon specification.

The icon is chosen using the following hierarchy:

1. The icon specified for the value, if it exists.
2. The icon specified for a parent node found by walking up the tree, if one exists.
3. The icon from a template, if a template exists and it returns a non-empty string.
4. The default category icon, which always exists.

[4] https://manual.calibre-ebook.com/db_api.html

Icons are per-user, not per-library, stored in the calibre configuration folder. Icons for item values are stored in the `tb_icons` subfolder. Icons used by templates are in the `template_icons` subfolder of `tb_icons`.

1.13 Mřížka obálek

Můžete nechat Calibre zobrazit mřížku obálek knih namísto seznamu knih, pokud dáváte přednost procházení své kolekce podle obálek. *Mřížka obálek* se aktivuje kliknutím na tlačítko *Rozložení* v pravém dolním rohu hlavního okna Calibre. Velikosti obálek a pozadí *Mřížky obálek* můžete přizpůsobit pomocí *Předvolby → Rozhraní → Vzhled a chování → Mřížka obálek*. Můžete dokonce nechat Calibre zobrazit jakékoliv zadané pole pod obálkami podle své vlastní vůle, jako je název, autoři, hodnocení nebo vlastní sloupec.

1.14 Prohlížeč obálek

Kromě *Mřížky obálek* popsané výše, můžete také nechat Calibre zobrazit obálky v jedné řadě. To se aktivuje prostřednictvím tlačítka *Rozložení* v pravém dolním rohu hlavního okna. V *Předvolby → Rozhraní → Vzhled a chování → Prohlížeč obálek* můžete změnit počet zobrazených obálek a dokonce nechat zobrazit samotný *Prohlížeč obálek* v samostatném automaticky otevíraném okně.

1.15 Adding notes for authors, series, etc.

You can add notes for an author/series/tag/publisher/etc. to your calibre library. To do so right click on the author name in the *Tag browser* on the left or the *Book details* panel on the right and choose *Create note* or *Edit note*.

A simple popup window will allow you to enter your notes using basic formatting and supporting links and images. Once a note for an author is created, it can be viewed easily from the *Book details* panel by clicking the little pencil icon next to the author name.

You can search through all the notes in your library using the *Browse notes* tool by pressing `Ctrl+Shift+N` or adding it to the toolbar via *Preferences → Toolbars & menus*.

1.16 Rychlé zobrazení

Někdy chcete vybrat knihu a rychle získat seznam knih se stejnou hodnotou v některé kategorii (autoři, štítky, vydavatel, série atd.) jako aktuálně vybraná kniha, ale beze změny aktuálního zobrazení knihovny. Můžete to udělat pomocí Rychlého zobrazení. Rychlé zobrazení otevře buď druhé okno nebo panel v seznamu knih zobrazující seznam knih odpovídajících hodnotě, o kterou máte zájem. Předpokládejme například, že chcete vidět seznam všech knih s jedním nebo více autory aktuálně vybrané knihy. Klikněte do buňky autora, která vás zajímá, a stiskněte klávesu ‚Q' nebo klikněte na ikonu *Rychlé zobrazení* v oddíle *Rozložení* v okně Calibre. Vlevo se otevře okno nebo panel se všemi autory pro tuto knihu a vpravo všechny knihy od vybraného autora.

Některé příklady použití Rychlého zobrazení: rychlé ukázání, které další knihy:

- mají některé štítky použité u aktuálně vybrané knihy,
- jsou ve stejné sérii jako aktuální kniha,
- mají stejné hodnoty ve vlastním sloupci jako aktuální kniha,
- jsou napsány jedním ze stejných autorů aktuální knihy
- sdílet hodnoty ve vlastním sloupci

Existují dvě možnosti, kde se informace Rychlého zobrazení objeví:

1. Mohou být otevřeny „neukotvené": nad oknem Calibre a zůstanou otevřeny, dokud je výslovně nezavřete.
2. Mohou být otevřeny „ukotvené": jako panel v oddíle seznamu knih hlavního okna Calibre.

Pomocí tlačítka „Ukotvit nebo zrušit ukotvení" můžete měnit okno z ukotveného na neukotvené podle potřeby.

Panel Rychlého zobrazení může být ponechán otevřen trvale, v takovém případě sleduje pohyby na seznamu knih. Například pokud v zobrazení knihovny Calibre kliknete na sloupec kategorie (štítky, série, vydavatel, autoři atd.) pro knihu, obsah okna Rychlého zobrazení se změní, aby vám ukázal v levém bočním panelu hodnoty v této kategorii pro vybranou knihu (např. štítky pro tuto knihu). Bude vybrána první položka v tomto seznamu a Rychlé zobrazení vám ukáže v pravém bočním panelu všechny knihy ve vaší knihovně, které používají tuto hodnotu. Klikněte na jinou hodnotu v levém panelu, abyste viděli knihy s touto jinou hodnotou.

Poklepejte na knihu v okně Rychlého zobrazení pro výběr této knihy v zobrazení knihovny. Toto také změní položky zobrazené v okně Rychlého zobrazení (levý panel), aby se zobrazily položky v nově vybrané knize.

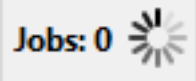
Poklepejte se `Shift`– nebo `Ctrl`– na knihu v okně Rychlého zobrazení pro otevření dialogového okna úpravy metadat této knihy v okně Calibre. Upravovaná kniha bude rychle zobrazena, když zavřete dialogové okno úpravy metadat.

Přejetím myší nad záhlavím sloupce a podíváním se na popis tlačítka pro toto záhlaví můžete vidět, zda sloupec může být zobrazen v Rychlém zobrazení. Také můžete zjistit kliknutím pravým tlačítkem myši na záhlaví sloupce, zda je volba „Rychlé zobrazení" uvedeno v nabídce, v takovém případě výběr volby Rychlého zobrazení odpovídá stisknutí ‚Q' v aktuální buňce.

Volby (v *Předvolby → Vzhled a chování → Rychlé zobrazení*):

- Respektovat (nebo ne) aktuální Virtuální knihovnu. Pokud je zaškrtnuto, Rychlé zobrazení ukáže pouze knihy v aktuální Virtuální knihovně. Výchozí: respektovat Virtuální knihovny
- Změňte obsah okna Rychlého zobrazení, když je změněn sloupec v seznamu knih pomocí kurzorových kláves. Výchozí: nesledovat změny prováděné kurzorovými klávesami
- Změnit sloupec, který je „rychle zobrazen", když je poklepáno v buňce v okně Rychlého náhledu. Jinak se změní kniha, ale prozkoumávaný sloupec ne. Výchozí: změnit sloupec
- Změnit sloupec, který je „rychle zobrazen", na aktuální sloupec, když je v panelu Rychlého náhledu stisknuta klávesa Enter. Jinak se změní kniha, ale prozkoumávaný sloupec ne. Výchozí: změnit sloupec
- Zvolte, které sloupce jsou zobrazeny v okně nebo panelu Rychlého zobrazení.

1.17 Úlohy



Panel Úlohy zobrazuje počet aktuálně spuštěných úloh. Úlohy jsou úkoly, které běží v samostatném procesu. Zahrnují převody e-knih a komunikaci s vaším čtecím zařízení. Pro přístup k seznamu úloh můžete kliknout na panel úloh. Jakmile je úloha dokončena, můžete si prohlédnout podrobný protokol z této úlohy poklepáním na něj v seznamu. To je užitečné pro úlohy ladění, které nemusely být úspěšně dokončeny.

1.18 Klávesové zkratky

calibre has several keyboard shortcuts to save you time and mouse movement. These shortcuts are active in the book list view (when you're not editing the details of a particular book), and most of them affect the title you have selected. The calibre E-book viewer *has its own shortcuts* (stránka 57) which can be customised in the viewer *Preferences*.

Poznámka

Poznámka: Klávesové zkratky Calibre nevyžadují modifikační klávesu (Command, Option, Control atd.), pokud není výslovně uvedeno jinak. Stačí pouze stisknout klávesu s písmenem, například E pro úpravu.

Tabulka 1: Klávesové zkratky pro hlavní program Calibre

Klávesová zkratka	Akce
`F2 (Enter for macOS)`	Upravit metadata aktuálně vybraného pole v seznamu knih
`A`	Přidat knihy
`Shift+A`	Přidat formáty k vybraným knihám
`C`	Převést vybrané knihy
`D`	Odeslat do zařízení
`Del`	Odebrat vybrané knihy
`E`	Upravit metadata vybraných knih
`G`	Získat knihy
`I`	Zobrazit Podrobnosti o knize
`K`	Upravit obsah
`M`	Sloučit vybrané záznamy
`Alt+M`	Sloučit vybrané záznamy, ponechat originály
`O`	Otevřít nadřazenou složku
`P`	Vylepšit knihy
`S`	Uložit na disk
`T`	Upravit knihu
`V`	Zobrazit
`Shift+V`	Zobrazit naposledy čtenou knihu

continues on next page

Tabulka 1 – pokračujte na předchozí stránce

Kláve-sová zkrat-ka	Akce
Alt+V/ Cmd+V for macOS	Zobrazit konkrétní formát
Alt+Shi	Přepnout seznam úloh
Alt+Shi	Přepnout Prohlížeč obálek
Alt+Shi	Přepnout panel Podrobnosti o knize
Alt+Shi	Přepnout Prohlížeč štítků
Alt+Shi	Přepnout Mřížku obálek
Alt+A	Zobrazit knihy od stejného autora jako aktuální kniha
Alt+T	Zobrazit knihy se stejnými štítky jako aktuální kniha
Alt+P	Zobrazit knihy od stejného vydavatele jako aktuální kniha
Alt+Shi	Zobrazit knihy ze stejné série jako aktuální kniha
/, Ctrl+F	Zaměřit Panel hledání
Shift+C	Otevřít dialogové okno Rozšířené hledání
Shift+A	Přepnout Panel hledání
Esc	Vymazat aktuální hledání
Shift+E	Zaměřit seznam knih
Ctrl+Es	Vymazat Virtuální knihovnu
Alt+Esc	Vymazat dodatečná omezení
Ctrl+*	Vytvořit dočasnou Virtuální knihovnu založenou na aktuálním hledání
Ctrl+Vp	Vybrat další kartu Virtuální knihovny
Ctrl+Vl	Vybrat předchozí kartu Virtuální knihovny
N nebo F3	Najít další knihu, která odpovídá aktuálnímu hledání (funguje, pouze pokud je zapnuto zvýrazňování hledání v předvolbách hledání)
Shift+N nebo Shift+F	Najít předchozí knihu, která odpovídá aktuálnímu hledání (funguje, pouze pokud je zapnuto zvýrazňování hledání v předvolbách hledání)
Ctrl+D	Stáhnout metadata a obálky
Ctrl+R	Restartovat Calibre
Ctrl+Sh	Restartovat Calibre v režimu ladění
Shift+C	Přidat prázdné knihy do Calibre
Ctrl+M	Přepnout stav označení nebo neoznačení pro vybrané knihy
Ctrl+/ or Ctrl+Al	Open the popup to search the full text of all books in the library
Q	Otevřít automaticky otevírané Rychlé zobrazení pro zobrazení knih v souvisejících sériích, štítcích atd.
Shift+Q	Zaměřit na otevřený panel Rychlého zobrazení
Shift+S	Provést hledání v panelu Rychlého zobrazení
F5	Znovu použít aktuální řazení
Ctrl+Q	Ukončit Calibre
X	Přepnout automatické posouvání seznamu knih
Ctrl+Al	Restrict the displayed books to only those books that are in a category currently displayed in the *Tag browser*
B	Browse annotations (highlights and bookmarks) made in the calibre E-book viewer for all books in the library
Ctrl+Sh	Browse notes associated with authors/series/tags/etc.

continues on next page

Tabulka 1 – pokračujte na předchozí stránce

Klávesová zkratka	Akce
Alt+Shi	Toggle the layout between wide and narrow views

KAPITOLA 2

Přidávání oblíbených webových stránek se zprávami

Calibre má výkonné, pružné a snadno použitelné rozhraní pro stahování zpráv z internetu a jejich převodu do e-knihy. Následující vám za pomoci příkladů ukáže, jak získat zprávy z různých webových stránek.

K pochopení, jak používat rozhraní, následujte příklady ve vypsaném pořadí níže:

- *Úplně automatické načítání* (stránka 33)
  - *Blog Calibre* (stránka 34)
  - *bbc.co.uk* (stránka 35)
- *Přizpůsobení procesu načítání* (stránka 35)
  - *Použití tiskové verze bbc.co.uk* (stránka 35)
  - *Nahrazování stylů článku* (stránka 36)
  - *Krájení a sekání* (stránka 37)
  - *Příklad ze skutečného života* (stránka 37)
- *Tipy pro vývoj nových předpisů* (stránka 40)
- *Další čtení* (stránka 41)
- *Dokumentace API* (stránka 41)

2.1 Úplně automatické načítání

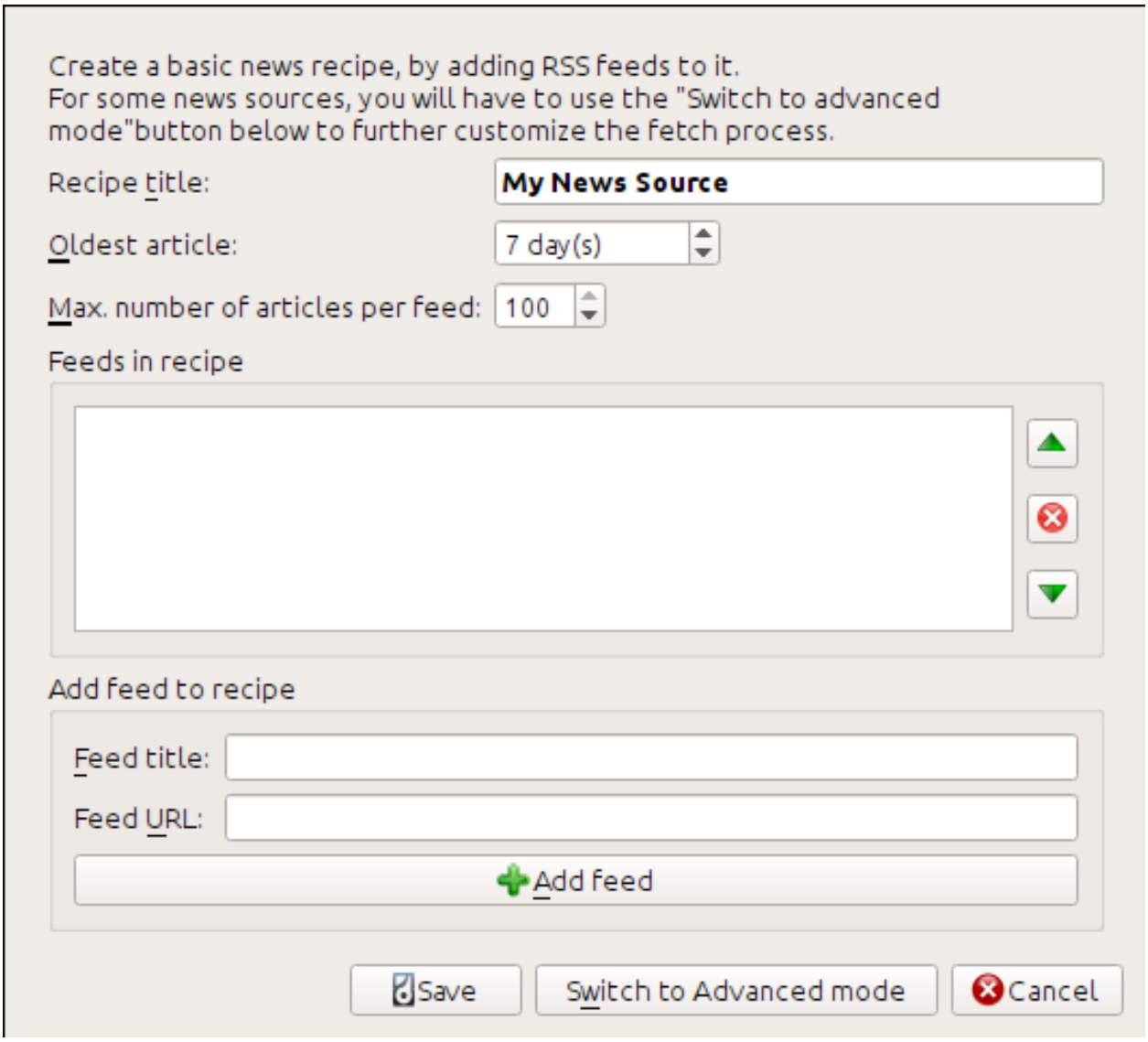
Pokud je váš zdroj zpráv dostatečně jednoduchý, Calibre může být schopné získat ho úplně automaticky, potřebujete jen zadat URL. Calibre shromáždí všechny informace potřebné ke stažení zdroje zpráv do předpisu. Abyste řekli Calibre o zdroji zpráv, musíte pro to vytvořit *předpis*. Pojďme se podívat na nějaké příklady:

2.1.1 Blog Calibre

Blog Calibre je blog příspěvků, které popisují mnoho užitečných funkcí Calibre jednoduchým a přístupným způsobem pro nové uživatele Calibre. Pro stažení tohoto blogu do e-knihy spoléháme na informační kanál *RSS* blogu:

```
http://blog.calibre-ebook.com/feeds/posts/default
```

URL RSS jsem získal podíváním se pod „Subscribe to“ dole na stránce blogu a výběrem *Posts → Atom*. Aby Calibre stáhlo informační kanály a převedlo je na e-knihu, měli byste kliknout pravým tlačítkem myši na tlačítko *Načíst zprávy* a pak na položku nabídky *Přidat vlastní zdroj zpráv* a pak na tlačítko :guilabel:`Nový předpis. Mělo by se otevřít dialogové okno podobné tomu níže zobrazenému.



Nejdříve zadejte `Blog Calibre` do pole *Název předpisu*. To bude název e-knihy, která bude vytvořena z článků ve výše uvedených informačních kanálech.

Další dvě pole (*Nejstarší článek* a *Max. počet článků*) vám umožní kontrolu nad tím, kolik článků by mělo být staženo z každého informačního kanálu, a jsou docela sebevysvětlující.

Pro přidání informačních kanálů do předpisu zadejte název a URL informačního kanálu a klikněte na tlačítko *Přidat informační kanál*. Jakmile jste přidali informační kanál, jednoduše klikněte na tlačítko *Uložit* a máte hotovo! Zavřete

dialogové okno.

Pro otestování nového *předpisu* klikněte na tlačítko *Načíst zprávy* a v podnabídce *Vlastní zdroje zpráv* klikněte na *Blog Calibre*. Po několika minutách se nově stažená e-kniha příspěvků blogu zobrazí v hlavním zobrazení knihovny (pokud máte připojenou čtečku, bude umístěna do čtečky namísto do knihovny). Vyberte ji a stiskněte tlačítko *Zobrazit* pro čtení!

Důvod, proč toto fungovalo tak dobře s tak málo úsilím, je ten, že blog poskytuje informační kanály *RSS* s *úplným obsahem*, t.j. obsah článku je vložen do samotného informačního kanálu. Pro většinu zdrojů zpráv, které poskytují zprávy tímto způsobem s informačními kanály s *úplným obsahem*, nepotřebujete žádné další úsilí, abyste je převedli na e-knihy. Nyní se podíváme na zdroje zpráv, které neposkytují informační kanály s úplným obsahem. V takových informačních kanálech je celý článek webová stránka a informační kanál obsahuje pouze odkaz na webovou stránku s krátkým souhrnem článku.

2.1.2 bbc.co.uk

Zkusme následující dva informační kanály z *BBC*:

1. Titulní sránka zpráv: https://newsrss.bbc.co.uk/rss/newsonline_world_edition/front_page/rss.xml
2. Věda a příroda: https://newsrss.bbc.co.uk/rss/newsonline_world_edition/science/nature/rss.xml

Postupujte podle pokynů navržených ve výše uvedeném *Blog Calibre* (stránka 34) pro vytvoření předpisu pro *BBC* (pomocí výše uvedených informačních kanálů). Při pohledu na staženou e-knihu vidíme, že Calibre udělalo slušnou práci vyextrahováním pouze obsahu, na kterém vám záleží, z každé webové stránky článku. Proces extrakce však není dokonalý. Někdy ponechá nežádoucí obsah, jako jsou nabídky a navigační pomůcky, nebo odebere obsah, který by měl být ponechán, jako jsou záhlaví článku. Abychom dosáhli dokonalé extrakce obsahu, budeme muset přizpůsobit proces načtení, jak je popsáno v následujícím oddíle.

2.2 Přizpůsobení procesu načítání

Pokud chcete zdokonalit proces stahování, nebo stáhnout obsah z obzvláště složité webové stránky, můžete využít celou sílu a pružnost rozhraní předpisů. Za tímto účelem v dialogovém okně *Přidat vlastní zdroje zpráv* jednoduše klikněte na tlačítko *Přepnout do rozšířeného režimu*.

Nejjednodušší a často nejproduktivnější přizpůsobení je použít tiskovou verzi online článků. Tisková verze má obvykle mnohem méně nadbytečných dat a překládá se mnohem plynuleji do e-knihy. Zkusme použít tiskovou verzi článků z *BBC*.

2.2.1 Použití tiskové verze bbc.co.uk

Prvním krokem je podívat se na e-knihu, kterou jsme dříve stáhli z *bbc.co.uk* (stránka 35). Na konci každého článku v e-knize je malá záložka oznamující, odkud byl článek stažen z. Zkopírujte a vložte tuto URL do prohlížeče. Nyní na webové stránce článku najděte odkaz, který odkazuje na „Tisknutelnou verzi“. Klikněte na něj pro zobrazení tiskové verze článku. Vypadá to mnohem čistěji! Nyní porovnejte obě URL. Pro mě byly:

URL článku
: https://news.bbc.co.uk/2/hi/science/nature/7312016.stm

URL tiskové verze
: https://newsvote.bbc.co.uk/mpapps/pagetools/print/news.bbc.co.uk/2/hi/science/nature/7312016.stm

Takže to vypadá, že pro získání tiskové verze musíme přidat ke každé URL článku předponu:

newsvote.bbc.co.uk/mpapps/pagetools/print/

Nyní byste měli v *Pokročilém režimu* dialogového okna Vlastní zdroje zpráv vidět něco podobného (nezapomeňte vybrat předpis *BBC* před přepnutím do pokročilého režimu):

Recipe source code (python)

```
class AdvancedUserRecipe1206418393(BasicNewsRecipe):
    title          = u'The BBC'
    oldest_article = 7
    max_articles_per_feed = 100

    feeds          = [(u'News Front Page', u'http://newsrss.bbc.co.uk/rss/newsonlir
```

Můžete vidět, že pole ze *Základního režimu* byla přeložena do kódu Pythonu přímočarým způsobem. Potřebujeme přidat pokyny k tomuto předpisu, aby použil tiskovou verzi článků. Vše, co je potřeba, je přidat následující dva řádky:

```
def print_version(self, url):
    return url.replace('https://', 'https://newsvote.bbc.co.uk/mpapps/pagetools/print/
↪')
```

Toto je Python, takže odsazení je důležité. Poté, co jste přidali řádky, mělo by to vypadat takto:

Recipe source code (python)

```
class AdvancedUserRecipe1206418393(BasicNewsRecipe):
    title          = u'The BBC'
    oldest_article = 7
    max_articles_per_feed = 100

    feeds          = [(u'News Front Page', u'http://newsrss.bbc.co.uk/rss/newsonlir

    def print_version(self, url):
        return url.replace('http://', 'http://newsvote.bbc.co.uk/mpapps/pagetools/p
```

Ve výše uvedeném `def print_version(self, url)` definuje *metodu*, která je volána Calibre pro každý článek. `url` je URL původního článku. `print_version` dělá to, že vezme tuto URL a nahradí ji novou URL, která odkazuje na tiskovou verzi článku. Abyste se naučili více o Pythonu[5], podívejte se na kurz[6].

Nyní klikněte na tlačítko *Přidat nebo aktualizovat předpis* a vaše změny budou uloženy. Znovu stáhněte e-knihu. Měli byste mít mnohem lepší e-knihu. Jeden z problémů s novou verzí je, že písma na tiskové verzi webové stránky jsou příliš malá. To je automaticky opraveno při převodu na e-knihu, ale i po procesu opravy bude velikost písma nabídek a navigačního panelu příliš velká vzhledem k textu článku. Abychom to opravili, provedeme nějaká další přizpůsobení v dalším oddíle.

2.2.2 Nahrazování stylů článku

V předchozím oddíle jsme viděli, že velikost písma pro články z tiskové verze *BBC* byla příliš malá. Ve většině webových stránek, včetně *BBC*, je tato velikost písma nastavena pomocí šablon stylů *CSS*. Můžeme zakázat načítání těchto šablon stylů přidáním řádku:

```
no_stylesheets = True
```

Předpis nyní vypadá takto:

[5] https://www.python.org
[6] https://docs.python.org/tutorial/

Recipe source code (python)

```
class AdvancedUserRecipe1206419520(BasicNewsRecipe):
    title          = u'The BBC'
    oldest_article = 7
    max_articles_per_feed = 100
    no_stylesheets = True

    feeds          = [(u'News Front Page', u'http://newsrss.bbc.co.uk/rss/newsonli

    def print_version(self, url):
        return url.replace('http://', 'http://newsvote.bbc.co.uk/mpapps/pagetools/
```

Nová verze vypadá docela dobře. Pokud jste perfekcionista, budete si chtít přečíst další oddíl, který se zabývá skutečnou úpravou staženého obsahu.

2.2.3 Krájení a sekání

Calibre obsahuje velice výkonné a pružné možnosti, pokud jde o zacházení se staženým obsahem. Abychom vám pár z nich ukázali, pojďme se znovu podívat na našeho starého přítele, na předpis *BBC* (stránka 36). Při pohledu na zdrojový kód (*HTML*) pár článků (tisková verze) vidíme, že mají patičku, která neobsahuje žádné užitečné informace, obsažené v

```
<div class="footer">
...
</div>
```

Toto může být odebráno přidáním:

```
remove_tags    = [dict(name='div', attrs={'class':'footer'})]
```

k předpisu. A nakonec nahraďme některé z *CSS*, které jsme dříve zakázali, naším vlastním *CSS*, které je vhodné pro převod na e-knihu:

```
extra_css      = '.headline {font-size: x-large;} \n .fact { padding-top: 10pt  }'
```

S těmito dodatky se náš předpis stal „kvalitou výroby“.

Tento *předpis* prozkoumá pouze špičku ledovce, pokud jde o sílu Calibre. Abychom prozkoumali více možností Calibre, prohlédneme si v dalším oddíle složitější příklad ze skutečného života.

2.2.4 Příklad ze skutečného života

Přiměřeně složitý příklad ze skutečného života, který odhaluje více z *API* `BasicNewsRecipe`, je *předpis* pro *The New York Times*

```
import string, re
from calibre import strftime
from calibre.web.feeds.recipes import BasicNewsRecipe
from calibre.ebooks.BeautifulSoup import BeautifulSoup

class NYTimes(BasicNewsRecipe):

    title       = 'The New York Times'
```

(continues on next page)

(pokračujte na předchozí stránce)

```
    __author__  = 'Kovid Goyal'
    description = 'Daily news from the New York Times'
    timefmt = ' [%a, %d %b, %Y]'
    needs_subscription = True
    remove_tags_before = dict(id='article')
    remove_tags_after  = dict(id='article')
    remove_tags = [dict(attrs={'class':['articleTools', 'post-tools', 'side_tool',
↪'nextArticleLink clearfix']}),
                dict(id=['footer', 'toolsRight', 'articleInline', 'navigation',
↪'archive', 'side_search', 'blog_sidebar', 'side_tool', 'side_index']),
                dict(name=['script', 'noscript', 'style'])]
    encoding = 'cp1252'
    no_stylesheets = True
    extra_css = 'h1 {font: sans-serif large;}\n.byline {font:monospace;}'

    def get_browser(self):
        br = BasicNewsRecipe.get_browser(self)
        if self.username is not None and self.password is not None:
            br.open('https://www.nytimes.com/auth/login')
            br.select_form(name='login')
            br['USERID']   = self.username
            br['PASSWORD'] = self.password
            br.submit()
        return br

    def parse_index(self):
        soup = self.index_to_soup('https://www.nytimes.com/pages/todayspaper/index.
↪html')

        def feed_title(div):
            return ''.join(div.findAll(text=True, recursive=False)).strip()

        articles = {}
        key = None
        ans = []
        for div in soup.findAll(True,
             attrs={'class':['section-headline', 'story', 'story headline']}):

             if ''.join(div['class']) == 'section-headline':
                 key = string.capwords(feed_title(div))
                 articles[key] = []
                 ans.append(key)

             elif ''.join(div['class']) in ['story', 'story headline']:
                 a = div.find('a', href=True)
                 if not a:
                     continue
                 url = re.sub(r'\?.*', '', a['href'])
                 url += '?pagewanted=all'
                 title = self.tag_to_string(a, use_alt=True).strip()
                 description = ''
                 pubdate = strftime('%a, %d %b')
```

(continues on next page)

(pokračujte na předchozí stránce)

```
                summary = div.find(True, attrs={'class':'summary'})
                if summary:
                    description = self.tag_to_string(summary, use_alt=False)

                feed = key if key is not None else 'Uncategorized'
                if feed not in articles:
                    articles[feed] = []
                if not 'podcasts' in url:
                    articles[feed].append(
                              dict(title=title, url=url, date=pubdate,
                                   description=description,
                                   content=''))
        ans = self.sort_index_by(ans, {'The Front Page':-1, 'Dining In, Dining Out':1,
↪ 'Obituaries':2})
        ans = [(key, articles[key]) for key in ans if key in articles]
        return ans

    def preprocess_html(self, soup):
        refresh = soup.find('meta', {'http-equiv':'refresh'})
        if refresh is None:
            return soup
        content = refresh.get('content').partition('=')[2]
        raw = self.browser.open('https://www.nytimes.com'+content).read()
        return BeautifulSoup(raw.decode('cp1252', 'replace'))
```

V tomto předpisu vidíme několik nových funkcí. Nejdříve máme:

```
timefmt = ' [%a, %d %b, %Y]'
```

To nastaví zobrazený čas na úvodní stránce vytvořené e-knihy, aby byl ve formátu `Day, Day_Number Month, Year`. Podívejte se na *timefmt* (stránka 51).

Pak vidíme skupinu směrnic pro vyčištění staženého *HTML*:

```
remove_tags_before = dict(name='h1')
remove_tags_after  = dict(id='footer')
remove_tags = ...
```

Tyto odeberou vše před první značkou `<h1>` a vše za první značkou, jejíž identifikátor je `footer`. Podívejte se na *remove_tags* (stránka 50), *remove_tags_before* (stránka 50), *remove_tags_after* (stránka 50).

Další zajímavá funkce je:

```
needs_subscription = True
...
def get_browser(self):
    ...
```

`needs_subscription = True` říká Calibre, že tento předpis potřebuje uživatelské jméno a heslo pro přístup k obsahu. To způsobí, že Calibre požádá o uživatelské jméno a heslo, kdykoliv se pokusíte použít tento předpis. Kód v *calibre.web.feeds.news.BasicNewsRecipe.get_browser()* (stránka 42) skutečně provede přihlášení na webovou stránku NYT. Jakmile se přihlásíte, Calibre použije stejnou, přihlášenou instance prohlížeče k načtení celého obsahu. Podívejte se na automatizace[7] k porozumění kódu v `get_browser`.

[7] https://mechanize.readthedocs.io/en/latest/

Další novou funkcí je metoda *calibre.web.feeds.news.BasicNewsRecipe.parse_index()* (stránka 44). Její úlohou je přejít na https://www.nytimes.com/pages/todayspaper/index.html a načíst seznam článků, které se objevují v *dnešních* novinách. I když je to složitější, než jednoduché použití *RSS*, předpis vytvoří e-knihu, která je velice blízká novinám. `parse_index` provádí těžké použití BeautifulSoup[8] pro analýzu webových stránek novin. Můžete také použít jiné, modernější analyzátory, pokud se vám nelíbí BeatifulSoup. Calibre přichází s lxml[9] a html5lib[10], které jsou doporučenými analyzátory. Pro jejich použití nahraďte volání `index_to_soup()` následujícím:

```
raw = self.index_to_soup(url, raw=True)
# For html5lib
import html5lib
root = html5lib.parse(raw, namespaceHTMLElements=False, treebuilder='lxml')
# For the lxml html 4 parser
from lxml import html
root = html.fromstring(raw)
```

Poslední novou funkcí je metoda *calibre.web.feeds.news.BasicNewsRecipe.preprocess_html()* (stránka 45). Může být použita k provedení libovolných transformací na každé stažené stránce HTML. Zde je použita k obejití reklam, které NYT zobrazuje před každým článkem.

2.3 Tipy pro vývoj nových předpisů

Nejlepším způsobem vývoje nových předpisů je použít rozhraní příkazového řádku. Vytvořte předpis pomocí svého oblíbeného editoru Pythonu a uložte ho do souboru `myrecipe.recipe`. Přípona *.recipe* je vyžadována. Pomocí tohoto předpisu můžete stáhnout obsah příkazem:

```
ebook-convert myrecipe.recipe .epub --test -vv --debug-pipeline debug
```

Příkaz **`ebook-convert`** stáhne celé webové stránky a uloží je do souboru EPUB `myrecipe.epub`. Volba `-vv` způsobí, že ebook-convert vyplivne spoustu informací o tom, co dělá. Volba `ebook-convert-recipe-input --test` umožní stáhnout pouze pár článků z nejvýše dvou informačních kanálů. Kromě toho ebook-convert umístí stažené HTML do složky `debug/input`, kde `debug` je složka, kterou jste zadali ve volbě *ebook-convert --debug-pipeline* (stránka 333).

Jakmile je stahování dokončeno, můžete se podívat na stažené *HTML* otevřením souboru `debug/input/index.html` v prohlížeči. Jakmile jste spokojeni se správností stahování a předzpracování, můžete generovat e-knihy v různých formátech, jak je zobrazeno níže:

```
ebook-convert myrecipe.recipe myrecipe.epub
ebook-convert myrecipe.recipe myrecipe.mobi
...
```

Pokud jste se svým předpisem spokojeni a máte pocit, že je dostatečná poptávka pro ospravedlnění jeho zahrnutí do sady vestavěných předpisů, zveřejněte svůj předpis ve fóru předpisů Calibre[11], abyste ho sdíleli s ostatními uživateli Calibre.

Poznámka

Na macOS jsou nástroje příkazového řádku uvnitř sady Calibre, například pokud jste nainstalovali Calibre do `/Applications`, nástroje příkazového řádku jsou v `/Applications/calibre.app/Contents/MacOS/`.

[8] https://www.crummy.com/software/BeautifulSoup/bs4/doc/
[9] https://lxml.de/
[10] https://github.com/html5lib/html5lib-python
[11] https://www.mobileread.com/forums/forumdisplay.php?f=228

Viz také

ebook-convert **(stránka 325)**
Rozhraní příkazového řádku pro všechny převody e-knih.

2.4 Další čtení

Abyste se dozvěděli více o psaní pokročilých předpisů pomocí některého z vybavení dostupného v `BasicNewsRecipe`, měli byste si pročíst následující zdroje:

Dokumentace API **(stránka 41)**
Dokumentace třídy `BasicNewsRecipe` a všechny její důležité metody a pole.

BasicNewsRecipe[12]
Zdrojový kód `BasicNewsRecipe`

Vestavěné předpisy[13]
Zdrojový kód vestavěných předpisů, které jsou dodávány s Calibre

Fórum předpisů Calibre[14]
Zde se nachází spousta znalostí pisatelů předpisů Calibre.

2.5 Dokumentace API

2.5.1 Dokumentace API pro předpisy

API pro psaní předpisů je definováno pomocí *BasicNewsRecipe* (stránka 41)

class calibre.web.feeds.news.**BasicNewsRecipe**(*options*, *log*, *progress_reporter*)

Base class that contains logic needed in all recipes. By overriding progressively more of the functionality in this class, you can make progressively more customized/powerful recipes. For a tutorial introduction to creating recipes, see *Přidávání oblíbených webových stránek se zprávami* (stránka 33).

classmethod adeify_images(*soup*)

If your recipe when converted to EPUB has problems with images when viewed in Adobe Digital Editions, call this method from within *postprocess_html()* (stránka 45).

classmethod image_url_processor(*baseurl*, *url*)

Perform some processing on image urls (perhaps removing size restrictions for dynamically generated images, etc.) and return the processed URL. Return None or an empty string to skip fetching the image.

classmethod print_version(*url*)

Take a *url* pointing to the webpage with article content and return the *URL* pointing to the print version of the article. By default does nothing. For example:

```
def print_version(self, url):
    return url + '?&pagewanted=print'
```

classmethod tag_to_string(*tag*, *use_alt=True*, *normalize_whitespace=True*)

Convenience method to take a BeautifulSoup[15] `Tag` and extract the text from it recursively, including any CDATA sections and alt tag attributes. Return a possibly empty Unicode string.

[12] https://github.com/kovidgoyal/calibre/blob/master/src/calibre/web/feeds/news.py
[13] https://github.com/kovidgoyal/calibre/tree/master/recipes
[14] https://www.mobileread.com/forums/forumdisplay.php?f=228

use_alt: If *True* try to use the alt attribute for tags that don't have any textual content

tag: BeautifulSoup[16] `Tag`

abort_article(*msg=None*)

Call this method inside any of the preprocess methods to abort the download for the current article. Useful to skip articles that contain inappropriate content, such as pure video articles.

abort_recipe_processing(*msg*)

Causes the recipe download system to abort the download of this recipe, displaying a simple feedback message to the user.

add_toc_thumbnail(*article*, *src*)

Call this from populate_article_metadata with the src attribute of an <img> tag from the article that is appropriate for use as the thumbnail representing the article in the Table of Contents. Whether the thumbnail is actually used is device dependent (currently only used by the Kindles). Note that the referenced image must be one that was successfully downloaded, otherwise it will be ignored.

canonicalize_internal_url(*url*, *is_link=True*)

Return a set of canonical representations of `url`. The default implementation uses just the server hostname and path of the URL, ignoring any query parameters, fragments, etc. The canonical representations must be unique across all URLs for this news source. If they are not, then internal links may be resolved incorrectly.

Parametry

is_link – Is True if the URL is coming from an internal link in an HTML file. False if the URL is the URL used to download an article.

cleanup()

Called after all articles have been download. Use it to do any cleanup like logging out of subscription sites, etc.

clone_browser(*br*)

Clone the browser br. Cloned browsers are used for multi-threaded downloads, since mechanize is not thread safe. The default cloning routines should capture most browser customization, but if you do something exotic in your recipe, you should override this method in your recipe and clone manually.

Cloned browser instances use the same, thread-safe CookieJar by default, unless you have customized cookie handling.

default_cover(*cover_file*)

Create a generic cover for recipes that don't have a cover

download()

Download and pre-process all articles from the feeds in this recipe. This method should be called only once on a particular Recipe instance. Calling it more than once will lead to undefined behavior. :return: Path to index.html

extract_readable_article(*html*, *url*)

Extracts main article content from ‚html', cleans up and returns as a (article_html, extracted_title) tuple. Based on the original readability algorithm by Arc90.

get_article_url(*article*)

Override in a subclass to customize extraction of the *URL* that points to the content for each article. Return the article URL. It is called with *article*, an object representing a parsed article from a feed. See feedparser[17]. By default it looks for the original link (for feeds syndicated via a service like FeedBurner or Pheedo) and if found, returns that or else returns article.link[18].

get_browser(**args*, ***kwargs*)

Return a browser instance used to fetch documents from the web. By default it returns a mechanize[19] browser instance that supports cookies, ignores robots.txt, handles refreshes and has a random common user agent.

To customize the browser override this method in your sub-class as:

```
def get_browser(self, *a, **kw):
    br = super().get_browser(*a, **kw)
    # Add some headers
    br.addheaders += [
        ('My-Header', 'one'),
        ('My-Header2', 'two'),
    ]
    # Set some cookies
    br.set_cookie('name', 'value')
    br.set_cookie('name2', 'value2', domain='.mydomain.com')
    # Make a POST request with some data
    br.open('https://someurl.com', {'username': 'def', 'password': 'pwd'}).
↪read()
    # Do a login via a simple web form (only supported with mechanize␣
↪browsers)
    if self.username is not None and self.password is not None:
        br.open('https://www.nytimes.com/auth/login')
        br.select_form(name='login')
        br['USERID']   = self.username
        br['PASSWORD'] = self.password
        br.submit()
    return br
```

get_cover_url()

Return a *URL* to the cover image for this issue or *None*. By default it returns the value of the member *self.cover_url* which is normally *None*. If you want your recipe to download a cover for the e-book override this method in your subclass, or set the member variable *self.cover_url* before this method is called.

get_extra_css()

By default returns *self.extra_css*. Override if you want to programmatically generate the extra_css.

get_feeds()

Return a list of *RSS* feeds to fetch for this profile. Each element of the list must be a 2-element tuple of the form (title, url). If title is None or an empty string, the title from the feed is used. This method is useful if your recipe needs to do some processing to figure out the list of feeds to download. If so, override in your subclass.

get_masthead_title()

Override in subclass to use something other than the recipe title

get_masthead_url()

Return a *URL* to the masthead image for this issue or *None*. By default it returns the value of the member *self.masthead_url* which is normally *None*. If you want your recipe to download a masthead for the e-book override this method in your subclass, or set the member variable *self.masthead_url* before this method is called. Masthead images are used in Kindle MOBI files.

get_obfuscated_article(*url*)

If you set *articles_are_obfuscated* this method is called with every article URL. It should return the path to a file on the filesystem that contains the article HTML. That file is processed by the recursive HTML fetching engine, so it can contain links to pages/images on the web. Alternately, you can return a dictionary of the form:

{‚data': <HTML data>, ‚url': <the resolved URL of the article>}. This avoids needing to create temporary files. The *url* key in the dictionary is useful if the effective URL of the article is different from the URL passed into this method, for example, because of redirects. It can be omitted if the URL is unchanged.

This method is typically useful for sites that try to make it difficult to access article content automatically.

get_url_specific_delay(*url*)

Return the delay in seconds before downloading this URL. If you want to programmatically determine the delay for the specified URL, override this method in your subclass, returning self.delay by default for URLs you do not want to affect.

Vrací

A floating point number, the delay in seconds.

index_to_soup(*url_or_raw*, *raw=False*, *as_tree=False*, *save_raw=None*)

Convenience method that takes an URL to the index page and returns a BeautifulSoup[20] of it.

url_or_raw: Either a URL or the downloaded index page as a string

is_link_wanted(*url*, *tag*)

Return True if the link should be followed or False otherwise. By default, raises NotImplementedError which causes the downloader to ignore it.

Parametry

- **url** – The URL to be followed
- **tag** – The tag from which the URL was derived

parse_feeds()

Create a list of articles from the list of feeds returned by *BasicNewsRecipe.get_feeds()* (stránka 43). Return a list of `Feed` objects.

parse_index()

This method should be implemented in recipes that parse a website instead of feeds to generate a list of articles. Typical uses are for news sources that have a „Print Edition" webpage that lists all the articles in the current print edition. If this function is implemented, it will be used in preference to *BasicNewsRecipe.parse_feeds()* (stránka 44).

It must return a list. Each element of the list must be a 2-element tuple of the form `('feed title', list of articles)`.

Each list of articles must contain dictionaries of the form:

```
{
'title'       : article title,
'url'         : URL of print version,
'date'        : The publication date of the article as a string,
'description' : A summary of the article
'content'     : The full article (can be an empty string). Obsolete
                do not use, instead save the content to a temporary
                file and pass a file:///path/to/temp/file.html as
                the URL.
}
```

For an example, see the recipe for downloading *The Atlantic*. In addition, you can add ‚author' for the author of the article.

If you want to abort processing for some reason and have calibre show the user a simple message instead of an error, call *abort_recipe_processing()* (stránka 42).

populate_article_metadata(*article*, *soup*, *first*)

Called when each HTML page belonging to article is downloaded. Intended to be used to get article metadata like author/summary/etc. from the parsed HTML (soup).

Parametry

- **article** – A object of class `calibre.web.feeds.Article`. If you change the summary, remember to also change the text_summary
- **soup** – Parsed HTML belonging to this article
- **first** – True iff the parsed HTML is the first page of the article.

postprocess_book(*oeb*, *opts*, *log*)

Run any needed post processing on the parsed downloaded e-book.

Parametry

- **oeb** – An OEBBook object
- **opts** – Conversion options

postprocess_html(*soup*, *first_fetch*)

This method is called with the source of each downloaded *HTML* file, after it is parsed for links and images. It can be used to do arbitrarily powerful post-processing on the *HTML*. It should return *soup* after processing it.

Parametry

- **soup** – A BeautifulSoup[21] instance containing the downloaded *HTML*.
- **first_fetch** – True if this is the first page of an article.

preprocess_html(*soup*)

This method is called with the source of each downloaded *HTML* file, before it is parsed for links and images. It is called after the cleanup as specified by remove_tags etc. It can be used to do arbitrarily powerful pre--processing on the *HTML*. It should return *soup* after processing it.

soup: A BeautifulSoup[22] instance containing the downloaded *HTML*.

preprocess_image(*img_data*, *image_url*)

Perform some processing on downloaded image data. This is called on the raw data before any resizing is done. Must return the processed raw data. Return None to skip the image.

preprocess_raw_html(*raw_html*, *url*)

This method is called with the source of each downloaded *HTML* file, before it is parsed into an object tree. raw_html is a unicode string representing the raw HTML downloaded from the web. url is the URL from which the HTML was downloaded.

Note that this method acts *before* preprocess_regexps.

This method must return the processed raw_html as a unicode object.

publication_date()

Use this method to set the date when this issue was published. Defaults to the moment of download. Must return a `datetime.datetime` object.

skip_ad_pages(*soup*)

This method is called with the source of each downloaded *HTML* file, before any of the cleanup attributes like remove_tags, keep_only_tags are applied. Note that preprocess_regexps will have already been applied. It is meant to allow the recipe to skip ad pages. If the soup represents an ad page, return the HTML of the real page. Otherwise return None.

soup: A BeautifulSoup[23] instance containing the downloaded *HTML*.

sort_index_by(*index*, *weights*)

Convenience method to sort the titles in *index* according to *weights*. *index* is sorted in place. Returns *index*.

index: A list of titles.

weights: A dictionary that maps weights to titles. If any titles in index are not in weights, they are assumed to have a weight of 0.

articles_are_obfuscated = False

Set to True and implement *get_obfuscated_article()* (stránka 43) to handle websites that try to make it difficult to scrape content.

auto_cleanup = False

Automatically extract all the text from downloaded article pages. Uses the algorithms from the readability project. Setting this to True, means that you do not have to worry about cleaning up the downloaded HTML manually (though manual cleanup will always be superior).

auto_cleanup_keep = None

Specify elements that the auto cleanup algorithm should never remove. The syntax is a XPath expression. For example:

```
auto_cleanup_keep = '//div[@id="article-image"]' will keep all divs with
                                                 id="article-image"
auto_cleanup_keep = '//*[@class="important"]' will keep all elements
                                              with class="important"
auto_cleanup_keep = '//div[@id="article-image"]|//span[@class="important"]'
                  will keep all divs with id="article-image" and spans
                  with class="important"
```

browser_type = 'mechanize'

The simulated browser engine to use when downloading from servers. The default is to use the Python mechanize browser engine, which supports logging in. However, if you don't need logging in, consider changing this to either ‚webengine' which uses an actual Chromium browser to do the network requests or ‚qt' which uses the Qt Networking backend. Both ‚webengine' and ‚qt' support HTTP/2, which mechanize does not and are thus harder to fingerprint for bot protection services.

center_navbar = True

If True the navigation bar is center aligned, otherwise it is left aligned

compress_news_images = False

Set this to False to ignore all scaling and compression parameters and pass images through unmodified. If True and the other compression parameters are left at their default values, images will be scaled to fit in the screen dimensions set by the output profile and compressed to size at most (w * h)/16 where w x h are the scaled image dimensions.

compress_news_images_auto_size = 16

The factor used when auto compressing JPEG images. If set to None, auto compression is disabled. Otherwise, the images will be reduced in size to (w * h)/compress_news_images_auto_size bytes if possible by reducing the quality level, where w x h are the image dimensions in pixels. The minimum JPEG quality will be 5/100 so it is possible this constraint will not be met. This parameter can be overridden by the parameter compress_news_images_max_size which provides a fixed maximum size for images. Note that if you enable scale_news_images_to_device then the image will first be scaled and then its quality lowered until its size is less than (w * h)/factor where w and h are now the *scaled* image dimensions. In other words, this compression happens after scaling.

compress_news_images_max_size = None

Set JPEG quality so images do not exceed the size given (in KBytes). If set, this parameter overrides auto compression via compress_news_images_auto_size. The minimum JPEG quality will be 5/100 so it is possible this constraint will not be met.

conversion_options = {}

Recipe specific options to control the conversion of the downloaded content into an e-book. These will override any user or plugin specified values, so only use if absolutely necessary. For example:

```
conversion_options = {
  'base_font_size'   : 16,
  'linearize_tables' : True,
}
```

cover_margins = (0, 0, '#ffffff')

By default, the cover image returned by get_cover_url() will be used as the cover for the periodical. Overriding this in your recipe instructs calibre to render the downloaded cover into a frame whose width and height are expressed as a percentage of the downloaded cover. cover_margins = (10, 15, ‘#ffffff‘) pads the cover with a white margin 10px on the left and right, 15px on the top and bottom. Color names are defined here[24]. Note that for some reason, white does not always work in Windows. Use #ffffff instead

delay = 0

The default delay between consecutive downloads in seconds. The argument may be a floating point number to indicate a more precise time. See *get_url_specific_delay()* (stránka 44) to implement per URL delays.

description = ''

A couple of lines that describe the content this recipe downloads. This will be used primarily in a GUI that presents a list of recipes.

encoding = None

Specify an override encoding for sites that have an incorrect charset specification. The most common being specifying `latin1` and using `cp1252`. If None, try to detect the encoding. If it is a callable, the callable is called with two arguments: The recipe object and the source to be decoded. It must return the decoded source.

extra_css = None

Specify any extra *CSS* that should be added to downloaded *HTML* files. It will be inserted into *<style>* tags, just before the closing *</head>* tag thereby overriding all *CSS* except that which is declared using the style attribute on individual *HTML* tags. Note that if you want to programmatically generate the extra_css override the *get_extra_css()* (stránka 43) method instead. For example:

```
extra_css = '.heading { font: serif x-large }'
```

feeds = None

List of feeds to download. Can be either `[url1, url2, ...]` or `[('title1', url1), ('title2', url2),...]`

filter_regexps = []

List of regular expressions that determines which links to ignore. If empty it is ignored. Used only if is_link_wanted is not implemented. For example:

```
filter_regexps = [r'ads\.doubleclick\.net']
```

will remove all URLs that have *ads.doubleclick.net* in them.

Only one of *BasicNewsRecipe.match_regexps* (stránka 48) or *BasicNewsRecipe.filter_regexps* (stránka 47) should be defined.

handle_gzip = True

Set to False if you do not want to use gzipped transfers with the mechanize browser. Note that some old servers flake out with gzip.

ignore_duplicate_articles = None

Ignore duplicates of articles that are present in more than one section. A duplicate article is an article that has the same title and/or URL. To ignore articles with the same title, set this to:

```
ignore_duplicate_articles = {'title'}
```

To use URLs instead, set it to:

```
ignore_duplicate_articles = {'url'}
```

To match on title or URL, set it to:

```
ignore_duplicate_articles = {'title', 'url'}
```

keep_only_tags = []

Keep only the specified tags and their children. For the format for specifying a tag see *BasicNewsRecipe.remove_tags* (stránka 50). If this list is not empty, then the *<body>* tag will be emptied and re-filled with the tags that match the entries in this list. For example:

```
keep_only_tags = [dict(id=['content', 'heading'])]
```

will keep only tags that have an *id* attribute of *„content“* or *„heading“*.

language = 'und'

The language that the news is in. Must be an ISO-639 code either two or three characters long

masthead_url = None

By default, calibre will use a default image for the masthead (Kindle only). Override this in your recipe to provide a URL to use as a masthead.

match_regexps = []

List of regular expressions that determines which links to follow. If empty, it is ignored. Used only if is_link_wanted is not implemented. For example:

```
match_regexps = [r'page=[0-9]+']
```

will match all URLs that have *page=some number* in them.

Only one of *BasicNewsRecipe.match_regexps* (stránka 48) or *BasicNewsRecipe.filter_regexps* (stránka 47) should be defined.

max_articles_per_feed = 100

Maximum number of articles to download from each feed. This is primarily useful for feeds that don't have article dates. For most feeds, you should use *BasicNewsRecipe.oldest_article* (stránka 49)

needs_subscription = False

If True the GUI will ask the user for a username and password to use while downloading. If set to „optional“ the use of a username and password becomes optional

no_stylesheets = False

Convenient flag to disable loading of stylesheets for websites that have overly complex stylesheets unsuitable for conversion to e-book formats. If True stylesheets are not downloaded and processed

oldest_article = 7.0

Oldest article to download from this news source. In days.

preprocess_regexps = []

List of *regexp* substitution rules to run on the downloaded *HTML*. Each element of the list should be a two element tuple. The first element of the tuple should be a compiled regular expression and the second a callable that takes a single match object and returns a string to replace the match. For example:

```
preprocess_regexps = [
   (re.compile(r'<!--Article ends here-->.*</body>', re.DOTALL|re.IGNORECASE),
    lambda match: '</body>'),
]
```

will remove everything from *<!–Article ends here–>* to *</body>*.

publication_type = 'unknown'

Publication type Set to newspaper, magazine or blog. If set to None, no publication type metadata will be written to the opf file.

recipe_disabled = None

Set to a non empty string to disable this recipe. The string will be used as the disabled message

recipe_specific_options = None

Specify options specific to this recipe. These will be available for the user to customize in the Advanced tab of the Fetch News dialog or at the ebook-convert command line. The options are specified as a dictionary mapping option name to metadata about the option. For example:

```
recipe_specific_options = {
    'edition_date': {
        'short': 'The issue date to download',
        'long':  'Specify a date in the format YYYY-mm-dd to download the␣
↪issue corresponding to that date',
        'default': 'current',
    }
}
```

When the recipe is run, self.recipe_specific_options will be a dict mapping option name to the option value specified by the user. When the option is unspecified by the user, it will have the value specified by ‚default'. If no default is specified, the option will not be in the dict at all, when unspecified by the user.

recursions = 0

Number of levels of links to follow on article webpages

remove_attributes = []

List of attributes to remove from all tags. For example:

```
remove_attributes = ['style', 'font']
```

remove_empty_feeds = False

If True empty feeds are removed from the output. This option has no effect if parse_index is overridden in the sub class. It is meant only for recipes that return a list of feeds using *feeds* or *get_feeds()* (stránka 43). It is also used if you use the ignore_duplicate_articles option.

remove_javascript = True

Convenient flag to strip all JavaScript tags from the downloaded HTML

remove_tags = []

List of tags to be removed. Specified tags are removed from downloaded HTML. A tag is specified as a dictionary of the form:

```
{
 name      : 'tag name',   #e.g. 'div'
 attrs     : a dictionary, #e.g. {'class': 'advertisement'}
}
```

All keys are optional. For a full explanation of the search criteria, see Beautiful Soup[25] A common example:

```
remove_tags = [dict(name='div', class_='advert')]
```

This will remove all * * tags and all their children from the downloaded *HTML*.

remove_tags_after = None

Remove all tags that occur after the specified tag. For the format for specifying a tag see *BasicNewsRecipe.remove_tags* (stránka 50). For example:

```
remove_tags_after = [dict(id='content')]
```

will remove all tags after the first element with *id="content"*.

remove_tags_before = None

Remove all tags that occur before the specified tag. For the format for specifying a tag see *BasicNewsRecipe.remove_tags* (stránka 50). For example:

```
remove_tags_before = dict(id='content')
```

will remove all tags before the first element with *id="content"*.

requires_version = (0, 6, 0)

Minimum calibre version needed to use this recipe

resolve_internal_links = False

If set to True then links in downloaded articles that point to other downloaded articles are changed to point to the downloaded copy of the article rather than its original web URL. If you set this to True, you might also need to implement *canonicalize_internal_url()* (stránka 42) to work with the URL scheme of your particular website.

reverse_article_order = False

Reverse the order of articles in each feed

scale_news_images = None

Maximum dimensions (w,h) to scale images to. If scale_news_images_to_device is True this is set to the device screen dimensions set by the output profile unless there is no profile set, in which case it is left at whatever value it has been assigned (default None).

scale_news_images_to_device = True

Rescale images to fit in the device screen dimensions set by the output profile. Ignored if no output profile is set.

simultaneous_downloads = 5

Number of simultaneous downloads. Set to 1 if the server is picky. Automatically reduced to 1 if *BasicNewsRecipe.delay* (stránka 47) > 0

summary_length = 500

Max number of characters in the short description

```
template_css = '\n .article_date {\n color: gray; font-family: monospace;\n }\n\n
.article_description {\n text-indent: 0pt;\n }\n\n a.article {\n font-weight:
bold; text-align:left;\n }\n\n a.feed {\n font-weight: bold;\n }\n\n
.calibre_navbar {\n font-family:monospace;\n }\n '
```

The CSS that is used to style the templates, i.e., the navigation bars and the Tables of Contents. Rather than overriding this variable, you should use *extra_css* in your recipe to customize look and feel.

timefmt = ' [%a, %d %b %Y]'

The format string for the date shown on the first page. By default: Day_Name, Day_Number Month_Name Year

timeout = 120.0

Timeout for fetching files from server in seconds

title = 'Neznámý zdroj zpráv'

The title to use for the e-book

use_embedded_content = None

Normally we try to guess if a feed has full articles embedded in it based on the length of the embedded content. If *None*, then the default guessing is used. If *True* then the we always assume the feeds has embedded content and if *False* we always assume the feed does not have embedded content.

[15] https://www.crummy.com/software/BeautifulSoup/bs4/doc/
[16] https://www.crummy.com/software/BeautifulSoup/bs4/doc/
[17] https://pythonhosted.org/feedparser/
[18] https://pythonhosted.org/feedparser/reference-entry-link.html
[19] https://mechanize.readthedocs.io/en/latest/
[20] https://www.crummy.com/software/BeautifulSoup/bs4/doc
[21] https://www.crummy.com/software/BeautifulSoup/bs4/doc/
[22] https://www.crummy.com/software/BeautifulSoup/bs4/doc/
[23] https://www.crummy.com/software/BeautifulSoup/bs4/doc/
[24] https://www.imagemagick.org/script/color.php
[25] https://www.crummy.com/software/BeautifulSoup/bs4/doc/#searching-the-tree

KAPITOLA 3

Prohlížeč e-knih

Calibre obsahuje vestavěný Prohlížeč e-knih, který umí zobrazit všechny hlavní formáty e-knih. Prohlížeč e-knih je vysoce přizpůsobitelný a má mnoho pokročilých funkcí.

- *Spuštění Prohlížeče e-knih* (stránka 54)
- *Navigace v e-knize* (stránka 54)
- *Text zvýraznění* (stránka 55)
- *Číst nahlas* (stránka 55)
- *Vyhledávání textu* (stránka 55)
- *Následování odkazů pouze pomocí klávesnice* (stránka 56)
- *Přizpůsobení vzhledu a chování vašeho zážitku ze čtení* (stránka 56)
- *Vyhledávání ve slovníku* (stránka 56)
- *Kopírování textu a obrázků* (stránka 57)
- *Přibližování obrázků* (stránka 57)
- *Syncing with a paper edition of the current book* (stránka 57)
- *Klávesové zkratky* (stránka 57)
- *Neplovoucí obsah* (stránka 59)
- *Designing your book to work well with the calibre E-book viewer* (stránka 59)

3.1 Spuštění Prohlížeče e-knih

Jakoukoliv z knih ve své knihovně Calibre můžete zobrazit výběrem knihy a stisknutím tlačítka *Zobrazit*. To otevře knihu v Prohlížeči e-knih. Můžete také spustit samotný Prohlížeč e-knih z nabídky Start ve Windows. Na macOS ho můžete připnout do doku a spouštět ho odtud. Na Linuxu můžete použít jeho spouštěč v menu nebo spuštěním příkazu **ebook-viewer**.

3.2 Navigace v e-knize

„Otáčet knihy“ v knize můžete budˇ:

- Kliknutím myší do levého nebo pravého okraje stránky
- Stisknutím `spacebar`, `page up`, `page down` nebo kurzorových kláves
- Na dotykové obrazovce klepnutím na text nebo přejetím prstem vlevo a vpravo

Přistupovat k ovládacím prvkům prohlížeče můžete budˇ:

- Kliknutím pravým tlačítkem myši na text
- Stisknutím kláves `Esc` nebo `Nabídka`
- Na dotykové obrazovce klepnutím na horní třetinu obrazovky

Prohlížeč má dva režimy, „stránkovaný“ a „plovoucí“. V režimu stránek je obsah knihy prezentován jako stránky, podobně jako v papírové knize. V plovoucím režimu je text zobrazován nepřetržitě, jako ve webovém prohlížeči. Můžete mezi nimi přepínat pomocí *Předvoleb* prohlížeče pod *Rozložení stránky* nebo stisknutím klávesové zkratky `Ctrl+M`.

3.2.1 Záložky

Když jste uprostřed knihy a zavřete Prohlížeč e-knih, zapamatuje si, kde jste přestali číst, a vrátí se tam, když příště otevřete knihu. Můžete také nastavit záložky v knize pomocí tlačítka *Záložky* v ovládacích prvcích Prohlížeče e-knih nebo stisknutím `Ctrl+B`. Při prohlížení knih ve formátu EPUB jsou tyto záložky skutečně ukládány do samotného souboru EPUB. Můžete přidat záložky, a pak odeslat soubor přátelům. Když otevřou soubor, uvidí vaše záložky. Toto chování můžete vypnout v oddílu *Různé* v předvolbách prohlížeče.

3.2.2 Obsah

Pokud kniha, kterou čtete, má definovaný obsah, můžete k němu přistoupit stisknutím tlačítka *Obsah*. Tím se zobrazí seznam oddílů v knize. Můžete kliknout na kterýkoliv z nich pro přejití na tuto část knihy.

3.2.3 Navigace podle místa

E-knihy, na rozdíl od papírových knih, nemají žádnou koncepci stránek. Na přesná umístění v e-knihách můžete odkazovat pomocí funkce *Přejít na → Umístění* v ovládacích prvcích prohlížeče.

Tuto informaci o umístění můžete použít pro jednoznačné odkazy na části knih, když o nich diskutujete s přáteli, nebo na ně odkazujete v jiných pracích. Tato umístění můžete zadat pod *Přejít na → Umístění* v ovládacích prvcích prohlížeče.

Existuje adresa URL, kterou můžete zkopírovat do schránky a vložit do jiných programů nebo dokumentů. Kliknutí na tuto adresu URL otevře knihu v Prohlížeči e-knih Calibre v aktuálním umístění.

Pokud kliknete na odkazy uvnitř e-knihy, aby vás přenesly na různé části knihy, jako jsou vysvětlivky, můžete použít tlačítka *Zpět* a *Vpřed* v levém horním rohu v ovládacích prvcích prohlížeče. Tato tlačítka se chovají stejně jako ta v internetovém prohlížeči.

3.2.4 Režim odkazu

Calibre má také velice šikovný *Režim odkazů*. Můžete ho zapnout kliknutím na tlačítko *Režim odkazů* v ovládacích prvcích prohlížeče. Jakmile to uděláte, tak při spuštění bude mít každý odstavec zobrazeno jedinečné číslo složené z čísel oddílu a odstavce.

Toto číslo můžete použít pro jednoznačné odkazy na části knih, když o nich diskutujete s přáteli, nebo na ně odkazujete v jiných pracích. Tato čísla můžete zadat do *Přejít na funkci* pro pro přejití na konkrétní umístění odkazu.

3.3 Text zvýraznění

Když vyberete text v prohlížeči, vedle výběru se objeví malý automaticky otevíraný panel. Klikněte na tlačítko zvýraznění na tomto panelu pro vytvoření zvýraznění. Můžete přidávat poznámky a měnit barvu zvýraznění. Na dotykové obrazovce dlouze klepněte na nějaké slovo pro jeho výběr a zobrazení automaticky otevírané lišty. Jakmile jste v režimu zvýraznění, můžete změnit, jaký text je vybrán, pomocí úchytů pro snadný výběr na dotykové obrazovce. Přetažením úchytů k horním nebo dolním okrajům můžete posouvat při výběru. Můžete také `Shift` a `kliknout` nebo `kliknout pravým tlačítkem` pro rozšíření výběru, což je obzvlášť užitečné pro vícestránkové výběry.

Pro zobrazení samostatného panelu se seznamem všech zvýraznění v knize seřazených podle kapitoly můžete použít tlačítko *Zvýraznění* v ovládacích prvcích prohlížeče.

Můžete procházet *všechna zvýraznění* v celé knihovně Calibre kliknutím pravým tlačítkem myši na tlačítko *Zobrazit* a výběrem *Procházet anotace*.

A konečně, pokud používáte Server s obsahem Calibre v prohlížeči prohlížeče, můžete si nechat prohlížeč synchronizovat jeho anotace s prohlížečem prohlížeče tak, že přejdete do *Předvolby → Různé* v předvolbách prohlížeče a zadáte uživatelské jméno Serveru s obsahem, se kterým má prohlížeč synchronizovat. Pro synchronizaci s anonymními uživateli použijte speciální hodnotu `*`.

3.4 Číst nahlas

The viewer can read book text aloud. To use it you can simply click the *Read aloud* button in the viewer controls to start reading book text aloud. The word or sentence being currently read is highlighted. Speech is synthesized from the text using either the Piper[26] neural text-to-speech engine or your operating system services for text-to-speech. You can change the backend and the voice being used by clicking the gear icon in the bar that is displayed while *Read aloud* is active.

Zvýrazněné pasáže můžete také číst nahlas přidáním tlačítka *Číst nahlas* do panelu výběru v předvolbách prohlížeče pod *Chování výběru*.

Poznámka

Support for text-to-speech in browsers is very incomplete and bug-ridden so how well *Read aloud* will work in the in-browser viewer is dependent on how well the underlying browser supports text-to-speech.

3.5 Vyhledávání textu

Prohlížeč má velmi výkonné možnosti vyhledávání. Stiskněte klávesu `Ctrl+F` nebo vstupte do ovládacích prvků prohlížeče a klikněte na vyhledávání. Nejjednodušší formou vyhledávání je prostě vyhledat jakýkoliv text, který zadáte do textového pole. Různé formy vyhledávání se volí pomocí pole režimu vyhledávání pod vstupem pro vyhledávání. Dostupné jsou tyto režimy:

[26] https://github.com/rhasspy/piper

1. *Obsahuje* - Nejjednodušší výchozí režim. Text zadaný do vyhledávacího pole se hledá kdekoliv. Veškerá interpunkce, diakritika a mezery jsou ignorovány. Například vyhledávání: `Pena` bude odpovídat všemu následujícímu: `penal, pen a, pen.a a Peña`. Pokud zaškrtnete políčko *Rozlišovat velikost písmen*, pak už diakritika, mezery a interpunkce nebudou ignorovány.
2. *Celá slova* - Vyhledává celá slova. Takže například vyhledávání `pena` bude odpovídat slovu `Peña`, ale ne slovu `Penal`. Stejně jako u výše uvedeného vyhledávání *Obsahuje* je diakritika a interpunkce ignorována, pokud není zaškrtnuto políčko *Rozlišovat velikost písmen*.
3. *Slova poblíž* - Vyhledává celá slova, která jsou blízko sebe. Takže například vyhledání `calibre skvělé` bude hledat místa, kde se slova `calibre` a `skvělé` vyskytují do šedesáti znaků od sebe. Pokud chcete změnit počet znaků, přidejte nové číslo na konec seznamu slov. Například `calibre skvělé úžasné 120` bude odpovídat místům, kde se tato tři slova vyskytují do 120 znaků od sebe. Pamatujte, že interpunkce a diakritika *není* při tomto vyhledávání ignorována.
4. *Regulární výraz* - Interpretuje hledaný text jako *regulární výraz*. Více informací o používání regulárních výrazů naleznete v *návodu* (stránka 214).

3.6 Následování odkazů pouze pomocí klávesnice

Prohlížeč e-knih má *Režim rad*, který umožňuje klikat na odkazy v textu bez použití myši. Stiskněte klávesy `Alt+F` a všechny odkazy na aktuální obrazovce budou zvýrazněny číslem nebo písmenem. Stisknutím písmene na klávesnici kliknete na odkaz. Stisknutím klávesy `Esc` přerušíte *Režim rad*, aniž byste vybrali jakýkoliv odkaz.

Pokud je na obrazovce více než třicet pět odkazů, některé z nich budou mít více písmen, v takovém případě zadejte první a druhé nebo první a aktivujte je stisknutím `Enter`. Klávesu `Backspace` můžete také použít k vrácení chyby při psaní.

3.7 Přizpůsobení vzhledu a chování vašeho zážitku ze čtení

Velikost písma můžete změnit za běhu pomocí *Velikost písma* v ovládacích prvcích prohlížeče nebo pomocí `Ctrl++` a `Ctrl+-` nebo držením klávesy `Ctrl` a použitím kolečka myši.

Barvy lze změnit v oddílu *Barvy* v předvolbách prohlížeče.

Počet stránek zobrazených na obrazovce stejně jako okraje stránky můžete změnit v *Rozložení stránky* v předvolbách prohlížeče.

Vlastní záhlaví a zápatí, jako je zbývající čas čtení, název aktuální kapitoly, pozice v knize atd., můžete zobrazit prostřednictvím oddílu *Záhlaví a zápatí* předvoleb prohlížeče.

Pokročilejší přizpůsobení může být dosaženo nastavením *Stylů*. Zde můžete zadat obrázek pozadí, který bude zobrazen pod textem, a také šablonu stylů, kterou můžete nastavit, aby se použila na každou knihu. Pomocí ní můžete dělat věci, jako jsou změna stylů odstavců, zarovnání textu do bloku atd. Pro příklady vlastních šablon stylů používaných uživateli Calibre navštivte fórum[27].

3.8 Vyhledávání ve slovníku

Význam slov v aktuální knize můžete vyhledat poklepáním nebo dlouhým klepnutím na slovo, které chcete vyhledat, a následným kliknutím na tlačítko vyhledávání, které vypadá jako knihovna.

[27] https://www.mobileread.com/forums/showthread.php?t=51500

3.9 Kopírování textu a obrázků

Text a obrázky můžete vybrat přetažením obsahu myší a potom kliknutím pravým tlačítkem myši a výběrem *Kopírovat* pro zkopírování do schránky. Zkopírovaný materiál může být vložen do jiné aplikace jako prostý text a obrázky.

3.10 Přibližování obrázků

Přiblížit obrázek pro zobrazení v plné velikosti v samostatném okně můžete buď poklepáním nebo dlouhým klepnutím na něj. Můžete na něj také kliknout pravým tlačítkem myši a zvolit *Zobrazit obrázek*.

3.11 Syncing with a paper edition of the current book

Some e-books, that have corresponding print editions, include metadata that marks the start of each paper page. For such e-books, the viewer allows you to jump to a particular paper edition page via the *Go to* button in the viewer controls. You can also optionally display the paper page corresponding to the current location in the book's headers or footers via the viewer settings, by adding *Pages from paper edition* to either the header or the footer.

3.12 Klávesové zkratky

The viewer has extensive keyboard shortcuts, like the rest of calibre. They can be customised in the viewer *Preferences*. The default shortcuts are listed below:

Tabulka 1: Keyboard shortcuts for the calibre E-book viewer

Klávesová zkratka	Akce
`Home`, `Ctrl+Ar` `Ctrl+Ar`	Scroll to the start of the current file in a multi file book
`Ctrl+Hc`	Posunout na začátek knihy
`Ctrl+En`	Posunout na konec knihy
`End`, `Ctrl+Ar` `Ctrl+Ar`	Scroll to the end of the current file in a multi file book
`ArrowUp`	Posouvat zpět, plynule v plovoucím režimu a po celých obrazovkách v režimu stránek
`Arrow-Down`	Posouvat vpřed, plynule v plovoucím režimu a po celých obrazovkách v režimu stránek
`ArrowLe`	Posouvat vlevo o trochu v plovoucím režimu a po celých obrazovkách v režimu stránek
`ArrowRi`	Posouvat vpravo o trochu v plovoucím režimu a po celých obrazovkách v režimu stránek
`PageUp`, `Shift+S`	Posouvat zpět po celých obrazovkách
`PageDow` `Spa-cebar`	Posouvat vpřed po celých obrazovkách
`Ctrl+Pa`	Posunout na předchozí oddíl

continues on next page

Tabulka 1 – pokračujte na předchozí stránce

Kláve-sová zkrat-ka	Akce
Ctrl+Pa	Posunout na další oddíl
Alt+Arr	Zpět
Alt+Arr	Vpřed
Ctrl+T	Toggle Table of Contents
Ctrl+S	Číst nahlas
Alt+P	Change settings quickly by creating and switching to *profiles*
Alt+f	Následovat odkazy pomocí klávesnice
Ctrl+C	Kopírovat do schránky
Alt+C	Kopírovat aktuální umístění do schránky
Ctrl+Sh	Kopírovat aktuální umístění do schránky jako URL calibre://
/, Ctrl+f, Cmd+f	Spustit hledání
F3, Enter	Najít další
Shift+F Shift+E	Najít předchozí
Ctrl+Pl Me-ta+Plus	Zvětšit velikost písma
Ctrl+Mi Me-ta+Minu	Zmenšit velikost písma
Ctrl+0	Obnovit výchozí velikost písma
Ctrl+]	Zvýšit počet stránek na obrazovku
Ctrl+[	Snížit počet stránek na obrazovku
Ctrl+Al	Nastavit počet stránek na obrazovku automaticky
F11, Ctrl+Sh	Přepnout na celou obrazovku
Ctrl+M	Přepnout mezi Režimem stránek a Plovoucím režimem pro rozložení textu
Ctrl+W	Přepnout posuvník
Ctrl+X	Přepnout Režim odkazu
Ctrl+B	Zobrazit nebo skrýt záložky
Ctrl+Al	Nová záložka
Ctrl+N, Ctrl+E	Zobrazit metadata knihy
Ctrl+Al Ctrl+Al	Znovu načíst knihu
Ctrl+Sh	Upravit aktuální výběr vpřed o slovo
Ctrl+Sh	Upravit aktuální výběr zpět o slovo
Shift+A	Upravit aktuální výběr vpřed o znak
Shift+A	Upravit aktuální výběr zpět o znak
Shift+A	Upravit aktuální výběr vpřed o řádek
Shift+H	Rozšířit aktuální výběr na začátek řádku

continues on next page

Tabulka 1 – pokračujte na předchozí stránce

Kláve-sová zkrat-ka	Akce
Shift+E	Rozšířit aktuální výběr na konec řádku
Ctrl+A	Vybrat vše
Shift+A	Upravit aktuální výběr zpět o řádek
Ctrl+Sh	Upravit aktuální výběr vpřed o odstavec
Ctrl+Sh	Upravit aktuální výběr zpět o odstavec
Esc, Menu-Key	Zobrazit ovládací prvky Prohlížeče e-knih
Ctrl+Cc	Zobrazit předvolby Prohlížeče e-knih
Ctrl+Es Me-ta+Esc, Me-ta+Comm	
Ctrl+G, ;, :	Přejít na zadané umístění nebo pozici v knize
Ctrl+Sp	Přepnout automatické posouvání
Alt+Arr	Automaticky posouvat rychleji
Alt+Arr	Automaticky posouvat pomaleji
Ctrl+I	Zobrazit nebo skrýt kontrolu
Ctrl+L	Zobrazit nebo skrýt panel vyhledání slova
Ctrl+Q (Cmd+Q on ma-cOS)	Ukončit
Ctrl+P	Tisknout knihu do PDF
Ctrl+F1	Přepnout panel nástrojů
Ctrl+H	Přepnout panel zvýraznění
Ctrl+D	Upravit tuto knihu

3.13 Neplovoucí obsah

Některé knihy mají velmi široký obsah, který nemůže být zalomen na ohraničeních stránky. Například tabulky nebo značky `<pre>`. V takových případech byste měli přepnout prohlížeč do *plovoucího režimu* stisknutím `Ctrl+M` pro čtení tohoto obsahu. Alternativně můžete také přidat následující CSS do oddílu *Styly* v předvolbách prohlížeče, abyste prohlížeč donutili rozdělit řádky textu ve značkách `<pre>`:

```
code, pre { white-space: pre-wrap }
```

3.14 Designing your book to work well with the calibre E-book viewer

The calibre E-book viewer will set the `is-calibre-viewer` class on the root element. So you can write CSS rules that apply only for it. Additionally, the viewer will set the following classes on the `body` element:

`body.calibre-viewer-dark-colors`
: Nastavit při použití tmavého barevného schématu

body.calibre-viewer-light-colors
: Nastavit při použití světlého barevného schématu

body.calibre-viewer-paginated
: Nastavit při použití režimu stránek

body.calibre-viewer-scrolling
: Nastavit při použití plovoucího režimu

body.calibre-footnote-container
: Set when displaying a popup footnote

Finally, you can use the calibre color scheme colors via CSS variables[28]. The calibre E-book viewer defines the following variables: `--calibre-viewer-background-color`, `--calibre-viewer-foreground-color` and optionally `--calibre-viewer-link-color` in color themes that define a link color.

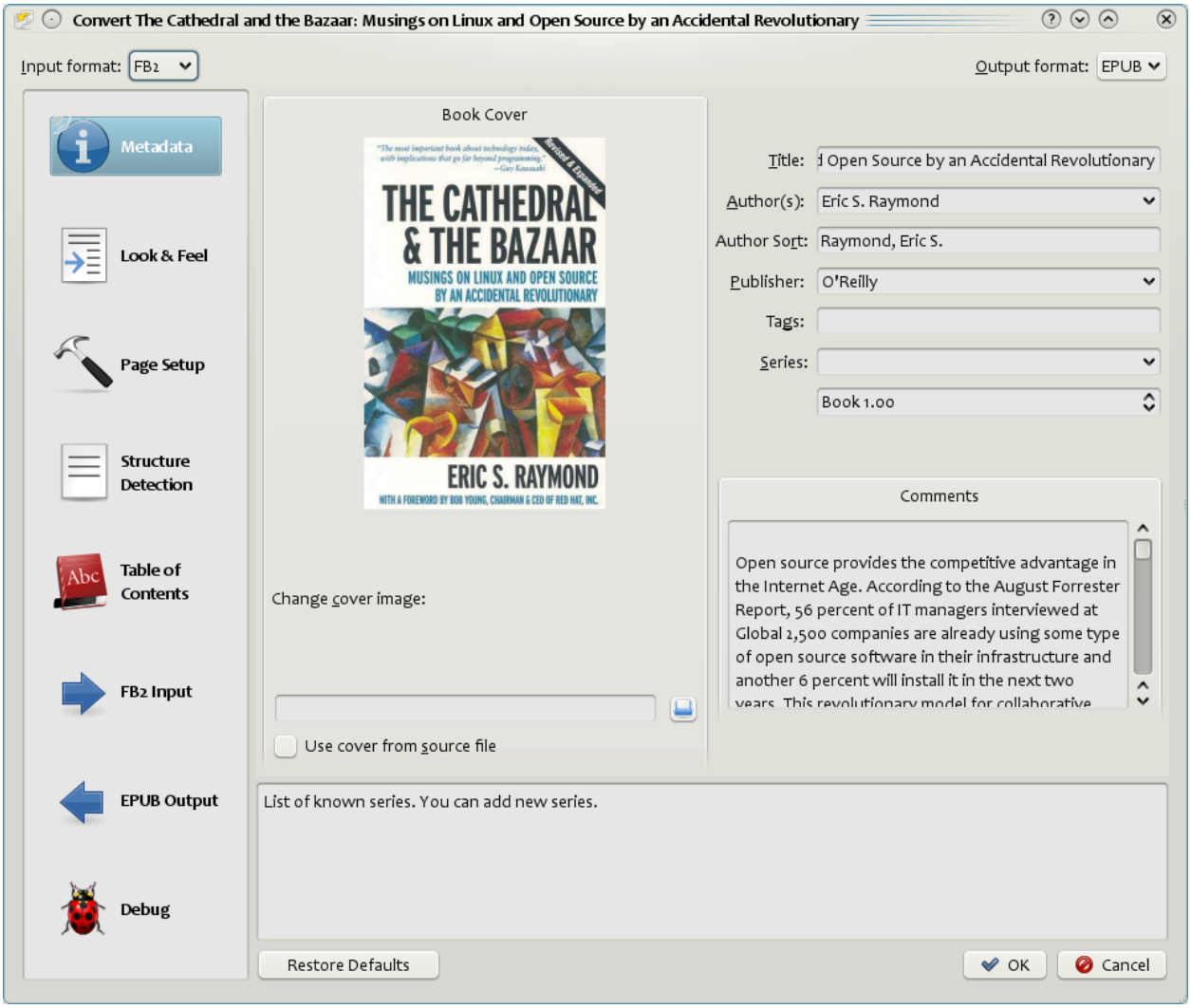
[28] https://developer.mozilla.org/en-US/docs/Web/CSS/Using_CSS_custom_properties

KAPITOLA 4

Převod e-knihy

Calibre má systém převodu, který je navržen tak, aby se velmi snadno používal. Obyčejně stačí přidat knihu do Calibre, kliknout na převod a Calibre se pokusí vygenerovat výstup, který je co nejpodobnější vstupu. Calibre však přijímá velké množství vstupních formátů, z nichž ne všechny jsou stejně vhodné jako jiné pro převod na e-knihy. V případě takových vstupních formátů, nebo pokud pouze chcete větší kontrolu nad systémem převodu, Calibre má mnoho voleb pro jemné ladění procesu převodu. Pamatujte však, že systému převodu Calibre není náhradou za plnohodnotný editor e-knih. Pro úpravu e-knih je doporučuji nejdříve pomocí Calibre převést do formátu EPUB nebo AZW3, a pak použít funkci *Upravit knihu*, abyste je dostali do dokonalé podoby. Pak můžete použít upravenou e-knihu jako vstup pro převod do jiných formátů v Calibre.

Tento dokument bude odkazovat hlavně na nastavení převodu, které naleznete v dialogovém okně převodu, jak je zobrazeno níže. Všechna tato nastavení jsou také dostupná prostřednictvím rozhraní příkazového řádku pro převod zdokumentovaného v *ebook-convert* (stránka 325). V Calibre můžete získat nápovědu k jakémukoliv individuálnímu nastavení podržením myši nad ním, objeví se popisek tlačítka popisující nastavení.



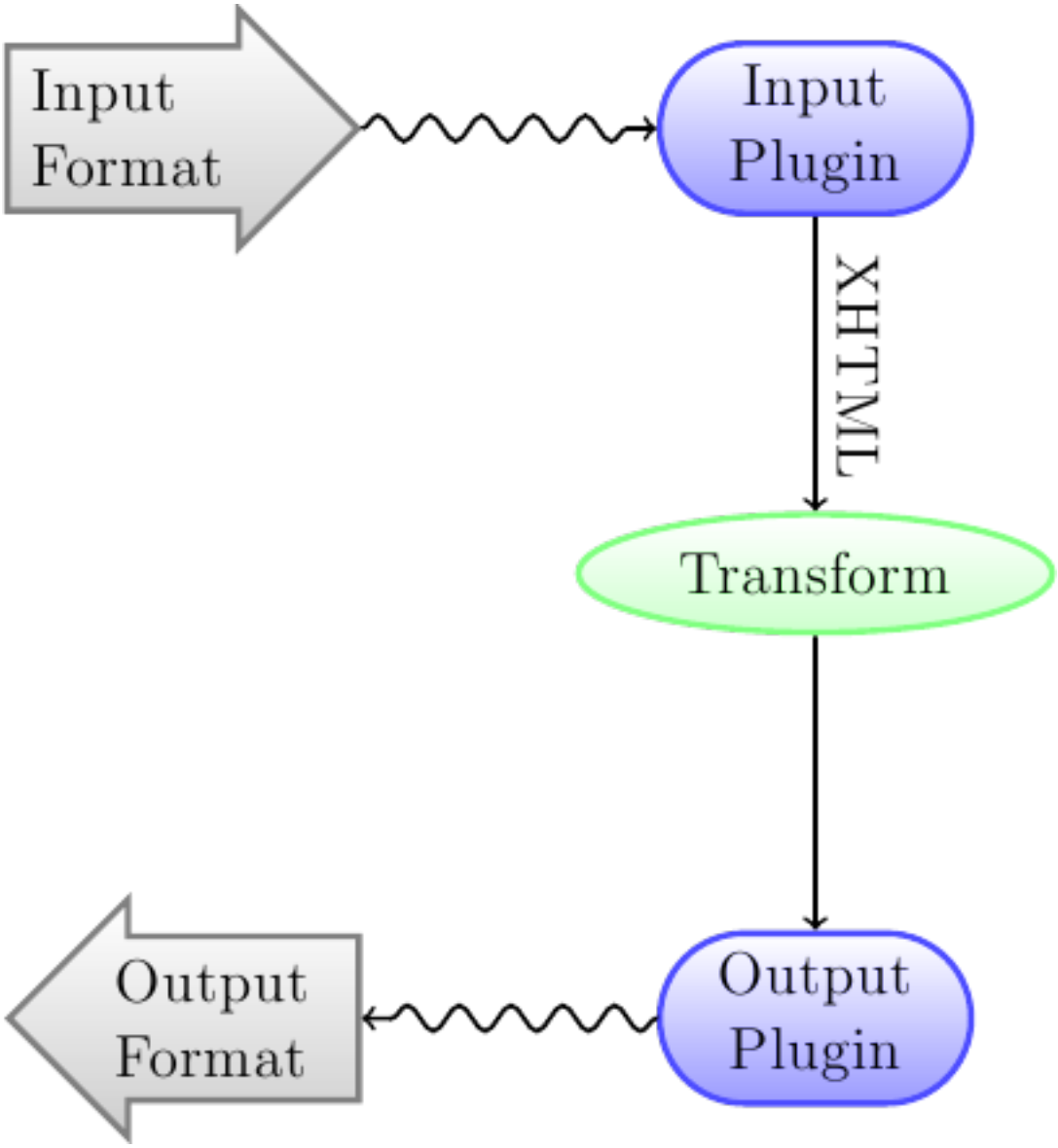
Obsah

- *Úvod* (stránka 63)
- *Vzhled a chování* (stránka 64)
- *Nastavení stránky* (stránka 67)
- *Heuristické zpracování* (stránka 67)
- *Hledat a nahradit* (stránka 69)
- *Rozpoznávání struktury* (stránka 69)
- *Obsah* (stránka 70)
- *Požití obrázku jako názvu kapitol pří konverzi HTML vstupních dokumentů* (stránka 72)
- *Používání atributů značek pro dodání textu pro položky v obsahu* (stránka 72)
- *Jak jsou nastaveny nebo uloženy volby pro převod* (stránka 72)

- *Tipy pro konkrétní formáty* (stránka 73)

4.1 Úvod

První věc, kterou musíte pochopit o systému převodu, je to, že je navržen jako řetězec. Schematicky to vypadá takto:



Vstupní formát se nejdříve převede na XHTML vhodným *vstupním modulem*. Toto HTML se pak *transformuje*. V posledním kroku se zpracované XHTML převede na zadaný výstupní formát vhodným *výstupním modulem*. Výsledky převodu se mohou značně lišit v závislosti na vstupním formátu. Některé formáty se převádějí mnohem lépe než ostatní. Seznam nejlepších zdrojových formátů pro převod je dostupný *zde* (stránka 130).

Transformace, které se provádějí na výstupu XHTML, je místo, kde se odehrává veškerá práce. Existují různé transformace, například pro vložení metadat knihy jako stránky na začátek knihy, pro rozpoznání nadpisů kapitol a automatického vytvoření obsahu, pro proporcionální změnu velikosti písma a tak dále. Je důležité si zapamatovat, že všechny transformace se provádějí na výstupním XHTML pomocí *vstupního modulu*, ne na samotném vstupním souboru. Takže

například když požádáte Calibre, aby převedlo soubor RTF na EPUB, tak bude nejdříve vnitřně převeden na XHTML, na toto XHTML budou použity různé transformace, a pak *výstupní modul* vytvoří soubor EPUB, automaticky vygeneruje všechna metadata, obsah a tak dále.

Tento proces můžete vidět v akci použitím volby ladění . Stačí zadat cestu ke složce pro výstup ladění.

Během převodu Calibre umístí XHTML generované různými fázemi převodního řetězce do různých podsložek. Čtyři podsložky jsou:

Tabulka 1: Fáze převodního řetězce

Složka	Popis
vstup	Obsahuje výstup HTML ze vstupního modulu. Použijte pro ladění vstupního modulu.
analyzováno	Výsledek předzpracování a převodu na XHTML výstupu ze vstupního modulu. Použijte pro ladění rozpoznávání struktury.
struktura	Po rozpoznání struktury, ale před vyhlazením CSS a převodem velikosti písma. Použijte pro ladění převodu velikosti písma a transformace CSS.
zpracováno	Těsně před tím, než je e-kniha předána výstupnímu modulu. Použijte pro ladění výstupního modulu.

Pokud chcete trochu upravit vstupní dokument před tím, než ho necháte Calibre převést, to nejlepší, co můžete udělat, je upravit soubory v podsložce `vstup`, pak je zazipovat a použít soubor ZIP jako vstupní formát pro následující převody. Abyste to udělali, použijte dialogové okno *Upravit metadata* pro přidání souboru ZIP jako formátu pro knihu, a pak v levém horním rohu dialogového okna převodu vyberte ZIP jako vstupní formát.

Tento dokument se bude zabývat hlavně různými transformacemi, které pracují na polotovaru XHTML, a jak je ovládat. Na konci je několik konkrétních tipů pro každý vstupní nebo výstupní formát.

4.2 Vzhled a chování

Obsah

- *Písma* (stránka 65)
- *Text* (stránka 65)
- *Rozložení* (stránka 66)
- *Stylování* (stránka 66)
- *Převod stylů* (stránka 67)
- *Převod HTML* (stránka 67)

Tato skupina voleb určuje různé aspekty vzhledu a chování převedené e-knihy.

4.2.1 Písma

Jedna z nejhezčích funkcí při čtení e-knih je schopnost snadno upravit velikost písma, aby vyhovovalo jednotlivým potřebám a světelným podmínkám. Calibre má propracované algoritmy, aby zajistilo, že všechny produkované knihy mají jednotné velikosti písma, bez ohledu na to, jaké velikosti písma jsou zadány ve vstupním dokumentu.

Základní velikost písma dokumentu je nejběžnější velikost písma v tomto dokumentu, t. j. velikost větší části textu v tomto dokumentu. Když zadáte *Základní velikost písma*, Calibre automaticky přiměřeně změní měřítko všech velikostí písma v dokumentu, takže nejběžnější velikost písma se stane zadanou základní velikostí písma a ostatním velikostem písma bude přiměřeně upraveno měřítko. Výběrem větší základní velikosti písma můžete písma v dokumentu zvětšit a naopak. Když nastavíte základní velikost písma, měli byste pro dosažení nejlepších výsledků také nastavit klíč velikosti písma.

Obyčejně Calibre automaticky zvolí základní velikost písma vhodnou pro výstupní profil, který jste zvolili (viz *Nastavení stránky* (stránka 67)). Toto však zde můžete přepsat, pokud pro vás výchozí nastavení není vhodné.

Volba *Klíč velikosti písma* vám umožňuje určit, jak se mění nezákladní velikosti písma. Algoritmus změny velikosti písma funguje pomocí klíče velikosti písma, což je jednoduše čárkami oddělený seznam velikostí písma. Klíč velikost písma říká Calibre, o kolik „kroků“ větší nebo menší by měla daná velikost písma být v porovnání s základní velikostí písma. Předpokládá se, že by v dokumentu měl být omezený počet velikostí písma. Například jedna velikost pro základní text, několik velikostí pro různé úrovně nadpisů a několik velikostí pro horní nebo dolní indexy a poznámky pod čarou. Klíč velikosti písma umožňuje Calibre rozčlenit velikosti písma ve vstupních dokumentech do samostatných „košů“ odpovídajících různým logickým velikostem písem.

Vysvětleme si to na příkladu. Předpokládejme, že zdrojový dokument, který převádíme, byl vytvořen někým s vynikajícím zrakem a má základní velikost písma 8pt. To znamená, že větší část textu v dokumentu má velikost 8pt, zatímco nadpisy jsou o něco větší (řekněme 10 a 12pt) a poznámky pod čarou o něco menší 6pt. Takže pokud použijeme následující nastavení:

```
Base font size : 12pt
Font size key  : 7, 8, 10, 12, 14, 16, 18, 20
```

výstupní dokument bude mít základní velikost písma 12pt, nadpisy 14 a 16pt a poznámky pod čarou 8pt. Nyní předpokládejme, že chceme, aby největší velikost nadpisu více vyčnívala a také aby poznámky pod čarou byly trochu větší. Abychom toho dosáhli, klíč písma by měl být změněn na:

```
New font size key : 7, 9, 12, 14, 18, 20, 22
```

Největší nadpisy budou mít nyní 18pt, zatímco poznámky pod čarou budou mít 9pt. Můžete si hrát s tímto nastavením, abyste se pokusili zjistit, co by pro vás bylo nejlepší pomocí průvodce změny měřítka písma, který můžete otevřít kliknutím na malé tlačítko vedle nastavení *Klíč velikosti písma*.

Může tu být také zakázána veškerá změna velikosti písma v převodu, pokud chcete zachovat velikost písma ve vstupním dokumentu.

Související nastavení je *Výška řádku*. Výška řádku určuje svislou výšku řádků. Ve výchozím nastavení (výška řádku 0) se neprovádí žádná manipulace s výškami řádků. Pokud zadáte nevýchozí hodnotu, budou výšky řádků nastaveny na všech místech, které nemají určenu vlastní výšky řádků. Toto je však taková tupá zbraň a mělo by to být používáno šetrně. Pokud chcete upravit výšky řádků pro nějaký oddíl vstupu, je lepší použít Dodatečné CSS.

V tomto oddíle můžete také říct Calibre, aby do knihy vložilo jakákoliv odkazovaná písma. To umožní, aby písma na čtecích zařízeních fungovala, i když nejsou v zařízení dostupná.

4.2.2 Text

Text může být buď zarovnaný, nebo ne. Zarovnaný text má dodatečné mezery mezi slovy, aby měl hladký pravý okraj. Někteří lidé upřednostňují zarovnaný text, jiní ne. Calibre obyčejně zachová zarovnání v původním dokumentu. Pokud ho chcete přepsat, můžete v tomto oddíle použít volbu *Zarovnání textu*.

Můžete také říct Calibre, aby *Vylepšilo interpunkci*, což nahradí prosté uvozovky, pomlčky a výpustky jejich typograficky správnými alternativami. Pamatujte, že tento algoritmus není dokonalý, takže stojí za to přezkoumat výsledky. Naopak je také dostupná *Nevylepšená interpunkce*.

Nakonec je zde *Vstupní kódování znaků*. Starší dokumenty někdy nezadávají své kódování znaků. Při převodu to může vést k neanglické znakům nebo speciálním znakům, jako je poškození chytrých uvozovek. Calibre zkouší automaticky rozpoznat kódování znaků zdrojového dokumentu, ale není vždy úspěšné. Pomocí tohoto nastavení můžete vynutit předpokládání konkrétního kódování znaků. *cp1252* je běžné kódování pro dokumenty vytvořené pomocí softwaru Windows. Měli byste si také přečíst *Jak převedu můj soubor obsahující neanglické znaky nebo chytré uvozovky?* (stránka 130) pro více informací o problematice kódování.

4.2.3 Rozložení

Obyčejně jsou odstavce v XHTML vykresleny s prázdným řádkem mezi nimi a bez úvodního odsazení textu. Calibre má pár voleb, jak toto určit. *Odebrat mezery mezi odstavci* vynutí zajištění, aby všechny odstavce neměly žádné vnitřní mezery mezi odstavci. Nastaví také odsazení textu na 1,5em (může být změněno) pro označení začátku každého odstavce. *Vložit prázdný řádek* dělá opak, což zaručuje, že je právě jeden prázdný řádek mezi každou dvojicí odstavců. Obě tyto volby jsou velice ucelené, odebírají mezery nebo je vkládají pro *všechny* odstavce (technicky vzato značky <p> a). Je to proto, abyste mohli jen nastavit volbu a být si jistí, že se to provede tak, jak bylo ohlášeno bez ohledu na to, jak je neuspořádaný vstupní soubor. Jedinou výjimkou je případ, kdy vstupní soubor používá pevné konce řádků pro doplnění mezer mezi odstavce.

Pokud chcete odebrat mezery mezi všemi odstavci kromě pár vybraných, nepoužívejte tyto volby. Místo toho přidejte následující kód CSS do Dodatečné CSS:

```
p, div { margin: 0pt; border: 0pt; text-indent: 1.5em }
.spacious { margin-bottom: 1em; text-indent: 0pt; }
```

Pak ve zdrojovém dokumentu označte odstavce, které potřebují mezery *class="spacious"*. Pokud váš vstupní dokument není v HTML, použijte volbu ladění popsanou v Úvodu, abyste získali HTML (použijte podsložku `input`).

Další užitečnou volbou je *Převést tabulky na řádky*. Některé chybně navržené dokumenty používají tabulky pro ovládání rozložení textu na stránce. Při převodu mají tyto dokumenty často text, který vybíhá mimo stránku a jiné artefakty. Tato volba extrahuje obsah z tabulek a zobrazí ho v lineárním tvaru. Pamatujte, že tato volba převede na řádky *všechny* tabulky, takže ji používejte, pouze pokud jste si jisti, že vstupní dokument nepoužívá tabulky pro oprávněné účely, jako je zobrazení tabulkových informací.

4.2.4 Stylování

Volba *Dodatečné CSS* vám umožňuje zadat libovolné CSS, které bude použito na všechny soubory HTML na vstupu. Toto CSS je použito s velice vysokou prioritou, a proto by mělo přepsat většinu CSS přítomného v samotném **vstupním dokumentu**. Toto nastavení můžete použít pro jemné ladění prezentace nebo rozložení dokumentu. Například pokud chcete, aby všechny odstavce třídy *endnote* byly zarovnány vpravo, stačí přidat:

```
.endnote { text-align: right }
```

nebo pokud chcete změnit odsazení všech odstavců:

```
p { text-indent: 5mm; }
```

Dodatečné CSS je velice výkonná volba, ale potřebujete pochopit, jak CSS funguje, pro použití jejího plného potenciálu. Můžete použít volbu ladění řetězce popsanou výše, abyste viděli, jaké CSS je přítomné ve vstupním dokumentu.

Jednodušší volbou je použít *Filtrovat informace o stylu*. To vám umožňuje odebrat všechny vlastnosti CSS zadaných typů z dokumentu. Můžete to například použít k odebrání všech barev nebo písem.

4.2.5 Převod stylů

Toto je nejvýkonnější vybavení související se stylováním. Můžete ho použít k definování pravidel, které mění styly na základě různých podmínek. Můžete ho například použít ke změně všech zelených barev na modré nebo k odebrání všech tučných stylů z textu nebo obarvení všech záhlaví určitou barvou apod.

4.2.6 Převod HTML

Podobné převodu stylů, ale umožňuje provádět změny v obsahu knihy ve formátu HTML. Můžete nahradit jednu značku jinou, přidat značkám třídy nebo jiné atributy na základě jejich obsahu atd.

4.3 Nastavení stránky

Volby *Nastavení stránky* jsou pro ovládání rozložení obrazovky, jako jsou okraje a velikosti obrazovky. Existují volby pro nastavení okrajů stránky, které budou použity Výstupním modulem, pokud vybraný Výstupní formát podporuje okraje stránky. Kromě toho byste měli vybrat Vstupní profil a Výstupní profil. Obě sady profilů se v podstatě zabývají tím, jak interpretovat rozměry ve vstupních a výstupních dokumentech, velikosti obrazovky a klíče úpravy měřítka výchozího písma.

Pokud víte, že soubor, který převádíte, byl zamýšlen k použití na konkrétním zařízení nebo softwarové platformě, zvolte odpovídající vstupní profil, jinak prostě zvolte výchozí vstupní profil. Pokud víte, že soubory, které vytváříte, jsou určeny pro konkrétní typ zařízení, zvolte odpovídající výstupní profil. Pokud víte, že soubory, které vytváříte, jsou určeny pro konkrétní typ zařízení, zvolte odpovídající výstupní profil. V opačném případě zvolte jeden z obecných výstupních profilů. Pokud převádíte na MOBI nebo AZW3, budete téměř vždy chtít zvolit jeden z výstupních profilů Kindle. V opačném případě je nejlepším řešením pro moderní zařízení pro čtení e-knih zvolit výstupní profil *Obecné HD zařízení e-ink*.

Výstupní profil také určuje velikost obrazovky. To například způsobí, že v některých výstupních formátech bude u obrázků automaticky změněna velikost, aby se přizpůsobily obrazovce. Takže zvolte profil zařízení, které má podobnou velikost obrazovky vašemu zařízení.

4.4 Heuristické zpracování

Heuristické zpracování poskytuje celou řadu funkcí, které mohou být použity pro pokus o rozpoznání a opravu běžných problémů ve špatně formátovaných vstupních dokumentech. Použijte tyto funkce, pokud váš vstupní dokument trpí špatným formátování Protože tyto funkce spoléhají na běžné vzory, uvědomte si, že v některých případech může volba vést k horším výsledkům, tak to používejte opatrně. Například několik z těchto voleb odebere všechny entity pevných mezer, nebo mohou zahrnovat falešně pozitivní shody souvisejících s funkcí.

Povolit heuristické zpracování
: Tato volba aktivuje fázi *Heuristického zpracování* převodního řetězce Calibre. Musí to být povoleno, aby byly použity různé dílčí funkce.

Zrušit zalomení řádků
: Povolení této volby způsobí, že se Calibre pokusí rozpoznat a opravit tvrdé pevné konce řádků existujících v dokumentu pomocí interpunkčních vodítek a délky řádku. Calibre se nejdříve pokusí rozpoznat, zda pevné konce řádků existují, pokud se zdá, že neexistují, Calibre se nepokusí zrušit zalomení řádků. Koeficient zrušení zalomení řádku může být zmenšen, pokud chcete ‚vynutit', aby Calibre zrušilo zalomení řádků.

Koeficient zrušení zalomení řádku
: Tato volba určuje algoritmus, který Calibre používá k odebrání pevných konců řádků. Například pokud je hodnota této volby 0,4, znamená to, že Calibre odebere pevné konce řádků z konců řádků, jejichž délky jsou menší než délka 40 % všech řádků v dokumentu. Pokud má váš dokument pouze několik konců řádků, které potřebují opravu, pak by tato hodnota měla být snížena někam mezi 0,1 a 0,2.

Rozpoznat a označit neformátované nadpisy a podnadpisy kapitol
: Pokud váš dokument neobsahuje nadpisy kapitol a názvy formátované odlišně od zbytku textu, Calibre může použít

tuto volbu, aby se je pokusilo rozpoznat a obklopit je značkami nadpisů. Značky <h2> se používají pro nadpisy kapitol; značky <h3> se používají pro jakékoliv rozpoznané názvy.

Tato funkce nevytvoří obsah, ale v mnoha případech způsobí, že výchozí nastavení rozpoznání kapitol Calibre správně rozpozná kapitoly a sestaví obsah. Upravte XPath v Rozpoznávání struktury, pokud není obsah automaticky vytvořen. Pokud v dokumentu nejsou použity žádné další nadpisy, pak nastavení „//h:h2“ v Rozpoznávání struktury bude nejsnadnější způsob, jak vytvořit obsah pro dokument.

Tyto vložené nadpisy nejsou naformátovány, pro použití formátování použijte volbu Dodatečné CSS pod Vzhledem a chováním v nastavení převodu. Například pro značky zarovnání nadpisů na střed použijte následující:

```
h2, h3 { text-align: center }
```

Přečíslovat sekvence značek <h1> nebo <h2>
Někteří vydavatelé formátují nadpisy kapitol pomocí více značek <h1> nebo <h2> postupně. Výchozí nastavení převodu Calibre způsobí, že takové nadpisy budou rozděleny na dvě části. Tato volba přečísluje značky nadpisů, aby se zabránilo rozdělení.

Odstranit prázdné řádky mezi odstavci
Tato volba způsobí, že Calibre analyzuje prázdné řádky obsažené v dokumentu. Pokud je každý odstavec proložený prázdným řádkem, pak Calibre odebere všechny tyto prázdné odstavce. Sekvence více prázdných řádků budou považovány za přerušení příběhu a budou zachovány jako jeden odstavec. Tato volba se liší od volby *Odebrat mezery mezi odstavci* pod *Vzhled a chování* v tom, že skutečně mění obsah HTML, zatímco druhá volba mění styly dokumentu. Tato volba může také odebrat odstavce, které byly vloženy pomocí volby Calibre *Vložit prázdný řádek*.

Zajistit konzistentní formátování přerušení příběhu
Pomocí této volby se Calibre pokusí rozpoznat běžné značky přerušení příběhu a zajistit, aby byly zarovnané na střed. Značkám ‚měkkého‘ přerušení příběhu, tj. přerušení příběhu definované pouze dodatečnými prázdnými znaky, jsou změněny styly, aby se zajistilo, že nebudou zobrazeny ve spojení s konci stránek.

Nahradit přerušení příběhu
Pokud je tato volba konfigurována, pak Calibre nahradí značky přerušení příběhu, které najde, nahrazovacím textem zadaným uživatelem. Pamatujte, prosím, že některé ozdobné znaky nemusí být podporovány na všech čtecích zařízení.

Obecně byste se měli vyhnout používání značek HTML, Calibre zahodí jakékoliv značky a použije předem definované značení. Značky <hr />, tj. vodorovné čáry, a značky <img> jsou výjimky. Vodorovné čáry mohou být volitelně zadány styly, pokud se rozhodnete přidat vlastní styl, nezapomeňte zahrnout nastavení šířky ‚width‘, jinak bude informace stylu zahozena. Můžete použít značky obrázků, ale Calibre neposkytuje možnost přidat obrázek během převodu, to musí být provedeno po faktickém použití funkce ‚Upravit knihu‘.

Příklad značky obrázku (po převodu umístí obrázek do složky ‚Images‘ uvnitř souboru EPUB):

Příklad vodorovné čáry se styly:
<hr style=“width:20%;padding-top: 1px;border-top: 2px ridge black;border-bottom: 2px groove black;“/>

Odebrat nadbytečné spojovníky
Calibre analyzuje veškerý obsah rozdělený spojovníky v dokumentu, když je tato volba povolena. Samotný dokument je použit jako slovník pro analýzu. To umožňuje Calibre přesně odebrat spojovníky pro jakákoliv slova v dokumentu v jakémkoliv jazyce, společně s vymyšlenými a neznámými vědeckými slovy. Primární nevýhodou je, že slova objevující se v dokumentu pouze jednou nebudou změněna. Analýza probíhá ve dvou průchodech, první průchod analyzuje konce řádků. Řádkům je zrušeno zalomení, pouze pokud v dokumentu slovo existuje s nebo bez spojovníku. Druhý průchod analyzuje všechna slova rozdělená spojovníky v celém dokumentu, spojovníky jsou odebrány, pokud slovo existuje někde jinde v dokumentu bez shody.

Převést na kurzívu obecná slova a vzory
Pokud je povoleno, bude Calibre hledat obecná slova a vzory, které označují kurzívu a převede je na kurzívu.

Příkladem jsou obecné textové zásady, jako je ~slovo~, nebo fráze, které by obecně měly být kurzívou, např. latinské fráze, jako jsou ‚etc.' nebo ‚et cetera'.

Nahradit odsazení entitami pomocí odsazeními CSS
Některé dokumenty používají zásadu definování odsazení textu pomocí entit pevných mezer. Když je tato volba povolena, Calibre se pokusí rozpoznat tento druh formátování a převést je na 3% odsazení textu pomocí CSS.

4.5 Hledat a nahradit

Tyto volby jsou užitečné především pro převody dokumentů PDF nebo převody OCR, i když mohou být také použity pro opravu mnoha konkrétních problémů dokumentů. Například některé převody mohou zanechat záhlaví a zápatí stránek v textu. Tyto volby používají regulární výrazy, aby se pokusily rozpoznat záhlaví, zápatí nebo jiný libovolný text a odebrat ho nebo nahradit. Nezapomeňte, že pracují s polotovarem XHTML vytvořeným převodním řetězcem. K dispozici je průvodce, aby vám pomohl přizpůsobit regulární výrazy pro váš dokument. Klikněte na kouzelnou hůlku vedle pole výrazu a klikněte na tlačítko ‚Tesovat' po sestavení vyhledávacího výrazu. Úspěšné shody budou zvýrazněny žlutě.

Vyhledávání funguje pomocí regulárního výrazu Pythonu. Všechny odpovídající texty jsou jednoduše odebrány z dokumentu nebo nahrazeny pomocí vzoru nahrazení. Vzor nahrazení je nepovinný, pokud zůstane prázdný, pak text odpovídající hledanému vzoru bude z dokumentu odstraněn. Další informace o regulárních výrazech a jejich syntaxi můžete zjistit v *Vše o používání regulárních výrazů v Calibre* (stránka 214).

4.6 Rozpoznávání struktury

Rozpoznávání struktury zahrnuje to, že se Calibre pokouší co nejlépe rozpoznat strukturální prvky ve vstupním dokumentu, pokud nejsou správně zadány. Například kapitoly, konce stránek, záhlaví, zápatí atd. Jak si dokážete představit, tento proces se velice liší knihu od knihy. Naštěstí má Calibre velice mocné možnosti, aby toto určilo. S mocí přichází složitost, ale pokud si jednou uděláte čas na to naučit se to složité, zjistíte, že to stálo za námahu.

4.6.1 Konce kapitol a stránek

Calibre má dvě sady možností pro *rozpoznávání kapitol* a *vkládání konců stránek*. To může být někdy trochu matoucí, protože ve výchozím nastavení Calibre vloží konce stránek před rozpoznané kapitoly, stejně jako před místa rozpoznaná možností konců stránek. Důvodem pro to je to, že často existují místa, kam by měly být vloženy konce stránek, která nejsou hranicemi kapitoly. Rozpoznané kapitoly mohou být také volitelně vloženy do automaticky generovaného obsahu.

Calibre používá *XPath*, výkonný jazyk, který umožňuje uživateli zadat hranice kapitol nebo konce stránek. XPath se může zdát nejdříve trochu skličující pro používání, naštěstí je zde *kurz XPath* (stránka 158) v uživatelské příručce. Nezapomeňte, že Rozpoznávání struktury pracuje na polotovaru XHTML vytvořeného převodním řetězcem. Použijte možnost ladění popsanou v *Úvod* (stránka 63), abyste určili vhodné nastavení pro svou knihu. Je zde také tlačítko pro Průvodce XPath, aby pomohl s generováním jednoduchých výrazů XPath.

Ve výchozím nastavení používá Calibre následující výraz pro rozpoznávání kapitol:

```
//*[(name()='h1' or name()='h2') and re:test(., 'chapter|book|section|part\s+', 'i
↪')) or @class = 'chapter']
```

Tento výraz je poměrně složitý, protože se snaží zpracovat velký počet běžných případů souběžně. To znamená, že Calibre bude předpokládat, že kapitoly začínají buď značkou *<h1>* nebo *<h2>*, které obsahují kterékoliv z těchto slov *(kapitola, kniha, oddíl nebo část)*, nebo které mají atribut *class="kapitola"*.

Související možností je *Označení kapitol*, který vám umožní určit, co Calibre udělá, když rozpozná kapitolu. Ve výchozím nastavení vloží konec stránky před kapitolu. Můžete ho nechat vložit linku namísto nebo navíc ke konci stránky. Můžete ho také nechat nedělat nic.

Výchozí nastavení pro rozpoznávání konců stránek je:

```
//*[name()='h1' or name()='h2']
```

což znamená, že Calibre ve výchozím nastavení vloží konce stránek před každou značku *<h1>* a *<h2>*.

Poznámka

Výchozí výrazy se mohou měnit v závislosti na vstupním formátu, který převádíte.

4.6.2 Různé

V tomto oddílu je několik dalších voleb.

Vložit metadata jako stránku na začátek knihy
: Jednou ze skvělých věcí na Calibre je to, že vám umožňuje udržovat velice kompletní metadata o všech vašich knihách, například hodnocení, štítky, komentáře atd. Tato možnost vytvoří jednoduchou stránku se všemi těmito metadaty a vloží ji do převedené e-knihy, obvykle hned za obálku. Berte to jako způsob, jak si vytvořit svůj vlastní přizpůsobený přebal knihy.

Odebrat první obrázek
: Někdy zdrojový dokument, který převádíte, obsahuje obálku jako součást knihy namísto samostatné obálky. Pokud jste také zadali obálku v Calibre, pak bude mít převedená kniha dvě obálky. Tato možnost jednoduše odebere první obrázek ze zdrojového dokumentu, čímž zajistí, že převedená kniha má pouze jednu obálku, tu zadanou v Calibre.

4.7 Obsah

Když má vstupní dokument ve svých metadatech obsah, Calibre ho prostě použije. Nicméně velké množství starších formátů buď nepodporuje obsah založený na metadatech nebo ho jednotlivé dokumenty nemají. V těchto případech vám volby v tomto oddílu mohou pomoct automaticky vygenerovat obsah v převáděné e-knize založený na skutečném obsahu ve vstupním dokumentu.

Poznámka

Používání těchto voleb může být trochu složitější, abyste získali přesně to pravé. Pokud upřednostňujete vytváření nebo úpravu obsahu ručně, převeďte na formáty EPUB nebo AZW3 a vyberte zaškrtávací políčko v dolní oddílu Obsah v dialogovém okně převodu, která říká: *Ručně jemně ladit obsah po dokončení převodu*. Tímto se po převodu spustí nástroj Upravit obsah. Umožní vám vytvářet položky v obsahu jednoduše kliknutím na místo v knize, kam chcete, aby položka ukazovala. Můžete také použít samotný nástroj Upravit obsah, aniž byste prováděli převod. Přejděte na *Předvolby → Rozhraní → Panely nástrojů* a přidejte *Upravit obsah* na hlavní panel nástrojů. Pak stačí vybrat knihu, kterou chcete upravit, a kliknout na tlačítko *Upravit obsah*.

První volba je *Vynutit použití automaticky generovaného obsahu*. Zaškrtnutím této volby můžete nechat Calibre přepsat jakýkoliv nalezený obsah v metadatech vstupního dokumentu automaticky vygenerovaným.

Výchozí způsob, jakým vytvoření automaticky vygenerovaného obsahu funguje, je ten, že Calibre nejdříve zkusí přidat všechny rozpoznané kapitoly do vygenerovaného obsahu. Jak přizpůsobit rozpoznávání kapitol můžete zjistit ve výše uvedeném oddílu *Rozpoznávání struktury* (stránka 69). Pokud nechcete zahrnout rozpoznané kapitoly do vygenerovaného obsahu, zaškrtněte volbu *Nepřidávat rozpoznané kapitoly do obsahu*.

Pokud bylo rozpoznáno méně kapitol, než je hodnota *Mezní hodnota kapitol*, Calibre pak přidá do obsahu všechny hypertextové odkazy, které najde ve vstupním dokumentu. Toto často funguje dobře na mnoha vstupních dokumentech, které obsahují hypertextové odkazy obsahu hned na začátku. Volba *Počet odkazů* může být použita k ovládání tohoto chování. Pokud je nastavena na nulu, nejsou přidány žádné odkazy. Pokud je nastavena na číslo větší než nula, je přidán nanejvýše tento počet odkazů.

Calibre automaticky vyfiltruje duplicity z generovaného obsahu. Pokud však jsou zde nějaké další nežádoucí položky, můžete je vyfiltrovat pomocí volby *Filtr obsahu*. To je regulární výraz, který porovná názvy položek ve vygenerovaném obsahu. Kdykoliv je nalezena shoda, bude odebrána. Například pro odebrání všech názvů položek „Další“ nebo „Předchozí“ použijte:

```
Next|Previous
```

Volby *Obsah 1., 2. a 3. úrovně* vám umožní vytvořit propracovaný víceúrovňový obsah. Jsou to výrazy XPath, které porovnají značky v polotovaru XHTML vytvořeného převodním řetězcem. Podívejte se na *Úvod* (stránka 63), jak získat přístup k tomuto XHTML. Přečtěte si také *Kurz XPath* (stránka 158), abyste zjistili, jak vytvářet výrazy XPath. Vedle každé volby je tlačítko, které spustí průvodce, který vám pomůže s vytvořením základních výrazů XPath. Následující jednoduchý příklad vysvětluje, jak používat tyto volby.

Předpokládejte, že máte vstupní dokument, jehož výstupem je XHTML, které vypadá takto:

```
<html xmlns="http://www.w3.org/1999/xhtml">
    <head>
        <title>Sample document</title>
    </head>
    <body>
        <h1>Chapter 1</h1>
        ...
        <h2>Section 1.1</h2>
        ...
        <h2>Section 1.2</h2>
        ...
        <h1>Chapter 2</h1>
        ...
        <h2>Section 2.1</h2>
        ...
    </body>
</html>
```

Potom, nastavíme možnosti jako:

```
Level 1 TOC : //h:h1
Level 2 TOC : //h:h2
```

Toto bude mít za výsledek automaticky generovaný dvouúrovňový obsah, který vypadá jako:

```
Chapter 1
    Section 1.1
    Section 1.2
Chapter 2
    Section 2.1
```

Varování

Ne všechny formáty podporují víceúrovňové obsahy. Nejdříve byste to měli zkusit s výstupem EPUB. Pokud to funguje, zkuste formát vlastního výběru.

4.8 Požití obrázku jako názvu kapitol pří konverzi HTML vstupních dokumentů

Předpokládejme, že chcete použít obrázek jako název kapitoly, ale stále chcete, aby Calibre bylo pro vás schopno automaticky generovat obsah z názvů kapitol. Použijte následující značení HTML, abyste toho dosáhli:

```
<html>
    <body>
        <h2>Chapter 1</h2>
        <p>chapter 1 text...</p>
        <h2 title="Chapter 2"><img src="chapter2.jpg" /></h2>
        <p>chapter 2 text...</p>
    </body>
</html>
```

Nastavte *Obsah 1. úrovně* na `//h:h2`. Pak pro druhou kapitolu Calibre vezme název z hodnoty atributu `title` značky `<h2>`, protože značka nemá žádný text.

4.9 Používání atributů značek pro dodání textu pro položky v obsahu

Pokud máte obzvlášť dlouhé názvy kapitol a chcete mít v obsahu zkrácené verze, můžete použít atribut názvu, abyste toho dosáhli, například:

```
<html>
    <body>
        <h2 title="Chapter 1">Chapter 1: Some very long title</h2>
        <p>chapter 1 text...</p>
        <h2 title="Chapter 2">Chapter 2: Some other very long title</h2>
        <p>chapter 2 text...</p>
    </body>
</html>
```

Nastavte *Obsah 1. úrovně* na `//h:h2/@title`. Pak Calibre vezme názvy z hodnot atributů `title` značek `<h2>`, namísto použití textu uvnitř značky. Pamatujte na koncové `/@title` ve výrazu XPath, tento formulář můžete použít, abyste řekli Calibre, aby získalo text z jakéhokoliv atributu chcete.

4.10 Jak jsou nastaveny nebo uloženy volby pro převod

V Calibre jsou dvě místa, kde mohou být nastaveny volby převodu. První je v Nastavení->Převod. Tato nastavení jsou výchozí pro volby převodu. Kdykoliv se pokusíte převést novou knihu, zde zadaná nastavení budou použita jako výchozí.

Můžete také změnit nastavení v dialogovém okně převodu pro převod každé knihy. Když převádíte nějakou knihu, Calibre si pamatuje nastavení použité pro tuto knihu, takže pokud ji převedete znovu, bude mít přednost uložené nastavení pro jednotlivou knihu před výchozími hodnotami nastavenými v předvolbách. Jednotlivé nastavení můžete obnovit na výchozí hodnoty pomocí tlačítka *Obnovit výchozí nastavení* v dialogovém okně převodu jednotlivé knihy. Uložené nastavení pro skupinu knih můžete odebrat výběrem všech knih a klepnutím na tlačítko *Upravit metadata*, které vyvolá dialogové okno hromadné úpravy metadat, v blízkosti spodní části dialogového okna je volba odebrat uložené nastavení převodu.

Když hromadně převádíte sadu knih, nastavení jsou brána v následujícím pořadí (poslední vyhrává):

- Z výchozího nastavení v Předvolby->Převod
- Z uloženého nastavení převodu pro každou převáděnou knihu (pokud existuje). Toto lze vypnout pomocí volby v levém horním rohu dialogového okna Hromadný převod.

- Z nastavení v dialogovém okně Hromadný převod

Pamatujte, že konečné nastavení pro každou knihu v Hromadném převodu bude uloženo a znovu použito, pokud je kniha znovu převáděna. Vzhledem k tomu, že nejvyšší priorita v Hromadném převodu je dána nastavením v dialogovém okně Hromadný převod, budou mít přednost před všemi konkrétními nastaveními knihy. Takže byste měli společně hromadně převádět pouze knihy, které potřebují podobná nastavení. Výjimkou jsou metadata a konkrétní nastavení formátu vstupu. Vzhledem k tomu, že dialogové okno Hromadný převod nemá nastavení pro tyto dvě kategorie, budou převzaty z konkrétního nastavení knihy (pokud existuje), nebo z výchozího nastavení.

Poznámka

Aktuální nastavení použité během jakéhokoliv převodu můžete zobrazit kliknutím na otáčející se ikonu v pravém dolním rohu a pak poklepáním na jednotlivou úlohu převodu. To zobrazí protokol převodu, který bude obsahovat aktuálně použitá nastavení na začátku.

4.11 Tipy pro konkrétní formáty

Zde najdete konkrétní tipy pro převod určitých formátů. Konkrétní volby pro určitý formát, zda je vstup nebo výstup dostupný v dialogovém okně převodu pod vlastním oddílem, například *Vstup TXT* nebo *Výstup EPUB*.

4.11.1 Převod dokumentů Microsoft Word

Calibre umí automaticky převést soubory `.docx` vytvořené v Microsoft Word 2007 a novějším. Stačí přidat soubor do Calibre a kliknout na tlačítko Převést.

Poznámka

Zde je ukázkový soubor .docx[29], který ukazuje schopnosti převodního modulu Calibre. Stačí ho stáhnout a převést do formátu EPUB nebo AZW3, abyste viděli, co Calibre umí.

Calibre automaticky vygeneruje Obsah na základě nadpisů, pokud označíte nadpisy ve Wordu pomocí stylů `Nadpis 1`, `Nadpis 2` atd. Otevřete výstupní e-knihu v prohlížeči Calibre a klikněte na tlačítko *Obsah* pro zobrazení vygenerovaného obsahu.

Starší soubory .doc

U starších souborů .doc můžete v Microsoft Word uložit dokument jako HTML a pak převést výsledný soubor HTML pomocí Calibre. Při ukládání jako HTML je nutné použít volbu „Uložit jako webovou stránku, zjednodušený formát“, protože to vytvoří čisté HTML, které se bude dobře převádět. Pamatujte, že Word vytváří opravdu chaotické HTML, jehož převod může trvat dlouho, takže buďte trpěliví. Pokud máte k dispozici novější verzi Wordu, můžete ho také přímo uložit jako .docx.

Jinou alternativou je využít bezplatné LibreOffice. Otevřete v LibreOffice svůj soubor .doc a uložte ho jako .docx, který může být přímo převeden v Calibre.

4.11.2 Převod dokumentů TXT

Dokumenty TXT nemají dobře definovaný způsob, jak zadávat formátování, jako je tučné, kurzíva atd, nebo strukturu dokumentu, jako jsou odstavce, nadpisy, oddíly a tak dále, ale existuje celá řada běžně používaných zásad. Ve výchozím nastavení se Calibre pokusí o automatické rozpoznání správného formátování a značení na základě těchto zásad.

[29] https://calibre-ebook.com/downloads/demos/demo.docx

Vstup TXT podporuje celou řadu voleb k odlišení, jak jsou rozpoznány odstavce.

Styl odstavců: Automaticky
Analyzuje textový soubor a pokusí se automaticky určit, jak jsou definovány odstavce. Tato volba bude obvykle fungovat dobře, pokud dosáhnete nežádoucí výsledky, zkuste jednu z ručních voleb.

Styl odstavců: Blok
Předpokládá, že jeden nebo více prázdných řádků jsou hranice odstavce:

```
This is the first.

This is the
second paragraph.
```

Styl odstavců: Jednoduchý
Předpokládá, že každý řádek je odstavec:

```
This is the first.
This is the second.
This is the third.
```

Styl odstavců: Tisk
Předpokládá, že každý odstavec začíná odsazením (buď tabulátor nebo 2 a více mezer). Odstavec končí, když je dosaženo dalšího řádku, který začíná odsazením:

```
  This is the
first.
  This is the second.

  This is the
third.
```

Styl odstavců: Neformátovaný
Předpokládá, že dokument nemá žádné formátování, ale používá pevné konce řádků. Pro pokus o znovu vytvoření odstavců je použita interpunkce a střední délka řádku.

Styl formátování: Automaticky
Pokusí se rozpoznat použitý typ značek formátování. Pokud nejsou požity žádné značky, bude použito heuristické formátování.

Styl formátování: Heuristiké
Analyzuje v dokumentu běžné nadpisy kapitol, přerušení příběhu a slova kurzívou a použije příslušné značky HTML během převodu.

Styl formátování: Značkování
Calibre také podporuje průběh vstupu TXT transformačním předzpracováním známým jako Značkování. Značkování umožňuje přidání základního formátování do dokumentů TXT, jako je tučné, kurzíva, nadpisy oddílů, tabulky, seznamy, obsahy, atd. Označení nadpisů kapitol úvodním znakem # a nastavení výrazu XPath pro rozpoznání kapitol na „//h:h1“ je nejjednodušší způsob, jak získat řádný obsah vygenerovaný z dokumentu TXT. Více o Syntaxi značkování se můžete dozvědět na daringfireball[30].

Styl formátování: Žádný
Nepoužije na text žádné zvláštní formátování, dokument je převeden na HTML bez žádných dalších změn.

[30] https://daringfireball.net/projects/markdown/syntax

4.11.3 Převod dokumentů PDF

Dokumenty PDF jsou jedním z nejhorších formátů, ze kterého převádět. Mají pevný formát velikosti stránky a umístění textu. To znamená, že je velice obtížné určit, kde končí jeden odstavec a začíná další. Calibre se pokusí zrušit zalomení odstavců pomocí konfigurovatelného *Koeficientu nezalamování řádků*. Je to měřítko použité k určení délky, při které by měl být řádek nezalomený. Platné hodnoty jsou desetinná čísla mezi 0 a 1. Výchozí hodnota je 0,45, těsně pod střední délkou řádku. Snižte tuto hodnotu pro zahrnutí více textu do nezalamování. Zvyšte pro zahrnutí méně. Tuto hodnotu můžete upravit v nastavení převodu pod *Vstup PDF*.

Často také mají záhlaví a zápatí jako součást dokumentu, které se zahrnou do textu. Použijte panel *Najít a nahradit* pro odebrání záhlaví a zápatí ke zmírnění tohoto problému. Pokud záhlaví a zápatí nejsou odebrány z textu, mohou rozhodit zrušení zalamování odstavců. Abyste se naučili, jak používat volbu odebírání záhlaví a zápatí, přečtěte si regexptutorial`.

Některá omezení vstupu PDF jsou:

- Nejsou podporovány dokumenty, které jsou složité, vícesloupcové a založené na obrázcích.
- Také není podporována extrakce vektorových obrázků a tabulek zevnitř dokumentu.
- Některá PDF používají speciální glyfy zastupující ll, ff nebo fi, atd. Jejich převod může nebo nemusí fungovat v závislosti na tom, jak jsou zastoupeny vnitřně v PDF.
- Nejsou podporovány odkazy a obsahy.
- PDF, která používají vložená písma nepodporující kód Unicode pro zastupování neanglických znaků, způsobí zkomolený výstup pro tyto znaky.
- Některá PDF jsou tvořena fotografiemi stránky s opticky rozpoznaným textem (OCR) za nimi. V takových případech Calibre použije opticky rozpoznaný text, což může být velice odlišné od toho, co vidíte, když si prohlížíte soubor PDF.
- PDF, která jsou použita pro zobrazení složitého textu, jako jsou jazyky zprava doleva a matematické vzorce, nebudou správně převedena.

Znovu opakujeme, že **PDF je opravdu, ale opravdu špatný** formát pro použití jako vstupu. Pokud musíte naprosto nutně použít PDF, pak buďte připraveni na výstup kdekoliv v rozmezí od slušného po nepoužitelný, v závislosti na vstupním PDF.

4.11.4 Sbírky komiksů

Sbírka komiksů je soubor .cbc. Soubor .cbc je soubor ZIP, který obsahuje jiné soubory CBZ nebo CBR. Soubor .cbc musí navíc obsahovat jednoduchý textový soubor nazvaný comics.txt kódovaný v UTF-8. Soubor comics.txt musí obsahovat seznam souborů komiksů uvnitř souboru .cbc v podobě názevsouboru:název, jak je zobrazeno níže:

```
one.cbz:Chapter One
two.cbz:Chapter Two
three.cbz:Chapter Three
```

Soubor .cbc pak bude obsahovat:

```
comics.txt
one.cbz
two.cbz
three.cbz
```

Calibre automaticky převede tento soubor .cbc na e-knihu s obsahem ukazujícím na každou položku v comics.txt.

4.11.5 Ukázka rozšířeného formátování EPUB

V tomto ukázkovém souboru[31] je znázorněno různé rozšířené formátování pro soubory EPUB. Soubor byl vytvořen z ručně kódovaného HTML pomocí Calibre a je určen pro použití jako šablony pro vaše vlastní úsilí o vytváření EPUB.

Zdrojové HTML, ze kterého byl vytvořen, je dostupné v demo.zip[32]. Nastavení použitá pro vytvoření EPUB ze souboru ZIP jsou:

```
ebook-convert demo.zip .epub -vv --authors "Kovid Goyal" --language en --level1-toc '/
↪/*[@class="title"]' --disable-font-rescaling --page-breaks-before / --no-default-
↪epub-cover
```

Pamatujte, že protože tento soubor prozkoumává potenciál EPUB, většina rozšířeného formátování nebude fungovat ve čtečkách méně schopných, než je vestavěný prohlížeč EPUB Calibre.

4.11.6 Převod dokumentů ODT

Calibre umí přímo převést soubory ODT (OpenDocument Text). Pro formátování dokumentu byste měli používat styly a minimalizovat použití přímého formátování. Při vkládání obrázků do dokumentu je musíte ukotvit k odstavci, obrázky ukotvené ke stránce skončí všechny na začátku převodu.

Pro povolení automatického rozpoznání kapitol je musíte označit vestavěnými styly nazvanými *Nadpis 1*, *Nadpis 2*, …, *Nadpis 6* (*Nadpis 1* se rovná značce HTML `<h1>`, *Nadpis 2* se rovná `<h2>` atd.). Při převodu v Calibre můžete zadat, který styl jste použili, do pole *Rozpoznat kapitoly v*. Příklad:

- Pokud označíte kapitoly stylem *Nadpis 2*, musíte nastavit pole ‚Rozpoznat kapitoly v' na `//h:h2`
- Pro vnořený obsah s oddíly označenými *Nadpis 2* a kapitolami označenými ‚*Nadpis 3* musíte zadat `//h:h2|//h:h3`. Na stránce Převod - Obsah nastavte pole *Obsah 1. úrovně* na `//h:h2` a pole *Obsah 2. úrovně* na `//h:h3`.

Jsou rozpoznány dobře známé vlastnosti dokumentu (název, klíčová slova, popis, autor) a Calibre použije první obrázek (ne příliš malý a s dobrým poměrem stran) jako obráek obálky.

K dispozici je také převodní režim rozšířených vlastností, který je aktivován nastavením vlastní vlastnosti `opf.metadata` (typ ‚Ano nebo ne') na Ano ve vašem dokumentu ODT (Soubor->Vlastnosti->Vlastní vlastnosti). Pokud Calibre tuto vlastnost rozpozná, jsou rozpoznány následující vlastní vlastnosti (`opf.authors` přepíše autora dokumentu):

```
opf.titlesort
opf.authors
opf.authorsort
opf.publisher
opf.pubdate
opf.isbn
opf.language
opf.series
opf.seriesindex
```

Kromě toho můžete zadat obrázek, který se použije jako obálka, jeho pojmenováním `opf.cover` (klikněte pravým tlačítkem myši, Obrázek->Volby->Název) v ODT. Pokud není nalezen žádný obrázek s tímto názvem, použije se ‚chytrá' metoda. Protože rozpoznávání obálky může u určitých výstupních formátů vést ke dvěma obálkám, zpracování odebere odstavec (pouze pokud jediný obsah je obálka!) z dokumentu. Ale funguje to pouze s pojmenovaným obrázkem!

Pro zakázání rozpoznávání obálky můžete v rozšířeném režimu nastavit vlastní vlastnost `opf.nocover` (typ ‚Ano nebo Ne') na Ano.

[31] https://calibre-ebook.com/downloads/demos/demo.epub
[32] https://calibre-ebook.com/downloads/demos/demo.zip

4.11.7 Převádění do PDF

První, nejdůležitější nastavení, o kterém musíte rozhodnout při převodu na PDF, je velikost stránky. Ve výchozím nastavení Calibre používá velikost stránky „US Letter“. Toto můžete změnit na jinou standardní velikost stránky nebo zcela vlastní velikost v oddílu *Výstup PDF* dialogového okna převodu. Pokud generujete PDF, které má být použito na konkrétním zařízení, můžete namísto toho zapnout volbu použít velikost stránky z *výstupního profilu* . Takže pokud je váš výstupní profil nastaven na Kindle, Calibre vytvoří PDF s velikostí stránky vhodné pro zobrazení na malé obrazovce Kindle.

Záhlaví a zápatí

Zadáním šablony záhlaví a zápatí můžete vložit libovolná záhlaví a zápatí na každou stránku PDF. Šablony jsou prosté fragmenty kódu HTML, které se vykreslí v místech záhlaví a zápatí. Například pro zobrazení čísel stránek uprostřed dolní části každé stránky zeleně, použijte následující šablonu zápatí:

```
<footer><div style="margin: auto; color: green">_PAGENUM_</div></footer>
```

Calibre automaticky nahradí `_PAGENUM_` aktuálním číslem stránky. Dokonce můžete dát různý obsah na sudé a liché stránky, například následující šablona záhlaví zobrazí název na lichých stránkách a autora na sudých stránkách:

```
<header style="justify-content: flex-end">
    <div class="even-page">_AUTHOR_</div>
    <div class="odd-page"><i>_TITLE_</i></div>
</header>
```

Calibre automaticky nahradí `_TITLE_` a `_AUTHOR_` názvem a autorem převáděného dokumentu. Nastavení `justify-content` na `flex-end` způsobí, že text bude zarovnán vpravo.

Můžete také zobrazit text na levých a pravých okrajích a měnit velikost písma, jak je zobrazeno na příkladu s touto šablonou záhlaví:

```
<header style="justify-content: space-between; font-size: smaller">
    <div>_TITLE_</div>
    <div>_AUTHOR_</div>
</header>
```

Toto zobrazí název vlevo a autora vpravo s velikostí písma menší než hlavní text.

V šablonách můžete také použít aktuální oddíl, jak je zobrazeno níže:

```
<header><div>_SECTION_</div></header>
```

`_SECTION_` je nahrazeno bez ohledu na to, jaký je název aktuálního oddílu. Tyto názvy jsou převzaty z metadat obsahu v dokumentu (osnovy PDF). Pokud dokument nemá žádný obsah, pak bude nahrazeno prázdným textem. Pokud jedna stránka PDF má několik oddílů, bude použit první oddíl na stránce. Podobně je zde proměnná `_TOP_LEVEL_SECTION_`, která může být použita pro získání názvu aktuálního oddílu nejvyšší úrovně.

Uvnitř šablon záhlaví a zápatí dokonce můžete použít JavaScript, například následující šablona způsobí, že čísla stránek začnou od 4 namísto od 1:

```
<footer>
    <div></div>
    <script>document.currentScript.parentNode.querySelector("div").innerHTML = "" + (_
↪PAGENUM_ + 3)</script>
</footer>
```

Níže jsou zdokumentovány některé další proměnné. které můžete použít v záhlavích a zápatích:

- _TOTAL_PAGES_ - celkový počet stránek v souboru PDF, užitečné například pro implementaci počítadla průběhu.
- _TOP_LEVEL_SECTION_PAGES_ - celkový počet stránek v aktuálním oddílu nejvyšší úrovně
- _TOP_LEVEL_SECTION_PAGENUM_ - číslo stránky aktuální stránky v aktuálním oddílu nejvyšší úrovně
- _WIDTH_PIXELS_ - the width of the header/footer area in pixels
- _HEIGHT_PIXELS_ - the height of the header/footer area in pixels

Poznámka

Při přidávání záhlaví a zápatí se ujistěte, že jste v oddíle *Výstup PDF* v dialogovém okně převodu nastavili horní a dolní okraje stránky na dostatečně velké hodnoty.

Tisknutelný obsah

Na konec PDF můžete také vložit tisknutelný obsah, který vypíše čísla stránek pro každý oddíl. To je velmi užitečné, pokud zamýšlíte vytisknout PDF na papír. Pokud si přejete použít PDF na elektronickém zařízení, pak Osnova PDF poskytuje tuto funkci a je generována ve výchozím nastavení.

Vzhled generovaného obsahu můžete přizpůsobit pomocí nastavení Dodatečné CSS pod částí Vzhled a chování dialogového okna převodu. CSS použité ve výchozím nastavení je uvedeno níže, jednoduše ho zkopírujte proveďte v něm libovolné změny.

```
.calibre-pdf-toc table { width: 100%% }

.calibre-pdf-toc table tr td:last-of-type { text-align: right }

.calibre-pdf-toc .level-0 {
    font-size: larger;
}

.calibre-pdf-toc .level-1 td:first-of-type { padding-left: 1.4em }
.calibre-pdf-toc .level-2 td:first-of-type { padding-left: 2.8em }
```

Vlastní okraje stránky pro jednotlivé soubory HTML

Pokud převádíte soubor EPUB nebo AZW3 s více jednotlivými soubory HTML uvnitř a chcete změnit okraje stránek pro konkrétní soubor HTML, můžete pomocí Editoru e-knih Calibre přidat do souboru HTML následující blok stylu:

```
<style>
@page {
    margin-left: 10pt;
    margin-right: 10pt;
    margin-top: 10pt;
    margin-bottom: 10pt;
}
</style>
```

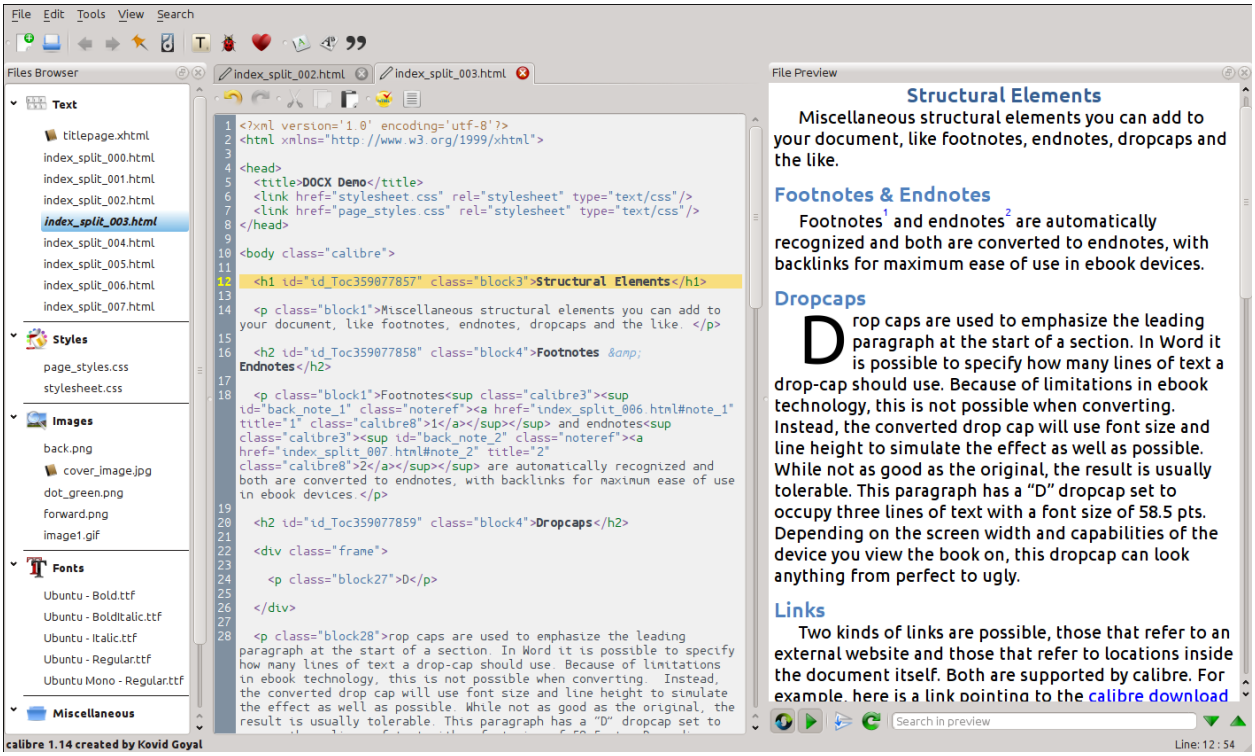
Pak v oddílu výstupu PDF v dialogovém okně převodu zapněte volbu na *Použít okraje stránek z převáděného dokumentu*. Nyní budou mít všechny stránky generované z tohoto souboru HTML okraje `10pt`.

KAPITOLA 5

Úprava e-knih

calibre has an integrated e-book editor that can be used to edit books in the EPUB, KEPUB (Kobo) and AZW3 (Kindle) formats. The editor shows you the HTML and CSS that is used internally inside the book files, with a live preview that updates as you make changes. It also contains various automated tools to perform common cleanup and fixing tasks.

Tento editor můžete použít kliknutím pravým tlačítkem myši na kteroukoliv knihu v Calibre a výběrem *Upravit knihu*.



Obsah

- *Základní pracovní postup* (stránka 81)
- *Prohlížeč souborů* (stránka 83)
  - *Přejmenovávání souborů* (stránka 84)
  - *Slučování souborů* (stránka 84)
  - *Změna pořadí textových souborů* (stránka 84)
  - *Označování obálky* (stránka 85)
  - *Odstraňování souborů* (stránka 85)
  - *Exportování souborů* (stránka 85)
  - *Přidávání nových obrázků, písem atd. nebo vytváření nových prázdných souborů* (stránka 85)
  - *Nahrazování souborů* (stránka 85)
  - *Efektivní propojování šablon stylů se soubory HTML* (stránka 85)
- *Hledat a nahradit* (stránka 86)
  - *Uložená hledání* (stránka 86)
  - *Režim funkcí* (stránka 86)
  - *Hledat a ignorovat značky HTML* (stránka 86)
- *Automatizované nástroje* (stránka 87)
  - *Upravování obsahu* (stránka 87)
  - *Kontrolování knihy* (stránka 88)
  - *Přidávání obálky* (stránka 89)
  - *Vkládání odkazovaných písem* (stránka 89)
  - *Částečné vkládání písem* (stránka 89)
  - *Vylepšení interpunkce* (stránka 89)
  - *Transformace vlastností CSS* (stránka 89)
  - *Odebírání nepoužívaných pravidel CSS* (stránka 89)
  - *Opravování HTML* (stránka 90)
  - *Zkrášlování souborů* (stránka 90)
  - *Vkládání vloženého obsahu* (stránka 90)
  - *Nastavování sémantiky* (stránka 90)
  - *Filtrování informací stylu* (stránka 90)
  - *Upgradování vnitřních prvků knihy* (stránka 91)
- *Kontrolní body* (stránka 91)
- *Panel Živý náhled* (stránka 93)
  - *Rozdělování souborů HTML* (stránka 94)
- *Panel Živé CSS* (stránka 95)

- *Různé nástroje* (stránka 96)
  - *Zobrazení obsahu* (stránka 96)
  - *Kontrola pravopisu slov v knize* (stránka 96)
  - *Vkládání speciálních znaků* (stránka 98)
  - *Zobrazení kontroly kódu* (stránka 99)
  - *Kontrola externích odkazů* (stránka 99)
  - *Stahování externích zdrojů* (stránka 99)
  - *Uspořádávání souborů do složek podle typu* (stránka 99)
  - *Importování souborů v jiných formátech e-knih než EPUB* (stránka 99)
  - *Nástroj Hlášení* (stránka 109)
- *Speciální funkce v editoru kódu* (stránka 110)
  - *Zvýraznění syntaxe* (stránka 110)
  - *Nápověda závislá na kontextu* (stránka 110)
  - *Automatické dokončování* (stránka 111)
  - *Fragmenty* (stránka 111)

5.1 Základní pracovní postup

Poznámka

Videoprohlídka Editoru e-knih Calibre je dostupná zde[33].

When you first open a book with the Edit book tool, you will be presented with a list of files on the left. These are the individual HTML files, stylesheets, images, etc. that make up the content of the book. Simply double click on a file to start editing it. Note that if you want to do anything more sophisticated than making a few small tweaks, you will need to know HTML Tutorial[34] and CSS Tutorial[35].

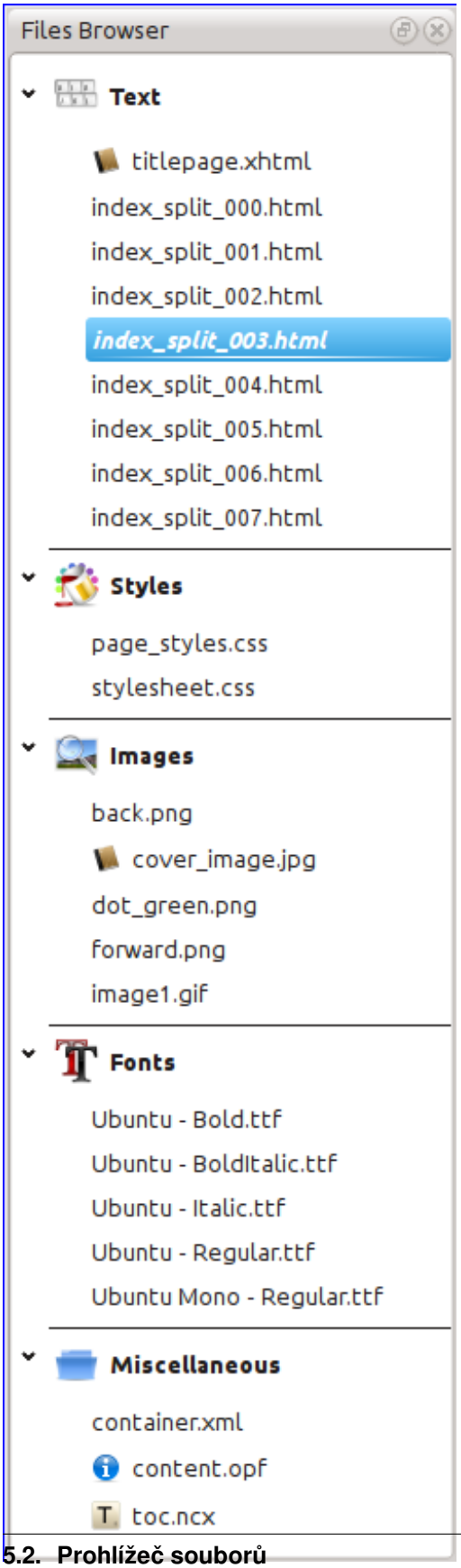
Zatímco provádíte v editoru změny v HTML nebo CSS, bude se vpravo v panelu náhledu zobrazovat živý náhled změn. Pokud jste spokojeni s tím, jak vypadají změny, které jste provedli, klikněte na tlačítko *Uložit* nebo použijte *Soubor → Uložit* pro uložení změn do e-knihy.

Jednou z užitečných funkcí jsou *Kontrolní body*. Než se pustíte do nějaké ambiciózní sady úprav, můžete vytvořit kontrolní bod. Kontrolní bod zachová aktuální stav vaší knihy, pak pokud se v budoucnu rozhodnete, že se vám nelíbí změny, které jste provedli, můžete se vrátit do stavu, kdy jste vytvořili kontrolní bod. Abyste vytvořili kontrolní bod, použijte *Úpravy → Vytvořit kontrolní bod*. Kontrolní body budou pro vás také automaticky vytvořeny, kdykoliv spustíte jakýkoliv automatizovaný nástroj, jako je globální hledání a nahrazování. Funkce kontrolních bodů je doplněk k normálnímu mechanismu zpět/znovu při úpravách jednotlivých souborů. Kontrolní body jsou vyžadovány, když jsou změny rozděleny do více souborů v knize.

Toto je základní pracovní postup pro úpravy knih – Otevřete soubor, proveďte změny, zkontrolujte je a uložte. Zbytek této příručky probere různé přítomné nástroje a funkce, které vám umožní efektivně provést konkrétní úkoly.

[33] https://calibre-ebook.com/demo#tutorials
[34] https://www.w3schools.com/Html/default.asp
[35] https://www.w3schools.com/css/default.asp

5.2 Prohlížeč souborů



Prohlížeč souborů vám poskytuje přehled o různých souborech uvnitř knihy, kterou upravujete. Soubory jsou uspořádány podle kategorie s textovými soubory (HTML) nahoře, následovanými soubory šablon stylů (CSS), obrázky a tak dále. Jednoduše poklepejte na soubor pro spuštění jeho úpravy. Úprava je podporována pro soubory HTML, CSS a obrázky. Pořadí textových souborů je stejné, v jakém by byly zobrazeny, pokud byste četli knihu. Všechny ostatní soubory jsou uspořádány abecedně.

Přejetím myší nad položkou můžete vidět její velikost a také úplnou cestu k souboru uvnitř knihy ve spodní části obrazovky. Pamatujte, že soubory uvnitř e-knih jsou komprimované, takže velikost konečné knihy není součtem jednotlivých velikostí souborů.

Mnoho souborů má v knize zvláštní význam. Ty budou mít obvykle vedle svých názvů ikonu označující zvláštní význam. Například na obrázku vlevo můžete vidět, že soubory *cover_image.jpg* a *titlepage.xhtml* mají vedle sebe ikonu obálky, to znamená, že jsou to obrázek obálky a titulní stránka knihy. Podobně soubor *content.opf* má vedle sebe ikonu metadat, která označuje, že obsahuje metadata knihy, a soubor *toc.ncx* má vedle sebe ikonu T označující, že je to Obsah.

Na jednotlivých souborech můžete provádět mnoho akcí kliknutím pravým tlačítkem myši na ně.

5.2.1 Přejmenovávání souborů

Jednotlivý soubor můžete přejmenovat kliknutím pravým tlačítkem myši na něj a výběrem *Přejmenovat*. Přejmenování souboru automaticky aktualizuje všechny odkazy na něj v celé knize. Takže musíte pouze zadat nový název, Calibre se postará o zbytek.

Můžete také hromadně přejmenovat mnoho souborů najednou. To je užitečné, pokud chcete, aby soubory měly nějaký jednoduchý vzor názvu. Například můžete chtít přejmenovat všechny soubory HTML, aby měly názvy Kapitola-1.html, Kapitola-2.html a tak dále. Vyberte soubory, které chcete hromadně přejmenovat, podržením klávesy `Shift` nebo `Ctrl` a kliknutím na soubory. Pak klikněte pravým tlačítkem myši a vyberte *Hromadně přejmenovat vybrané soubory*. Zadejte předponu a číslo, kterým má začínat automatické číslování, klikněte na OK a je hotovo. Dialogové okno pro hromadné přejmenování také umožňuje přejmenování souborů podle pořadí, v jakém se objevují v knize namísto pořadí, v jakém jste je vybrali, což je užitečné například pro přejmenování všech obrázků podle pořadí, v jakém se zobrazí.

A konečně můžete hromadně změnit příponu souboru pro všechny vybrané soubory. Vyberte více souborů, jak je uvedeno výše, klikněte pravým tlačítkem myši a zvolte *Změnit příponu souboru pro vybrané soubory*.

5.2.2 Slučování souborů

Někdy můžete chtít sloučit dohromady dva soubory HTML nebo dva soubory CSS. Někdy může být užitečné mít vše v jednom souboru. Ale buďte opatrní, uložení velkého množství obsahu do jednoho souboru způsobí problémy s výkonem při zobrazování knihy v typické čtečce e-knih.

Pro sloučení více souborů dohromady je vyberte podržením klávesy `Ctrl` a kliknutím na ně (ujistěte se, že vybíráte pouze soubory jednoho typu, buď všechny soubory HTML nebo všechny soubory CSS a tak dále). Pak klikněte pravým tlačítkem myši a vyberte Sloučit. To je všechno, Calibre sloučí soubory, automaticky se postará o migraci všech odkazů na sloučené soubory. Pamatujte, že slučování souborů může někdy způsobit změnu stylu textů, protože jednotlivé soubory mohly používat různé šablony stylů.

Můžete také vybrat textové soubory a poté je přetáhnout na jiný textový soubor pro sloučení přetažených textových souborů do cílového textového souboru.

5.2.3 Změna pořadí textových souborů

Pořadí, v jakém se textové (HTML) soubory při čtení knihy otevírají, můžete jednoduše změnit přetažením v *Prohlížeči souborů* nebo kliknutím na soubor, který chcete přesunout, a následným stisknutím modifikátorů `Ctrl`+`Shift` s klávesami `Page Up`, `Page Down`, `Home` nebo `End`. Pro technicky zdatné se tomu říká změna pořadí hřbetu knihy.

Pamatujte, že položky je potřeba přetáhnout *mezi* jiné položky, nikoliv na ně, což může být trochu obtížné, dokud si na to nezvyknete. Přetažení na jiný soubor způsobí, že se soubory sloučí.

5.2.4 Označování obálky

E-knihy mají obvykle obrázek obálky. Tento obrázek je označen v *Prohlížeči souborů* ikonou hnědé knihy vedle názvu obrázku. Pokud chcete určit nějaký jiný obrázek jako obálku, můžete to provést kliknutím pravým tlačítkem myši na soubor a výběrem *Označit jako obrázek obálky*.

Navíc soubory EPUB mají koncept *titulní stránky*. Titulní stránka je soubor HTML, který se chová jako titulní stránka nebo obálka knihy. Označit soubor HTML jako titulní stránku při úpravě souborů EPUB můžete kliknutím pravým tlačítkem myši. Dávejte pozor na to, aby označený soubor obsahoval pouze informace o obálce. Pokud obsahuje jiný obsah, jako je první kapitola, pak bude tento obsah ztracen, pokud uživatel někdy převede soubor EPUB v Calibre do jiného formátu. To je proto, že při převodu Calibre předpokládá, že označená titulní stránka obsahuje pouze obálku a žádný jiný obsah.

5.2.5 Odstraňování souborů

Soubory můžete odstranit buď kliknutím pravým tlačítkem myši na ně, nebo jejich výběrem a stisknutím klávesy Delete. Odstranění souboru odebere všechny odkazy na soubor ze souboru OPF, což vám ušetří tu práci. Odkazy na jiných místech však odebrány nejsou, pro jejich snadné nalezení a odebrání nebo nahrazení můžete použít nástroj Zkontrolovat knihu.

5.2.6 Exportování souborů

Soubor zevnitř knihy můžete exportovat někam jinam ve vašem počítači. To je užitečné, pokud chcete pracovat na souboru izolovaně pomocí specializovaných nástrojů. Abyste to udělali, jednoduše klikněte pravým tlačítkem myši na soubor a zvolte *Exportovat*.

Jakmile dokončíte práci na exportovaném souboru, můžete ho znovu importovat do knihy novým kliknutím pravým tlačítkem myši na soubor a výběrem *Nahradit souborem…*, což vám umožní nahradit soubor v knize dříve exportovaným souborem.

Soubory můžete také kopírovat mezi více instancemi editoru. V *Prohlížeči souborů* vyberte soubory, které chcete zkopírovat, klikněte pravým tlačítkem myši a zvolte *Kopírovat vybrané soubory do jiné instance editoru*. Potom v jiné instanci editoru klikněte pravým tlačítkem myši v *Prohlížeči souborů* a zvolte *Vložit soubor z jiné instance editoru*.

5.2.7 Přidávání nových obrázků, písem atd. nebo vytváření nových prázdných souborů

Přidat nový obrázek, písmo, šablonu stylů atd. z vašeho počítače do knihy můžete kliknutím na *Soubor → Nový soubor*. To vám umožní buď importovat soubor kliknutím na tlačítko *Importovat zdrojový soubor*, nebo vytvořit nový prázdný soubor HTML nebo šablonu stylů jednoduše zadáním názvu souboru do pole pro nový soubor.

Můžete také importovat více souborů do knihy najednou pomocí Soubor-> Importovat soubory do knihy.

5.2.8 Nahrazování souborů

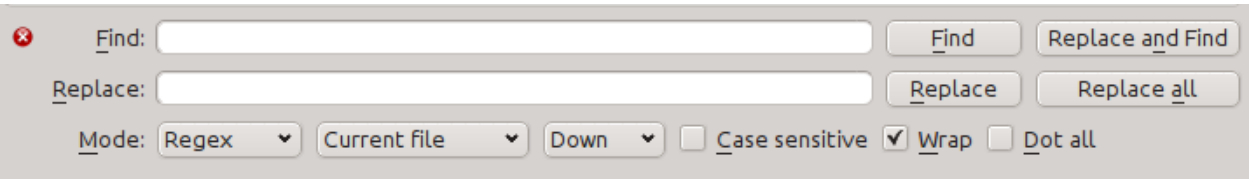
Existující soubory v knize můžete snadno nahradit kliknutím pravým tlačítkem myši na soubor a výběrem nahradit. Tím se automaticky aktualizují všechny odkazy a reference, pokud má nahrazující soubor jiný název než nahrazený soubor.

5.2.9 Efektivní propojování šablon stylů se soubory HTML

Pro větší pohodlí můžete v Prohlížeči souborů vybrat více souborů HTML, kliknout pravým tlačítkem myši a vybrat Propojit šablony stylů, aby Calibre automaticky vložilo značky <link> pro tyto šablony stylů do všech vybraných souborů HTML.

5.3 Hledat a nahradit

Upravit knihu má velice výkonné rozhraní hledání a nahrazení, které vám umožňuje hledat a nahradit text v aktuálním souboru, napříč všemi soubory a dokonce i v označené oblasti aktuálního souboru. Hledat můžete pomocí normálního hledání nebo pomocí regulárních výrazů. Abyste zjistili, jak používat regulární výrazy pro pokročilé vyhledávání, podívejte se na *Vše o používání regulárních výrazů v Calibre* (stránka 214).



Hledání a nahrazení spusťte pomocí položky nabídky *Hledat → Najít a nahradit* (musíte upravovat soubor HTML nebo CSS).

Napište text, který chcete najít, do pole Najít a jeho nahrazení do pole Nahradit. Můžete kliknout na příslušná tlačítka pro nalezení další shody, nahrazení aktuální shody a nahrazení všech shod.

Pomocí rozevíracích nabídek v dolní části okna můžete nechat hledání pracovat nad aktuálním souborem, všemi textovými soubory, všemi soubory stylů nebo všemi soubory. Můžete také zvolit režim hledání, aby byl normální (řetězec) nebo regulární výraz.

Všechny shody pro hledaný výraz můžete spočítat pomocí *Hledat → Spočítat vše*. Počítání se spustí nad všemi soubory nebo oblastmi, které jste vybrali v rozevíracím poli.

Můžete také přejít na konkrétní řádek v aktuálně otevřeném editoru pomocí *Hledat → Přejít na řádek*.

Poznámka

Nezapomeňte, že pro využití plného výkonu hledání a nahrazení budete muset použít regulární výrazy. Podívejte se na *Vše o používání regulárních výrazů v Calibre* (stránka 214).

5.3.1 Uložená hledání

Často používané výrazy hledání nebo nahrazení (včetně výrazů režimu funkcí) můžete uložit a vícekrát je znovu použít . Pro uložení hledání jednoduše klikněte pravým tlačítkem myši do pole Najít a vyberte *Uložit aktuální hledání*.

Uložená hledání můžete vyvolat pomocí *Hledat → Uložená hledání*. To zobrazí seznam výrazů hledání a nahrazení, které můžete použít. Můžete dokonce vybrat více položek v seznamu podržením klávesy `Ctrl` při kliknutí, aby se spustilo více výrazů hledání a nahrazení v jedné operaci.

5.3.2 Režim funkcí

Režim funkcí vám umožňuje psát libovolně výkonné funkce Pythonu, které jsou spuštěny na každém Najít a nahradit. V režimu funkcí můžete provést skoro jakékoliv zpracování textu, jaké chcete. Další informace naleznete v *Režim funkcí pro Hledat a nahradit v editoru* (stránka 99).

5.3.3 Hledat a ignorovat značky HTML

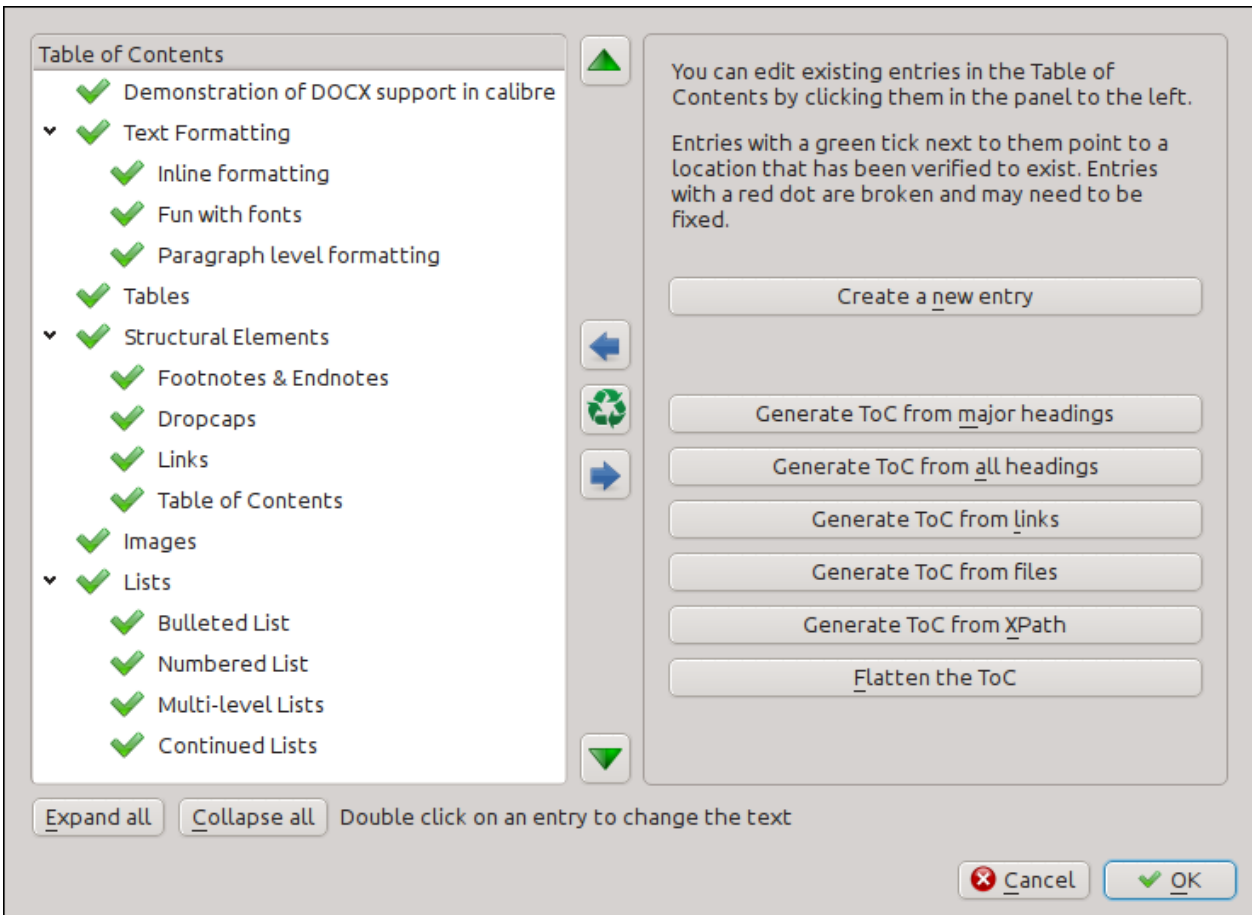
Je zde také vyhrazený nástroj pro hledání textu, který ignoruje jakékoliv značky HTML v něm obsažené. Například pokud kniha obsahuje HTML `Přízvuk na <i>slově</i>.`, můžete hledat `na slově` a bude nalezeno, přestože je uprostřed značka `<i>`. Použijte tento nástroj prostřednictvím položky nabídky *Hledat → Hledat a ignorovat značky HTML*.

5.4 Automatizované nástroje

Editor knih má různé nástroje pro pomoc s běžnými úkoly. Ty jsou přístupné pomocí nabídky *Nástroje*.

5.4.1 Upravování obsahu

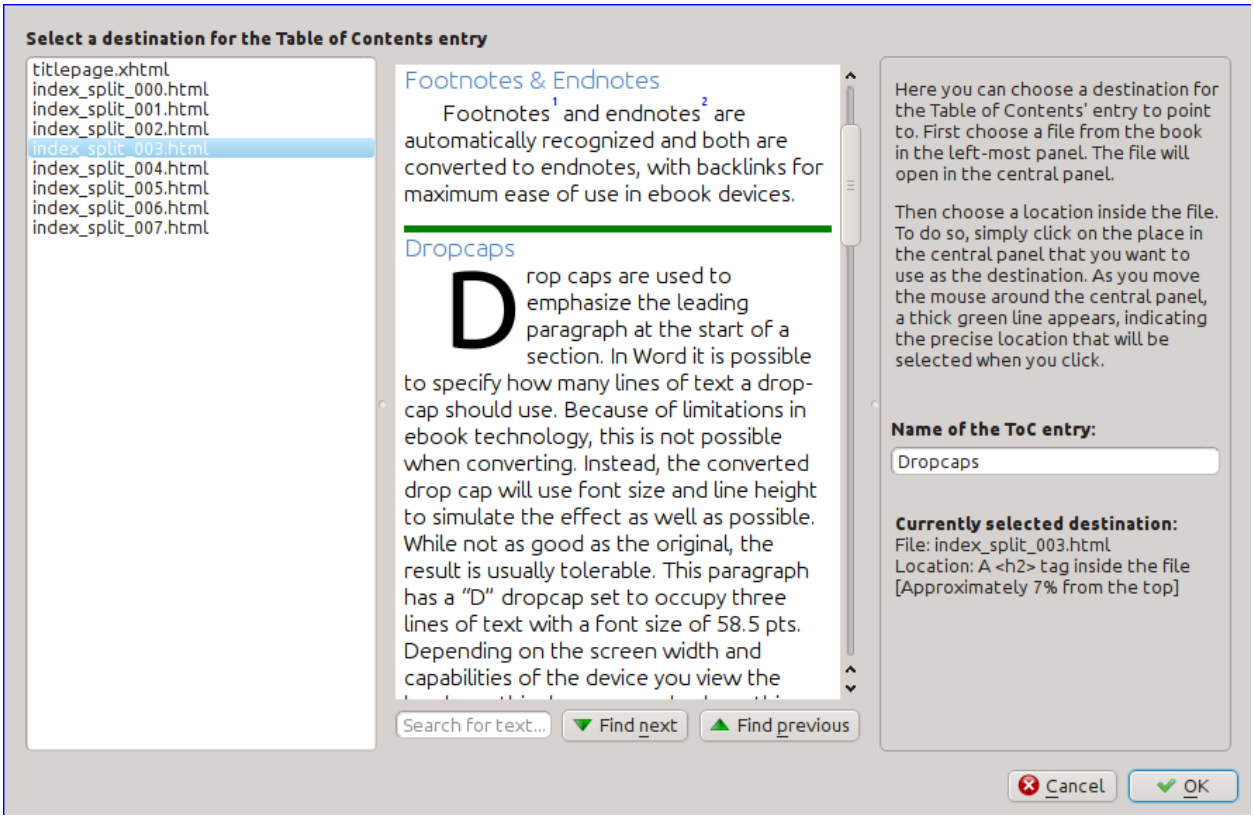
Pro usnadnění úpravy obsahu je zde specializovaný nástroj. Spustíte ho pomocí *Nástroje → Obsah → Upravit obsah*.



Nástroj Upravit obsah zobrazí aktuální Obsah (pokud existuje) vlevo. Jednoduše poklepejte na jakoukoliv položku pro změnu jejího textu. Můžete také přeuspořádat položky přetažením nebo pomocí tlačítek vpravo.

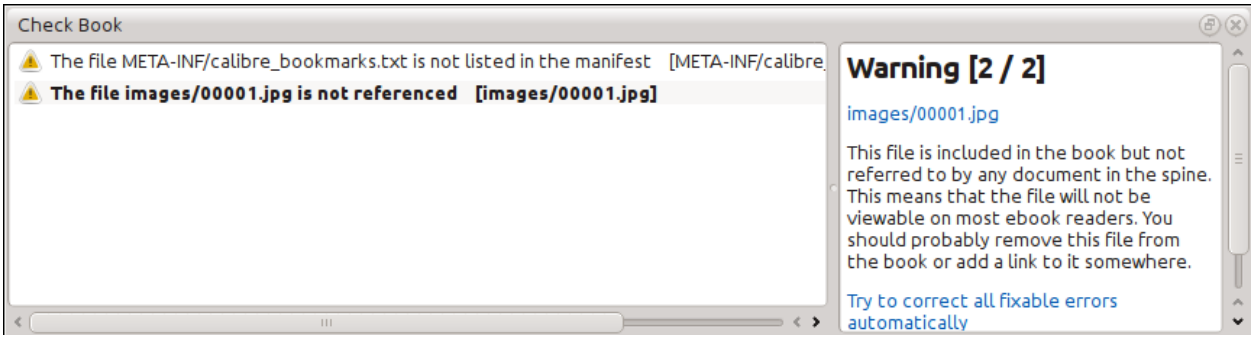
U knih, které nemají již existující Obsah, vám nástroj dává různé možnosti automatického vygenerování obsahu z textu. Můžete ho vygenerovat z nadpisů v dokumentu, z odkazů, z jednotlivých souborů a tak dále.

Jednotlivé položky můžete upravit kliknutím na ně a pak kliknutím na tlačítko *Změnit umístění, na které tato položka ukazuje*. To otevře malý náhled knihy, jednoduše přesuňte kurzor myši nad panel zobrazení knihy a klikněte na umístění, kam chcete, aby položka ukazovala. Tlustá zelená čára vám ukáže umístění. Klikněte na OK, jakmile jste spokojeni s umístěním.



5.4.2 Kontrolování knihy

Nástroj *Zkontrolovat knihu* vyhledá ve vaší knize problémy, které by mohly bránit zamýšlenému fungování na skutečných čtecích zařízeních. Aktivujte ho pomocí *Nástroje → Zkontrolovat knihu*.



Všechny nalezené problémy jsou nahlášeny v pěkném, snadno použitelném seznamu. Kliknutí na jakoukoliv položku v seznamu zobrazí nějakou nápovědu k této chybě a také vám dá možnost tuto chybu automaticky opravit, pokud chyba může být automaticky opravena. Můžete také poklepat na chybu pro otevření umístění chyby v editoru, abyste ji mohli opravit sami.

Některé z prováděných kontrol jsou:

- Poškozené značení HTML. Je nahlášeno jakékoliv značení HTML, které není analyzována jako dobře vytvořené XML. Oprava zajistí, že vaše značení funguje ve všech kontextech tak, jak má. Calibre také může automaticky opravit tyto chyby, ale automatická oprava může mít někdy neočekávané účinky, proto používejte opatrně. Jako vždy je před automatickou opravou vytvořen kontrolní bod, takže můžete snadno vrátit všechny změny. Automatická oprava funguje na základě analýzy značení pomocí algoritmu HTML5, který je vysoce odolný proti chybám, a pak převodem na dobře vytvořené XML.

- Poškozené nebo neznámé styly CSS. Je nahlášeno jakékoliv CSS, které není platné, nebo které má vlastnosti nedefinované ve standardu CSS 2.1 (a několik z CSS 3). CSS je kontrolováno ve všech šablonách stylů, vložených atributů stylů a značkách <style> v souborech HTML.
- Nefunkční odkazy. Jsou nahlášeny odkazy, které ukazují na soubory uvnitř knihy, které chybí.
- Neodkazované soubory. Jsou nahlášeny soubory v knize, na které neodkazuje žádný jiný soubor, nebo nejsou ve hřbetu.
- Různé běžné problémy v souborech OPF, jako jsou duplicitní položky hřbetu nebo manifestu, poškozené atributy idref nebo metaznačky obálek, chybějící požadované oddíly a tak dále.
- Různé kontroly kompatibility na známé problémy, které mohou způsobit nefunkčnost knihy ve čtecích zařízeních.

5.4.3 Přidávání obálky

Přidat obálku knize můžete snadno pomocí *Nástroje → Přidat obálku*. To vám umožní vybrat si jako obálku existující obrázek v knize nebo importovat nový obrázek do knihy a udělat z něj obálku. Při úpravě souborů EPUB je pro obálku automaticky vygenerováno obalující HTML. Pokud je v knize nalezena existující obálka, je nahrazena. Nástroj se také automaticky postará o správné označení souborů obálek jako obálek v OPF.

5.4.4 Vkládání odkazovaných písem

Přístupné pomocí *Nástroje → Vložit odkazovaná písma*. Tento nástroj najde všechna písma odkazovaná v knize, a pokud již nejsou vložena, vyhledá je ve vašem počítači a vloží je do knihy, pokud jsou nalezena. Ujistěte se, že máte potřebná autorská práva pro vkládání komerčně licencovaných písem, než to uděláte.

5.4.5 Částečné vkládání písem

Přístupné pomocí *Nástroje → Částečně vložit písma*. Tento nástroj zmenší všechna písma v knize tak, aby obsahovala pouze glyfy pro text skutečně přítomný v knize. Toto běžně zmenší velikost souborů písem o asi 50 %. Je však třeba si uvědomit, že jakmile jsou písma vložena částečně, tak pokud přidáte nový text, jehož znaky nejsou přítomné v částečně vloženém písmu, písmo nebude pro nový text fungovat. Takže toto dělejte pouze jako poslední krok ve vašem pracovním postupu.

5.4.6 Vylepšení interpunkce

Převede pomlčky, tři tečky, uvozovky, vícenásobné spojovníky atd. v prostém textu na jejich typograficky správné ekvivalenty. Pamatujte, že algoritmus může někdy generovat nesprávné výsledky, obzvláště pokud jde o jednoduché uvozovky na začátku zkrácených tvarů. Přístupné pomocí *Nástroje → Vylepšit interpunkci*.

5.4.7 Transformace vlastností CSS

Vytvořte pravidla pro transformaci stylování knihy. Například vytvořte pravidlo pro převod veškerého červeného textu na zelený nebo zdvojnásobte velikost písma veškerého textu v knize nebo změňte text určité rodiny písma na kurzívu atd.

Vytváření pravidel je jednoduché, pravidla dodržují formát přirozeného jazyka, který vypadá takto:

- Pokud vlastnost *color* (barva) je *red* (červená), *změň* ji na *green* (zelená)
- Pokud vlastnost *font-size* (velikost písma) je *jakákoliv hodnota*, *vynásob* hodnotu číslem *2*

Dostupné prostřednictvím *Nástroje → Transformace stylů*.

5.4.8 Odebírání nepoužívaných pravidel CSS

Odebere všechna nepoužívaná pravidla CSS ze šablon stylů a značek <style>. Některé knihy vytvořené pomocí výrobních šablon mohou mít velký počet nadbytečných pravidel CSS, která neodpovídají žádnému skutečnému obsahu. Tato

nadbytečná pravidla mohou zpomalit čtečky, které je musejí všechny zpracovat. Přístupné pomocí *Nástroje → Odebrat nepoužívaná pravidla CSS*.

5.4.9 Opravování HTML

Tento nástroj jednoduše převádí HTML, které nemůže být analyzováno jako XML, na dobře vytvořené XML. U e-knih je velice běžné, že nemají dobře vytvořené XML, takže tento nástroj jednoduše automatizuje proces opravy takového HTML. Nástroj funguje analýzou HTML pomocí algoritmu HTML5 (algoritmus použitý ve všech moderních prohlížečích) a poté převedením výsledek na XML. Uvědomte si, že automatické opravy mohou někdy mít neintuitivní výsledky. Pokud tomu dáváte přednost, můžete použít výše probíraný nástroj Zkontrolovat knihu k nalezení a ruční opravě problémů v HTML. Přístupné pomocí *Nástroje → Opravit HTML*.

5.4.10 Zkrášlování souborů

Tento nástroj se používá pro automatické formátování všech souborů HTML a CSS tak, aby „vypadali pěkně“. Kód je automaticky odsazený, takže je pěkně řazený, v případě potřeby jsou vloženy prázdné řádky a tak dále. Pamatujte, že zkrášlování také automaticky opraví poškozené HTML nebo CSS. Proto pokud nechcete, aby byly provedeny jakékoliv automatické opravy, použijte nejdříve nástroj Zkontrolovat knihu pro opravu všech problémů, a teprve pak spusťte zkrášlení. Přístupné pomocí *Nástroje → Zkrášlit všechny soubory*.

Poznámka

V HTML může mít jakýkoliv text významové prázdné znaky podle směrnice CSS o prázdných znacích. Proto by mohlo zkrášlení změnit vykreslení HTML. Aby se tomu zabránilo, pokud je to možné, algoritmus zkrášlení zkrášluje pouze značky na úrovni bloků, které obsahují jiné značky na úrovni bloků. Takže například text uvnitř značky <p> nebude mít změněny prázdné znaky. Ale značka <body>, která obsahuje pouze další značky <p> a , bude zkrášlena. To může někdy znamenat, že určitý soubor nebude zkrášlením ovlivněn, protože nemá žádné vhodné značky na úrovni bloků. V takových případech můžete zkusit různé nástroje zkrášlení, které jsou méně opatrné, například: *HTML Tidy <https://infohound.net/tidy/>* _.

5.4.11 Vkládání vloženého obsahu

Normálně je v e-knihách obsah oddělen od hlavního textu a je obvykle přístupný prostřednictvím speciálního tlačítka nebo nabídky Obsah ve čtecím zařízení e-knih. Můžete také nechat Calibre automaticky vygenerovat *vložený* obsah, který se stane součástí textu knihy. Je generovaný na základě aktuálně definovaného obsahu.

Pokud tento nástroj použijete vícekrát, každé vyvolání způsobí že dříve vytvořený vložený obsah bude nahrazen. Nástroj je přístupný pomocí *Nástroje → Obsah → Vložit vložený obsah*.

5.4.12 Nastavování sémantiky

Tento nástroj se používá k nastavení *sémantiky* v souborech EPUB. Sémantika jsou jednoduše odkazy v souboru OPF, které identifikují určitá umístění v knize, že mají zvláštní význam. Můžete je použít k identifikaci předmluvy, věnování, obálky, obsahu atd. Jednoduše zvolte typ sémantické informace, který chcete zadat, a pak vyberte umístění v knize, na které má odkaz směřovat. Tento nástroj je přístupný pomocí *Nástroje → Nastavit sémantiku*.

5.4.13 Filtrování informací stylu

Tento nástroj může být použit pro snadné odebrání zadaných vlastností stylů CSS z celé knihy. Můžete mu říct, které vlastnosti chcete odebrat, například `color, background-color, line-height`, a budou odebrány ze všech míst, kde se vyskytují — šablony stylů, značky `<style>` a vložené atributy `style`. Po odebrání informací o stylu se zobrazí souhrn všech provedených změn, takže můžete vidět, co přesně se změnilo. Nástroj je přístupný pomocí *Nástroje → Filtrovat informace o stylu*.

5.4.14 Upgradování vnitřních prvků knihy

Tento nástroj může být použit pro upgrade vnitřních prvků knihy. Například upgraduje knihy EPUB 2 na knihy EPUB 3. Nástroj je dostupný prostřednictvím *Upgradovat vnitřní prvky knihy*.

5.5 Kontrolní body

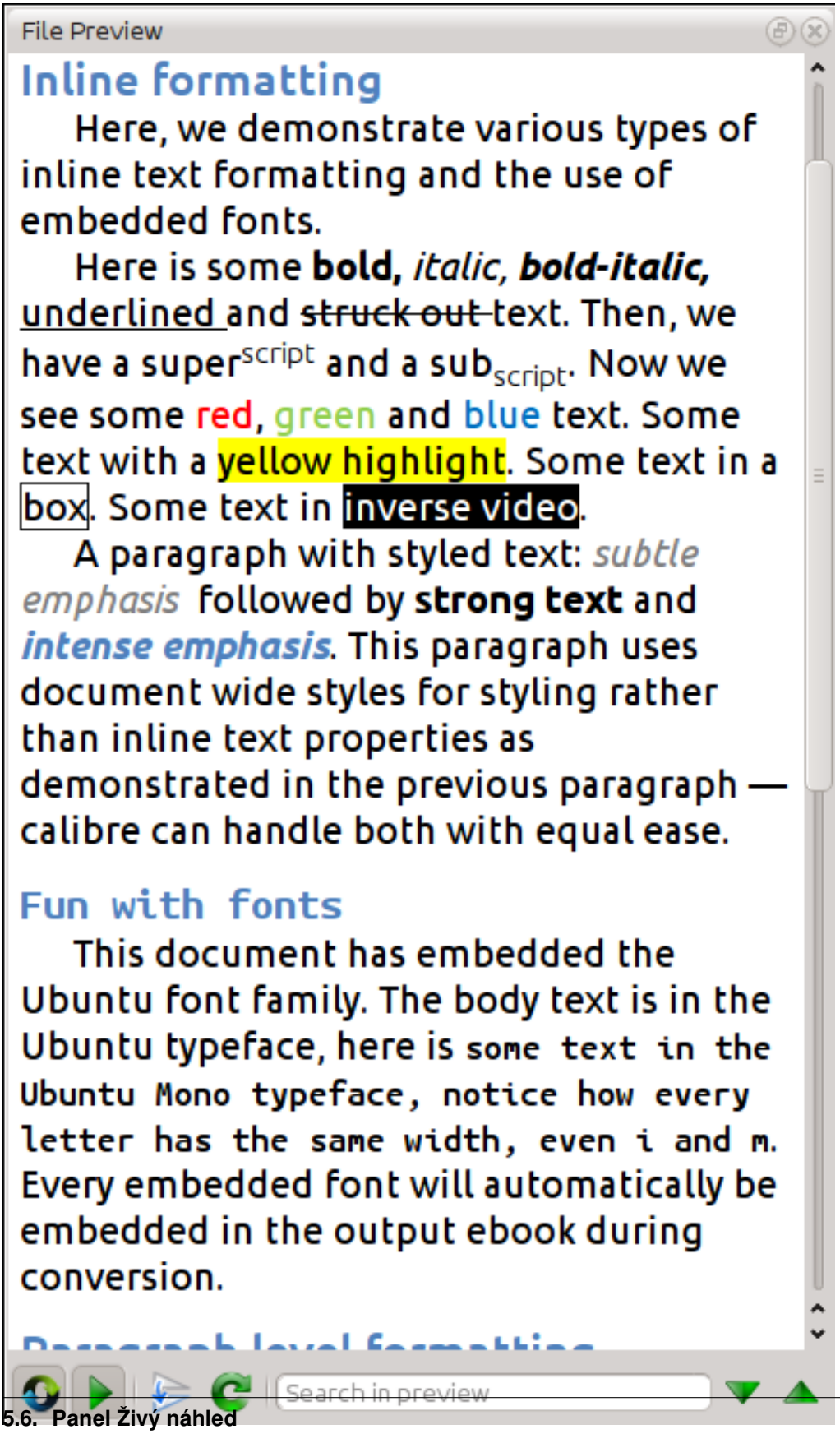
Kontrolní body jsou způsob, jak označit aktuální stav knihy jako „speciální“. Pak můžete pokračovat v provádění jakýchkoliv změn v knize, a pokud se vám nelíbí výsledky, můžete se vrátit do stavu kontrolního bodu. Kontrolní body jsou automaticky vytvořeny při každém spuštění některého z automatizovaných nástrojů popsaných v předchozí části.

Kontrolní bod můžete vytvořit pomocí *Úpravy → Vytvořit kontrolní bod*. A vrátit se k předchozímu kontrolnímu bodu pomocí *Úpravy → Vrátit …*

Funkce kontrolních bodů je doplněk k normálnímu mechanismu Zpět/Znovu při úpravách jednotlivých souborů. Kontrolní body jsou vyžadovány, když jsou změny rozděleny do více souborů v knize, nebo pokud chcete být schopni vrátit velkou skupinu souvisejících změn jako celku.

Seznam dostupných kontrolních bodů můžete zobrazit pomocí *Zobrazení → Kontrolní body*. Porovnat současný stav knihy se zadaným kontrolním bodem můžete pomocí nástroje *Porovnání e-knih* (stránka 121) – výběrem kontrolního bodu a kliknutím na tlačítko *Porovnat*. Tlačítko *Vrátit* obnoví knihu na vybraný kontrolní bod a zruší všechny změny od vytvoření kontrolního bodu.

5.6 Panel Živý náhled



Náhled souboru vám poskytuje přehled o různých souborech uvnitř knihy. Panel Živý náhled vám živě ukazuje změny, které provádíte (se zpožděním jedné nebo dvou sekund). Jak upravujete soubory HTML nebo CSS, panel náhledu se automaticky aktualizuje, aby odrážel změny. Když pohybujete kurzorem v editoru, panel náhledu sleduje jeho umístění a ukazuje vám odpovídající umístění v knize. Kliknutí do panelu náhledu způsobí, že kurzor v editoru bude umístěn nad prvek, na který jste kliknuli. Pokud kliknete na odkaz směřující na jiný soubor v knize, bude tento soubor automaticky otevřen panelu úprav a náhledu.

Automatickou synchronizaci pozice a živého náhledu změn můžete vypnout pomocí tlačítek pod panelem náhledu. K živé aktualizaci panelu náhledu dochází pouze tehdy, pokud aktivně nepíšete v editoru, abyste nebyli rušeni nebo zpomalováni čekáním na vykreslení náhledu.

Panel náhledu vám ukazuje, jak bude text vypadat při zobrazení. Panel náhledu však není náhradou za skutečné testování vaší knihy ve skutečném čtecím zařízení. Je jak více, tak méně schopný než skutečná čtečka. Bude tolerovat chyby a nedbalé značení mnohem lépe než většina čtecích zařízení. Také nezobrazí okraje stránek, konce stránek a vložená písma, která používají aliasy názvu písma. Používejte panel náhledu při práci na knize, ale jakmile skončíte, zkontrolujte ji ve skutečném čtecím zařízení nebo softwarovém emulátoru.

Poznámka

Panel náhledu nepodporuje vložená písma, pokud název písma uvnitř souboru písma neodpovídá názvu v pravidle CSS @font-face. Můžete použít nástroj Zkontrolovat knihu pro rychlé nalezení a opravu jakýchkoliv takových problémových písem.

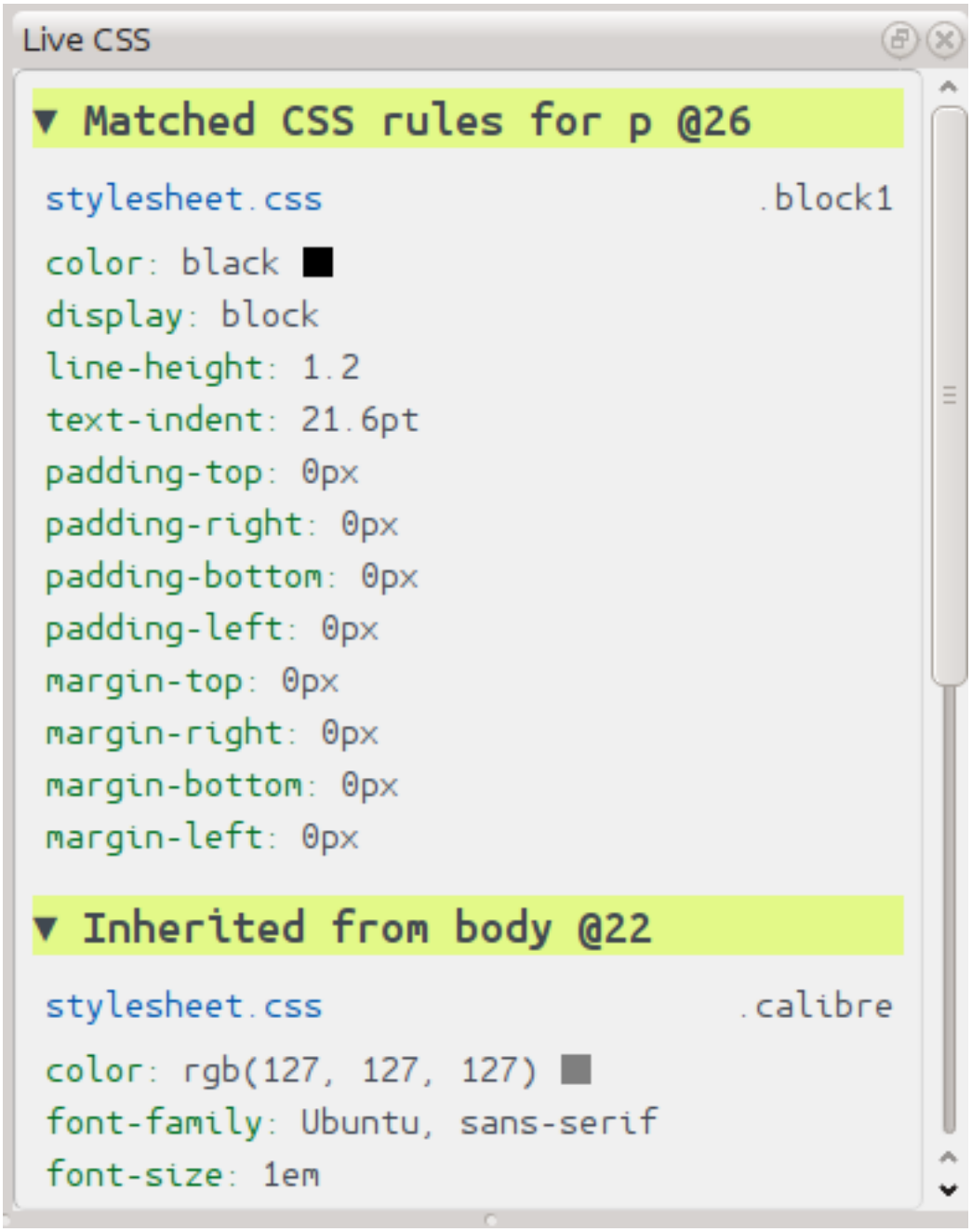
5.6.1 Rozdělování souborů HTML

Jedno možná ne tak zřejmé použití panelu náhledu je rozdělení dlouhých souborů HTML. Během zobrazení souboru, který chcete rozdělit, klikněte na tlačítko *Režim rozdělení* pod panelem náhledu . Pak jednoduše přesuňte myš na místo, kde chcete rozdělit soubor, a klikněte. Tlustá zelená čára vám ukáže, kde přesně dojde k rozdělení, když budete pohybovat myší. Jakmile najdete umístění, které chcete, jednoduše klikněte a rozdělení bude provedeno.

Rozdělení souboru automaticky aktualizuje všechny odkazy, které odkazují na dolní polovinu souboru, a otevře nově rozdělený soubor v editoru.

Jeden soubor HTML můžete také automaticky rozdělit na více místech kliknutím pravým tlačítkem myši uvnitř souboru v editoru a výběrem *Rozdělit na více místech*. To vám umožní snadno rozdělit velký soubor na všech značkách nadpisů nebo všech značkách, které mají určitou třídu, a tak dále.

5.7 Panel Živé CSS



Panel *Živé CSS* zobrazí všechna pravidla stylů, která jsou použita na značku, kterou právě upravujete. Je zobrazen název značky společně s jejím číslem řádku v editoru následovaný seznamem odpovídajících pravidel stylů.

Je to skvělý způsob, jak rychle zjistit, která pravidla stylů jsou použita na jakoukoliv značku. Zobrazení má také kliknutelné odkazy (modře), které vás přenesou přímo na místo, kde byl styl definován, pokud chcete provést jakékoliv změny v pravidlech stylu. Jsou zobrazen pravidla stylů, která jsou použita přímo na značku, stejně jako pravidla, která jsou zděděna z nadřazených značek.

Panelu také ukazuje, jaké jsou konečné vypočítané styly pro značku. Vlastnosti v seznamu, které jsou nahrazeny pravidly s vyšší prioritou, jsou zobrazeny přeškrtnuty čárou.

Panel Živé CSS můžete povolit pomocí *Zobrazení → Živé CSS*.

5.8 Různé nástroje

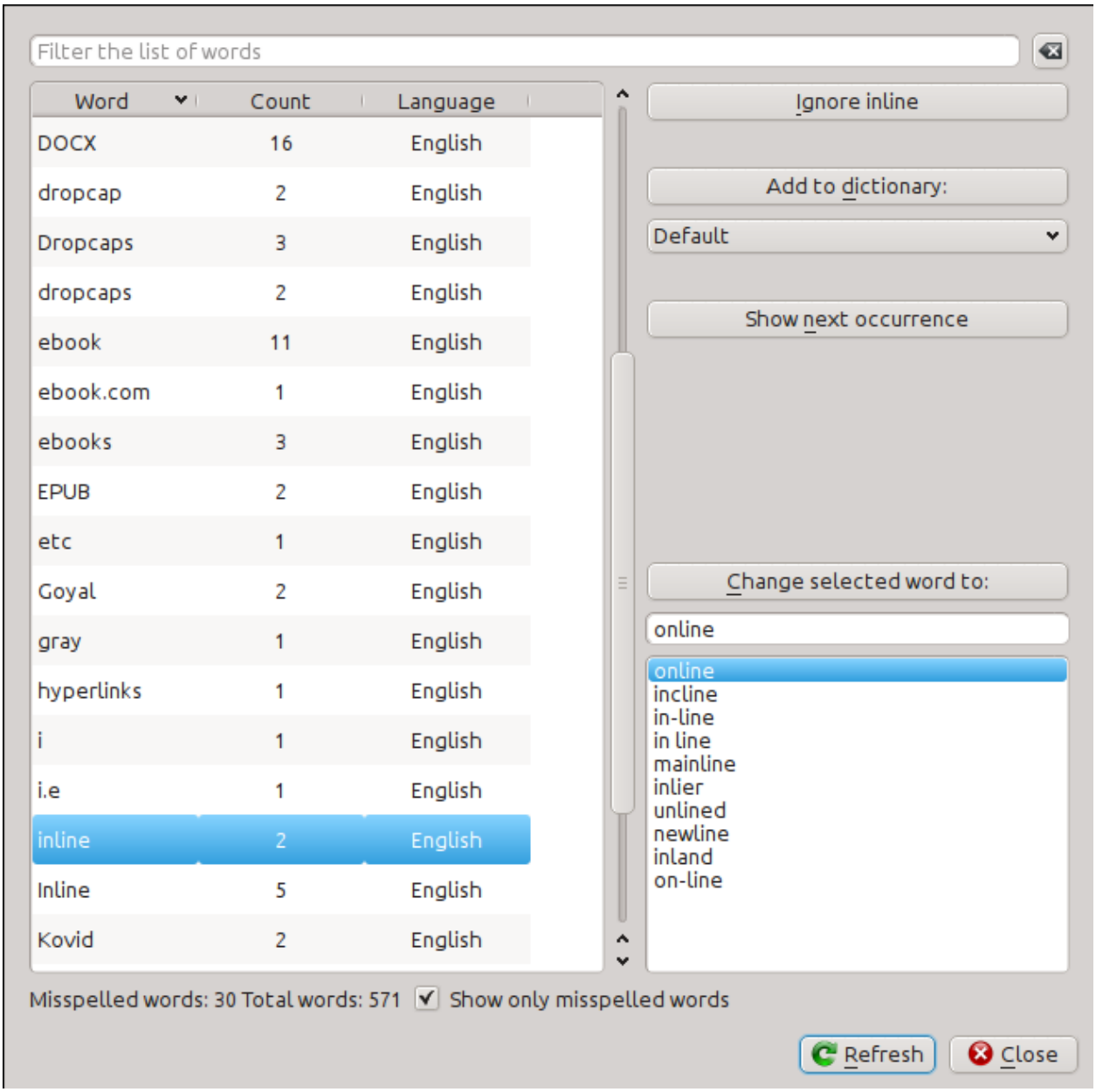
Je zde několik dalších nástrojů, které mohou být užitečné při úpravách knihy.

5.8.1 Zobrazení obsahu

Zobrazení obsahu vám ukazuje aktuální obsah v knize. Poklepáním na jakoukoliv položku se v editoru otevře místo, na které položka ukazuje. Pro úpravu obsahu, aktualizaci zobrazení nebo rozbalení a sbalení všech položek můžete kliknout pravým tlačítkem myši. Toto zobrazení je přístupné pomocí *Zobrazení → Obsah*.

5.8.2 Kontrola pravopisu slov v knize

Kontrolu pravopisu můžete spustit pomocí *Nástroje → Zkontrolovat pravopis*.



Slova jsou zobrazena s počtem výskytů v knize a jazykem, do kterého slovo patří. Informace o jazyku je převzata z metadat knihy a z atributů `lang` v souborech HTML. To umožňuje, aby kontrola pravopisu fungovala dobře i s knihami, které obsahují text ve více jazycích. Například v následujícím úryvku HTML, bude slovo `color` zkontrolováno pomocí americké angličtiny a slovo `colour` pomocí britské angličtiny:

```
<div lang="en_US">color <span lang="en_GB">colour</span></div>
```

Poznámka

Pro zvýraznění dalšího výskytu slova můžete poklepat na toto slovo v editoru. To je užitečné, pokud chcete ručně upravit slovo, nebo vidět, v jakém je v kontextu.

Pro změnu slova jednoduše poklepejte na jeden z nabízených alternativních pravopisů vpravo, nebo napište svůj vlastní

opravený pravopis a klikněte na tlačítko *Změnit vybrané slovo na*. To nahradí všechny výskyty slova v knize. Můžete také kliknout pravým tlačítkem myši na slovo v hlavním seznamu slov pro pohodlnou změnu slova z nabídky zobrazené po kliknutí pravým tlačítkem myši.

Můžete nechat kontrolu pravopisu ignorovat slovo pro aktuální relaci kliknutím na tlačítko *Ignorovat*. Můžete také přidat slovo do uživatelského slovníku kliknutím na tlačítko *Přidat slovo do slovníku*. Kontrola pravopisu podporuje více uživatelských slovníků, takže můžete vybrat slovníku, do kterého chcete slovo přidat.

Můžete také nechat kontrolu pravopisu zobrazit všechna slova v knize, ne jen ta špatně napsaná. To je užitečné, abyste viděli, která slova jsou v knize nejčastější a ke spuštění jednoduchého hledání a nahrazení na jednotlivých slovech.

Poznámka

Pokud provedete jakékoliv změny v knize úpravou souborů, když je otevřen nástroj kontroly pravopisu, měli byste kliknout na tlačítko *Aktualizovat* v nástroji Kontrola pravopisu. Pokud to neuděláte a budete pokračovat v používání nástroje Kontrola pravopisu, mohli byste přijít o změny provedené v editoru.

Poznámka

Pro vyloučení jednotlivého souboru z kontroly pravopisu při spuštění nástroje kontroly pravopisu můžete použít tlačítko *Vyloučit soubory* nebo přidat následující komentář těsně pod otevírací značku v souboru:

```
<!-- calibre-no-spell-check -->
```

Přidávání nových slovníků

Kontrola pravopisu obsahuje vestavěné slovníky pro angličtinu a španělštinu. Vlastní slovníky můžete nainstalovat pomocí *Úpravy → Předvolby → Nastavení editoru → Spravovat pravopisné slovníky*. Kontrola pravopisu může použít slovníky z programu LibreOffice (ve formátu .oxt). Tyto slovníky si můžete stáhnout z Repozitáře rozšíření LibreOffice[36].

5.8.3 Vkládání speciálních znaků

Znaky, které jsou obtížně napsat, můžete vložit pomocí nástroje *Úpravy → Vložit speciální znak*. Ten vám zobrazí všechny znaky Unicode, jednoduše klikněte na znak, který chcete zadat. Pokud při kliknutí podržíte klávesu `Ctrl`, okno se samo zavře po vložení vybraného znaku. Tento nástroj může být použit k vložení speciálních znaků do hlavního textu, nebo do jakékoliv jiné oblasti uživatelského rozhraní, jako je například nástroj Hledat a nahradit.

Protože existuje spousta znaků, můžete definovat své *Oblíbené* znaky, které budou zobrazeny jako první. Jednoduše klikněte pravým tlačítkem myši na znak pro jeho označení jako oblíbeného. Můžete také kliknout pravým tlačítkem myši na znak v oblíbených pro jeho odebrání z oblíbených. A konečně můžete přeuspořádat pořadí znaků v oblíbených kliknutím na tlačítko *Přeuspořádat oblíbené* a přetažením znaku v oblíbených.

Můžete také přímo zadat speciální znaky pomocí klávesnice. Abyste to udělali, zadejte kód Unicode pro znak (šestnáctkově) a pak stiskněte `Alt+X`, což převede dříve zadaný kód na odpovídající znak. Například abyste napsali ÿ, napsali byste ff a pak `Alt+X`. Pro zadání pevné mezery byste použili a0 a pak `Alt+X`, pro zadání vodorovných tří teček byste použili 2026 a `Alt+X` a tak dále.

A nakonec speciální znaky můžete zadat pomocí pojmenovaných entit HTML. Například zadání   bude nahrazeno pevnou mezerou, když napíšete středník. K nahrazení dojde, pouze když napíšete středník.

[36] https://extensions.libreoffice.org/?Tags%5B%5D=50

5.8.4 Zobrazení kontroly kódu

Toto zobrazení vám ukáže kódování HTML a CSS, které jsou použity na aktuální prvek. Otevřete ho kliknutím pravým tlačítkem myši na místo v panelu náhledu a výběrem *Zkontrolovat prvek*. To vám umožní zobrazit kódování HTML pro tento prvek a, co je důležitější, styly CSS, které jsou na něj použity. Můžete dokonce dynamicky upravovat styly a okamžitě uvidíte, jaký vliv mají vaše změny. Pamatujte, že úprava stylů ve skutečnosti neprovádí změny obsahu knihy, umožňuje to pouze rychlé experimentování. Schopnost živé úpravy uvnitř kontroly prvku je ve vývoji.

5.8.5 Kontrola externích odkazů

Tento nástroj můžete použít ke kontrole všech odkazů ve své knize, které odkazují na externí webové stránky. Nástroj se pokusí navštívit každou externě odkazovanou webovou stránku, a pokud návštěva selže, nahlásí všechny poškozené odkazy v pro vás vhodném formátu, abyste je opravili.

5.8.6 Stahování externích zdrojů

Tento nástroj můžete použít pro automatické stáhnutí jakýchkoliv obrázků, šablon stylů atd. v knize, které nejsou v sadě s knihou (tj. mají adresy URL ukazující na umístění na internetu). Nástroj najde všechny takové zdroje a automaticky je stáhne, přidá je ke knize a nahradí všechny odkazy na ně, aby používaly stažené soubory.

5.8.7 Uspořádávání souborů do složek podle typu

Při úpravách souborů EPUB, které jste odněkud získali, často zjistíte, že soubory uvnitř EPUB jsou uspořádány nahodile v různých podsložkách. Tento nástroj vám umožňuje automaticky přesunout všechny soubory do podsložek na základě jejich typů. Přístupné je to pomocí *Nástroje → Uspořádat do složek*. Pamatujte, že tento nástroj pouze mění způsob, jakým jsou soubory uspořádány uvnitř EPUB, ale nemění to, jak jsou zobrazeny v Prohlížeči souborů.

5.8.8 Importování souborů v jiných formátech e-knih než EPUB

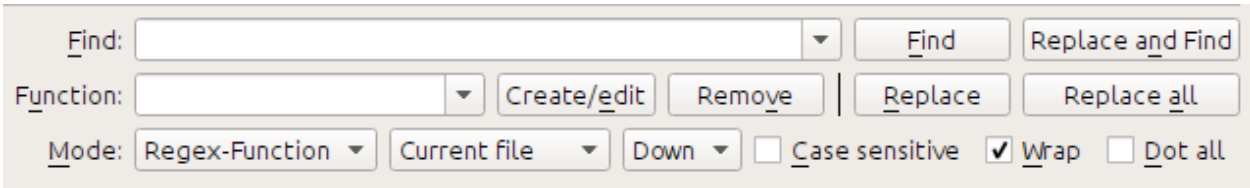
Editor obsahuje schopnost přímo importovat soubory v některých jiných formátech e-knih jako nový EPUB, aniž by prošly úplným převodem. To je obzvlášť užitečné pro přímé vytvoření souborů EPUB z vašich ručně upravených souborů HTML. Můžete to udělat pomocí *Soubor → Importovat soubor HTML nebo DOCX jako novou knihu*.

Režim funkcí pro Hledat a nahradit v editoru

Nástroj *Hledat a nahradit* v editoru podporuje *režim funkcí*. V tomto režimu můžete kombinovat regulární výrazy (viz: *Vše o používání regulárních výrazů v Calibre* (stránka 214)) s libovolnými výkonnými funkcemi Pythonu pro provádění všech druhů rozšířeného zpracování textu.

Ve standardním režimu *regulární výraz* pro hledání a nahrazení zadáváte jak regulární výraz pro hledání, tak šablonu, která je použita pro nahrazení všech nalezených shod. V režimu funkcí, namísto použití pevné šablony, zadáte libovolnou funkci v programovacím jazyce Python[37]. Umožní vám to udělat spoustu věcí, které nejsou možné u jednoduchých šablon.

Techniky pro použití režimu funkcí a syntaxe budou popsány pomocí příkladů, které vám ukáží, jak vytvořit funkce pro provádění postupně složitějších úloh.



[37] https://docs.python.org/

Automatická oprava velikosti písmen nadpisů v dokumentu

Zde budeme využívat jednu z vestavěných funkcí v editoru pro automatickou změnu velikosti písmen celého textu uvnitř značek nadpisů, aby byla všechna první písmena velká:

```
Find expression: <([Hh][1-6])[^>]*>.+?</\1>
```

For the function, simply choose the *Title-case text (ignore tags)* builtin function. The will change titles that look like: `<h1>some titLE</h1>` to `<h1>Some Title</h1>`. It will work even if there are other HTML tags inside the heading tags.

Vaše první vlastní funkce – vylepšení pomlček

Skutečná síla režimu funkcí pochází ze schopnosti vytvářet vaše vlastní funkce pro zpracování textu libovolnými způsoby. Nástroj Vylepšit interpunkci v editoru nechává samostatné pomlčky být, takže můžete použít tuto funkci pro jejich nahrazení dlouhými pomlčkami.

Pro vytvoření nové funkce jednoduše klikněte na tlačítko *Vytvořit nebo upravit* pro vytvoření nové funkce a zkopírujte níže uvedený kód Pythonu.

```
def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    return match.group().replace('--', '—').replace('-', '—')
```

Každá vlastní funkce *Najít a nahradit* musí mít jedinečný název a skládat se z funkce Pythonu s názvem replace, která přijímá všechny výše uvedené parametry. Pro tuto chvíli se nebudeme starat o všechny různé parametry funkce `replace()`. Soustředíme se pouze na parametr `match`. Zastupuje shodu při spuštění hledat a nahradit. Jeho úplná dokumentace je dostupná zde[38]. `match.group()` jednoduše vrátí všechny odpovídající texty a vše, co uděláme je, že nahradíme pomlčky v tomto textu dlouhými pomlčkami, nejdříve se dvojité pomlčky a potom jednotlivé pomlčky.

Použijte tuto funkci s hledáním regulárního výrazu:

```
>[^<>]+<
```

A nahradí to všechny pomlčky dlouhými pomlčkami, ale pouze ve skutečném textu a ne uvnitř definicí značek HTML.

Síla režimu funkcí – používání pravopisného slovníku pro opravu chybně rozdělených slov spojovníkem

E-knihy vytvořené naskenováním tištěných knih často obsahují chybně rozdělená slova spojovníkem – slova, která byla rozdělena na konci řádku na vytištěné stránce. Napíšeme si jednoduchou funkci pro automatické nalezení a opravení takových slov.

```
import regex
from calibre import replace_entities
from calibre import prepare_string_for_xml

def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):

    def replace_word(wmatch):
        # Try to remove the hyphen and replace the words if the resulting
        # hyphen free word is recognized by the dictionary
        without_hyphen = wmatch.group(1) + wmatch.group(2)
```

(continues on next page)

[38] https://docs.python.org/library/re.html#match-objects

(pokračujte na předchozí stránce)

```
        if dictionaries.recognized(without_hyphen):
            return without_hyphen
        return wmatch.group()

    # Search for words split by a hyphen
    text = replace_entities(match.group()[1:-1])  # Handle HTML entities like &amp;
    corrected = regex.sub(r'(\w+)\s*-\s*(\w+)', replace_word, text, flags=regex.
↪VERSION1 | regex.UNICODE)
    return '>%s<' % prepare_string_for_xml(corrected)  # Put back required entities
```

Použijte tuto funkci se stejným vyhledávacím výrazem jako předtím, a to:

```
>[^<>]+<
```

A jako kouzlem to opraví všechna chybně rozdělená slova v textu knihy. Hlavní trik je použít jeden z užitečných dodatečných parametrů pro funkci nahrazení, `dictionaries`. Ten odkazuje na slovníky, které samotný editor používá pro kontrolu pravopisu textu v knize. Tato funkce hledá slova oddělená spojovníkem, odebere spojovník a zkontrolujte, zda slovník rozpozná složené slovo. Pokud ano, původní slova jsou nahrazena složeným slovem bez spojovníku.

Pamatujte, že jediné omezení této techniky je, že to bude fungovat pouze pro jednojazyčné knihy, protože `dictionaries.recognized()` ve výchozím nastavení používá hlavní jazyk knihy.

Automatické číslování oddílů

Nyní uvidíme něco trochu jiného. Předpokládejme, že váš soubor HTML má mnoho oddílů, každý s nadpisem ve značce `<h2>`, která vypadají takto `<h2>Nějaký text</h2>`. Můžete si vytvořit vlastní funkci, která automaticky očísluje tyto nadpisy za sebou následujícími čísly oddílů, takže budou vypadat takto `<h2>1. Nějaký text</h2>`.

```
def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    section_number = '%d. ' % number
    return match.group(1) + section_number + match.group(2)

# Ensure that when running over multiple files, the files are processed
# in the order in which they appear in the book
replace.file_order = 'spine'
```

Použijte vyhledávací výraz:

```
(?s)(<h2[^<>]*>)(.+?</h2>)
```

Umístěte kurzor na začátek souboru a klikněte na *Nahradit vše*.

Tato funkce využívá další z užitečných dodatečných parametrů pro `replace()`: parametr `number`. Při provádění *Nahradit vše* je číslo automaticky zvýšeno pro každou následující shodu.

Další novinkou je použití `replace.file_order` – nastavení, které pro `'páteř'` znamená, že pokud je toto hledání spuštěno na více souborech HTML, soubory jsou zpracovávány v pořadí, v jakém se objevují v knize. Pro podrobnosti viz *Volba pořadí souborů při spuštění na více souborech HTML* (stránka 105).

Automatické vytvoření obsahu

Nakonec zkusme něco trochu náročnějšího. Předpokládejme, že vaše kniha má nadpisy ve značkách `h1` a `h2`, které vypadají takto `<h1 id="nejakeid">Nějaký text</h1>`. Automaticky vygenerujeme HTML obsah na základě těchto nadpisů. Vytvořte níže uvedenou vlastní funkci:

```
from calibre import replace_entities
from calibre.ebooks.oeb.polish.toc import TOC, toc_to_html
from calibre.gui2.tweak_book import current_container
from calibre.ebooks.oeb.base import xml2str

def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    if match is None:
        # All matches found, output the resulting Table of Contents.
        # The argument metadata is the metadata of the book being edited
        if 'toc' in data:
            toc = data['toc']
            root = TOC()
            for (file_name, tag_name, anchor, text) in toc:
                parent = root.children[-1] if tag_name == 'h2' and root.children else␣
↪root
                parent.add(text, file_name, anchor)
            toc = toc_to_html(root, current_container(), 'toc.html', 'Table of␣
↪Contents for ' + metadata.title, metadata.language)
            print(xml2str(toc))
        else:
            print('No headings to build ToC from found')
    else:
        # Add an entry corresponding to this match to the Table of Contents
        if 'toc' not in data:
            # The entries are stored in the data object, which will persist
            # for all invocations of this function during a 'Replace All' operation
            data['toc'] = []
        tag_name, anchor, text = match.group(1), replace_entities(match.group(2)),␣
↪replace_entities(match.group(3))
        data['toc'].append((file_name, tag_name, anchor, text))
        return match.group()  # We don't want to make any actual changes, so return␣
↪the original matched text

# Ensure that we are called once after the last match is found so we can
# output the ToC
replace.call_after_last_match = True
# Ensure that when running over multiple files, this function is called,
# the files are processed in the order in which they appear in the book
replace.file_order = 'spine'
```

A použijte ji s vyhledávacím výrazem:

```
<(h[12]) [^<>]* id=['"]([^'"]+)['"][^<>]*>([^<>]+)
```

Spusťte hledání nad *Všechny textové soubory* a na konci vyhledávání se otevře okno s „Výstup ladění z vaší funkce“, které bude mít HTML obsah připravený k vložení do `toc.html`.

Výše uvedená funkce je silně okomentovaná, takže by měla být snadno pochopitelná. Klíčovou novinkou je použití dalšího

užitečného dodatečného parametru funkce `replace()`, objektu `data`. Objekt `data` je *slovník* Pythonu, který přetrvává mezi všemi po sobě následujícími voláními funkce `replace()` během jedné operace *Nahradit vše*`.

Další novinkou je použití `call_after_last_match` – nastavení, které pro `True` nad funkcí `replace()` znamená, že editor vyvolá `replace()` ještě jednou poté, co byly nalezeny všechny shody. Pro toto dodatečné vyvolání bude shoda objektu `None`.

To byla jen ukázka, která vám předvedla sílu režimu funkcí, pokud opravdu potřebujete vygenerovat obsah z nadpisů ve vaší knize, měli byste raději použít vyhrazený nástroj pro obsah v *Nástroje → Obsah*.

API pro režim funkcí

Všechny funkce režimu funkcí musí být funkce Pythonu pojmenované replace s následujícím podpisem:

```
def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    return a_string
```

Když je spuštěno najít a nahradit, pro každou shodu, která je nalezena, bude vyvolána funkce `replace()`, která musí vrátit nahrazovací řetězec pro tuto shodu. Pokud není třeba provést žádná nahrazení, mělo by se vrátit `match.group()`, což je původní řetězec. Jednotlivé parametry pro funkci `replace()` jsou zdokumentovány níže.

Parametr `match`

Parametr `match` představuje aktuálně nalezenou shodu. Je to objekt Match Pythonu[39]. Jeho nejužitečnější metoda je `group()`, která může být použita k získání textu odpovídajícího jednotlivým skupin zachycování v hledání regulárního výrazu.

Parametr `number`

Parametr `number` je číslo aktuální shody. Když spustíte *Nahradit vše*, každá po sobě jdoucí shoda způsobí, že bude vyvoláno``replace()`` se zvyšujícím se číslem. První shoda má číslo 1.

Parametr `file_name`

Toto je název souboru, ve kterém byla nalezena aktuální shoda. Při hledání uvnitř označeného textu je `file_name` prázdný. `file_name` je v kanonické podobě, cesta relativní ke kořenu knihy, s `/` jako oddělovačem cesty.

Parametr `metadata`

Toto představuje metadata aktuální knihy, jako je název, autoři, jazyk atd. Je to objekt třídy: *`calibre.ebooks.metadata.book.base.Metadata`* (stránka 210). Užitečné atributy zahrnují `title`, `authors` (seznam autorů) a `language` (kód jazyka).

Parametr `dictionaries`

Toto představuje sbírku slovníků používaných ke kontrole pravopisu aktuální knihy. Jeho nejužitečnější metoda je `dictionaries.recognized(word)`, která vrátí `True`, pokud je předané slovo rozpoznáno slovníkem pro aktuální jazyk knihy.

[39] https://docs.python.org/library/re.html#match-objects

Parametr `data`

Jedná se o jednoduchý `slovník` Pythonu. Když spustíte *Nahradit vše*`, každá následná shoda způsobí, že se zavolá `replace()` se stejným `slovníkem` jako data. Můžete jej tedy použít k uložení libovolných dat mezi voláními `replace()` během operace *Nahradit vše*`.

Parametr `functions`

Parametr `functions` vám dá přístup ke všem ostatním uživatelem definovaným funkcím. To je užitečné pro opětovné použití kódu. Můžete definovat obslužné funkce na jednom místě a znovu je použít ve všech svých ostatních funkcích. Předpokládejme například, že vytvoříte funkci pojmenovanou `My Function` takto:

```
def utility():
    # do something

def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    ...
```

Potom v jiné funkci můžete přistupovat k funkci `utility()` takto:

```
def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    utility = functions['My Function']['utility']
    ...
```

Můžete také použít objekt funkce pro ukládání trvalých dat, která mohou být znovu použita jinými funkcemi. Například byste mohli mít jednu funkci, která při spuštění s *Nahradit vše* shromažďuje některá data, a další funkci, která je používá, když je spuštěna později. Vezměme si následující dvě funkce:

```
# Function One
persistent_data = {}

def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    ...
    persistent_data['something'] = 'some data'

# Function Two
def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    persistent_data = functions['Function One']['persistent_data']
    ...
```

Ladění vašich funkcí

Funkce, které vytvoříte, můžete ladit pomocí standardní funkce `print()` z Pythonu. Výstup tisku se zobrazí v automaticky otevíraném okně po dokončení Najít a nahradit. Příklad použití `print()` pro výstup celého obsahu jste viděli výše.

Volba pořadí souborů při spuštění na více souborech HTML

Když spustíte *Nahradit vše* na více souborech HTML, pořadí, ve kterém jsou soubory zpracovány, závisí na tom, jaké soubory máte otevřené pro úpravy. Můžete vynutit, aby vyhledávání zpracovalo soubory v pořadí, ve kterém se objeví, nastavením parametru `file_order` ve vaší funkci takto:

```
def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    ...

replace.file_order = 'spine'
```

`file_order` přijímá dvě hodnoty, `spine` a `spine-reverse`, které způsobují, že hledání zpracuje více souborů v pořadí, v jakém jsou uvedeny v knize, a to buď vpřed nebo zpět v uvedeném pořadí.

Vyvolání vaší funkce ještě jednou poté, co je nalezena poslední shoda

Někdy, jako ve výše uvedeném příkladu automatického generování obsahu, je užitečné vyvolat vaši funkci ještě jednou poté, co je nalezena poslední shoda. To můžete provést nastavením parametru ``call_after_last_match` ve vaší funkci takto:

```
def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    ...

replace.call_after_last_match = True
```

Připojování výstupu z funkce pro označený text

Při spuštění hledání a nahrazení na označeném textu je někdy užitečné připojit nějaký text na konec označeného textu. Můžete to udělat nastavením atributu `append_final_output_to_marked` na vaši funkci (pamatujte, že také musíte nastavit `call_after_last_match`) takto:

```
def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    ...
    return 'some text to append'

replace.call_after_last_match = True
replace.append_final_output_to_marked = True
```

Potlačení dialogového okna výsledku při provádění vyhledávání na označeném textu

Můžete také potlačit dialogové okno výsledku (což může zpomalit opakovanou aplikaci hledání a nahrazení na mnoha blocích textu) nastavením atributu `suppress_result_dialog` na vaší funkci takto:

```
def replace(match, number, file_name, metadata, dictionaries, data, functions, *args,␣
↪**kwargs):
    ...

replace.suppress_result_dialog = True
```

Další příklady

Další užitečné příklady, které vytvořili uživatelé Calibre, můžete nalézt ve fóru Editoru e-knih Calibre[40].

Fragmenty

Editor e-knih Calibre podporuje *fragmenty*. Fragment je kus textu, který je buď znovu často použit, nebo obsahuje velké množství nadbytečného textu. Editor vám umožňuje vložit fragment pouhými několika stisky kláves. Předpokládejme například, že zjistíte, že často vkládáte značky odkazů při úpravě souborů HTML, pak můžete jednoduše v editoru napsat `<a` a stiskněte `Control+J`. Editor to rozbalí na:

```
<a href="filename"></a>
```

Nejen to, slovo `filename` bude vybráno a kurzor umístěn nad ním, abyste mohli snadno zadat skutečný název souboru pomocí šikovné funkce editoru *Automatické dokončování* (stránka 111). A jakmile zadáte název souboru, stiskněte znovu `Control+J` a kurzor se přesune na pozici mezi značkami `<a>`, abyste mohli snadno zadat text pro odkaz.

Systém fragmentů v editoru je velice propracovaný, je zde několik předdefinovaných fragmentů a můžete vytvářet své vlastní, aby vyhovovali vašemu stylu úprav.

Následující diskuze o předdefinovaných fragmentech by měla pomoci vysvětlit sílu systému fragmentů.

Poznámka

Fragmenty můžete také použít ve vstupních polích textu v panelu *Hledat a nahradit*, zástupné symboly (pomocí `Control+J` pro přeskakování) však nebudou fungovat.

Předdefinované fragmenty

Předdefinované fragmenty jsou popsány níže. Pamatujte, že je můžete přepsat vytvořením svých vlastních fragmentů se stejným textem spouštěče.

Vkládání výplňového textu [Lorem]

První a nejjednodušší předdefinovaný fragment je používán pro vkládání výplňového textu do dokumentu. Výplňový text je brán z *De finibus bonorum et malorum <https://en.wikipedia.org/wiki/De_finibus_bonorum_et_malorum>*, filozofického díla od Cicera (přeloženého do angličtiny). Pro použití jednoduše zadejte `Lorem` do souboru HTML a stiskněte `Control+J`. Bude to nahrazeno několika odstavci výplně.

Definice tohoto fragmentu je velice jednoduchá, text spouštěče je definován jako `Lorem` a šablona je definována jednoduše jako literární text, který má být vložen. Můžete ho snadno upravit pro použití vámi oblíbené podoby výplňového textu.

Vkládání samouzavírací značky HTML [<>]

Nyní se podívejme na jednoduchý příklad výkonného konceptu *zástupných symbolů*. Řekněme, že chcete vložit samouzavírací značku `<hr/>`. Jednoduše napište `<>` a stiskněte `Control+J`, editor rozbalí fragment na:

```
<|/>
```

Symbol `|` zde představuje aktuální pozici kurzoru. Pak můžete napsat `hr` a stisknout `Control+J` pro přesun kurzoru za konec značky. Fragment je definován jako:

[40] https://www.mobileread.com/forums/showthread.php?t=237181

```
Trigger: <>
Template: <$1/>$2
```

Zástupné symboly jsou jednoduše symboly dolaru ($) následované číslem. Když je fragment rozbalen stiskem `Control+J`, kurzor je umístěn na první zástupný symbol (zástupný symbol s nejnižším číslem). Když znovu stisknete `Control+J`, kurzor přeskočí na další zástupný symbol (zástupný symbol s nejbližším vyšším číslem).

Vkládání značky odkazu HTML [<a]

Všechny značky odkazů HTML sdílejí společnou strukturu. Mají atribut `href` a nějaký text mezi otevíracími a uzavíracími značkami. Fragment, který učiní jejich psaní efektivnějším nás seznámí s některými dalšími funkcemi zástupných symbolů. Pro použití tohoto fragmentu jednoduše napište `<a` a stiskněte `Control+J`. Editor to rozbalí na:

```
<a href="filename|"></a>
```

Nejen to, slovo `filename` bude vybráno a kurzor umístěn nad ním, abyste mohli snadno zadat skutečný název souboru pomocí šikovné funkce editoru *Automatické dokončování* (stránka 111). A jakmile zadáte název souboru, stiskněte znovu `Control+J` a kurzor se přesune na pozici mezi značkami `<a>`, abyste mohli snadno zadat text pro odkaz. Po dokončení psaní textu stiskněte znovu `Control+J` pro přeskočení na místo za ukončovací značku. Tento fragment je definován jako:

```
Trigger: <a
Template: <a href="${1:filename}">${2*}</a>$3
```

Zde je pár nových funkcí. Zaprvé se zástupný symbol `$1` stal složitějším. Nyní obsahuje nějaký *výchozí text* (slovo `filename`). Pokud zástupný symbol obsahuje výchozí text, tak výchozí text nahradí zástupný symbol, když je fragment rozbalen. Také když skočíte na zástupný symbol s výchozím textem pomocí `Control+J`, výchozí text bude vybrán. Tímto způsobem můžete použít výchozí text, aby vám připomenul, abyste vyplnili důležité části šablony. Výchozí text pro zástupný symbol můžete zadat pomocí syntaxe: `${<number>:výchozí text}`.

Druhou novou funkcí je, že druhý zástupný symbol má za sebou hvězdičku (`${2*}`). To znamená, že jakýkoliv text, který byl vybrán před rozbalením šablony nahradí zástupný symbol. Abyste toto viděli v akci, vyberte v editoru nějaký text, stiskněte `Control+J`, napište `<a` a stiskněte znovu `Control+J`, šablona se rozbalí na:

```
<a href="filename">whatever text you selected</a>
```

Vkládání značky obrázku HTML [<i]

Toto je velice podobné vkládání odkazu HTML, což jsme viděli výše. Umožní vám to rychle zadat značku `<img src="filename" alt="description" />` a přeskakovat mezi atributy `src` a `alt`:

```
Trigger: <i
Template: <img src="${1:filename}" alt="${2*:description}" />$3
```

Vkládání libovolné značky HTML [<<]

Toto vám umožní vložit libovolnou úplnou značku HTML (nebo obalit dříve vybraný text značkou). Pro použití jednoduše napište `<<` a stiskněte `Control+J`. Editor to rozbalí na:

```
<|></>
```

Napište název značky, například: `span`, a stiskněte `Control+J`, výsledkem bude:

```
<span>|</span>
```

Všimněte si, že do ukončovací značky bylo automaticky vyplněno `span`. Toho je dosaženo dalšího funkcí zástupných symbolů, *zrcadlením*. Zrcadlení jednoduše znamená, že pokud zadáte do šablony ukázkový zástupný symbol více než jednou, druhá a všechny pozdější pozice budou automaticky vyplní tím, co napíšete do první pozici, když stisknete `Control+J`. Definice tohoto fragmentu je:

```
Trigger: <<
Template: <$1>${2*}</$1>$3
```

Jak můžete vidět, první zástupný symbol ($1) byl zadán dvakrát, podruhé do ukončovací značky, což jednoduše zkopíruje cokoliv, co napíšete do otevírací značky.

Vkládání libovolné značky HTML s atributem třídy [<c]

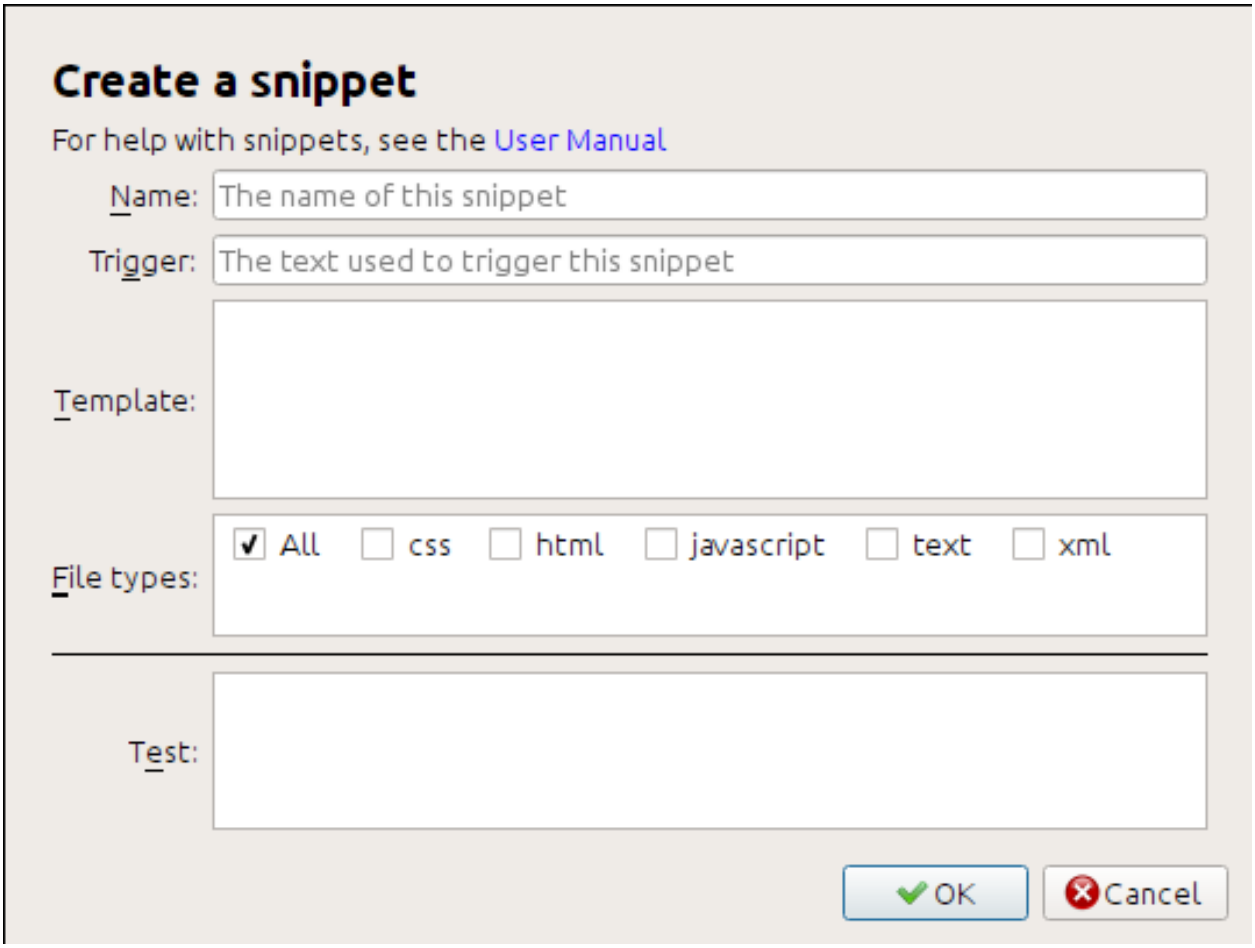
Toto je velice podobné výše uvedenému příkladu vkládání libovolné značky kromě toho, že se předpokládá, že chcete pro značku zadat třídu:

```
Trigger: <c
Template: <$1 class="${2:classname}">${3*}</$1>$4
```

Umožní vám to nejdříve zadat název značky, stiskněte `Control+J`, zadejte název třídy, stiskněte `Control+J`, zadejte obsah značky a stiskněte naposledy `Control+J`, abyste vyskočili ze značky. Ukončovací značka bude vyplněna automaticky.

Vytváření vlastních fragmentů

Fragmenty jsou opravdu senzační, protože si můžete vytvořit svůj vlastní, aby vyhovoval vašemu stylu úprav. Abyste vytvořili svůj vlastní fragment, přejděte v editoru do *Úpravy → Předvolby → Nastavení editoru → Spravovat fragmenty*. Objeví se snadno použitelné dialogové okno, které vám pomůže vytvořit si vlastní fragmenty. Jednoduše klikněte na tlačítko *Přidat fragment* a uvidíte dialogové okno, které vypadá takto:



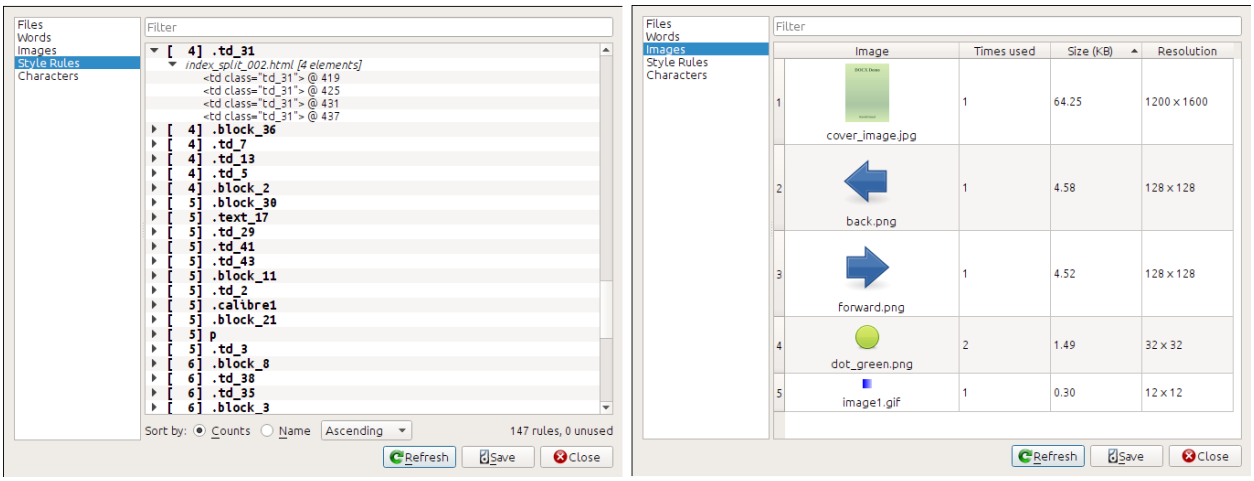
Nejdříve svůj fragment pojmenujte, něco popisného, co vám pomůže v budoucnu fragment poznat. Pak zadejte *spouštěč*. Spouštěč je jednoduše text, který musíte zadat v editoru před stisknutím `Control+J`, aby se rozbalil fragment.

Pak zadejte šablonu fragmentu. Měli byste začít jedním z výše uvedených příkladů a upravit ho podle svých potřeb. Nakonec zadejte, pro které typy souborů chcete, aby byl fragment aktivní. Tímto způsobem můžete mít více fragmentů se stejným spouštěcím textem, které pracují odlišně pro různé typy souborů.

Dalším krokem je otestovat váš nově vytvořený fragment. Použijte pole *Test* v dolní části. Zadejte spouštěcí text a stiskněte `Control+J` pro rozbalení fragmentu a přeskočení mezi zástupné symboly.

5.8.9 Nástroj Hlášení

Editor obsahuje pěkný nástroj *Hlášení* (přes *Nástroje → Hlášení*), který zobrazuje souhrny souborů, obrázků, odkazů, slov, znaků a stylů použitých v knize. Každý řádek v hlášení je propojený. Poklepáním na řádek skočíte na místo v knize, kde je tato položka použita nebo definována (podle potřeby). Například v zobrazení *Odkazy* můžete poklepat na položky sloupce *Zdroj*, abyste skočili tam, kde je odkaz definován, a na položky ve sloupci *Cíl*, abyste skočili, kam odkaz ukazuje.



5.9 Speciální funkce v editoru kódu

Editor HTML v Calibre je velice výkonný. Má mnoho funkcí, díky kterým je úprava HTML (a CSS) jednodušší.

5.9.1 Zvýraznění syntaxe

Editor HTML má velice propracované zvýraznění syntaxe. Funkce zahrnují:

- Text uvnitř značek pro tučné písmo, kurzívu a nadpisy jsou tučné nebo kurzívou
- Jak přesouváte kurzor přes HTML, zvýrazní se odpovídající značky HTML jsou zvýrazněny a pomocí klávesových zkratek můžete přeskočit na otevírací nebo ukončovací značku `Ctrl+{` a **:kdb:`Ctrl+}`**. Podobně můžete vybrat obsah značky pomocí `Ctrl+Alt+T` nebo `Ctrl+Shift+T`.
- Neplatné HTML je zvýrazněno červeným podtržením
- Pravopisné chyby v textu uvnitř značek HTML a atributy, jako jsou názvy, jsou zvýrazněny. Kontrola pravopisu je schopná rozpoznat jazyk na základě hodnoty atributu `lang` aktuální značky a celkového jazyka knihy.
- CSS vložené uvnitř značek `<style>` je zvýrazněno
- Speciální znaky, které může být těžké rozlišit, jako jsou pevné mezery, různé druhy pomlček atd., jsou zvýrazněny.
- Odkazy na jiné soubory ve značkách `<a>`, `<img>` a `<link>` mají všechny zvýrazněné názvy souborů. Pokud odkazují na název souboru, který neexistuje, je název souboru označen červeným podtržením.

5.9.2 Nápověda závislá na kontextu

Pro získání nápovědy ke značce HTML nebo vlastnosti CSS můžete kliknout pravým tlačítkem myši na název této značky nebo vlastnosti.

Můžete také podržet klávesu `Ctrl` a kliknout na jakýkoliv název souboru uvnitř značky odkazu pro automatické otevření tohoto souboru v editoru. Podobně `Ctrl` a kliknutím na název třídy se dostanete k prvnímu pravidlu stylu, které odpovídá značce a třídě.

Kliknutí pravým tlačítkem na název třídy v souboru HTML vám umožní přejmenovat třídu a změnit tak všechny výskyty třídy v celé knize a ve všech jejích šablonách stylů.

5.9.3 Automatické dokončování

Při úpravě e-knihy je jedním z nejobtížnějších úkolů vytvořit odkazy na jiné soubory uvnitř knihy, šablony stylů CSS nebo obrázky. Musíte vyřešit správný název souboru a relativní cestu k souboru. Editor má automatické dokončování, aby toto zjednodušil.

Během psaní názvu souboru editor automaticky nabízí návrhy. Jednoduše použijte klávesu `Tab` pro výběr správného názvu souboru. Editor dokonce nabízí návrhy pro odkazy ukazující na kotvu uvnitř jiného souboru HTML. Po napsání znaku `#` zobrazí editor seznam všech kotev v cílovém souboru s malým úryvkem textu, aby vám pomohl zvolit správnou kotvu.

Pamatujte, že na rozdíl od většiny ostatních systémů dokončování, systém dokončování editoru používá podposloupnou shodu. To znamená, že pro dokončení názvu souboru můžete zadat jen dvě nebo tři písmena odkudkoliv z názvu souboru . Řekněme například, že když chcete název souboru `../images/arrow1.png`, můžete jednoduše napsat `ia1` a stisknout `Tab` pro dokončení názvu souboru. Při hledání shod systém dokončování upřednostňuje písmena, která jsou na začátku slova, nebo ihned po oddělovači cesty. Jakmile si zvyknete na tento systém, zjistíte, že vám ušetří spoustu času a úsilí.

5.9.4 Fragmenty

Editor e-knih Calibre podporuje *fragmenty*. Fragment je kus textu, který je buď často znovu použit nebo obsahuje hodně nadbytečného textu. Editor vám umožňuje vložit fragment pomocí jen několika stisků kláves. Fragmenty jsou velice výkonné, s mnoha funkcemi, jako jsou zástupné symboly, mezi kterými můžete přeskakovat, automatické zrcadlení opakované textu a tak dále. Pro více informací navštivte *Fragmenty* (stránka 106).

KAPITOLA 6

Server s obsahem Calibre

Server s obsahem Calibre vám umožňuje přístup k vašim knihovnám Calibre a čtení knih přímo v prohlížeči na vašem oblíbeném mobilním telefonu nebo tabletu. Následkem toho nemusíte do telefonu instalovat žádné speciální aplikace pro čtení a správu knih. Stačí použít prohlížeč. Server stahuje a ukládá knihu, kterou čtete, do off-line mezipaměti, takže ji můžete číst, i když nejste připojeni k internetu.

Obsah

- *Přístup k Serveru s obsahem z jiných zařízení* (stránka 114)
  - *Přístup k serveru ze zařízení ve vaší domácí síti* (stránka 114)
  - *Přístup k serveru odkudkoliv z internetu* (stránka 115)
- *Rozhraní serveru* (stránka 115)
  - *Seznam knih* (stránka 116)
  - *The book viewer* (stránka 116)
- *Podpora prohlížečů* (stránka 116)
- *Povolení offline podpory* (stránka 116)
- *Správa uživatelských účtů pouze z příkazového řádku* (stránka 117)
- *Integrace Serveru s obsahem Calibre do jiných serverů* (stránka 117)
  - *Použití plně virtuálního hostitele* (stránka 117)
  - *Používání předpony adresy URL* (stránka 118)
- *Vytváření služby pro server Calibre na moderním systému Linux* (stránka 119)

Pro spuštění serveru klikněte na tlačítko *Připojit nebo sdílet* a zvolte *Spustit Server s obsahem*. Z brány firewall počítače nebo antivirového programu se může zobrazit zpráva s dotazem, zda je v pořádku povolit přístup k souboru `calibre.exe`. Klikněte na tlačítko `Povolit` nebo `OK`. Pak v počítači otevřete prohlížeč (nejlépe Chrome nebo Firefox) a zadejte

následující adresu:

http://127.0.0.1:8080

This will open a page in the browser showing you your calibre libraries, click on any one and browse the books in it. Click on a book, and it will show you all the metadata about the book, along with buttons to *Read* and *Download* the book. Click the *Read* button to start reading the book.

Poznámka

Výše uvedená adresa `http: //127.0.0.1: 8080` bude fungovat pouze na počítači, kde je spuštěno Calibre. Pro přístup k serveru z jiných počítačů, telefonů, tabletů atd. budete muset udělat trochu více práce, jak je popsáno v následující části.

6.1 Přístup k Serveru s obsahem z jiných zařízení

Existují dva typy přístupu ze vzdáleného zařízení, které budete obvykle potřebovat. První, jednodušší typ, je z vaší domácí sítě. Pokud používáte Calibre na počítači ve vaší domácí síti a vaše ostatní zařízení jsou také připojena ke stejné domácí síti, měl by být přístup k serveru na těchto zařízeních snadný.

6.1.1 Přístup k serveru ze zařízení ve vaší domácí síti

Po spuštění serveru v Calibre, jak je popsáno výše, klikněte znovu na tlačítko *Připojit nebo sdílet*. Namísto akce *Spustit Server s obsahem* byste měli vidět akci *Zastavit Server s obsahem*. Vpravo od této akce bude uvedena IP adresa a číslo portu. Vypadají jako mnoho čísel oddělených tečkami. Například:

```
Stop Content server [192.168.1.5, port 8080]
```

Tato čísla vám sdělují, jakou adresu použít pro připojení k serveru ve vašem zařízení. Podle výše uvedeného příkladu bude adresa:

```
http://192.168.1.5:8080
```

První část adresy je vždy `http://` další část je adresa IP, což jsou čísla před čárkou a nakonec máme číslo portu, které musí být přidáno k IP adrese s dvojtečkou (`:`). Pokud máte štěstí, mělo by to být vše, co potřebujete, a můžete se dívat na knihovny Calibre na svém zařízení. Pokud ne, čtěte dál.

Odstraňování problémů s připojením v domácí síti

Pokud se vám nedaří připojit se k serveru z vašeho zařízení, zkuste následující kroky:

1. Zkuste, zda je server spuštěn, otevřením adresy `http://127.0.0.1:8080` v prohlížeči spuštěném na stejném počítači jako server.
2. Zkontrolujte, zda vaše brána firewall nebo antivirus umožňují připojení k vašemu počítači na portu `8080` a k programu Calibre. Nejjednodušší způsob, jak odstranit firewall nebo antivirus jako zdroj problémů, je dočasně je oba vypnout a pak se zkusit připojit. Před vypnutím brány firewall byste se měli nejprve odpojit od internetu, aby byl váš počítač v bezpečí.
3. Zkontrolujte, zda jsou vaše zařízení a počítač ve stejné síti. To znamená, že by oba měly být připojeny ke stejnému bezdrátovému směrovači. Zvláště nesmí být používán mobilní připojení nebo poskytovatel internetového připojení poskytující přímé WiFi připojení.
4. Pokud máte nestandardní nastavení sítě, může se stát, že je adresa IP uvedená v nabídce *Připojit nebo sdílet* nesprávná. V takovém případě budete muset sami zjistit, jaká je správná adresa IP. Bohužel, vzhledem k nekonečné rozmanitosti možných síťových konfigurací, není možné vám poskytnout přesný návod, jak to udělat.

5. Pokud máte nastaveno uživatelské jméno a heslo, zkuste to nejdříve bez nich, abyste zjistili, zda to způsobuje problémy. Některá zařízení s elektronickým inkoustem mají prohlížeče, které nezvládají ověřování. Někdy můžete tento problém vyřešit zahrnutím uživatelského jména a hesla do adresy URL, například: `http://uzivatelskejmeno:heslo@192.168.1.2:8080`.
6. Pokud si nebudete vědět rady, vždy můžete požádat o pomoc v calibre user forums[41].

6.1.2 Přístup k serveru odkudkoliv z internetu

Varování

Než to uděláte, měli byste na serveru zapnout ochranu uživatelským jménem a heslem, jinak bude mít k vašim knihám přístup kdokoliv na světě. Přejděte na *Předvolby → Sdílení → Sdílení po síti* a povolte volbu *Vyžadovat uživatelské jméno a heslo pro přístup k serveru s obsahem*.

Zatímco se konkrétní podrobnosti o nastavení přístupu k internetu liší v závislosti na konfiguraci sítě a typu používaného počítače, základní schéma je následující.

1. Zjistěte externí IP adresu počítače, na kterém chcete spustit server. Můžete to udělat tak, že v prohlížeči spuštěném na tomto počítači navštívíte stránku What is my IP address[42].
2. Pokud je počítač za směrovačem, povolte přesměrování portů na směrovači, aby přesměrovával port `8080` (nebo jakýkoliv port, na kterém se rozhodnete spustit Server s obsahem Calibre) do počítače.
3. Ujistěte se, že je server Calibre povolen ve všech firewallech a antivirových programech ve vašem počítači.
4. Nyní byste měli mít možnost přistupovat k serveru na jakémkoli zařízení připojeném k internetu pomocí adresy IP, kterou jste zjistili v prvním kroku. Pokud je například adresa IP, kterou jste zjistili, `123.123.123.123` a port, který používáte pro server Calibre, je `8080`, adresa, kterou použijete ve svém zařízení, bude: `http://123.123.123.123:8080`.
5. Volitelně použijte službu, jako je no-ip[43], abyste nastavili snadno zapamatovatelnou adresu, kterou chcete použít namísto adresy IP, kterou jste zjistili v prvním kroku.

Poznámka

Pro maximální zabezpečení byste také měli na Serveru s obsahem povolit protokol HTTPS. Můžete tak učinit buď přímo na serveru poskytnutím cesty k certifikátu HTTPS, který se má použít, v rozšířených volbách konfigurace pro server, nebo můžete nastavit reverzní proxy, jak je popsáno níže, pro použití existujícího nastavení HTTPS.

6.2 Rozhraní serveru

Rozhraní serveru je zjednodušená verze hlavního rozhraní Calibre, optimalizovaná pro použití s dotykovými obrazovkami. Na domovské obrazovce se zobrazují knihy, které právě čtete, a také možnost vybrat knihovnu Calibre, kterou chcete procházet. Server v Calibre vám poskytuje přístup ke všem vašim knihovnám, ne jen k jedné jako dříve.

[41] https://www.mobileread.com/forums/forumdisplay.php?f=166
[42] https://www.whatismyip.com/
[43] https://www.noip.com/free

6.2.1 Seznam knih

Seznam knih na serveru je jednoduchá mřížka obálek. Klepněte na obálku pro zobrazení podrobných metadat knihy nebo přečtení knihy. Pokud dáváte přednost podrobnějšímu seznamu, můžete změnit výchozí zobrazení kliknutím na tři svislé tečky v pravém horním rohu.

Řazení a vyhledávání v seznamu knih by mělo být uživatelům Calibre známé. Jsou přístupné kliknutím na jejich ikony v pravé horní oblasti. Obě pracují přesně stejně jako v hlavním programu Calibre. Stránka vyhledávání vám dokonce umožňuje vytvářet vyhledávací dotazy kliknutím na autory, štítky atd., stejně jako pomocí Prohlížeče štítků v hlavním programu.

V rozhraní serveru je také přítomna velmi oblíbená funkce hlavního programu *Virtuální knihovny*. Klikněte na tři svislé tečky v pravém horním rohu pro výběr Virtuální knihovny.

6.2.2 The book viewer

You can read any book in your calibre library by simply tapping on it and then tapping the *Read* button. The book viewer is very simple to operate. You can both tap and swipe to turn pages. Swiping up/down skips between chapters. Tapping the top quarter of the screen gets you the detailed controls and viewer preferences.

Pokud ponecháte Server s obsahem spuštěný, můžete dokonce otevřít stejnou knihu na více zařízeních a bude si pamatovat poslední pozici čtení. Pokud ne, můžete vynutit synchronizaci klepnutím na v horní čtvrtinu obrazovky a výběrem *Synchronizovat*.

6.3 Podpora prohlížečů

Nový server Calibre využívá mnoho pokročilých funkcí HTML 5 a CSS 3. Proto vyžaduje použití aktuálního prohlížeče. Byl testován na Chrome pro Android a Safari pro iOS, stejně jako Chrome a Firefox pro počítač.

Server dbá na to, aby používal funkce, které buď již byly standardizovány, nebo jsou na cestě ke standardizaci. Pokud tedy v současné době nefunguje s vaším oblíbeným prohlížečem, pravděpodobně jednou bude, až ten prohlížeč dožene ztrátu.

Pokud používáte obzvláště zastaralý nebo omezený prohlížeč nebo nechcete spouštět JavaScript, můžete použít *mobilní* zobrazení jednoduše přidáním `/mobile` k adrese serveru.

Poznámka

Apple na iOS umožňuje pouze jeden modul prohlížeče, takže Firefox, Chrome a Safari jsou vlastně pod kapotou stejný prohlížeč. Nové rozhraní serveru vyžaduje iOS 10.3.2 nebo novější. Na Androidu byl server testován s Chrome verze 58 a novější.

6.4 Povolení offline podpory

Browser makers have been trying to force people to use SSL by disabling advanced features in their browsers for plain HTTP connections. One such casualty is ApplicationCache, which was what was used in calibre for offline support. As a result now-a-days sadly, offline mode works only as long as you keep the browser tab open. In addition, in Firefox on Android, you will need to type `about:config` and create a preference called `browser.tabs.useCache` and set it to `true`.

6.5 Správa uživatelských účtů pouze z příkazového řádku

Program Calibre má v *Předvolby* pěknou sekci, která vám umožní spravovat uživatelské účty pro server. Pokud však chcete spustit samostatný server a nemůžete spustit hlavní program Calibre na stejném počítači nebo pod stejným uživatelským účtem, můžete také spravovat uživatele pouze pomocí příkazového řádku.

Uživatelské účty můžete spravovat pomocí volby `--manage-users` v samostatném programu `calibre-server`. Předpokládejme, že chcete uložit databázi uživatelů do složky `/srv/calibre`, pak ji vytvoříte spuštěním:

```
calibre-server --userdb /srv/calibre/users.sqlite --manage-users
```

Postupujte podle pokynů a vytvořte uživatelské účty, nastavte jejich oprávnění atd. Po dokončení můžete spustit server takto:

```
calibre-server --userdb /srv/calibre/users.sqlite --enable-auth
```

Použije uživatelské účty, které jste vytvořili v předchozím kroku.

6.6 Integrace Serveru s obsahem Calibre do jiných serverů

Zde vám ukážeme, jak integrovat Server s obsahem Calibre do jiného serveru. Nejčastějším důvodem je využití SSL nebo aby knihovna Calibre sloužila jako součást větší stránky. Základní technikou je spuštění serveru Calibre a nastavení reverzní proxy tomuto serveru z hlavního serveru.

Reverzní proxy je, když váš normální server přijímá příchozí požadavky a předává je serveru Calibre. Pak přečte odpověď ze serveru Calibre a předá ji klientovi. To znamená, že můžete jednoduše spustit server Calibre, aniž byste se ho snažili integrovat s hlavním serverem.

6.6.1 Použití plně virtuálního hostitele

Nejjednodušší konfigurací je vyhradit pro server Calibre plně virtuálnho hostitele. V takovém případě spusťte server Calibre takto:

```
calibre-server
```

Nyní nastavte virtuálního hostitele na hlavním serveru, například pro nginx:

```
http {
    client_max_body_size 64M;  # needed to upload large books
}

server {
    listen [::]:80;
    server_name myserver.example.com;

    location / {
        proxy_pass http://127.0.0.1:8080;
    }
}
```

Nebo pro Apache:

```
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_http_module modules/mod_proxy_http.so
```

(continues on next page)

(pokračujte na předchozí stránce)

```
<VirtualHost *:80>
    ServerName myserver.example.com
    AllowEncodedSlashes On
    ProxyPreserveHost On
    ProxyPass "/"  "http://localhost:8080/"
</VirtualHost>
```

6.6.2 Používání předpony adresy URL

Pokud nechcete pro Calibre vyhradit plně virtuálního hostitele, můžete použít předponu adresy URL. Spusťte server Calibre takto:

```
calibre-server --url-prefix /calibre --port 8080
```

Klíčovým parametrem je zde `--url-prefix /calibre`. Způsobí, že Server s obsahem bude sloužit všem adresám URL s předponou `/calibre`. Abyste to viděli v akci, navštivte ve svém prohlížeči adresu `http://localhost:8080/calibre`. Měli byste vidět normální webovou stránku Serveru s obsahem, ale nyní bude spuštěna pod `/calibre`.

S nginx je vyžadována konfigurace:

```
http {
    client_max_body_size 64M;  # needed to upload large books
}

proxy_set_header X-Forwarded-For $remote_addr;
location /calibre/ {
    proxy_buffering off;
    proxy_pass http://127.0.0.1:8080$request_uri;
}
location /calibre {
    # we need a trailing slash for the Application Cache to work
    rewrite /calibre /calibre/ permanent;
}
```

Pro Apache nejdříve povolte moduly proxy v Apache přidáním následujícího textu do `httpd.conf`:

```
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_http_module modules/mod_proxy_http.so
```

Přesná technika pro povolení modulů proxy se bude lišit v závislosti na instalaci Apache. Jakmile povolíte moduly proxy, přidejte následující pravidla do `httpd.conf` (nebo pokud používáte virtuální hostitele, tak do konfiguračního souboru pro daného virtuálního hostitele):

```
AllowEncodedSlashes On
RewriteEngine on
RewriteRule ^/calibre/(.*) http://127.0.0.1:8080/calibre/$1 [proxy]
RedirectMatch permanent ^/calibre$ /calibre/
```

To je vše, nyní budete mít přístup k Serveru s obsahem Calibre pro URL `/calibre` na vašem hlavním serveru. Výše uvedená pravidla předávají všechny požadavky pod `/calibre` serveru Calibre běžícímu na portu 8080 a díky výše uvedené volbě `--url-prefix` je server Calibre zpracovává transparentně.

Poznámka

Pokud používáte reverzní proxy, měli byste říct Serveru s obsahem Calibre, aby naslouchal pouze na localhost pomocí `--listen-on 127.0.0.1`. Tímto způsobem bude server naslouchat pouze spojením přicházejícím ze stejného počítače, tj. z reverzní proxy.

Poznámka

Pokud máte pro váš hlavní server nastaveno SSL, měli byste říci serveru Calibre, aby používal základní ověřování namísto ověřování algoritmem digest, protože je rychlejší. Abyste to udělali, předejte volbu `--auth-mode=basic` pro `calibre-server`.

6.7 Vytváření služby pro server Calibre na moderním systému Linux

Můžete snadno vytvořit službu, která spustí Calibre při spuštění moderního systému založeného na Linuxu (systemd[44]). Stačí vytvořit soubor `/etc/systemd/system/calibre-server.service` s níže zobrazeným obsahem:

```
[Unit]
Description=calibre Content server
After=network.target

[Service]
Type=simple
User=mylinuxuser
Group=mylinuxgroup
ExecStart=/opt/calibre/calibre-server "/path/to/calibre library folder"

[Install]
WantedBy=multi-user.target
```

Změňte `mylinuxuser` a `mylinuxgroup` na libovolného uživatele a skupinu, kterou má server spouštět. Měli by to být stejný uživatel a skupina, která vlastní soubory ve složce knihovny Calibre. Pamatujte, že není obecně dobrý nápad spouštět server jako root. Také změňte cestu ke složce knihovny Calibre, aby vyhovovala vašemu systému. V případě potřeby můžete přidat více knihoven. Viz nápověda k příkazu `calibre-server`.

Nyní spusťte:

```
sudo systemctl start calibre-server
```

pro spuštění serveru. Zkontrolujte jeho stav pomocí:

```
sudo systemctl status calibre-server
```

Aby se spustil při spouštění, spusťte:

```
sudo systemctl enable calibre-server
```

[44] https://www.freedesktop.org/wiki/Software/systemd/

Poznámka

Server Calibre *nepotřebuje* spuštěný server X, ale potřebuje mít nainstalované knihovny X, protože některé komponenty, které používá, na ně odkazují.

Poznámka

Server Calibre také podporuje aktivaci systemd socketu, takže ji můžete použít, pokud je to potřeba.

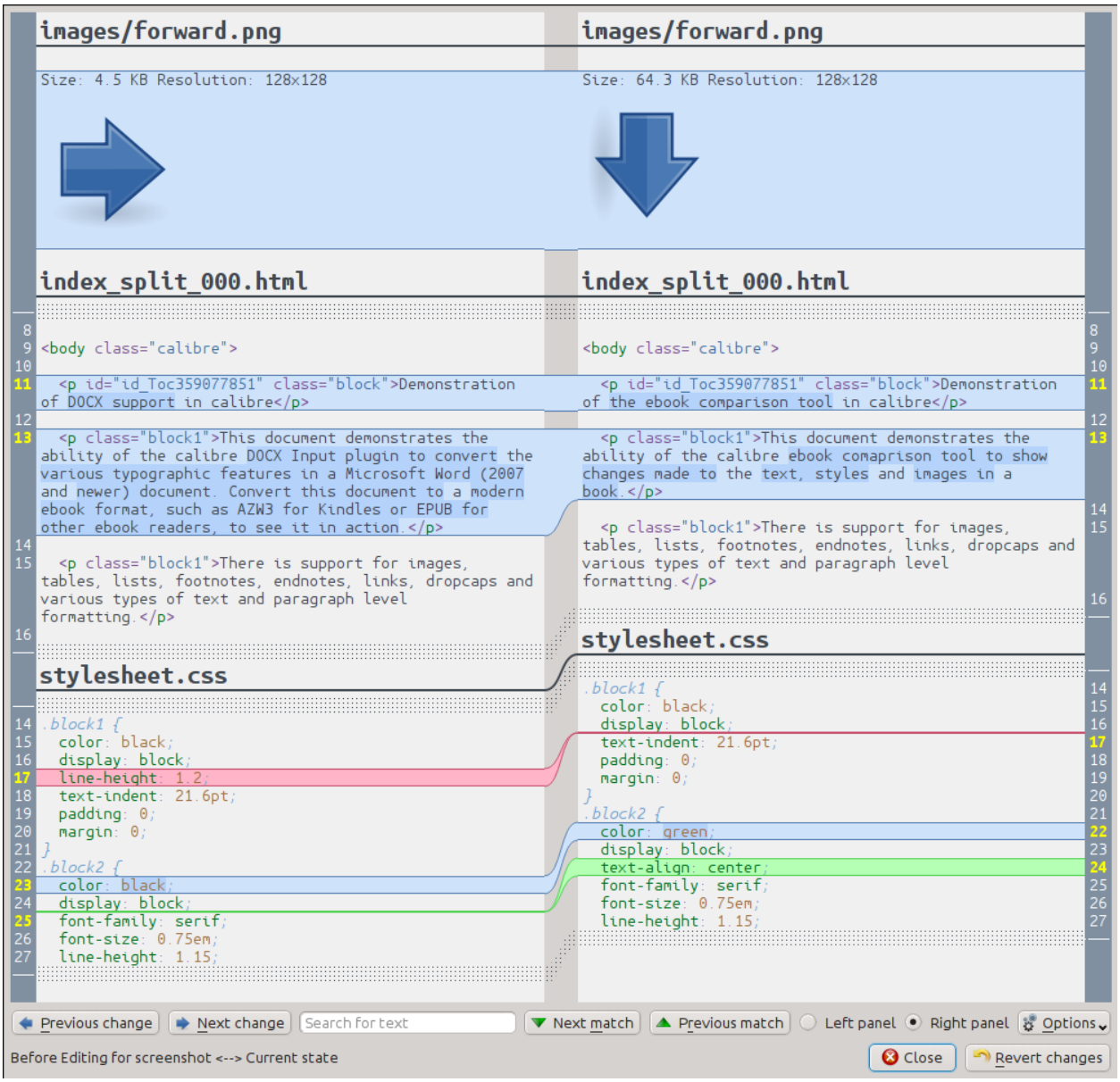
KAPITOLA 7

Porovnání e-knih

Calibre zahrnuje integrovaný nástroj porovnání e-knih, který může být použit pro zobrazení, co se změnilo uvnitř e-knihy po úpravě nebo převodu. Umí porovnat knihy ve formátech EPUB a AZW3.

Pro jeho použití buď otevřete e-knihu v nástroji pro *Úprava e-knih* (stránka 79) a klikněte na *Soubor → Porovnat s jinou knihou* nebo použijte panel *Podrobnosti o knize* (stránka 20). Pokud provádíte převod z EPUB do EPUB, původní soubor EPUB bude uložen jako ORIGINAL_EPUB. Jednoduše klikněte pravým tlačítkem myši na položku ORIGINAL_EPUB v panelu Podrobnosti o knize a zvolte *Porovnat s formátem EPUB*.

Nástroj porovnání, který se otevře, bude vypadat jako snímek obrazovky níže. Ukáže vám rozdíly v textu, stylech a obrázcích ve zvolených knihách.



7.1 Pochopení zobrazení porovnání

Jak můžete vidět na snímku obrazovky výše, zobrazení porovnání ukazuje rozdíly mezi dvěma knihami vedle sebe. Jsou zobrazeny pouze rozdíly s několika řádky kontextu kolem nich. Tak můžete snadno vidět na první pohled pouze to, co se změnilo uvnitř velkého dokumentu, jako je kniha.

Přidaný text je zobrazen se zeleným pozadím, odebraný text s červeným pozadím a změněný text s modrým pozadím.

Čísla řádků všeho změněného textu jsou zobrazena po stranách, umožňují snadno přejít na konkrétní změnu v editoru. Když otevřete nástroj porovnání z editoru, můžete také poklepat na řádek v pravém panelu pro automatické přejití na tento řádek v editoru.

Užitečná technika při porovnávání knih je říct nástroji porovnání, aby zkrášlil textové soubory a soubory stylů, než vypočítá rozdíly. To může mít často za výsledek čistější a snadněji sledovatelné rozdíly. Abyste to udělali, klikněte na tlačítko *Volby* vpravo dole a zvolte *Zkrášlit soubory před jejich porovnáním*. Pamatujte, že zkrášlení může mít někdy nežádoucí účinky, protože může způsobit, že jsou neplatné značky změněny, aby byly platné. Pomocí tlačítka *Volby* můžete také

změnit počet řádků kontextu zobrazovaného kolem rozdílů.

Prostřednictvím Panelu hledání v dolní části můžete hledat jakýkoliv text v rozdílech. Budete muset zadat, ve kterém panelu hledat, *Levý* nebo *Pravý*.

7.2 Spuštění nástroje porovnání

Nástroj porovnání je nejužitečnější, když máte dvě verze stejné knihy a chcete vidět, jaký je mezi nimi rozdíl. Za tímto účelem je několik způsobů spuštění nástroje.

7.2.1 Porovnání dvou souborů e-knih

Otevřete první soubor v nástroji *Úprava e-knih* (stránka 79). Nyní klikněte na *Soubor → Porovnat s jinou knihou* a zvolte druhý soubor (musí být ve stejném formátu jako první). Otevře se zobrazení porovnání s upravovaným souborem vpravo a druhým souborem vlevo.

7.2.2 Porovnání ORIGINAL_FMT s FMT

Když provádíte v calibre převod z FMT do něho samého, původní soubor je uložen jako ORIGINAL_FMT. Co se změnilo převodem, můžete zobrazit kliknutím pravým tlačítkem myši na položku ORIGINAL_FMT v panelu *Podrobnosti o knize* (stránka 20) v hlavním okně calibre a výběrem *Porovnat s FMT*. Otevře se zobrazení porovnání s ORIGINAL_FMT vlevo a FMT vpravo.

7.2.3 Porovnání kontrolního bodu s aktuálním stavem knihy během úpravy

Nástroj *Úprava e-knih* (stránka 79) má velice užitečnou funkci zvanou *Kontrolní body* (stránka 91). Ta vám umožňuje uložit aktuální stav knihy jako pojmenovaný *kontrolní bod*, do kterého se můžete vrátit, pokud se vám nelíbí změny, které jste udělali od vytvoření kontrolního bodu. Kontrolní body jsou také vytvářeny automaticky, když provedete různé automatizované akce v editoru. Seznam kontrolních bodů můžete vidět přejitím na *Zobrazení → Kontrolní body* a pak použitím tlačítka *Porovnat* pro porovnání knihy ve vybraném kontrolním bodu s aktuálním stavem. Nástroj porovnání zobrazí kontrolní bod vlevo a aktuální stav vpravo.

KAPITOLA 8

Úprava metadat e-knihy

Obsah

- *Úprava metadat jedné knihy najednou* (stránka 125)
  - *Stahování metadat* (stránka 126)
  - *Správa formátů knihy* (stránka 126)
  - *Vše o obálkách* (stránka 126)
- *Úprava metadat mnoha knih najednou* (stránka 126)
  - *Hledat a nahradit* (stránka 126)
  - *Hromadné stahování metadat* (stránka 127)
- *Přidání dodatečných souborů dat do knihy* (stránka 127)

E-knihy se objevují ve všech tvarech a velikostech a spíše častěji než ne jsou jejich metadata (věci, jako jsou název, autor, série, vydavatel) neúplná nebo nesprávná. Nejjednodušší způsob, jak změnit metadata v calibre, je jednoduše poklepat na položku a zadat správné nahrazení. Pro propracovanější „výkonné úpravy“ použijte nástroje pro úpravu metadat probrané níže.

8.1 Úprava metadat jedné knihy najednou

Klikněte na knihu, kterou chcete upravit, a pak klikněte na tlačítko *Upravit metadata* nebo stiskněte klávesu E. Otevře se dialogové okno, které vám umožní upravit všechny aspekty metadat. Má různé funkce, aby byly úpravy rychlejší a efektivnější. Seznam běžně používaných tipů:

- Můžete kliknout na tlačítko mezi názvem a autory pro jejich automatické prohození.
- Můžete kliknout na tlačítko vedle řazení autora, aby ho calibre automaticky vyplnilo pomocí hodnot řazení uložených u každého autora. Použije dialogové okno *Spravovat autory*, abyste zobrazili a změnili hodnoty řazení autorů. Toto dialogové okno lze otevřít kliknutím a podržením tlačítka vedle řazení autora.

- Můžete kliknout na tlačítko vedle štítků pro použití *Editoru štítků* pro správu štítků přidružených ke knize.
- Pole „Identifikátory“ může být použito pro zadání ISBN (a mnoho jiných typů identifikátorů), bude mít červené pozadí, pokud zadáte neplatné ISBN. Pro platná ISBN bude zelené.
- Pole řazení autora bude červené, pokud se hodnota řazení autora liší od té, kterou calibre předpokládá.

8.1.1 Stahování metadat

Nejhezčí funkce dialogového okna úpravy metadat je jeho schopnost automaticky vyplnit mnoho polí metadat získáním metadat z různých webových stránek. Calibre aktuálně používá Knihy Google a Amazon. Stažená metadata za vás mohou vyplnit název, autora, sérii, štítky, hodnocení, popis a ISBN.

Pro použití stahování vyplňte pole název a autor a klikněte na tlačítko *Stáhnout metadata*. calibre vám nabídne seznam knih, které nejvíce odpovídají názvu a autorovi. Pokud vyplníte nejdříve pole ISBN, bude použito přednostně před názvem a autorem. Pokud nejsou nalezeny žádné shody, zkuste učinit hledání trochu méně konkrétní zahrnutím pouze některých klíčových slov v názvu a použitím pouze příjmení autora.

8.1.2 Správa formátů knihy

V calibre může mít jedna kniha přidruženo mnoho různých *formátů*. Například jste mohli získat kompletní shakespearovo dílo ve formátu EPUB a později ho převést do MOBI, abyste ho mohli číst na Kindle. calibre pro vás automaticky spravuje více formátů. V oddílu *Dostupné formáty* dialogového okna Upravit metadata můžete tyto formáty spravovat. Můžete přidat nový formát, odstranit existující formát a také požádat calibre o nastavení metadat a obálky pro položku knihy z metadat v jednom z formátů.

8.1.3 Vše o obálkách

Můžete požádat calibre, aby pro vás stáhlo obálky knih, za předpokladu, že kniha má známé ISBN. Případně můžete zadat soubor ve svém počítači, který chcete použít jako obálku. calibre pro vás může dokonce vygenerovat výchozí obálku se základními metadaty. Můžete přetahovat obrázky na obálku, abyste ji změnili, a také kliknutím pravým tlačítkem myši kopírovat a vložit obrázky obálky.

Kromě toho tu je tlačítko pro automatické oříznutí rámečků obálky v případě, že má vaše obálka ošklivý rámeček.

8.2 Úprava metadat mnoha knih najednou

Nejdříve vyberte knihy, které chcete upravit, podržením `Ctrl` or `Shift` a kliknutím na ně. Pokud vyberete více než jednu knihu, kliknutí na tlačítko *Upravit metadata* způsobí otevření dialogového okna *hromadné* úpravy metadat. Pomocí tohoto dialogového okna můžete rychle nastavit autora, vydavatele, hodnocení, štítky, sérii atd. mnoha knih na stejnou hodnotu. To je obzvlášť užitečné, pokud jste právě importovali řadu knih, které mají některá společná metadata. Toto dialogové okno je velice výkonné, například má kartu *Hledat a nahradit*, kterou můžete použít k provedení hromadných operací na metadatech a dokonce i ke kopírování metadat z jednoho sloupce do jiného.

Normální dialogové okno úpravy metadat má také tlačítka *Další* a *Předchozí*, která můžete použít k úpravě metadat několika knih jednu po druhé.

8.2.1 Hledat a nahradit

Dialogové okno *Upravit metadata pro hodně knih* vám umožňuje provádět libovolně výkonné operace hledání a nahrazení na vybraných knihách. Ve výchozím nastavení používá jednoduché textové hledání a nahrazení, ale také podporuje *regulární výrazy*. Pro více informací o regulárních výrazech se podívejte na *Vše o používání regulárních výrazů v Calibre* (stránka 214).

Jak již bylo zmíněno výše, existují dva režimy hledání a nahrazení: znaková shoda a regulární výraz. Znaková shoda bude hledat ve *Vyhledávacím poli*, které zvolíte, znaky, které zadáte do pole *Hledat*, a nahradí tyto znaky tím, co zadáte do pole

Nahradit čím. Bude nahrazen každý výskyt hledaných znaků. Předpokládejme například, že vyhledávací pole obsahuje *o zlobivé kočičce*. Pokud hledáte *o*, které má být nahrazeno *AHOJ*, pak výsledek bude *AHOJ zlAHOJbivé kAHOJčičce*.

Pokud pole, ve kterém hledáte, je *vícenásobné* pole, jako jsou štítky, pak se každý štítek posuzuje samostatně. Například pokud vaše štítky obsahují *Horor, Strašidelné*, vyhledávací výraz *r,* nebude odpovídat ničemu, protože výraz bude nejdříve použit na *Horor* a pak na *Strašidelné*.

Pokud chcete, aby hledání ignorovalo rozdíly mezi velkými a malými písmeny, zrušte zaškrtnutí pole *Rozlišovat velikost písmen*.

Můžete nechat calibre, aby změnilo velikost písmen výsledku (informace po provedení nahrazení) výběrem jedné z funkcí z pole *Použít funkci po nahrazení*. Dostupné operace jsou:

- *Malá písmena* – změnit všechny znaky v poli na malá písmena
- *Velká písmena* – změnit všechny znaky v poli na velká písmena
- *Všechna první velká* – převést každé slovo ve výsledku na velká počáteční písmena

Pole *Váš test* je vám k dispozici pro zadání textu, abyste si ověřili, že hledání a nahrazení dělá to, co chcete. Ve většině případů budou stačit testovací pole knih, ale je možné, že existuje případ, který chcete zkontrolovat, a který není zobrazen v těchto polích. Zadejte tento případ do *Váš test*.

Režim regulární výraz má některé rozdíly od znakového režimu, kromě (samozřejmě) používání regulárních výrazů. První je, že funkce jsou použity na části řetězce odpovídající hledanému řetězci, nikoliv na celá pole. Druhý je, že se funkce použijí na řetězec nahrazení, nikoliv na celé pole.

Třetí a nejdůležitější je, že řetězec nahrazení může odkazovat na části hledaného řetězce pomocí zpětných odkazů. Zpětný odkaz je `\\n`, kde n je celé číslo, které odkazuje na ntou skupinu v závorkách ve vyhledávacím výrazu. Například, se stejným příkladem jako výše, *o zlobivé kočičce*, vyhledávací výraz *o (…) (…)* a výraz nahrazení *o \2 \1*, bude výsledek *o kočičce zlobivé*. Pro více informací o zpětných odkazech se podívejte na *Vše o používání regulárních výrazů v Calibre* (stránka 214).

Jeden užitečný vzor: předpokládejme, že chcete změnit velikost písmen celého pole. Nejsnadnější způsob, jak to udělat, je použít znakový režim, ale předpokládejme dále, že chcete použít režim regulárního výrazu. Vyhledávací výraz by měl být *(^.*$)*, výraz nahrazení by měl být *\1* a měla by být vybrána požadovaná funkce změny velikosti písmen.

A konečně, v režimu regulárního výrazu můžete kopírovat hodnoty z jednoho pole do druhého. Stačí, aby se zdrojové a cílová pole lišilo. Kopie může nahradit cílové pole, předřadit se poli (přidat na začátek) nebo se připojit k poli (přidat na konec). Zaškrtávací políčko ‚použít čárku' říká calibre, aby přidalo (nebo nepřidalo) čárku mezi text a cílové pole v režimech předřazení a připojení. Pokud je cíl vícenásobný (např. štítky), pak nemůžete zrušit zaškrtnutí tohoto políčka.

Hledat a nahradit je provedeno po použití všech ostatních změn metadat na ostatních kartách. To může vést k určitému zmatku, protože testovací pole zobrazí informace před ostatními změnami, ale operace bude použita po ostatních změn. Pokud máte jakékoliv pochybnosti o tom, co se stane, nemíchejte hledání a nahrazení s ostatními změnami.

8.2.2 Hromadné stahování metadat

Pokud chcete stáhnout metadata pro více knih najednou, klikněte pravým tlačítkem myši na tlačítko *Upravit metadata* a vyberte *Stáhnout metadata*. Můžete si zvolit stáhnout pouze metadata, pouze obálky nebo obojí.

8.3 Přidání dodatečných souborů dat do knihy

Calibre může ke knize uložit libovolný počet dodatečných souborů dat. Mohou to být alternativní obálky, doplňkový materiál atd. Nelze je přímo prohlížet ani používat jako zdroje pro převod. Nejsou ani indexovány fulltextovým vyhledávačem v Calibre. Pokud je chcete zobrazit, přidat nebo odstranit, vyberte knihu a klikněte pravým tlačítkem myši na tlačítko *Upravit metadata* a zvolte *Soubory dat*. Tím se zobrazí okno, ve kterém můžete s těmito soubory provádět operace. Alternativně můžete kliknout pravým tlačítkem myši na tlačítko *Přidat knihy* a zvolit *Přidat soubory dat k vybraným záznamům knihy* pro rychlejší přidání souborů dat.

KAPITOLA 9

Nejčastější dotazy

Obsah

- *Převod formátu e-knihy* (stránka 129)
- *Integrace zařízení* (stránka 132)
- *Library management* (stránka 139)
- *Různé* (stránka 143)

9.1 Převod formátu e-knihy

Obsah

- *Ze kterých a do kterých formátů podporuje Calibre převod?* (stránka 130)
- *Jaké jsou nejlepší zdrojové formáty pro převod?* (stránka 130)
- *Převedl jsem soubor PDF, ale výsledek má různé problémy?* (stránka 130)
- *Jak převedu můj soubor obsahující neanglické znaky nebo chytré uvozovky?* (stránka 130)
- *O co jde s obsahem v souborech MOBI?* (stránka 130)
- *Jak převedu kolekci souborů HTML v konkrétním pořadí?* (stránka 131)
- *EPUB vytvořený pomocí Calibre není platný?* (stránka 132)
- *Jak použiji některé pokročilé funkce nástrojů převodu?* (stránka 132)

9.1.1 Ze kterých a do kterých formátů podporuje Calibre převod?

Calibre podporuje převod mnoha vstupních formátů do mnoha výstupních formátů. Umí převést každý vstupní formát z následujícího seznamu do každého výstupního formátu.

Input Formats: AZW, AZW3, AZW4, CBZ, CBR, CB7, CBC, CHM, DJVU, DOCX, EPUB, FB2, FBZ, HTML, HTMLZ, KEPUB, LIT, LRF, MOBI, ODT, PDF, PRC, PDB, PML, RB, RTF, SNB, TCR, TXT, TXTZ

Output Formats: AZW3, EPUB, DOCX, FB2, HTMLZ, KEPUB, OEB, LIT, LRF, MOBI, PDB, PMLZ, RB, PDF, RTF, SNB, TCR, TXT, TXTZ, ZIP

Poznámka

PRC je všeobecný formát, Calibre podporuje soubory PRC s hlavičkami TextRead a MOBIBook. PDB je také všeobecný formát. Calibre podporuje soubory PDB eReader, Plucker (pouze vstup), PML a zTxt. Podpora DJVU je pouze pro převod souborů DJVU, které obsahují vložený text. Ty jsou obvykle generovány programy pro OCR. Knihy MOBI mohou být dvou typů, Mobi6 a KF8. Calibre plně podporuje oba. Soubory MOBI mají často přípony .azw nebo .azw3. Soubory DOCX z Microsoft Word 2007 a novější jsou podporovány.

9.1.2 Jaké jsou nejlepší zdrojové formáty pro převod?

In order of decreasing preference: LIT, MOBI, AZW, EPUB, KEPUB, AZW3, FB2, FBZ, DOCX, HTML, PRC, ODT, RTF, PDB, TXT, PDF

9.1.3 Převedl jsem soubor PDF, ale výsledek má různé problémy?

PDF je hrozný formát, ze kterého převádět. Seznam různých problémů, se kterými se setkáte při převodu PDF, naleznete v *Převod dokumentů PDF* (stránka 75).

9.1.4 Jak převedu můj soubor obsahující neanglické znaky nebo chytré uvozovky?

Existují dva aspekty tohoto problému:

1. Znalost kódování zdrojového souboru: Calibre se pokusí odhadnout, jaké kódování znaků vaše zdrojové soubory používají, ale často je to nemožné, takže mu musíte říct, jaké kódování použít. To lze provést v grafickém rozhraní prostřednictvím pole *Vstupní kódování znaků* v oddílu *Vzhled a chování → Text* dialogového okna převodu. Nástroje příkazového řádku mají volbu `ebook-convert-txt-input --input-encoding`.
2. Při přidávání souborů HTML do Calibre možná budete muset říct Calibre, v jakém kódování soubory jsou. Abyste to udělali, přejděte do *Předvolby → Rozšířené → Moduly → Typ souboru* a přizpůsobte modul *HTML do ZIP* a řekněte mu, v jakém kódování jsou vaše soubory HTML. Nyní, když přidáte do Calibre soubory HTML, budou správně zpracovány. Soubory HTML z různých zdrojů mají často různá kódování, takže možná budete muset změnit toto nastavení opakovaně. Běžné kódování mnoha souborů z webu je `cp1252` a navrhuji, abyste ho vyzkoušeli jako první. Pamatujte, že při převodu souborů HTML máte výše zmíněné nastavení vstupního kódování ponechat prázdné. Je to proto, že modul *HTML do ZIP* automaticky převede soubory HTML na standardní kódování (utf-8).

9.1.5 O co jde s obsahem v souborech MOBI?

Nejprve si musíte uvědomit, že většina e-knih má dva obsahy. Jedním z nich je obvyklý obsah, který najdete v papírových knihách. Tento obsah je součástí hlavního dokumentu a může mít jakýkoliv styl. Tato obsah se nazývá *obsahový obsah*.

Pak existuje *metadatový obsah*. Metadatový obsah je obsah, který není součástí textu knihy a je obvykle přístupný některým speciálním tlačítkem ve čtečce. Například v Prohlížeči e-knih Calibre použijete tlačítko Zobrazit obsah, abyste viděli tento obsah. U tohoto obsahu nemůže tvůrce knihy měnit styl. Jak je zobrazen, závisí na programu prohlížeče.

In the MOBI format, the situation is a little confused. This is because the MOBI format, alone amongst mainstream e-book formats, *does not* have decent support for a metadata ToC. A MOBI book simulates the presence of a metadata ToC by putting an *extra* content ToC at the end of the book. When you click Go to Table of Contents on your Kindle, it is to this extra content ToC that the Kindle takes you.

Může se vám zdát, že kniha MOBI má dva totožné obsahy. Nezapomeňte, že jeden je sémanticky obsahový obsah a druhý je metadatový obsah, přestože oba mohou mít přesně stejné položky a vypadat stejně. Jeden může být přístupný přímo z nabídky Kindle, druhý ne.

Při převodu do MOBI Calibre rozpoznává ve vstupním dokumentu *metadatový obsah* a generuje ve výstupním souboru MOBI obsah na konci souboru. Toto můžete vypnout volbou v nastavení Výstup MOBI. Můžete také říct Calibre, zda ho má umístit na začátek nebo konec knihy prostřednictvím volby v nastavení Výstup MOBI. Nezapomeňte, že tento obsah je sémanticky *metadatový obsah*, v jakémkoliv jiném formátu než MOBI *nemůže být součástí textu*. Skutečnost, že v MOBI je součástí textu, je nehoda způsobená omezeními MOBI. Pokud chcete mít obsah v konkrétním místě v textu vašeho dokumentu, vytvořte ho ručně. Důrazně doporučujeme, abyste výchozí ponechali tak, jak je, tj. s metadatovým obsahem na konci knihy. Pamatujte také, že pokud zakážete generování obsahu na konci souboru, výsledný soubor MOBI nemusí na Kindle správně fungovat, protože Kindle používá metadatový obsah pro mnoho věcí, včetně funkce překlápění stránky.

Pokud máte ve vstupním dokumentu ručně upravený obsah, můžete v Calibre použít možnost rozpoznání obsahu, které z něj automaticky vygeneruje metadatový obsah. Více informací o tom, jak tyto volby používat, naleznete v oddílu o převodu v uživatelské příručce.

Nakonec vám doporučuji, abyste ve svých e-knihách vynechali obsahový obsah a měli pouze metadatový obsah. Metadatové obsahy poskytnou lidem čtoucím vaše e-knihy mnohem lepší navigaci (kromě Kindle, kde jsou v podstatě stejné jako obsahový obsah).

Poznámka

Novější formát AZW3 má správnou podporu pro metadatový obsah. Firmware Kindle však má sklon k selhání, pokud zakážete generování vloženého obsahu na konci souboru. Takže je doporučeno, abyste nechali vygenerovaný obsah být. Pokud vytvoříte soubor AZW3 s metadatovým obsahem a žádným vygenerovaným obsahem na konci souboru, některé funkce v Kindle nebudou fungovat, jako je funkce překlápění stránek.

9.1.6 Jak převedu kolekci souborů HTML v konkrétním pořadí?

Abyste převedli kolekci souborů HTML v konkrétním pořadí, musíte vytvořit soubor s obsahem. To znamená další soubor HTML, který obsahuje odkazy na všechny ostatní soubory v požadovaném pořadí. Takový soubor vypadá takto:

```
<html>
   <body>
     <h1>Table of Contents</h1>
     <p style="text-indent:0pt">
        <a href="file1.html">First File</a><br/>
        <a href="file2.html">Second File</a><br/>
        .
        .
        .
     </p>
   </body>
</html>
```

Pak jednoduše přidejte tento soubor HTML do grafického rozhraní a použijte tlačítko *Převést* pro vytvoření vaší elektronické knihy. Pro ovládání toho, jak je obsah generován, můžete použít volbu v oddílu Obsah v dialogovém okně převodu.

Poznámka

Ve výchozím nastavení při přidávání souborů HTML Calibre sleduje odkazy v souborech v pořadí *do hloubky*. To znamená, že pokud soubor A.html odkazuje na B.html a C.html a D.html, ale B.html také odkazuje na D.html, pak budou soubory v pořadí A.html, B.html, D.html, C.html. Pokud místo toho chcete, aby pořadí bylo A.html, B.html, C.html, D.html, pak musíte říct Calibre, aby přidalo vaše soubory v pořadí *do šířky*. Udělejte to přejitím na *Předvolby → Rozšířené → Moduly → Typ souboru* a úpravou modulu *HTML do ZIP*.

9.1.7 EPUB vytvořený pomocí Calibre není platný?

Calibre nezaručuje, že jím vytvořený EPUB je platný. Jedinou zárukou je to, že pokud mu dodáte platné XHTML 1.1 + CSS 2.1, bude výstupem platný EPUB. Calibre se snaží zajistit, aby soubory EPUB, které vytváří, skutečně fungovaly tak, jak je zamýšleno na široké škále zařízení, což je cíl, který není kompatibilní s vytvářením platných souborů EPUB, a to je mnohem důležitější pro drtivou většinu jeho uživatelů. Pokud potřebujete nástroj, který vždy vytváří platné soubory EPUB, Calibre není pro vás. To znamená, že pokud chcete poslat EPUB vytvořený v Calibre do online úložiště používajícího kontrolu platnosti EPUB, musíte se sami ujistit, že je EPUB platný, Calibre to za vás neudělá – jinými slovy musíte jako vstupní dokumenty dodat Calibre platné XHTML + CSS.

9.1.8 Jak použiji některé pokročilé funkce nástrojů převodu?

Pomoc s jakoukoli jednotlivou funkcí převaděčů můžete získat tak, že na ni najedete myší v grafickém rozhraní nebo v terminálu spustíte `ebook-convert dummy.html .epub -h`. Dobrým místem pro začátek je podívat se na následující ukázkový soubor, který ukazuje některé pokročilé funkce html-demo.zip[45].

9.2 Integrace zařízení

Obsah

- *Která zařízení Calibre podporuje?* (stránka 133)
- *Jak mohu pomoci, aby Calibre podporovalo mé zařízení?* (stránka 133)
- *Moje zařízení není v Calibre rozpoznáno?* (stránka 133)
- *Mé zařízení je nestandardní nebo neobvyklé. Co můžu udělat pro jeho připojení?* (stránka 134)
- *Jak použiji Calibre s mým zařízením iPad, iPhone nebo iPod touch?* (stránka 134)
- *How do I use calibre with my Android phone/tablet or Kindle Fire?* (stránka 135)
- *Can I access my calibre books using the web browser in my Kindle or other reading device?* (stránka 136)
- *I cannot send emails using calibre?* (stránka 136)
- *My device is getting mounted read-only in Linux, so calibre cannot connect to it?* (stránka 137)
- *Why does calibre not support collections on the Kindle or shelves on the Nook?* (stránka 137)
- *I am getting an error when I try to use calibre with my Kobo Touch/Glo/etc.?* (stránka 137)
- *Covers for books I send to my e-ink Kindle show up momentarily and then are replaced by a generic cover?* (stránka 138)
- *Covers for books sent to my Kindle ColorSoft and newer do not show up in the book list?* (stránka 138)

[45] https://calibre-ebook.com/downloads/html-demo.zip

- *Obálky pro mé soubory MOBI se přestaly zobrazovat v Kindle pro PC nebo Kindle pro Android, iPad apod.* (stránka 138)
- *I transferred some books to my Kindle using calibre and they did not show up?* (stránka 139)

9.2.1 Která zařízení Calibre podporuje?

Calibre se umí přímo připojit ke všem hlavním (a většině menších) čtecím zařízením e-knih, chytrým telefonům, tabletům atd. Kromě toho pomocí funkce *Připojit ke složce* ho můžete použít s jakoukoliv čtečkou elektronických knih, která se exportuje jako disk USB. A nakonec se můžete bezdrátově připojit k jakémukoliv zařízení, které má webový prohlížeč, pomocí Serveru s obsahem Calibre.

9.2.2 Jak mohu pomoci, aby Calibre podporovalo mé zařízení?

Pokud se vaše zařízení v operačním systému zobrazí jako disk USB, je přidání jeho podpory do Calibre velice snadné. Potřebujeme od vás jen pár informací:

- Úplný seznam formátů e-knih, které vaše zařízení podporuje.
- Je v zařízení nějaká zvláštní složka, do které by měly být umístěny všechny soubory e-knih? Také jestli zařízení rozpoznává soubory umístěné do podsložek.
- Potřebujeme také informace o vašem zařízení, které bude Calibre shromažďovat automaticky. Nejprve, pokud vaše zařízení podporuje SD karty, vložte je. Pak připojte zařízení k počítači. V Calibre přejděte na *Předvolby → Různé* a klikněte na tlačítko „Ladit rozpoznávání zařízení“. Tím se vytvoří nějaký výstup ladění. Zkopírujte ho do souboru a opakujte postup, tentokrát se zařízením odpojeným od počítače.
- Zašlete nám oba výše uvedené výstupy s ostatními informacemi a my pro vás napíšeme ovladač zařízení.

Jakmile nám odešlete výstup pro konkrétní operační systém, podpora pro zařízení v tomto operačním systému se objeví v příštím vydání Calibre. Abyste nám zaslali výstup, otevřete hlášení o chybě a připojte k němu výstup. Viz jak hlásit chyby[46].

9.2.3 Moje zařízení není v Calibre rozpoznáno?

Následujte tyto kroky pro nalezení problému:

- Ujistěte se, že k počítači připojujete současně pouze jedno zařízení. Nepřipojujte současně další zařízení podporované Calibre, jako je iPhone nebo iPad atd.
- Pokud připojujete zařízení Apple iDevice (iPad, iPod Touch, iPhone), společnost Apple již nepovoluje připojení softwaru třetích stran k jejich zařízením pomocí kabelu USB. Místo toho použijte bezdrátové připojení prostřednictvím Serveru s obsahem Calibre.
- If you are connecting a 2024 Kindle or newer or an Android device, and are on macOS or Linux, read the note under *Pomocí kabelu USB* (stránka 135).
- Pokud v systému MacOS zjistíte chyby oprávnění při připojování zařízení ke Calibre, můžete to napravit pod *Předvolby systému* > *Zabezpečení a soukromí* > *Soukromí* > *Soubory a složky*.
- Ujistěte se, že používáte nejnovější verzi Calibre (aktuálně 8.7.0). Nejnovější verzi lze vždy stáhnout z webové stránky Calibre[47]. Jakou verzi Calibre právě používáte, zjistíte na dolním řádku hlavního okna Calibre.
- Ujistěte se, že váš operační systém zařízení vidí. To znamená, že by se zařízení mělo zobrazit v Průzkumníkovi Windows (v systému Windows) nebo Finderu (v systému MacOS).

[46] https://calibre-ebook.com/bugs
[47] https://calibre-ebook.com/download

- V Calibre přejděte na *Předvolby → Ignorovaná zařízení* a zkontrolujte, zda vaše zařízení není ignorováno.
- Pokud všechny výše uvedené kroky selžou, přejděte na *Předvolby → Různé* a klikněte na *Ladění rozpoznávání zařízení* zatímco je zařízení připojeno a odešlete výstup jako ticket na hlášení chyb Calibre[48].

9.2.4 Mé zařízení je nestandardní nebo neobvyklé. Co můžu udělat pro jeho připojení?

Kromě funkce *Připojit ke složce*, kterou naleznete pod tlačítkem *Připojit nebo sdílet* Calibre poskytuje modul zařízení `Definováno uživatelem`, který lze použít pro připojení k libovolnému zařízení USB, které se zobrazí jako disková jednotka ve vašem operačním systému. Poznámka: ve Windows musí mít zařízení pro používání v Calibre přiřazeno písmeno jednotky. Další informace naleznete v modulu zařízení `Předvolby -> Moduly -> Moduly zařízení -> Definováno uživatelem` a `Předvolby -> Různé -> Získat informace pro nastavení zařízení definovaného uživatelem`. Pamatujte, že pokud používáte uživatelem definovaný modul pro zařízení normálně rozpoznané vestavěným modulem Calibre, musíte nejdříve zakázat vestavěný modul, aby se místo toho použil uživatelem definovaný modul.

9.2.5 Jak použiji Calibre s mým zařízením iPad, iPhone nebo iPod touch?

Snadným způsobem procházení vaší kolekce Calibre ze zařízení Apple je použití serveru, díky kterému je kolekce dostupná přes síť. Nejprve proveďte následující kroky v Calibre

- Nastavte Upřednostňovaný výstupní formát v Calibre na EPUB (výstupní formát lze nastavit pod *Předvolby → Rozhraní → Chování*)
- Nastavte výstupní profil na iPad (bude to fungovat i pro iPhone nebo iPod) pod *Předvolby → Převod → Běžné volby → Nastavení stránky*.
- Převeďte knihy, které chcete na svém iZařízení číst, do formátu EPUB tak, že je vyberete a kliknete na tlačítko *Převést*.
- Zapněte Server s obsahem kliknutím na tlačítko *Připojit/sdílet* a nechte Calibre spuštěné. Můžete také Calibre říct, aby automaticky spustilo Server s obsahem prostřednictvím *Předvolby → Sdílení → Sdílení po síti*.

Server s obsahem umožňuje číst knihy přímo v samotném prohlížeči Safari. Kromě toho existuje mnoho aplikací pro vaše iZařízení, které se mohou připojit k Serveru s obsahem Calibre. Příklady zahrnují: Marvin, Mapleread a samotné iBooks.

Používání Serveru s obsahem

Spusťte prohlížeč Safari a zadejte IP adresu a port počítače, na kterém je spuštěn server Calibre:

```
http://192.168.1.2:8080/
```

Nahraďte `192.168.1.2` místní IP adresou počítače, na kterém je spuštěno Calibre. Podrobnosti o spuštění serveru a nalezení správné IP adresy naleznete na *Server s obsahem Calibre* (stránka 113).

V prohlížeči Safari uvidíte seznam knih, klepněte na kteroukoli knihu a budete mít možnost si ji buď stáhnout, nebo si ji přečíst v samotném prohlížeči. Pokud se rozhodnete ji stáhnout, Safari se vás zeptá, jestli ji chcete otevřít pomocí iBooks.

Mnoho čtecích aplikací podporuje procházení knihovny Calibre přímo prostřednictvím její podpory OPDS. V takových aplikacích můžete přejít na obrazovku online katalogu a přidat IP adresu serveru Calibre pro procházení a stahování knih z knihovny Calibre v rámci aplikace.

[48] https://bugs.launchpad.net/calibre

9.2.6 How do I use calibre with my Android phone/tablet or Kindle Fire?

Zařízení Android můžete připojit ke Calibre dvěma způsoby. Pomocí kabelu USB nebo bezdrátově, vzduchem. Prvním krokem pro použití zařízení Android je nainstalovat do něj aplikaci pro čtení e-knih. Existuje mnoho bezplatných a placených aplikací pro čtení e-knih pro Android: Některé příklady (v žádném konkrétním pořadí): FBReader[49], Moon+[50], Mantano[51], Aldiko[52], Kindle[53].

Pomocí kabelu USB

Jednoduše připojte zařízení k počítači pomocí kabelu USB. Calibre by mělo automaticky rozpoznat zařízení a pak do něj můžete přenést knihy kliknutím na tlačítko *Odeslat do zařízení*. Pamatujte, že v MacOS a Linuxu lze k zařízení Android připojit pouze jeden program najednou, takže se ujistěte, že zařízení není otevřeno ve Správci souborů OS nebo v nástroji pro přenos souborů Android atd.

Poznámka

U novějších zařízení Android bude možná nutné provést několik věcí, aby připojení fungovalo, protože Google opravdu nechce, abyste byli nezávislí na jeho cloudu. Nejprve odemkněte obrazovku před připojením kabelu USB. Když připojíte kabel USB, zobrazí se automaticky otevírané oznámení. Ujistěte se, že říká něco jako „Přenos mediálních souborů“ nebo „MTP (Režim přenosu médií)“. Pokud tomu tak není, klepněte na oznámení a změňte režim na Přenos médií (MTP). V tomto bodě možná bude nutné restartovat Calibre, aby bylo zařízení rozpoznáno. Nakonec klepněte na OK. Automaticky otevírané okno se možná v zařízení bude objevovat pokaždé, když se k němu Calibre nebo operační systém skutečně pokusí připojit, a bude požádat o povolení.

Poznámka

U zařízení Kindle Fire 8 a novějších se po připojení kabelu USB zobrazí ikona , která ukazuje, že se zařízení nabíjí. Klepněte na ni a přepněte zařízení do režimu přenosu dat a poté spusťte Calibre, mělo být pak být rozpoznáno.

Vzduchem

calibre has a builtin web server, the *calibre Content server* (stránka 113). It makes your calibre collection available over the net. You can browse it on your device using a simple browser or a dedicated application. First perform the following steps in calibre:

- Nastavte *Upřednostňovaný výstupní formát* v Calibre na EPUB pro normální zařízení Android nebo MOBI pro Kindle (Výstupní formát lze nastavit pod *Předvolby → Rozhraní → Chování*)
- Převeďte knihy, které chcete na svém zařízení číst, do formátu EPUB nebo MOBI tak, že je vyberete a kliknete na tlačítko *Převést*.
- Zapněte *Server s obsahem* v předvolbách Calibre a nechte Calibre spuštěné.

Nyní ve svém zařízení Android otevřete prohlížeč a přejděte na

http://192.168.1.2:8080/

Nahraďte `192.168.1.2` místní IP adresou počítače, na kterém je spuštěno Calibre. Podrobnosti o spuštění serveru a nalezení správné IP adresy naleznete na *Server s obsahem Calibre* (stránka 113).

[49] https://play.google.com/store/apps/details?id=org.geometerplus.zlibrary.ui.android&hl=en
[50] https://play.google.com/store/apps/details?id=com.flyersoft.moonreader&hl=en
[51] https://play.google.com/store/apps/details?id=com.mantano.reader.android.lite&hl=en
[52] https://play.google.com/store/apps/details?id=com.aldiko.android&hl=en
[53] https://play.google.com/store/apps/details?id=com.amazon.kindle&feature=related_apps

Nyní můžete procházet svou kolekci knih a stahovat knihy z Calibre do vašeho zařízení a otevírat je pomocí jakéhokoliv softwaru pro čtení e-knih, který máte na svém zařízení Android.

Mnoho čtecích aplikací podporuje procházení knihovny Calibre přímo prostřednictvím její podpory OPDS. V takových aplikacích můžete přejít na obrazovku online katalogu a přidat IP adresu serveru Calibre pro procházení a stahování knih z knihovny Calibre v rámci aplikace.

9.2.7 Can I access my calibre books using the web browser in my Kindle or other reading device?

calibre has a *Content server* that exports the books in calibre as a web page. See *Server s obsahem Calibre* (stránka 113) for details.

Some devices, like the Kindle (1/2/DX), do not allow you to access port 8080 (the default port on which the content server runs). In that case, change the port in the calibre Preferences to 80. (On some operating systems, you may not be able to run the server on a port number less than 1024 because of security settings. In this case the simplest solution is to adjust your router to forward requests on port 80 to port 8080).

Also some devices do not have browsers advanced enough to run the app-like interface used by the Content server. For such devices, you can simply add `/mobile` to the server URL to get a simplified, non-JavaScript interface.

9.2.8 I cannot send emails using calibre?

Because of the large amount of spam in email, sending email can be tricky, as different mail servers use different strategies to block email. The most common problem is if you are sending email directly (without a mail relay) in calibre. Many servers (for example, Amazon) block email that does not come from a well known relay. The most robust way to setup email sending in calibre is to do the following:

- Create a free GMX account at GMX[54].
- Go to *Preferences → Sharing → Sharing books by email* in calibre and click the *Use GMX* button and fill in the information asked for.
- Log into your GMX account on the website and enable SMTP sending (*Settings->POP3 & IMAP->Send and receive emails via external program*)
- calibre will then be able to use GMX to send the mail.
- If you are sending to your Kindle, remember to update the email preferences on your Amazon Kindle page to allow email sent from your GMX email address. Also note that Amazon does not allow email delivery of AZW3 and new style (KF8) MOBI files. Finally, Amazon recently started sending confirmation emails that you have to click on back to your GMX account before the book is actually delivered.

Even after doing this, you may have problems. One common source of problems is that some poorly designed antivirus programs block calibre from opening a connection to send email. Try adding an exclusion for calibre in your antivirus program.

> **Poznámka**
>
> Microsoft/GMX can disable your account if you use it to send large amounts of email. So, when using these services to send mail calibre automatically restricts itself to sending one book every five minutes. If you don't mind risking your account being blocked you can reduce this wait interval by going to *Preferences → Advanced → Tweaks* in calibre.

[54] https://www.gmx.com

Poznámka

Google recently deliberately broke their email sending protocol (SMTP) support in an attempt to force everyone to use their web interface so they can show you more ads. They are trying to claim that SMTP is insecure, that is incorrect and simply an excuse. Use some other email provider instead.

Poznámka

If you are concerned about giving calibre access to your email account, simply create a new free email account with GMX or Outlook and use it only for calibre.

9.2.9 My device is getting mounted read-only in Linux, so calibre cannot connect to it?

Linux kernels mount devices read-only when their filesystems have errors. You can repair the filesystem with:

```
sudo fsck.vfat -y /dev/sdc
```

Replace /dev/sdc with the path to the device node of your device. You can find the device node of your device, which will always be under /dev by examining the output of:

```
mount
```

9.2.10 Why does calibre not support collections on the Kindle or shelves on the Nook?

Neither the Kindle nor the Nook provide any way to manipulate collections over a USB connection. If you really care about using collections, I would urge you to sell your Kindle/Nook and get a Kobo. Only Kobo seems to understand that life is too short to be entering collections one by one on an e-ink screen

Note that in the case of the Kindle, there is a way to manipulate collections via USB, but it requires that the Kindle be rebooted *every time* it is disconnected from the computer, for the changes to the collections to be recognized. As such, it is unlikely that any calibre developers will ever feel motivated enough to support it. There is however, a calibre plugin that allows you to create collections on your Kindle from the calibre metadata. It is available from here[55].

Poznámka

Amazon have removed the ability to manipulate collections completely in their newer models, like the Kindle Touch and Kindle Fire, making even the above plugin useless, unless you root your Kindle and install custom firmware.

9.2.11 I am getting an error when I try to use calibre with my Kobo Touch/Glo/etc.?

The Kobo has very buggy firmware. Connecting to it has been known to fail at random. Certain combinations of motherboard, USB ports/cables/hubs can exacerbate this tendency to fail. If you are getting an error when connecting to your touch with calibre try the following, each of which has solved the problem for *some* calibre users.

- Connect the Kobo directly to your computer, not via USB Hub
- Try a different USB cable and a different USB port on your computer
- Log out of the Kobo and log in again, this causes it to rebuild the database, fixing corrupted database errors.

[55] https://www.mobileread.com/forums/showthread.php?t=244202

- Try upgrading the firmware on your Kobo Touch to the latest
- Try resetting the Kobo (sometimes this cures the problem for a little while, but then it re-appears, in which case you have to reset again and again)
- Try only putting one or two books onto the Kobo at a time and do not keep large collections on the Kobo

9.2.12 Covers for books I send to my e-ink Kindle show up momentarily and then are replaced by a generic cover?

This happens because of an Amazon bug. They try to download a cover for the book from their servers and when that fails, they replace the existing cover that calibre created with a generic cover. For details see this forum thread[56]. As of version 4.17, calibre has a workaround, where if you connect the Kindle to calibre after the covers have been destroyed by Amazon, calibre will restore them automatically. So in order to see the covers on your Kindle, you have to:

1) Send the book to the Kindle with calibre
2) Disconnect the Kindle and wait for Amazon to destroy the cover
3) Reconnect the Kindle to calibre

Note that this workaround only works for books sent with calibre 4.17 or later. Alternately, simply keep your Kindle in airplane mode, you don't really want Amazon knowing every book you read anyway. I encourage you to contact Amazon customer support and complain loudly about this bug. Maybe Amazon will listen.

Poznámka

If the workaround is not working for you make sure you Kindle firmware is at least version 5.12.5, released in April 2020.

9.2.13 Covers for books sent to my Kindle ColorSoft and newer do not show up in the book list?

Amazon deliberately broke this functionality in their ColorSoft and newer devices in order to discourage you from reading non Amazon books on their devices. See this forum thread[57] for details. The only known workaround is to send the books as „Personal documents" to the Kindle which will fix the covers not showing up but break other features such as Whispersync and the books will show up under „Personal documents" not „Books" on the device. To enable this in calibre go to *Preferences → Output options → MOBI output* and enable the check box that says *Enable sharing of book content*. This will cause all future books sent to the Kindle by calibre to be marked as personal documents.

9.2.14 Obálky pro mé soubory MOBI se přestaly zobrazovat v Kindle pro PC nebo Kindle pro Android, iPad apod.

Je to způsobeno chybou v softwaru Amazonu. Můžete to obejít přejitím do *Předvolby → Převod → Volby výstupu → Výstup MOBI* a nastavením volby *Povolit sdílení obsahu knihy…*. Pokud převádíte dříve převedenou knihu, budete muset tuto volbu také povolit v dialogovém okně převodu pro danou jednotlivou knihu (protože nastavení převodu knihy se ukládají a mají přednost).

Pamatujte, že pokud to uděláte, bude to znamenat, že vygenerované MOBI se v Kindle Fire zobrazí pod osobními dokumenty namísto Books a nebude fungovat synchronizace Amazonu, ale obálky ano. Je to vaše volba, která funkce je pro vás důležitější. Doporučuji, abyste se obrátili na Amazon a požádali je o opravu této chyby.

Chyba v softwaru Amazonu je, že když do Kindle vložíte soubor MOBI, pokud není soubor označen jako osobní dokument, Amazon předpokládá, že jste knihu zakoupili od něj a pokusí se stáhnout náhled obálky ze svých serverů. Když

[56] https://www.mobileread.com/forums/showthread.php?t=329945
[57] https://www.mobileread.com/forums/showthread.php?t=364350

se stahování nezdaří, odmítne se vrátit zpět k obálce definované v souboru MOBI. Je pravděpodobné, že se tak Amazon snaží zkusit donutit autory, aby prodávali pouze prostřednictvím něj. Jinými slovy, Kindle zobrazí obálky pouze pro knihy označené jako osobní dokumenty nebo knihy zakoupené přímo od Amazonu.

Pokud odešlete soubor MOBI do čtečky Kindle z Calibre pomocí připojení USB, Calibre obejde tuto chybu Amazonu nahráním samotného náhledu obálky. Toto řešení je však možné pouze při použití připojení USB a odesílání z Calibre. Pamatujte, že pokud odesíláte e-mailem, Amazon automaticky označí soubor MOBI jako osobní dokument a obálka bude fungovat, ale kniha se zobrazí v osobních dokumentech.

9.2.15 I transferred some books to my Kindle using calibre and they did not show up?

Books sent to the Kindle only show up on the Kindle after they have been *indexed* by the Kindle. This can take some time. If the book still does not show up after some time, then it is likely that the Kindle indexer crashed. Sometimes a particular book can cause the indexer to crash. Unfortunately, Amazon has not provided any way to deduce which book is causing a crash on the Kindle. Your only recourse is to either reset the Kindle, or delete all files from its memory using Windows Explorer (or whatever file manager you use) and then send the books to it again, one by one, until you discover the problem book. Once you have found the problem book, delete it off the Kindle and do a MOBI to MOBI or MOBI to AZW3 conversion in calibre and then send it back. This will most likely take care of the problem.

9.3 Library management

Obsah

- *Where are the book files stored?* (stránka 139)
- *How does calibre manage author names and sorting?* (stránka 140)
- *Why doesn't calibre let me store books in my own folder structure?* (stránka 141)
- *Why doesn't calibre have a column for foo?* (stránka 142)
- *Can I have a column showing the formats or the ISBN?* (stránka 142)
- *How do I move my calibre data from one computer to another?* (stránka 142)
- *The list of books in calibre is blank!* (stránka 143)
- *I am getting errors with my calibre library on a networked drive/NAS?* (stránka 143)

9.3.1 Where are the book files stored?

When you first run calibre, it will ask you for a folder in which to store your books. Whenever you add a book to calibre, it will copy the book into that folder. Books in the folder are nicely arranged into sub-folders by Author and Title. Note that the contents of this folder are automatically managed by calibre, **do not** add any files/folders manually to this folder, as they may be automatically deleted. If you want to add a file associated to a particular book, use the top right area of *Edit metadata* dialog to do so. Then, calibre will automatically put that file into the correct folder and move it around when the title/author changes.

Metadata about the books is stored in the file `metadata.db` at the top level of the library folder. This file is a sqlite database. When backing up your library make sure you copy the entire folder and all its sub-folders.

The library folder and all its contents make up what is called a calibre library. You can have multiple such libraries. To manage the libraries, click the calibre icon on the toolbar. You can create new libraries, remove/rename existing ones and switch between libraries easily.

You can copy or move books between different libraries (once you have more than one library setup) by right clicking on a book and selecting the *Copy to library* action.

9.3.2 How does calibre manage author names and sorting?

Author names are complex, especially across cultures, see this note[58] for some of the complexities. calibre has a very flexible strategy for managing author names. The first thing to understand is that books and authors are separate entities in calibre. A book can have more than one author, and an author can have more than one book. You can manage the authors of a book by the edit metadata dialog. You can manage individual authors by right clicking on the author in the Tag browser on the left of the main calibre window and selecting *Manage authors*. Using this dialog you can change the name of an author and also how that name is sorted. This will automatically change the name of the author in all the books of that author. When a book has multiple authors, separate their names using the & character.

Now coming to author name sorting:

- When a new author is added to calibre (this happens whenever a book by a new author is added), calibre automatically computes a sort string for both the book and the author.
- Authors in the Tag browser are sorted by the sort value for the **authors**. Remember that this is different from the Author sort field for a book.
- By default, this sort algorithm assumes that the author name is in `First name Last name` format and generates a `Last name, First name` sort value.
- You can change this algorithm by going to *Preferences → Advanced → Tweaks* and setting the *author_sort_copy_method* tweak.
- You can force calibre to recalculate the author sort values for every author by right clicking on any author and selecting *Manage authors*, then pushing the *Recalculate all author sort values* button. Do this after you have set the author_sort_copy_method tweak to what you want.
- You can force calibre to recalculate the author sort values for all books by using the bulk metadata edit dialog (select all books and click edit metadata, check the *Automatically set author sort* checkbox, then press OK).
- When recalculating the author sort values for books, calibre uses the author sort values for each individual author. Therefore, ensure that the individual author sort values are correct before recalculating the books' author sort values.
- You can control whether the Tag browser display authors using their names or their sort values by setting the *categories_use_field_for_author_name* tweak in *Preferences → Advanced → Tweaks*

Note that you can set an individual author's sort value to whatever you want using *Manage authors*. This is useful when dealing with names that calibre will not get right, such as complex multi-part names like Miguel de Cervantes Saavedra or when dealing with Asian names like Sun Tzu.

With all this flexibility, it is possible to have calibre manage your author names however you like. For example, one common request is to have calibre display author names LN, FN. To do this, and if the note below does not apply to you, then:

- Set the `author_sort_copy_method` tweak to `copy` as described above.
- Restart calibre. Do not change any book metadata before doing the remaining steps.
- Change all author names to LN, FN using the Manage authors dialog.
- After you have changed all the authors, press the *Recalculate all author sort values* button.
- Press OK, at which point calibre will change the authors in all your books. This can take a while.

[58] https://www.w3.org/International/questions/qa-personal-names.en.php?changelang=en

Poznámka

When changing from FN LN to LN, FN, it is often the case that the values in author_sort are already in LN, FN format. If this is your case, then do the following:

- Set the `author_sort_copy_method` tweak to `copy` as described above.
- Restart calibre. Do not change any book metadata before doing the remaining steps.
- Open the Manage authors dialog. Press the `copy all author sort values to author` button.
- Check through the authors to be sure you are happy. You can still press Cancel to abandon the changes. Once you press OK, there is no undo.
- Press OK, at which point calibre will change the authors in all your books. This can take a while.

9.3.3 Why doesn't calibre let me store books in my own folder structure?

The whole point of calibre's library management features is that they provide a search and sort based interface for locating books that is *much* more efficient than any possible folder scheme you could come up with for your collection. Indeed, once you become comfortable using calibre's interface to find, sort and browse your collection, you won't ever feel the need to hunt through the files on your disk to find a book again. By managing books in its own folder structure of Author -> Title -> Book files, calibre is able to achieve a high level of reliability and standardization. To illustrate why a search/tagging based interface is superior to folders, consider the following. Suppose your book collection is nicely sorted into folders with the following scheme:

```
Genre -> Author -> Series -> ReadStatus
```

Now this makes it very easy to find for example all science fiction books by Isaac Asimov in the Foundation series. But suppose you want to find all unread science fiction books. There's no easy way to do this with this folder scheme, you would instead need a folder scheme that looks like:

```
ReadStatus -> Genre -> Author -> Series
```

In calibre, you would instead use tags to mark genre and read status and then just use a simple search query like `tag:scifi and not tag:read`. calibre even has a nice graphical interface, so you don't need to learn its search language instead you can just click on tags to include or exclude them from the search.

To those of you that claim that you need access to the filesystem, so that you can have access to your books over the network, calibre has an excellent Content server that gives you access to your calibre library over the net.

If you are worried that someday calibre will cease to be developed, leaving all your books marooned in its folder structure, explore the powerful *Save to disk* feature in calibre that lets you export all your files into a folder structure of arbitrary complexity based on their metadata.

Finally, the reason there are numbers at the end of every title folder, is for *robustness*. That number is the id number of the book record in the calibre database. The presence of the number allows you to have multiple records with the same title and author names. It is also part of what allows calibre to magically regenerate the database with all metadata if the database file gets corrupted. Given that calibre's mission is to get you to stop storing metadata in filenames and stop using the filesystem to find things, the increased robustness afforded by the id numbers is well worth the uglier folder names.

If you are still not convinced, then I'm afraid calibre is not for you. Look elsewhere for your book cataloguing needs. Just so we're clear, **this is not going to change**. Kindly do not contact us in an attempt to get us to change this.

9.3.4 Why doesn't calibre have a column for foo?

calibre is designed to have columns for the most frequently and widely used fields. In addition, you can add any columns you like. Columns can be added via *Preferences → Interface → Add your own columns*. Watch the tutorial UI Power tips[59] to learn how to create your own columns, or read this blog post[60].

You can also create „virtual columns" that contain combinations of the metadata from other columns. In the add column dialog use the *Quick create* links to easily create columns to show the book ISBN or formats. You can use the powerful calibre template language to do much more with columns. For more details, see *Jazyk šablon v programu Calibre* (stránka 160).

9.3.5 Can I have a column showing the formats or the ISBN?

Yes, you can. Follow the instructions in the answer above for adding custom columns.

9.3.6 How do I move my calibre data from one computer to another?

You can export all calibre data (books, settings and plugins) and then import it on another computer. First let's see how to export the data:

- Right click the calibre icon in the main calibre toolbar and select *Export/import all calibre data*. Note that if there is currently a device connected, this menu option will not be available – so, disconnect any connected devices. Then click the button labelled *Export all your calibre data*. You will see a list of all your calibre libraries. Click OK and choose an empty folder somewhere on your computer. The exported data will be saved in this folder. Simply copy this folder to your new computer and follow the instructions below to import the data.
- Install calibre on your new computer and run through the *Welcome wizard*, it does not matter what you do there, as you will be importing your old settings in the next step. You will now have an empty calibre, with just the *Getting Started* guide in your library. Once again, right click the calibre button and choose *Export/import all calibre data*. Then click the button labelled *Import previously exported data*. Select the folder with the exported data that you copied over earlier. You will now have a list of libraries you can import. Go through the list one by one, and select the new location for each library (a location is just an empty folder somewhere on your computer). Click OK. After the import completes, calibre will restart, with all your old libraries, settings and calibre plugins.

Poznámka

This import/export functionality is only available from calibre version 2.47 onwards. If you have an older version of calibre, or if you encounter problems with the import/export, you can just copy over your calibre library folder manually, as described in the next paragraph.

Simply copy the calibre library folder from the old to the new computer. You can find out what the library folder is by clicking the calibre icon in the toolbar. Choose the *Switch/create calibre library* action and you will see the path to the current calibre library.

Now on the new computer, start calibre for the first time. It will run the *Welcome wizard* asking you for the location of the calibre library. Point it to the previously copied folder. If the computer you are transferring to already has a calibre installation, then the *Welcome wizard* won't run. In that case, right-click the calibre icon in the toolbar and point it to the newly copied folder. You will now have two calibre libraries on your computer and you can switch between them by clicking the calibre icon on the toolbar. Transferring your library in this manner preserves all your metadata, tags, custom columns, etc.

[59] https://calibre-ebook.com/demo#tutorials
[60] https://blog.calibre-ebook.com/calibre-custom-columns/

9.3.7 The list of books in calibre is blank!

In order to understand why that happened, you have to understand what a calibre library is. At the most basic level, a calibre library is just a folder. Whenever you add a book to calibre, that book's files are copied into this folder (arranged into sub folders by author and title). Inside the calibre library folder, at the top level, you will see a file called metadata.db. This file is where calibre stores the metadata like title/author/rating/tags etc. for *every* book in your calibre library. The list of books that calibre displays is created by reading the contents of this metadata.db file.

There can be two reasons why calibre is showing a empty list of books:

- Your calibre library folder changed its location. This can happen if it was on an external disk and the drive letter for that disk changed. Or if you accidentally moved the folder. In this case, calibre cannot find its library and so starts up with an empty library instead. To remedy this, do a right-click on the calibre icon in the calibre toolbar and select Switch/create library. Click the little blue icon to select the new location of your calibre library and click OK. If you don't know the new location search your computer for the file `metadata.db`.
- Your metadata.db file was deleted/corrupted. In this case, you can ask calibre to rebuild the metadata.db from its backups. Right click the calibre icon in the calibre toolbar and select Library maintenance->Restore database. calibre will automatically rebuild metadata.db.

9.3.8 I am getting errors with my calibre library on a networked drive/NAS?

Do not put your calibre library on a networked drive.

A filesystem is a complex beast. Most network filesystems lack various filesystem features that calibre uses. Some don't support file locking, some don't support hardlinking, some are just flaky. Additionally, calibre is a single user application, if you accidentally run two copies of calibre on the same networked library, bad things will happen. Finally, different OSes impose different limitations on filesystems, so if you share your networked drive across OSes, once again, bad things *will happen*.

Consider using the calibre Content server to make your books available on other computers. Run calibre on a single computer and access it via the Content server or a Remote Desktop solution.

If you must share the actual library, use a file syncing tool like DropBox or rsync instead of a networked drive. If you are using a file-syncing tool it is **essential** that you make sure that both calibre and the file syncing tool do not try to access the calibre library at the same time. In other words, **do not** run the file syncing tool and calibre at the same time.

Even with these tools there is danger of data corruption/loss, so only do this if you are willing to live with that risk. In particular, be aware that **Google Drive** is incompatible with calibre, if you put your calibre library in Google Drive, **you will suffer data loss**. See this thread[61] for details.

9.4 Různé

Obsah

- *Amazon is stopping email delivery of MOBI files?* (stránka 144)
- *I want calibre to download news from my favorite news website.* (stránka 145)
- *Why the name calibre?* (stránka 145)
- *Why does calibre show only some of my fonts on macOS?* (stránka 145)
- *calibre is not starting on Windows?* (stránka 145)
- *calibre freezes/crashes occasionally?* (stránka 146)

[61] https://www.mobileread.com/forums/showthread.php?t=205581

- *The calibre E-book viewer and Edit book tools do not work on Windows?* (stránka 146)
- *Using the viewer or doing any conversions results in a permission denied error on Windows* (stránka 147)
- *calibre is not starting/crashing on macOS?* (stránka 147)
- *I get only a black or white screen when running the calibre E-book viewer?* (stránka 147)
- *Stáhnul jsem instalační soubor, ale nefunguje?* (stránka 147)
- *My antivirus program claims calibre is a virus/trojan?* (stránka 148)
- *How do I backup calibre?* (stránka 148)
- *How do I use purchased EPUB books with calibre (or what do I do with .acsm files)?* (stránka 148)
- *I am getting a „Permission Denied" error?* (stránka 149)
- *Can I have the comment metadata show up on my reader?* (stránka 149)
- *How do I get calibre to use my HTTP proxy?* (stránka 149)
- *I want some feature added to calibre. What can I do?* (stránka 149)
- *Why doesn't calibre have an automatic update?* (stránka 150)
- *How is calibre licensed?* (stránka 150)
- *How do I run calibre from my USB stick?* (stránka 150)
- *How do I run parts of calibre like news download and the Content server on my own Linux server?* (stránka 150)

9.4.1 Amazon is stopping email delivery of MOBI files?

Amazon have announced[62] that they will stop accepting MOBI files emailed to `@kindle.com` email addresses. You can instruct calibre to send EPUB instead of MOBI by going to *Preferences → Sharing books by email* and then removing MOBI from the list of formats to send to your `@kindle.com` email address and adding EPUB instead.

Note however, that Amazon's EPUB intake is very flawed, they will reject a number of EPUB files that work everywhere else. In such cases you can try the following trick:

1. Convert the EPUB file to MOBI
2. Then convert the MOBI file back to EPUB and send the resulting EPUB file

This will remove all advanced formatting, embedded fonts, etc., but greatly increase the chances of Amazon accepting the EPUB.

Poznámka

If you were previously using email delivery of periodicals downloaded by calibre, you will be better off sending those by USB cable or downloading them from the calibre Content server via the Kindle's built-in browser. However, if you want to continue using email delivery you can try changing the output format in Preferences->Behavior to EPUB, then calibre will download the news in EPUB format. Whether Amazon will accept the EPUB or not is a whole other question.

[62] https://blog.the-ebook-reader.com/2022/05/03/amazon-dropping-mobi-support-on-send-to-kindle-apps/

9.4.2 I want calibre to download news from my favorite news website.

If you are reasonably proficient with computers, you can teach calibre to download news from any website of your choosing. To learn how to do this see *Přidávání oblíbených webových stránek se zprávami* (stránka 33).

Otherwise, you can request a particular news site by posting in the calibre Recipes forum[63].

9.4.3 Why the name calibre?

Take your pick:

- Converter And LIBRary for E-books
- A high *calibre* product
- A tribute to the SONY Librie which was the first e-ink based e-book reader
- My wife chose it ;-)

calibre is pronounced as cal-i-ber *not* ca-li-bre. If you're wondering, calibre is the British/commonwealth spelling for caliber. Being Indian, that's the natural spelling for me.

9.4.4 Why does calibre show only some of my fonts on macOS?

calibre embeds fonts in e-book files it creates. E-book files support embedding only TrueType and OpenType (.ttf and .otf) fonts. Most fonts on macOS systems are in .dfont format, thus they cannot be embedded. calibre shows only TrueType and OpenType fonts found on your system. You can obtain many such fonts on the web. Simply download the .ttf/.otf files and add them to the Library/Fonts folder in your home folder.

9.4.5 calibre is not starting on Windows?

There can be several causes for this:

- If you get no errors but the calibre window does not appear, it has probably just appeared off screen. You can gather all windows onto the current screen using one of the techniques described here[64].
- Some software has been known to interfere with calibre, try rebooting in Safe mode and see if it works. A known culprit is the Sunshine[65] screen sharing software.
- If you get an error about calibre not being able to open a file because it is in use by another program, do the following:
  - Uninstall calibre
  - Reboot your computer
  - Re-install calibre. But do not start calibre from the installation wizard.
  - Temporarily disable your antivirus program (disconnect from the Internet before doing so, to be safe)
  - Look inside the folder you chose for your calibre library. If you see a file named metadata.db, delete it.
  - Start calibre
  - From now on you should be able to start calibre normally.
- If you get an error about a Python function terminating unexpectedly after upgrading calibre, first uninstall calibre, then delete the folders (if they exists) `C:\Program Files\Calibre` and `C:\Program Files\Calibre2`. Now re-install and you should be fine.

[63] https://www.mobileread.com/forums/forumdisplay.php?f=228
[64] https://www.wikihow.com/Bring-an-Off-Screen-Window-Back-on-Windows
[65] https://github.com/LizardByte/Sunshine

- If you get an error in the *Welcome wizard* on an initial run of calibre, try choosing a folder like `C:\library` as the calibre library (calibre sometimes has trouble with library locations if the path contains non-English characters, or only numbers, etc.)
- Try running it as administrator (Right click on the icon and select *Run as administrator*)

If it still won't launch, start a command prompt (press the Windows key and R; then type **`cmd.exe`** in the Run dialog that appears). At the command prompt type the following command and press Enter:

```
calibre-debug -g
```

Post any output you see in a help message on the Forum[66].

9.4.6 calibre freezes/crashes occasionally?

There are several possible things I know of, that can cause this:

- You recently connected an external monitor or TV to your computer. In this case, whenever calibre opens a new window like the edit metadata window or the conversion dialog, it appears on the second monitor where you don't notice it and so you think calibre has frozen. Disconnect your second monitor and restart calibre.
- The following programs have been reported to cause crashes in calibre: If you are running any of these, close them before starting calibre, or uninstall them: *RoboForm*, *Logitech SetPoint Settings*, *Constant Guard Protection by Xfinity*, *Spybot*, *Killer Network Manager*, *Nahimic UI Interface*, *Acronis True Image*.
- You are using a Wacom branded USB mouse/tablet. There is an incompatibility between Wacom drivers and the graphics toolkit calibre uses. Try using a non-Wacom mouse.
- On some 64 bit versions of Windows there are security software/settings that prevent 64-bit calibre from working properly. If you are using the 64-bit version of calibre try switching to the 32-bit version.
- If the crash happens when you are trying to copy text from the calibre E-book viewer, it is most likely caused by some clipboard monitoring/managing application you have running. Turn it off and you should be fine.
- If the crashes happen specifically when you are using a file dialog, like clicking on the *Add books* button or the *Save to Disk* button, then you have some software that has installed broken Shell extensions on your computer. Known culprits include: *SpiderOak*, *odrive sync* and *Dell Backup and Recovery* and *NetDrive*. If you have one of these, uninstall them and you will be fine. You can also use the NirSoft Shell Extension Viewer[67] to see what shell extensions are installed on your system and disable them individually, if you don't want to uninstall the full program. Remember to use „Restart Explorer" or reboot your computer after disabling the shell extensions.

If none of the above apply to you, then there is some other program on your computer that is interfering with calibre. First reboot your computer in safe mode, to have as few running programs as possible, and see if the crashes still happen. If they do not, then you know it is some program causing the problem. The most likely such culprit is a program that modifies other programs' behavior, such as an antivirus, a device driver, something like RoboForm (an automatic form filling app) or an assistive technology like Voice Control or a Screen Reader.

The only way to find the culprit is to eliminate the programs one by one and see which one is causing the issue. Basically, stop a program, run calibre, check for crashes. If they still happen, stop another program and repeat.

9.4.7 The calibre E-book viewer and Edit book tools do not work on Windows?

These two programs use hardware acceleration as they embed a version of the Chrome browser to render HTML. If they do not work it will be because of incompatibility with your system's GPU (graphics) drivers. Try updating these first, and reboot. If that does not fix it, you can set the `QTWEBENGINE_CHROMIUM_FLAGS` environment variable to the value `--disable-gpu` to turn off hardware acceleration. See this page[68] for details.

[66] https://www.mobileread.com/forums/forumdisplay.php?f=166
[67] https://www.nirsoft.net/utils/shexview.html
[68] https://doc.qt.io/qt-6/qtwebengine-debugging.html

9.4.8 Using the viewer or doing any conversions results in a permission denied error on Windows

Something on your computer is preventing calibre from accessing its own temporary files. Most likely the permissions on your `Temp` folder are incorrect. Go to the folder file:*C:\Users\USERNAME\AppData\Local* in Windows Explorer and then right click on the file:*Temp* folder, select *Properties* and go to the *Security* tab. Make sure that your user account has full control for this folder.

Some users have reported that running the following command in an Administrator Command Prompt fixed their permissions. To get an Administrator Command Prompt search for cmd.exe in the start menu, then right click on the command prompt entry and select *Run as administrator*. At the command prompt type the following command and press Enter:

```
icacls "%appdata%\..\Local\Temp" /reset /T
```

Alternately, you can run calibre as Administrator, but doing so will cause some functionality, such as drag and drop to not work.

Finally, some users have reported that disabling UAC fixes the problem.

9.4.9 calibre is not starting/crashing on macOS?

One common cause of failures on macOS is the use of accessibility technologies that are incompatible with the graphics toolkit calibre uses. Try turning off VoiceOver if you have it on. Also go to System Preferences->System->Universal Access and turn off the setting for enabling access for assistive devices in all the tabs. Another cause can be some third party apps that modify system behavior, such as Smart Scroll and Default Folder X.

You can obtain debug output about why calibre is not starting by running *Console.app*. Debug output will be printed to it. If the debug output contains a line that looks like:

```
Qt: internal: -108: Error ATSUMeasureTextImage text/qfontengine_mac.mm
```

then the problem is probably a corrupted font cache. You can clear the cache by following these instructions[69]. If that doesn't solve it, look for a corrupted font file on your system, in ~/Library/Fonts or the like. An easy way to check for corrupted fonts in macOS is to start the „Font Book" application, select all fonts and then in the File menu, choose „Validate fonts".

9.4.10 I get only a black or white screen when running the calibre E-book viewer?

This will be because of an incompatibility between Qt WebEngine, which the viewer uses to render and the GPU drivers on your system. First try upgrading the GPU drivers. If that does not help, you can try turning off hardware acceleration in Qt WebEngine by setting the environment variable `QTWEBENGINE_CHROMIUM_FLAGS` to the value `--disable-gpu`. See *Proměnné prostředí* (stránka 287) for how to change environment variables.

9.4.11 Stáhnul jsem instalační soubor, ale nefunguje?

Downloading from the Internet can sometimes result in a corrupted download. If the calibre installer you downloaded is not opening, try downloading it again. If re-downloading it does not work, download it from an alternate location[70]. If the installer still doesn't work, then something on your computer is preventing it from running.

- Try temporarily disabling your antivirus program (Microsoft Security Essentials, or Kaspersky or Norton or McAfee or whatever). This is most likely the culprit if the upgrade process is hanging in the middle.
- Similarly, if the installer is failing/rolling back and you have Microsoft PowerToys running, quit it.
- If you have installed to a non-standard location, try running the installer as Administrator

[69] https://typefaceapp.com/help/troubleshooting/macos-font-cache
[70] https://github.com/kovidgoyal/calibre/releases/latest

- Try rebooting your computer and running a registry cleaner like Wise registry cleaner[71].
- Try a clean install. That is, uninstall calibre, delete `C:\Program Files\Calibre2` (or wherever you previously chose to install calibre). Then re-install calibre. Note that uninstalling does not touch your books or settings.
- Try downloading the installer with an alternate browser. For example if you are using Microsoft Edge, try using Firefox or Chrome instead.
- If you get an error about a missing DLL on Windows, then most likely, the permissions on your temporary folder are incorrect. Go to the folder `C:\Users\USERNAME\AppData\Local` in Windows Explorer and then right click on the `Temp` folder and select *Properties* and go to the *Security* tab. Make sure that your user account has full control for this folder.

If you still cannot get the installer to work and you are on Windows, you can use the calibre portable install[72], which does not need an installer (it is just a ZIP file).

9.4.12 My antivirus program claims calibre is a virus/trojan?

The first thing to check is that you are downloading calibre from the official website[73]. Make sure you are clicking the download links on the left, not the advertisements on the right. calibre is a very popular program and unscrupulous people try to setup websites offering it for download to fool the unwary.

If you have the official download and your antivirus program is still claiming calibre is a virus, then, your antivirus program is wrong. Antivirus programs use heuristics, patterns of code that „look suspicious“ to detect viruses. It's rather like racial profiling. calibre is a completely open source product. You can actually browse the source code yourself (or hire someone to do it for you) to verify that it is not a virus. Please report the false identification to whatever company you buy your antivirus software from. If the antivirus program is preventing you from downloading/installing calibre, disable it temporarily, install calibre and then re-enable it.

9.4.13 How do I backup calibre?

The most important thing to backup is the calibre library folder, that contains all your books and metadata. This is the folder you chose for your calibre library when you ran calibre for the first time. You can get the path to the library folder by clicking the calibre icon on the main toolbar. You must backup this complete folder with all its files and sub-folders.

You can switch calibre to using a backed up library folder by simply clicking the calibre icon on the toolbar and choosing your backup library folder. A backed up library folder backs up your custom columns and saved searches as well as all your books and metadata.

If you want to backup the calibre configuration/plugins, you have to backup the config folder. You can find this config folder via *Preferences → Miscellaneous*. Note that restoring configuration folders is not officially supported, but should work in most cases. Just copy the contents of the backup folder into the current configuration folder to restore.

9.4.14 How do I use purchased EPUB books with calibre (or what do I do with .acsm files)?

Most purchased EPUB books have *DRM* (stránka 373). This prevents calibre from opening them. You can still use calibre to store and transfer them to your e-book reader. First, you must authorize your reader on a Windows machine with Adobe Digital Editions. Once this is done, EPUB books transferred with calibre will work fine on your reader. When you purchase an epub book from a website, you will get an „.acsm“ file. This file should be opened with Adobe Digital Editions, which will then download the actual „.epub“ e-book. The e-book file will be stored in the folder „My Digital Editions“, from where you can add it to calibre.

[71] https://www.wisecleaner.com
[72] https://calibre-ebook.com/download_portable
[73] https://calibre-ebook.com/download

9.4.15 I am getting a „Permission Denied“ error?

A permission denied error can occur because of many possible reasons, none of them having anything to do with calibre.

- You can get permission denied errors if you are using an SD card with write protect enabled.
- Pokud v systému MacOS zjistíte chyby oprávnění při připojování zařízení ke Calibre, můžete to napravit pod *Předvolby systému* > *Zabezpečení a soukromí* > *Soukromí* > *Soubory a složky*.
- If you, or some program you used changed the file permissions of the files in question to read only.
- If there is a filesystem error on the device which caused your operating system to mount the filesystem in read only mode or mark a particular file as read only pending recovery.
- If the files have their owner set to a user other than you.
- If your file is open in another program.
- If the file resides on a device, you may have reached the limit of a maximum of 256 files in the root of the device. In this case you need to reformat the device/sd card referred to in the error message with a FAT32 filesystem, or delete some files from the SD card/device memory.

You will need to fix the underlying cause of the permissions error before resuming to use calibre. Read the error message carefully, see what file it points to and fix the permissions on that file or its containing folders.

9.4.16 Can I have the comment metadata show up on my reader?

Most readers do not support this. You should complain to the manufacturer about it and hopefully if enough people complain, things will change. In the meantime, you can insert the metadata, including comments into a „Jacket page“ at the start of the e-book, by using the option to „Insert metadata as page at start of book“ during conversion. The option is found in the *Structure detection* section of the conversion settings. Note that for this to have effect you have to *convert* the book. If your book is already in a format that does not need conversion, you can convert from that format to the same format.

Another alternative is to create a catalog in e-book form containing a listing of all the books in your calibre library, with their metadata. Click-and-hold the *Convert* button to access the catalog creation tool. And before you ask, no you cannot have the catalog „link directly to“ books on your reader.

9.4.17 How do I get calibre to use my HTTP proxy?

By default, calibre uses whatever proxy settings are set in your OS. Sometimes these are incorrect, for example, on Windows if you don't use Microsoft Edge then the proxy settings may not be up to date. You can tell calibre to use a particular proxy server by setting the `http_proxy` and `https_proxy` environment variables. The format of the variable is: `http://username:password@servername` you should ask your network administrator to give you the correct value for this variable. Note that calibre only supports HTTP proxies not SOCKS proxies. You can see the current proxies used by calibre in Preferences->Miscellaneous.

9.4.18 I want some feature added to calibre. What can I do?

You have two choices:

1. Create a patch by hacking on calibre and send it to me for review and inclusion. See Development[74].
2. Open a bug requesting the feature[75]. Remember that while you may think your feature request is extremely important/essential, calibre developers might not agree. Fortunately, calibre is open source, which means you always have the option of implementing your feature yourself, or hiring someone to do it for you. Furthermore, calibre has a comprehensive plugin architecture, so you might be able to develop your feature as a plugin, see *Psaní vlastních modulů pro rozšíření funkcí Calibre* (stránka 224).

[74] https://calibre-ebook.com/get-involved
[75] https://calibre-ebook.com/bugs

9.4.19 Why doesn't calibre have an automatic update?

Z mnoha důvodů:

- *There is no need to update every week*. If you are happy with how calibre works turn off the update notification and be on your merry way. Check back to see if you want to update once a year or so. There is a check box to turn off the update notification, on the update notification itself.
- calibre downloads currently use about 150TB of bandwidth a month[76]. Implementing automatic updates would greatly increase that and end up costing thousands of dollars a month, which someone has to pay.
- If I implement a dialog that downloads the update and launches it, instead of going to the website as it does now, that would save the most ardent calibre updater, *at most five clicks a week*. There are far higher priority things to do in calibre development.
- If you really, really hate downloading calibre every week but still want to be up to the latest, I encourage you to run from source, which makes updating trivial. Instructions are *available here* (stránka 343).
- There are third party automatic updaters for calibre made by calibre users in the calibre forum[77].
- Additionally, some third-party updaters such as Norton or Avast may update software behind the user's back. If you find calibre has updated unexpectedly, check for the presence of one.

9.4.20 How is calibre licensed?

calibre is licensed under the GNU General Public License v3 (an open source license). This means that you are free to redistribute calibre as long as you make the source code available. So if you want to put calibre on a CD with your product, you must also put the calibre source code on the CD. The source code is available for download[78]. You are free to use the results of conversions from calibre however you want. You cannot use either code or libraries from calibre in your software without making your software open source. For details, see The GNU GPL v3[79].

9.4.21 How do I run calibre from my USB stick?

A portable version of calibre is available here[80].

9.4.22 How do I run parts of calibre like news download and the Content server on my own Linux server?

First, you must install calibre onto your Linux server. If your server is using a modern Linux distribution, you should have no problems installing calibre onto it.

> **Poznámka**
>
> calibre needs GLIBC >= 2.31 and libstdc++ >= 6.0.28. If you have an older server, you will either need to compile these from source, or use calibre 3.48 which requires GLIBC >= 2.17 or 2.85.1 which requires GLIBC >= 2.13 or calibre 1.48 which requires only GLIBC >= 2.10. In addition, although the calibre command line utilities do not need a running X server, some of them do require the X server libraries to be installed on your system. This is because of Qt, which is used for various image processing tasks, and links against these libraries. If you get an ImportError about some Qt modules, you are likely missing some X libraries. Typical candidates are: `libxcb-cursor0`, `libxcb-xinerama0`, `libegl1`, `libopengl0`.

You can run the calibre server via the command:

[76] https://calibre-ebook.com/dynamic/downloads
[77] https://www.mobileread.com/forums/forumdisplay.php?f=238
[78] https://download.calibre-ebook.com
[79] https://www.gnu.org/licenses/gpl.html
[80] https://calibre-ebook.com/download_portable

```
/opt/calibre/calibre-server /path/to/the/library/you/want/to/share
```

You can download news and convert it into an e-book with the command:

```
/opt/calibre/ebook-convert "Title of news source.recipe" outputfile.epub
```

If you want to generate MOBI, use `outputfile.mobi` instead and use `--output-profile kindle`.

You can email downloaded news with the command:

```
/opt/calibre/calibre-smtp
```

I leave figuring out the exact command line as an exercise for the reader.

Finally, you can add downloaded news to the calibre library with:

```
/opt/calibre/calibredb add --with-library /path/to/library outfile.epub
```

Remember to read the *Rozhraní příkazového řádku* (stránka 303) section of the calibre User Manual to learn more about these, and other commands.

KAPITOLA 10

Kurzy

Zde naleznete kurzy, abyste mohli začít používat pokročilejší funkce Calibre, jako jsou XPath a šablony.

10.1 Správa podskupin knih, například „žánr

Někteří lidé si chtějí uspořádat knihy ve své knihovně do podskupin podobných podsložkám. Nejběžněji udávaný důvod je vytvořit hierarchie žánrů, ale je mnoho dalších. Jeden uživatel požadoval způsob, jak uspořádat učebnice podle předmětu a čísla kurzu. Další uživatel chtěl sledovat dárky podle předmětu a příjemce. Tento kurz bude používat po zbytek tohoto příspěvku příklad žánru.

Než budete pokračovat, pamatujte, že nemluvíme o složkách na pevném disku. Podskupiny nejsou složky souborů. Knihy nebudou nikam kopírovány. Struktura souborů knihovny Calibre není ovlivněna. Namísto toho představujeme způsob, jak uspořádat a zobrazit podskupiny knih v knihovně Calibre.

- *Nastavení* (stránka 155)
- *Hledání* (stránka 157)
- *Omezení* (stránka 157)
- *Užitečné funkce šablon* (stránka 158)

Běžně udávané požadavky pro podskupiny, jako žánry, jsou:

- Podskupina (např. žánr) musí obsahovat (ukazovat na) knihy, ne kategorie knih. To odlišuje podskupiny od uživatelských kategorií Calibre.
- Kniha může být ve více podskupinách (žánrech). To odlišuje podskupiny od fyzických složek souborů.
- Podskupiny (žánry) musí tvořit hierarchii; podskupiny mohou obsahovat podskupiny.

Štítky vám poskytují první dva. Když označíte knihu štítkem žánru, můžete pak použít Prohlížeč štítků (nebo hledat) pro nalezení knih s tímto žánrem, což je první. Mnoho knih může mít stejný štítek nebo štítky, to je druhý. Problém je, že štítky nevyhovují třetímu požadavku. Neposkytují hierarchii.

Funkce hierarchie Calibre vám poskytuje třetí – schopnost vidět žánry ve ‚stromu' a schopnost snadno hledat knihy v žánru nebo podžánru. Například předpokládejme, že vaše struktura žánrů je podobná následující:

```
Genre
    . History
    .. Japanese
    .. Military
    .. Roman
    . Mysteries
    .. English
    .. Vampire
    . Science Fiction
    .. Alternate History
    .. Military
    .. Space Opera
    . Thrillers
    .. Crime
    .. Horror
    etc.
```

Použitím funkce hierarchie můžete vidět tyto žánry v Prohlížeči štítků v podobě stromu, jak je zobrazeno na obrázku. V tomto příkladu je nejkrajnější úroveň (Žánr) vlastním sloupcem, který obsahuje žánry. Žánry obsahující podžánry se zobrazují s malým trojúhelníkem vedle sebe. Kliknutí na tento trojúhelník otevře položku a zobrazí podžánry, jak můžete vidět u Historie a Sci-fi.

Kliknutí na žánr může vyhledat všechny knihy s tímto žánrem nebo podřízené položky tohoto žánru. Například kliknutí na Sci-fi může předat všechny tři podřízené žánry, Alternativní historie, Vojenská, Vesmírná opera. Kliknutí na Alternativní historie předá knihy v tomto žánru, ignoruje ty v žánrech Vojenská a Vesmírná opera. Knihy samozřejmě mohou mít více žánrů. Pokud mát kniha žánr jak Vesmírná opera, tak Vojenská, pak uvidíte tuto knihu při kliknutí na oba žánry. Hledání je probráno podrobněji níže.

Další věc, kterou můžete vidět na obrázku, je, že žánr Vojenská se objevuje dvakrát, jednou pod Historie a jednou pod Sci-fi. Protože jsou žánry v hierarchii, jsou to dva samostatné žánry. Kniha může být v jednom, v druhém nebo

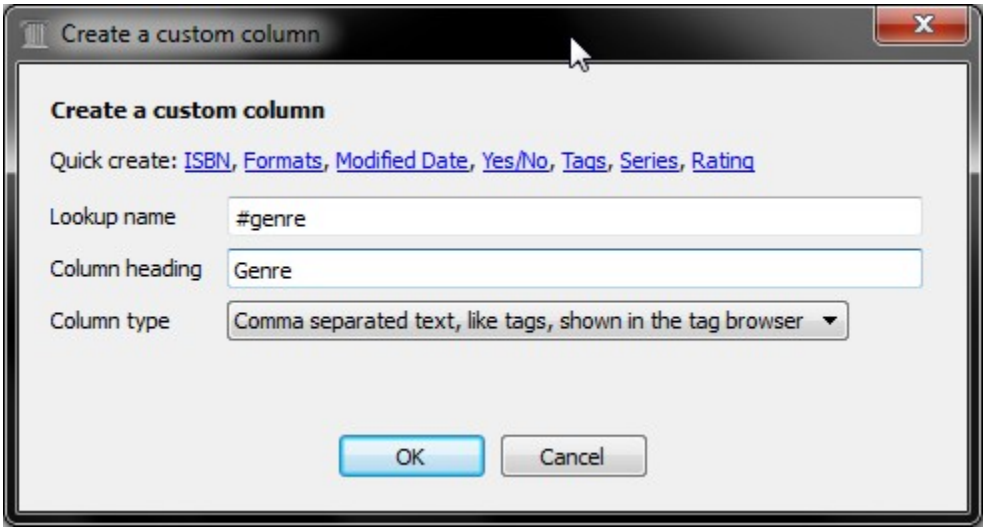
(s pochybami v tomto případě) v obou. Například knihy Winstona Churchilla ze série „Druhá světová válka“ by mohly být v „Historie.Vojenská“. Knihy Davida Weber o Honor Harringtonové by mohly být v „Sci-fi.Vojenská“ a ostatně také v „Sci-fi.Vesmírná opera“.

Jakmile žánr existuje, tj. alespoň jedna kniha má tento žánr, můžete ho snadno použít na další knihy přetažením knih ze zobrazení knihovny na žánr, který chcete, aby knihy měly. Žánry můžete také použít v editorech metadat; více o tomto níže.

10.1.1 Nastavení

Nyní se možná ptáte: „Jak bylo tohle všechno nastaveno?“ Jsou tři kroky: 1) vytvořte vlastní sloupec, 2) řekněte Calibre, že tento nový sloupec má být považován za hierarchii a 3) přidejte žánry.

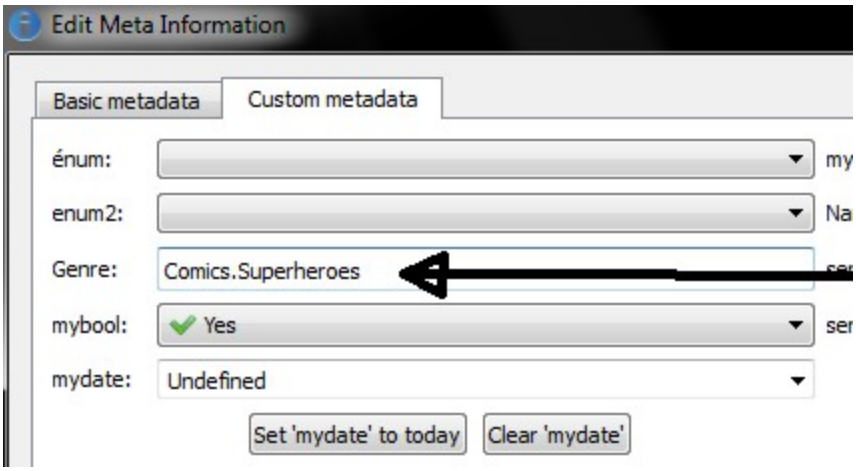
You create the custom column in the usual way, using Preferences -> Add your own columns. This example uses „#genre“ as the lookup name and „Genre“ as the column heading. It is important that the column type is set to *Comma-separated text, like tags, shown in the Tag browser*.



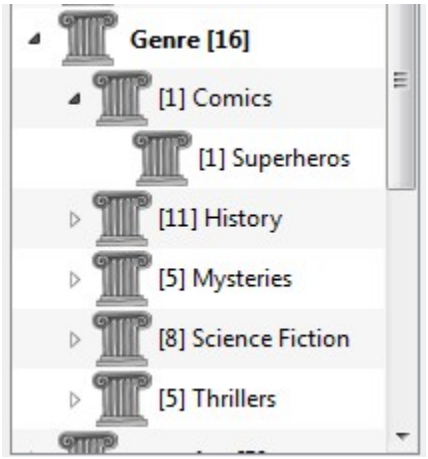
Then after restarting calibre, you must tell calibre that the column is to be treated as a hierarchy. Go to *Preferences → Look & feel → Tag browser → Hierarchy and searching* and choose the new Genre column as having hierarchical items.

V tuto chvíli nejsou v tomto sloupci žádné žánry. Zbývá nám poslední krok: jak použít žánr na knihu. Žánr v Calibre neexistuje, dokud se neobjeví alespoň u jedné knihy. Pro naučení se, jak poprvé použít žánr, musíme probrat podrobnosti o tom, jak žánr vypadá v metadatech knihy.

Hierarchie ‚věcí‘ je vystavěna vytvořením položky skládající se z frází oddělených tečkami. V pokračování příkladu se žánrem by tyto položky byly „Historie.Vojenská“, „Mysteriózní.Upíři“, „Sci-fi.Vesmírná opera“ atd. Tudíž pro vytvoření nového žánru vyberte knihu, která by měla mít tento žánr, upravte její metadata a zadejte nový žánr do sloupce, který jste vytvořili. V pokračování našeho příkladu, pokud chcete ke knize přiřadit nový žánr „Komiks“ s podžánrem „Superhrdinové“, otevřeli byste ‚Upravit metadata‘ pro tuto knihu (komiks), zvolili kartu Vlastní metadata a pak zadali „Komiks.Superhrdinové“, jak je zobrazeno na následujícím obrázku (ignorujte ostatní vlastní sloupce):



Po provedení výše uvedeného uvidíte v Prohlížeči štítků:

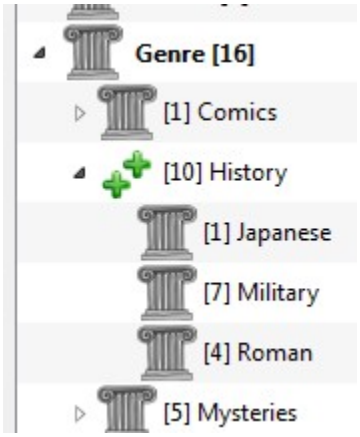


Od teď pro použití tohoto nového žánru na knihu (pravděpodobně komiksu) můžete buď přetáhnout knihu na žánru, nebo ho přidat do knihy pomocí úpravy metadat přesně stejným způsobem, jak je provedeno výše.

Poznámka

Hierarchické zobrazení funguje pouze pokud je Prohlížeč štítků nastaven na řazení položek podle názvu. Toto je výchozí nastavení a lze jej zkontrolovat kliknutím na tlačítko *Konfigurovat* v dolní části Prohlížeče štítků.

10.1.2 Hledání



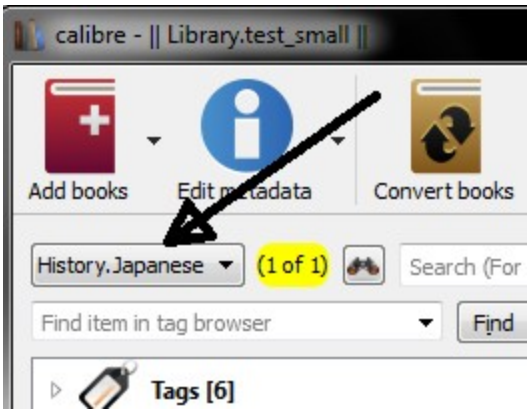
Nejsnadnější způsob hledání žánrů je pomocí Prohlížeče štítků, kliknutím na žánr, který chcete vidět. Kliknutím na žánr s podřízenými položkami vám zobrazí knihy s tímto žánrem a se všemi podřízenými žánry. To však může vyvolat otázku. Jen proto, že má žánr podřízené položky, neznamená to, že to není žánr sám o sobě. Například kniha může mít žánr „Historie“, ale ne „Historie.Vojenská“. Jak hledat knihy, které mají pouze „Historie“?

Mechanismus hledání Prohlížeče štítku ví, pokud má položka podřízené položky. Pokud ano, klepnutí na položku proběhne 5 hledáními namísto normálních tří. První je normální zelené plus, které vám zobrazí knihy pouze s tímto žánrem(např. Historie). Druhé je dvojité plus (zobrazeno výše), která vám ukáže knihy s tímto žánrem a všemi podžánry (např. Historie a Historie.Vojenská). Třetí je normální červené minus, které vám ukáže knihy bez tohoto přesného žánru. Čtvrté je dvojité minus, které vám ukáže knihy, bez tohoto žánru nebo podžánrů. Páté je zpět na začátek, bez označení, což znamená žádné hledání.

10.1.3 Omezení

Pokud hledáte žánr, pak k tomu vytvořte uložené hledání, můžete použít pole ‚omezit na‘ pro vytvoření Virtuální knihovny knih s tímto žánrem. To je užitečné, pokud chcete provést další hledání v rámci žánru, nebo spravovat nebo aktualizovat metadata knih v žánru. V pokračování našeho příkladu můžete vytvořit Uložené hledání s názvem „Historie.Japonská“ nejdříve kliknutím na žánr Japonská v Prohlížeči štítků pro získání hledání do pole hledání, zadáním Historie.Japonská do pole uloženého hledání, a pak stisknutím tlačítka „Uložit hledání“ (zelené pole s bílým plus na pravé straně).

Po vytvoření uloženého hledání ho můžete použít jako omezení.



10.1.4 Užitečné funkce šablon

Možná budete chtít použít informace o žánru v šabloně, jako u uložení na disk nebo odeslat do zařízení. Otázkou by pak mohlo být „Jak získám nejkrajnější název nebo názvy žánru?“ K dispozici je funkce šablony Calibre, podpoložky, aby to udělala snadněji.

Předpokládejme například, že chcete přidat nejkrajnější úroveň žánru do šablony uložit na disk pro vytvoření složek žánrů, jako v „Historie/Blížící se bouře - Churchill, Winston“. Abyste to provedli, musíte extrahovat první úroveň hierarchie a přidat ji před spolu s lomítkem pro označení, že by se měla vytvořit složka. Níže uvedená šablona toto splňuje:

```
{#genre:subitems(0,1)||/}{title} - {authors}
```

Pro více informací o šablonách a funkci `subitems()` se podívejte na *Jazyk šablony* (stránka 160).

10.2 Kurz XPath

V tomto kurzu vám bude poskytnut mírný úvod do XPath[81], dotazovací jazyk, který může být použit pro výběr libovolných částí dokumentů HTML[82] v Calibre. XPath je široce používaný standard a jeho vyhledání na Google vám poskytne mnoho informací. Tento kurz se však zaměřuje na používání XPath pro úkoly spojené s e-knihami, jako je hledání nadpisů kapitol v nestrukturovaném dokumentu HTML.

Obsah

- *Výběr podle názvu značky* (stránka 159)
- *Výběr podle atributů* (stránka 159)
- *Výběr podle obsahu značky* (stránka 159)
- *Ukázková e-kniha* (stránka 160)
- *Vestavěné funkce XPath* (stránka 160)

[81] https://en.wikipedia.org/wiki/XPath
[82] https://en.wikipedia.org/wiki/HTML

10.2.1 Výběr podle názvu značky

Nejjednodušší forma výběru je vybrat značky podle názvu. Například předpokládejme, že chcete vybrat v dokumentu všechny značky `<h2>`. Dotaz XPath pro toto je jednoduše:

```
//h:h2        (Selects all <h2> tags)
```

Předpona *//* znamená *hledat na jakékoliv úrovni dokumentu*. Nyní předpokládejme, že chcete hledat značky `<span>`, které jsou uvnitř značek `<a>`. Toho může být dosaženo pomocí:

```
//h:a/h:span    (Selects <span> tags inside <a> tags)
```

Pokud chcete hledat značky na určité úrovni v dokumentu, změňte předponu:

```
/h:body/h:div/h:p (Selects <p> tags that are children of <div> tags that are
            children of the <body> tag)
```

Toto porovná pouze `<p>Velice krátká e-kniha pro ukázání použití XPath.</p>` v *Ukázková e-kniha* (stránka 160), ale ne žádné další značky `<p>`. Předpona `h:` ve výše uvedených příkladech je zapotřebí pro porovnání značek XHTML. To je proto, že Calibre vnitřně představuje celý obsah jako XHTML. V XHTML mají značky *obor názvů* a `h:` je předpona oboru názvů pro značky HTML.

Nyní předpokládejme, že chcete vybrat jak značky `<h1>`, tak `<h2>`. K tomu potřebujeme konstrukci XPath zvanou *predikát*. *Predikát* je jednoduše test, který je použitý pro výběr značek. Testy mohou být libovolně výkonné, a jak bude tento kurz postupovat, uvidíte výkonnější příklady. Predikát je vytvořen uzavřením testovacího výrazu do hranatých závorek:

```
//*[name()='h1' or name()='h2']
```

V tomto výrazu XPath je několik nových funkcí. První je použití zástupného znaku `*`. Ten znamená *porovnat jakoukoliv značku*. Nyní se podívejte na testovací výraz `name()='h1' or name()='h2'`. *name()* je příklad *vestavěné funkce*. Jednoduše vyhodnocuje podle názvu značky. Takže jejím použitím můžeme vybrat značky, jejichž názvy jsou buď *h1* nebo *h2*. Pamatujte, že funkce *name()* ignoruje obory názvů, takže není potřeba předpona `h:`. XPath má několik užitečných vestavěných funkcí. V tomto kurzu bude představeno několik dalších.

10.2.2 Výběr podle atributů

Pro výběr značek na základě jejich atributů je požadováno použití predikátů:

```
//*[@style]              (Select all tags that have a style attribute)
//*[@class="chapter"]    (Select all tags that have class="chapter")
//h:h1[@class="bookTitle"] (Select all h1 tags that have class="bookTitle")
```

Operátor `@` zde odkazuje na atributy značky. Můžete použít některé z *vestavěných funkcí XPath* (stránka 160) pro provedení propracovanějšího porovnání hodnot atributů.

10.2.3 Výběr podle obsahu značky

Pomocí XPath můžete dokonce vybrat značky na základě textu, který obsahují. Nejlepší způsob, jak to udělat, je použít sílu *regulárních výrazů* pomocí vestavěné funkce *re:test()*:

```
//h:h2[re:test(., 'chapter|section', 'i')] (Selects <h2> tags that contain the words␣
↪chapter or
                                        section)
```

Operátor `.` zde odkazuje na obsah značky, stejně jako operátor `@` odkazoval na její atributy.

10.2.4 Ukázková e-kniha

```
<html>
    <head>
        <title>A very short e-book</title>
        <meta name="charset" value="utf-8" />
    </head>
    <body>
        <h1 class="bookTitle">A very short e-book</h1>
        <p style="text-align:right">Written by Kovid Goyal</p>
        <div class="introduction">
            <p>A very short e-book to demonstrate the use of XPath.</p>
        </div>

        <h2 class="chapter">Chapter One</h2>
        <p>This is a truly fascinating chapter.</p>

        <h2 class="chapter">Chapter Two</h2>
        <p>A worthy continuation of a fine tradition.</p>
    </body>
</html>
```

10.2.5 Vestavěné funkce XPath

name()
: Název aktuální značky.

contains()
: `contains(s1, s2)` vrací *true*, pokud s1 obsahuje s2.

re:test()
: `re:test(zdroj, vzor, příznaky)` vrací *true*, pokud řetězec *zdroj* odpovídá regulárnímu výrazu *vzor*. Obzvlášť užitečný příznak je `i`, porovnání je pak bez rozlišení malých a velkých písmen. Dobré základy syntaxe pro regulární výrazy můžete nalézt na syntaxe regexp[83]

10.3 Jazyk šablon v programu Calibre

The calibre template language is a calibre-specific language used throughout calibre for tasks such as specifying file paths, formatting values, and computing the value for user-specified columns. Examples:

- Specify the folder structure and file names when saving files from the calibre library to the disk or e-book reader.
- Define rules for adding icons and colors to the calibre book list.
- Define *virtual columns* that contain data from other columns.
- Advanced library searching.
- Advanced metadata search and replace.

The language is built around the notion of a *template*, which specifies which book metadata to use, computations on that metadata, and how it is to be formatted.

[83] https://docs.python.org/library/re.html

10.3.1 Basic templates

A basic template consists one or more `template expressions`. A `template expression` consists of text and names in curly brackets (`{}`) that is replaced by the corresponding metadata from the book being processed. For example, the default template in calibre used for saving books to device has 4 `template expressions`:

```
{author_sort}/{title}/{title} - {authors}
```

For the book „The Foundation“ by „Isaac Asimov“ the will become:

```
Asimov, Isaac/The Foundation/The Foundation - Isaac Asimov
```

The slashes are not `template expressions` because they are in between in `{}`. Such text is left where it appears. For example, if the template is:

```
{author_sort} Some Important Text {title}/{title} - {authors}
```

then for „The Foundation“ the template produces:

```
Asimov, Isaac Some Important Text The Foundation/The Foundation - Isaac Asimov
```

A `template expression` can access all the metadata available in calibre, including custom columns (columns you create yourself), by using a column's `lookup name`. To find the lookup name for a *column* (sometimes called *fields*), hover your mouse over the column header in calibre's book list. Lookup names for custom columns always begin with `#`. For series type columns there is an additional field named `#lookup name_index` that is the series index for that book in the series. For example, if you have a custom series column named `#myseries`, there will also be a column named `#myseries_index`. The standard series column's index is named `series_index`.

In addition to the standard column based fields, you also can use:

- `{formats}` - A list of formats available in the calibre library for a book
- `{identifiers:select(isbn)}` - The ISBN of the book

If the metadata for the field for a given book is not defined then the field in the template is replaced by the empty string (`''`). For example, consider the following template:

```
{author_sort}/{series}/{title} {series_index}
```

If Asimov's book „Second Foundation“ is in the series „Foundation“ then the template produces:

```
Asimov, Isaac/Foundation/Second Foundation 3
```

If a series has not been entered for the book then the template produces:

```
Asimov, Isaac/Second Foundation
```

The template processor automatically removes multiple slashes and leading or trailing spaces.

10.3.2 Advanced formatting

In addition to metadata substitution, templates can conditionally include additional text and control how substituted data is formatted.

Conditionally including text

Sometimes you want text to appear in the output only if a field is not empty. A common case is `series` and `series_index` where you want either nothing or the two values separated by a hyphen. calibre handles this case using a special `template expression` syntax.

For example and using the above Foundation example, assume you want the template to produce *Foundation - 3 - Second Foundation*. This template produces that output:

```
{series} - {series_index} - {title}
```

However, if a book has no series the template will produce - - *the title*, which is probably not what you want. Generally, people want the result be the title without the extraneous hyphens. You can accomplish this using the following template syntax:

```
{field:|prefix_text|suffix_text}
```

This `template expression` says that if `field` has the value *XXXX* then the result will be *prefix_textXXXXXsuffix_text*. If `field` is empty (has no value) then the result will be the empty string (nothing) because the prefix and suffix are ignored. The prefix and suffix can contain blanks.

Do not use subtemplates (`{ … }`) or functions (see below) in the prefix or the suffix.

Using this syntax, we can solve the above no-series problem with the template:

```
{series}{series_index:| - | - }{title}
```

The hyphens will be included only if the book has a series index, which it has only if it has a series. Continuing the Foundation example again, the template will produce *Foundation - 1 - Second Foundation*.

Notes:

- You must include the colon after the `lookup name` if you are using a prefix or a suffix.
- You must either use either no or both `|` characters. Using one, as in `{field:| - }`, is not allowed.
- It is OK to provide no text for either the prefix or the suffix, such as in `{series:|| - }`. The template `{title:||}` is the same as `{title}`.

Formatting

Suppose you want the `series_index` to be formatted as three digits with leading zeros. This does the trick:

`{series_index:0>3s}` - Three digits with leading zeros

For trailing zeros, use:

`{series_index:0<3s}` - Three digits with trailing zeros

If you use series indices with fractional values, e.g., 1.1, you might want the decimal points to line up. For example, you might want the indices 1 and 2.5 to appear as 01.00 and 02.50 so that they will sort correctly on a device that does lexical sorting. To do this, use:

`{series_index:0>5.2f}` - Five characters consisting of two digits with leading zeros, a decimal point, then 2 digits after the decimal point.

If you want only the first two letters of the data, use:

`{author_sort:.2}` - Only the first two letters of the author sort name

Much of the calibre template language formatting comes from Python. For more details on the syntax of these advanced formatting operations see the Python documentation[84].

10.3.3 Using templates to define custom columns

Templates can be used to display information that isn't in calibre metadata, or to display metadata differently from calibre's normal format. For example, you might want to show the `ISBN`, a field that calibre does not display. You can accomplish this creating a custom column with the type *Column built from other columns* (hereafter called *composite columns*) and providing a template to generate the displayed text. The column will display the result of evaluating the template. For

[84] https://docs.python.org/3/library/string.html#formatstrings

example, to display the ISBN, create the column and enter `{identifiers:select(isbn)}` in the template box. To display a column containing the values of two series custom columns, separated by a comma, use `{#series1:||, }{#series2}`.

Composite columns can use any template option, including formatting.

Note: You cannot edit the data displayed in a composite column. Instead you edit the source columns. If you edit a composite column, for example by double-clicking it, calibre will open the template for editing, not the underlying data.

10.3.4 Templates and plugboards

Plugboards are used for changing the metadata written into books during send-to-device and save-to-disk operations. A plugboard permits you to specify a template to provide the data to write into the book's metadata. You can use plugboards to modify the following fields: authors, author_sort, language, publisher, tags, title, title_sort. This feature helps people who want to use different metadata in books on devices to solve sorting or display issues.

When you create a plugboard, you specify the format and device for which the plugboard is to be used. A special device is provided, `save_to_disk`, that is used when saving formats (as opposed to sending them to a device). Once you have chosen the format and device, you choose the metadata fields to change, providing templates to supply the new values. These templates are *connected* to their destination fields, hence the name *plugboards*. You can of course use composite columns in these templates.

Plugboards are quite flexible and can be written in Single Function Mode, Template Program Mode, General Program Mode, or Python Template mode.

When a plugboard might apply (Content server, save to disk, or send to device), calibre searches the defined plugboards to choose the correct one for the given format and device. For example, to find the appropriate plugboard for an EPUB book being sent to an ANDROID device, calibre searches the plugboards using the following search order:

- a plugboard with an exact match on format and device, e.g., `EPUB` and `ANDROID`
- a plugboard with an exact match on format and the special `any device` choice, e.g., `EPUB` and `any device`
- a plugboard with the special `any format` choice and an exact match on device, e.g., `any format` and `ANDROID`
- a plugboard with `any format` and `any device`

The tags and authors fields have special treatment, because both of these fields can hold more than one item. A book can have many tags and many authors. When you specify that one of these two fields is to be changed, the template's result is examined to see if more than one item is there. For tags, the result is cut apart wherever calibre finds a comma. For example, if the template produces the value `Thriller, Horror`, then the result will be two tags, `Thriller` and `Horror`. There is no way to put a comma in the middle of a tag.

The same thing happens for authors, but using a different character for the cut, a *&* (ampersand) instead of a comma. For example, if the template produces the value `Blogs, Joe&Posts, Susan`, then the book will end up with two authors, `Blogs, Joe` and `Posts, Susan`. If the template produces the value `Blogs, Joe;Posts, Susan`, then the book will have one author with a rather strange name.

Plugboards affect the metadata written into the book when it is saved to disk or written to the device. Plugboards do not affect the metadata used by `save to disk` and `send to device` to create the file names. Instead, file names are constructed using the templates entered on the appropriate preferences window.

10.3.5 Using functions in templates - Single Function Mode

Suppose you want to display the value of a field in upper case when that field is normally in title case. You can do this using *template functions*. For example, to display the title in upper case use the `uppercase` function, as in `{title:uppercase()}`. To display it in title case, use `{title:titlecase()}`.

Functions go into the format part of the template, after the `:` and before the first `|` or the closing `}` if no prefix/suffix is used. If you have both a format and a function reference, the function comes after a second `:`. Functions return the value of the column specified in the template, suitably modified.

The syntax for using functions is one of:

```
{lookup_name:function(arguments)}
{lookup_name:format:function(arguments)}
{lookup_name:function(arguments)|prefix|suffix}
{lookup_name:format:function(arguments)|prefix|suffix}
```

Function names must always be followed by opening and closing parentheses. Some functions require extra values (arguments), and these go inside the parentheses. Arguments are separated by commas. Literal commas (commas as text, not argument separators) must be preceded by a backslash (`\`) . The last (or only) argument cannot contain a textual closing parenthesis.

Functions are evaluated before format specifications and the prefix/suffix. See further down for an example of using both a format and a function.

Important: If you have programming experience, please note that the syntax in *Single Function Mode* is not what you expect. Strings are not quoted and spaces are significant. All arguments are considered to be constants; there are no expressions.

Do not use subtemplates (`{ … }`) as function arguments. Instead, use *Template Program Mode* (stránka 171) and *General Program Mode* (stránka 166).

Notes on calling functions in Single Function Mode:

- When functions are used in Single Function Mode, the first parameter, `value`, is automatically replaced by the content of the field specified in the template. For example, when the template `{title:capitalize()}` is processed, the content of the `title` field is passed as the parameter `value` to the capitalize function.
- In the function documentation, the notation `[something]*` means that `something` can be repeated zero or more times. The notation `[something]+` means that the `something` is repeated one or more times (must exist at least one time).
- Some functions use regular expressions. In the template language regular expression matching is case-insensitive.

Functions are documented in *Odkaz na funkci šablony* (stránka 180). The documentation tells you what arguments the functions require and what the functions do. For example, here is the documentation of the *ifempty* (stránka 206) function.

- `ifempty(value, text_if_empty)` – if the `value` is not empty then return that `value`, otherwise return `text_if_empty`.

You see that the function requires two arguments, `value` and `text_if_empty`. However, because we are using Single Function Mode, we omit the `value` argument, passing only `text_if_empty`. For example, this template:

```
{tags:ifempty(No tags on this book)}
```

shows the tags for a book, if any. If it has no tags then it show *No tags on this book*.

The following functions are usable in Single Function Mode because their first parameter is `value`.

- *capitalize* (stránka 186) `(value)` – returns the `value` with the first letter in upper case and the rest lower case.
- *ceiling* (stránka 184) `(value)` – returns the smallest integer greater than or equal to `value`.
- *cmp* (stránka 204) `(value, y, lt, eq, gt)` – compares `value` and `y` after converting both to numbers.
- *contains* (stránka 205) `(value, pattern, text_if_match, text_if_not_match)` – checks if the value is matched by the regular expression `pattern`
- *date_arithmetic* (stránka 189) `(value, calc_spec, fmt)` – Calculate a new date from `value` using `calc_spec`.

- *encode_for_url* (stránka 208) `(value, use_plus)` – returns the `value` encoded for use in a URL as specified by `use_plus`. The value is first URL-encoded. Next, if `use_plus` is `0` then spaces are replaced by `'+'` (plus) signs. If it is `1` then spaces are replaced by `%20`.
- *floor* (stránka 185) `(value)` – returns the largest integer less than or equal to `value`.
- *format_date* (stránka 190) `(value, format_string)` – format the `value`, which must be a date string, using the `format_string`, returning a string.
- *format_duration* (stránka 192) `(value, template, [largest_unit])` – format the value, a number of seconds, into a string showing weeks, days, hours, minutes, and seconds. If the value is a float then it is rounded to the nearest integer.
- *format_number* (stránka 193) `(value, template)` – interprets the `value` as a number and formats that number using a Python formatting template such as `{0:5.2f}` or `{0:,d}` or `${0:5,.2f}`.
- *fractional_part* (stránka 185) `(value)` – returns the part of the value after the decimal point.
- *human_readable* (stránka 193) `(value)` – expects the `value` to be a number and returns a string representing that number in KB, MB, GB, etc.
- *ifempty* (stránka 206) `(value, text_if_empty)` – if the `value` is not empty then return that `value`, otherwise return `text_if_empty`.
- *language_strings* (stránka 196) `(value, localize)` – return the language names for the language codes (see here for names and codes[85]) passed in `value`.
- *list_contains* (stránka 198) `(value, separator, [ pattern, found_val, ]* not_found_val)` – interpret the `value` as a list of items separated by `separator`, checking the `pattern` against each item in the list.
- *list_count* (stránka 199) `(value, separator)` – interprets the value as a list of items separated by `separator` and returns the number of items in the list.
- *list_count_matching* (stránka 199) `(value, pattern, separator)` – interprets `value` as a list of items separated by `separator`, returning the number of items in the list that match the regular expression `pattern`.
- *list_item* (stránka 198) `(value, index, separator)` – interpret the `value` as a list of items separated by `separator`, returning the ‚index'th item.
- *list_sort* (stránka 201) `(value, direction, separator)` – return `value` sorted using a case-insensitive lexical sort.
- *lookup* (stránka 198) `(value, [ pattern, key, ]* else_key)` – The patterns will be checked against the `value` in order
- *lowercase* (stránka 186) `(value)` – returns the `value` in lower case.
- *mod* (stránka 185) `(value, y)` – returns the `floor` of the remainder of `value / y`.
- *rating_to_stars* (stránka 193) `(value, use_half_stars)` – Returns the `value` as string of star (★) characters.
- *re* (stránka 206) `(value, pattern, replacement)` – return the `value` after applying the regular expression.
- *re_group* (stránka 206) `(value, pattern [, template_for_group]*)` – return a string made by applying the regular expression `pattern` to `value` and replacing each matched instance
- *round* (stránka 185) `(value)` – returns the nearest integer to `value`.
- *select* (stránka 199) `(value, key)` – interpret the `value` as a comma-separated list of items with each item having the form `id:id_value` (the calibre `identifier` format).

[85] https://www.loc.gov/standards/iso639-2/php/code_list.php

- *shorten* (stránka 206) `(value, left_chars, middle_text, right_chars)` – Return a shortened version of the `value`
- *str_in_list* (stránka 199) `(value, separator, [ string, found_val, ]+ not_found_val)` – interpret the `value` as a list of items separated by `separator` then compare `string` against each value in the list.
- *subitems* (stránka 202) `(value, start_index, end_index)` – This function breaks apart lists of tag-like hierarchical items such as genres.
- *sublist* (stránka 203) `(value, start_index, end_index, separator)` – interpret the `value` as a list of items separated by `separator`, returning a new list made from the items from `start_index` to `end_index`.
- *substr* (stránka 207) `(value, start, end)` – returns the `start`'th through the `end`'th characters of `value`
- *swap_around_articles* (stránka 207) `(value, separator)` – returns the `value` with articles moved to the end, separated by a semicolon.
- *swap_around_comma* (stránka 207) `(value)` – given a `value` of the form `B, A`, return `A B`.
- *switch* (stránka 198) `(value, [patternN, valueN,]+ else_value)` – for each `patternN, valueN` pair, checks if the `value` matches the regular expression `patternN`
- *test* (stránka 207) `(value, text_if_not_empty, text_if_empty)` – return `text_if_not_empty` if the value is not empty, otherwise return `text_if_empty`.
- *titlecase* (stránka 186) `(value)` – returns the `value` in title case.
- *transliterate* (stránka 207) `(value)` – Return a string in a latin alphabet formed by approximating the sound of the words in `value`.
- *uppercase* (stránka 186) `(value)` – returns the `value` in upper case.

Using functions and formatting in the same template

Suppose you have an integer custom column `#myint` that you want displayed with leading zeros, as in `003`. One way to do this is to use a format of `0>3s`. However, by default if a number (integer or float) equals zero then the value is displayed as the empty string so zero values will produce the empty string, not `000`. If you want to see `000` values then you use both the format string and the `ifempty` function to change the empty value back to a zero. The template would be:

```
{#myint:0>3s:ifempty(0)}
```

Note that you can use the prefix and suffix as well. If you want the number to appear as `[003]` or `[000]`, then use the template:

```
{#myint:0>3s:ifempty(0)|[|]}
```

10.3.6 General Program Mode

General Program Mode (*GPM*) replaces *template expressions* with a program written in the *template language*. The syntax of the language is defined by the following grammar:

```
program         ::= 'program:' expression_list
expression_list ::= top_expression [ ';' top_expression ]*
top_expression  ::= or_expression
or_expression   ::= and_expression [ '||' and_expression ]*
and_expression  ::= not_expression [ '&&' not_expression ]*
not_expression  ::= [ '!' not_expression ]* | concatenate_expr
concatenate_expr::= compare_expr [ '&' compare_expr ]*
compare_expr    ::= add_sub_expr [ compare_op add_sub_expr ]
compare_op      ::= '==' | '!=' | '>=' | '>' | '<=' | '<' |
```

(continues on next page)

(pokračujte na předchozí stránce)

```
                    'in' | 'inlist' | 'inlist_field' |
                    '==#' | '!=#' | '>=#' | '>#' | '<=#' | '<#'
add_sub_expr     ::= times_div_expr [ add_sub_op times_div_expr ]*
add_sub_op       ::= '+' | '-'
times_div_expr   ::= unary_op_expr [ times_div_op unary_op_expr ]*
times_div_op     ::= '*' | '/'
unary_op_expr    ::= [ add_sub_op unary_op_expr ]* | expression
expression       ::= identifier | constant | function | assignment | field_reference |
                    if_expr | for_expr | break_expr | continue_expr |
                    '(' expression_list ')' | function_def
field_reference  ::= '$' [ '$' ] [ '#' ] identifier
identifier       ::= id_start [ id_rest ]*
id_start         ::= letter | underscore
id_rest          ::= id_start | digit
constant         ::= " string " | ' string ' | number
function         ::= identifier '(' expression_list [ ',' expression_list ]* ')'
function_def     ::= 'def' identifier '(' top_expression [ ',' top_expression ]* ')' ':
↪'
                    expression_list 'fed'
assignment       ::= identifier '=' top_expression
if_expr          ::= 'if' condition 'then' expression_list
                    [ elif_expr ] [ 'else' expression_list ] 'fi'
condition        ::= top_expression
elif_expr        ::= 'elif' condition 'then' expression_list elif_expr | ''
for_expr         ::= for_list | for_range
for_list         ::= 'for' identifier 'in' list_expr
                    [ 'separator' separator_expr ] ':' expression_list 'rof'
for_range        ::= 'for' identifier 'in' range_expr ':' expression_list 'rof'
range_expr       ::= 'range' '(' [ start_expr ',' ] stop_expr
                    [ ',' step_expr [ ',' limit_expr ] ] ')'
list_expr        ::= top_expression
break_expr       ::= 'break'
continue_expr    ::= 'continue'
separator_expr   ::= top_expression
start_expr       ::= top_expression
stop_expr        ::= top_expression
step_expr        ::= top_expression
limit_expr       ::= top_expression
```

Notes:

- a `top_expression` always has a value. The value of an `expression_list` is the value of the last `top_expression` in the list. For example, the value of the expression list `1;2;'foobar';3` is `3`.
- In a logical context, any non-empty value is `True`
- In a logical context, the empty value is `False`
- Strings and numbers can be used interchangeably. For example, `10` and `'10'` are the same thing.
- Comments are lines starting with a ‚#‘ character, possibly preceded by blanks or tabs.

Operator precedence

The operator precedence (order of evaluation) from highest (evaluated first) to lowest (evaluated last) is:

- Function calls, constants, parenthesized expressions, statement expressions, assignment expressions, field references.

- Unary plus (+) and minus (-). These operators evaluate right to left.

  These and all the other arithmetic operators return integers if the expression results in a fractional part equal to zero. For example, if an expression returns `3.0` it is changed to `3`.

- Multiply (`*`) and divide (`/`). These operators are associative and evaluate left to right. Use parentheses if you want to change the order of evaluation.

- Add (+) and subtract (-). These operators are associative and evaluate left to right.

- Numeric and string comparisons. These operators return `'1'` if the comparison succeeds, otherwise the empty string (`''`). Comparisons are not associative: `a < b < c` is a syntax error.

- String concatenation (`&`). The `&` operator returns a string formed by concatenating the left-hand and right-hand expressions. Example: `'aaa' & 'bbb'` returns `'aaabbb'`. The operator is associative and evaluates left to right.

- Unary logical not (`!`). This operator returns `'1'` if the expression is False (evaluates to the empty string), otherwise `''`.

- Logical and (`&&`). This operator returns ‚1' if both the left-hand and right-hand expressions are True, or the empty string `''` if either is False. It is associative, evaluates left to right, and does short-circuiting[86].

- Logical or (`||`). This operator returns `'1'` if either the left-hand or right-hand expression is True, or `''` if both are False. It is associative, evaluates left to right, and does short-circuiting[87]. It is an *inclusive or*, returning `'1'` if both the left- and right-hand expressions are True.

Field references

A `field_reference` evaluates to the value of the metadata field named by lookup name that follows the `$` or `$$`. Using `$` is equivalent to using the *field* (stránka 195) function. Using `$$` is equivalent to using the *raw_field* (stránka 197) function. Examples:

```
* $authors ==> field('authors')
* $#genre ==> field('#genre')
* $$pubdate ==> raw_field('pubdate')
* $$#my_int ==> raw_field('#my_int')
```

If expressions

`If` expressions first evaluate the `condition`. If the `condition` is True (a non-empty value) then the `expression_list` in the `then` clause is evaluated. If it is False then if present the `expression_list` in the `elif` or `else` clause is evaluated. The `elif` and `else` parts are optional. The words `if`, `then`, `elif`, `else`, and `fi` are reserved; you cannot use them as identifier names. You can put newlines and white space wherever they make sense. The `condition` is a `top_expression` not an `expression_list`; semicolons are not allowed. The `expression_lists` are semicolon-separated sequences of `top_expressions`. An `if` expression returns the result of the last `top_expression` in the evaluated `expression_list`, or the empty string if no expression list was evaluated.

Examples:

```
* program: if field('series') then 'yes' else 'no' fi
* program:
      if field('series') then
          a = 'yes';
          b = 'no'
      else
          a = 'no';
          b = 'yes'
```

(continues on next page)

[86] https://chortle.ccsu.edu/java5/Notes/chap40/ch40_2.html
[87] https://chortle.ccsu.edu/java5/Notes/chap40/ch40_2.html

(pokračujte na předchozí stránce)

```
    fi;
    strcat(a, '-', b)
```

Nested `if` example:

```
program:
  if field('series') then
    if check_yes_no(field('#mybool'), '', '', '1') then
      'yes'
    else
      'no'
    fi
  else
    'no series'
  fi
```

As said above, an `if` produces a value. This means that all the following are equivalent:

```
* program: if field('series') then 'foo' else 'bar' fi
* program: if field('series') then a = 'foo' else a = 'bar' fi; a
* program: a = if field('series') then 'foo' else 'bar' fi; a
```

For example, this program returns the value of the `series` column if the book has a series, otherwise the value of the `title` column:

```
program: field(if field('series') then 'series' else 'title' fi)
```

For expressions

The `for` expression iterates over a list of values, processing them one at a time. The `list_expression` must evaluate either to a metadata field `lookup name` e.g., `tags` or `#genre`, or to a list of values. The *range* (stránka 202) generates a list of numbers. If the result is a valid `lookup name` then the field's value is fetched and the separator specified for that field type is used. If the result isn't a valid lookup name then it is assumed to be a list of values. The list is assumed to be separated by commas unless the optional keyword `separator` is supplied, in which case the list values must be separated by the result of evaluating the `separator_expr`. A separator cannot be used if the list is generated by `range()`. Each value in the list is assigned to the specified variable then the `expression_list` is evaluated. You can use `break` to jump out of the loop, and `continue` to jump to the beginning of the loop for the next iteration.

Example: This template removes the first hierarchical name for each value in Genre (`#genre`), constructing a list with the new names:

```
program:
  new_tags = '';
  for i in '#genre':
    j = re(i, '^.*?\.(.*)$', '\1');
    new_tags = list_union(new_tags, j, ',')
  rof;
  new_tags
```

If the original Genre is *History.Military, Science Fiction.Alternate History, ReadMe* then the template returns *Military, Alternate History, ReadMe*. You could use this template in calibre's *Edit metadata in bulk* → *Search & replace* with *Search for* set to `template` to strip off the first level of the hierarchy and assign the resulting value to Genre.

Note: the last line in the template, `new_tags`, isn't strictly necessary in this case because `for` returns the value of the last top_expression in the expression list. The value of an assignment is the value of its expression, so the value of the `for`

statement is what was assigned to `new_tags`.

Function definition

If you have repeated code in a template then you can put that code into a local function. The `def` keyword starts the definition. It is followed by the function name, the argument list, then the code in the function. The function definition ends with the `fed` keyword.

Arguments are positional. When a function is called the supplied arguments are matched left to right against the defined parameters, with the value of the argument assigned to the parameter. It is an error to provide more arguments than defined parameters. Parameters can have default values, such as `a = 25`. If an argument is not supplied for that parameter then the default value is used, otherwise the parameter is set to the empty string.

The `return` statement can be used in a local function.

A function must be defined before it can be used.

Example: This template computes an approximate duration in years, months, and days from a number of days. The function `to_plural()` formats the computed values. Note that the example also uses the `&` operator:

```
program:
      days = 2112;
      years = floor(days/360);
      months = floor(mod(days, 360)/30);
      days = days - ((years*360) + (months * 30));

      def to_plural(v, str):
              if v == 0 then return '' fi;
              return v & ' ' & (if v == 1 then str else str & 's' fi) & ' '
      fed;

      to_plural(years, 'year') & to_plural(months, 'month') & to_plural(days,'day')
```

Relational operators

Relational operators return `'1'` if the comparison is true, otherwise the empty string (`''`).

There are two forms of relational operators: string comparisons and numeric comparisons.

String comparisons do case-insensitive string comparison using lexical order. The supported string comparison operators are `==`, `!=`, `<`, `<=`, `>`, `>=`, `in`, `inlist`, and `inlist_field`. For the `in`, `inlist`, and `inlist_field` operators, the result of the left hand expression is interpreted as a regular expression pattern. They are true if the value of left-hand regular expression matches the value of the right hand expression. The regular expressions are case-insensitive.

The `inlist` operator is true if the left hand regular expression matches any one of the items in the right hand list where the items in the list are separated by commas. The `inlist_field` operator is true if the left hand regular expression matches any of the items in the field (column) named by the right hand expression, using the separator defined for the field. NB: the `inlist_field` operator requires the right hand expression to evaluate to a field name, while the `inlist` operator requires the right hand expression to evaluate to a string containing a comma-separated list. Because of this difference, `inlist_field` is substantially faster than `inlist` because no string conversions or list constructions are done.

The numeric comparison operators are `==#`, `!=#`, `<#`, `<=#`, `>#`, `>=#`. The left and right expressions must evaluate to numeric values with two exceptions: both the string value „None“ (undefined field) and the empty string evaluate to the value zero.

Příklady:

- `program: field('series') == 'foo'` returns `'1'` if the book's series is *foo*, otherwise `''`.

- `program: 'f.o' in field('series')` returns `'1'` if the book's series matches the regular expression `f.o` (e.g., *foo*, *Off Onyx*, etc.), otherwise `''`.
- `program: 'science' inlist $#genre` returns `'1'` if any of the values retrieved from the book's genres match the regular expression `science`, e.g., *Science*, *History of Science*, *Science Fiction* etc., otherwise `''`.
- `program: '^science$' inlist $#genre` returns `'1'` if any of the book's genres exactly match the regular expression `^science$`, e.g., *Science*, otherwise `''`. The genres *History of Science* and *Science Fiction* don't match.
- `program: 'asimov' inlist $authors` returns `'1'` if any author matches the regular expression `asimov`, e.g., *Asimov, Isaac* or *Isaac Asimov*, otherwise `''`.
- `program: 'asimov' inlist_field 'authors'` returns `'1'` if any author matches the regular expression `asimov`, e.g., *Asimov, Isaac* or *Isaac Asimov*, otherwise `''`.
- `program: 'asimov$' inlist_field 'authors'` returns `'1'` if any author matches the regular expression `asimov$`, e.g., *Isaac Asimov*, otherwise `''`. It doesn't match *Asimov, Isaac* because of the `$` anchor in the regular expression.
- `program: if field('series') != 'foo' then 'bar' else 'mumble' fi` returns `'bar'` if the book's series is not *foo*. Otherwise it returns `'mumble'`.
- `program: if field('series') == 'foo' || field('series') == '1632' then 'yes' else 'no' fi` returns `'yes'` if series is either *foo* or *1632*, otherwise `'no'`.
- `program: if '^(foo|1632)$' in field('series') then 'yes' else 'no' fi` returns `'yes'` if series is either *foo* or *1632*, otherwise `'no'`.
- `program: if 11 > 2 then 'yes' else 'no' fi` returns `'no'` because the `>` operator does a lexical comparison.
- `program: if 11 ># 2 then 'yes' else 'no' fi` returns `'yes'` because the `>#` operator does a numeric comparison.

Functions in General Program Mode

See *Odkaz na funkci šablony* (stránka 180) for the list of functions built into the template language.

Notes:

- As opposed to *Single Function Mode* (stránka 163), in General Program Mode you must specify the first parameter `value`.
- All parameters are expression_lists (see the grammar above).

10.3.7 More complex programs in template expressions - Template Program Mode

Template Program Mode (*TPM*) is a blend of *General Program Mode* (stránka 166) and *Single Function Mode* (stránka 163). *TPM* differs from Single Function Mode in that it permits writing template expressions that refer to other metadata fields, use nested functions, modify variables, and do arithmetic. It differs from *General Program Mode* in that the template is contained between `{` and `}` characters and doesn't begin with the word `program:`. The program portion of the template is a General Program Mode expression list.

Example: assume you want a template to show the series for a book if it has one, otherwise show the value of a custom field #genre. You cannot do this in the *Single Function Mode* (stránka 163) because you cannot make reference to another metadata field within a template expression. In *TPM* you can, as the following expression demonstrates:

```
{series:'ifempty($, $#genre)'}
```

The example shows several things:

- *TPM* is used if the expression begins with `:'` and ends with `'}`. Anything else is assumed to be in *Single Function Mode* (stránka 163).

  If the template contains a prefix and suffix, the expression ends with `'|` where the `|` is the delimiter for the prefix. Example:

```
{series:'ifempty($, $#genre)'|prefix | suffix}
```

- Functions must be given all their arguments. For example, the standard built-in functions must be given the initial parameter `value`.
- The variable `$` is usable as the `value` argument and stands for the value of the field named in the template, `series` in this case.
- white space is ignored and can be used anywhere within the expression.
- constant strings are enclosed in matching quotes, either `'` or `"`.

In *TPM*, using `{` and `}` characters in string literals can lead to errors or unexpected results because they confuse the template processor. It tries to treat them as template expression boundaries, not characters. In some but not all cases you can replace a `{` with `[[` and a `}` with *]]*. Advice: if your program contains `{` and `}` characters then you should use *General Program Mode*.

10.3.8 Python Template Mode

Python Template Mode (PTM) lets you write templates using native Python and the calibre API[88]. The database API will be of most use; further discussion is beyond the scope of this manual. PTM templates are faster and can do more complicated operations but you must know how to write code in Python using the calibre API.

A PTM template begins with:

```
python:
def evaluate(book, context):
    # book is a calibre metadata object
    # context is an instance of calibre.utils.formatter.PythonTemplateContext,
    # which currently contains the following attributes:
    # db: a calibre legacy database object.
    # globals: the template global variable dictionary.
    # arguments: is a list of arguments if the template is called by a GPM template,␣
↪otherwise None.
    # funcs: used to call Built-in/User functions and Stored GPM/Python templates.
    # Example: context.funcs.list_re_group()

    # your Python code goes here
    return 'a string'
```

You can add the above text to your template using the context menu, usually accessed with a right click. The comments are not significant and can be removed. You must use python indenting.

The context object supports `str(context)` that returns a string of the context's contents, and `context.attributes` that returns a list of the attribute names in the context.

The `context.funcs` attribute allows calling Built-in and User template functions, and Stored GPM/Python templates, so that you can execute them directly in your code. The functions are retrieved using their names. If the name conflicts with a Python keyword, add an underscore to the end of the name. Examples:

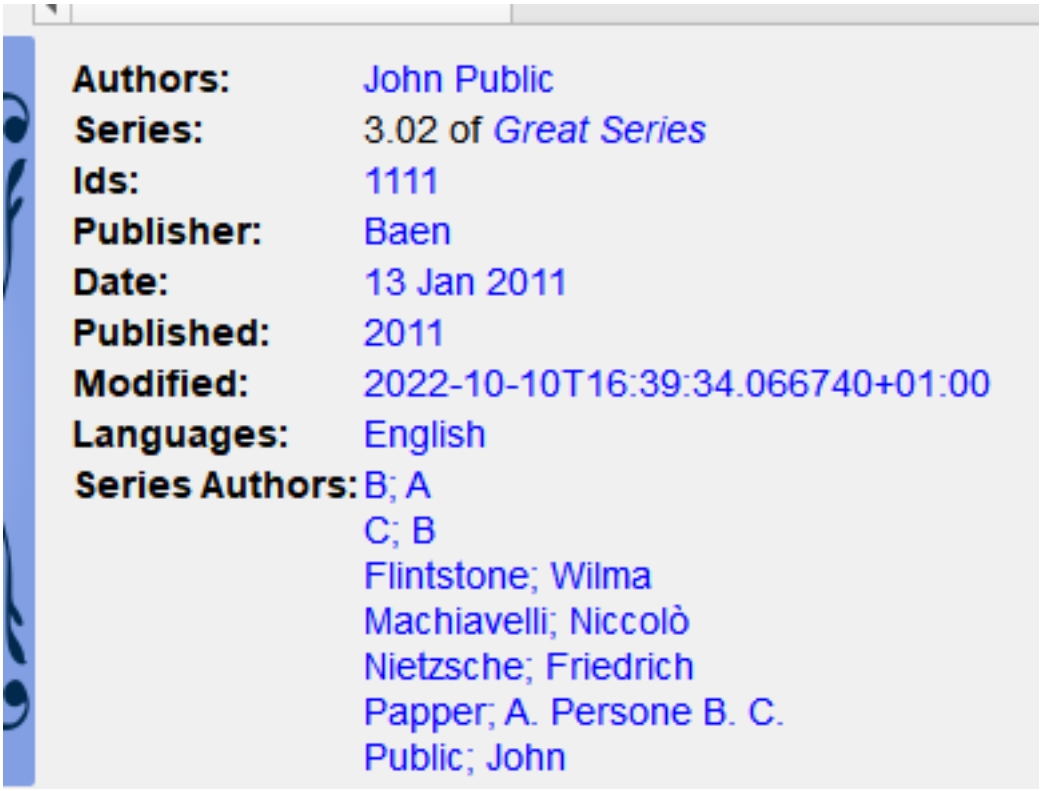
[88] https://manual.calibre-ebook.com/develop.html#api-documentation-for-various-parts-of-calibre

```
context.funcs.list_re_group()
context.funcs.assert_()
```

Here is an example of a PTM template that produces a list of all the authors for a series. The list is stored in a *Column built from other columns, behaves like tags*. It shows in *Book details* and has the *on separate lines* checked (in *Preferences → Look & feel → Book details*). That option requires the list to be comma-separated. To satisfy that requirement the template converts commas in author names to semicolons then builds a comma-separated list of authors. The authors are then sorted, which is why the template uses author_sort.

```
python:
def evaluate(book, context):
    if book.series is None:
        return ''
    db = context.db.new_api
    ans = set()
    # Get the list of books in the series
    ids = db.search(f'series:"={book.series}"', '')
    if ids:
        # Get all the author_sort values for the books in the series
        author_sorts = (v for v in db.all_field_for('author_sort', ids).values())
        # Add the names to the result set, removing duplicates
        for aus in author_sorts:
            ans.update(v.strip() for v in aus.split('&'))
    # Make a sorted comma-separated string from the result set
    return ', '.join(v.replace(',', ';') for v in sorted(ans))
```

The output in *Book details* looks like this:



10.3.9 Templates and URLs

You can use templates to construct URLs. Two cases are described here:

- Custom column *Book details* search URLs
- The calibre URL scheme

Custom column book details search URLs

When you create a custom column you can provide a URL to be used in *Book details* using a template. For example, if you have a custom column for *Translators* you can define a URL to take you to a site for translators. Book details search URLs can be provided for *Text*, *Enumerated*, *Series*, and *Column built from other column* column types.

When an item with a *search template* is clicked in *Book details* the template is evaluated. It is provided the normal book metadata. It is also provided three additional fields:

- `item_value`: the value of the clicked item.
- `item_value_quoted`: the value of clicked item, URL-encoded. Special characters are escaped to make them valid in URLs and spaces are replaced by `'+'` (plus) signs.
- `item_value_no_plus`: the value of clicked item, URL-encoded. Special characters are escaped to make them valid in URLs and spaces are replaced by the `%20`, not plus.

There are several ways to construct the URL. The following use Wikipedia as an example.

The simplest is a basic template:

```
https://en.wikipedia.org/w/index.php?search={item_value_encoded}
```

In some cases you might want to do more processing. There are four template functions you can use, depending on the complexity of the processing.

- *make_url* (stránka 208) `(path, [query_name, query_value]+)` – this function is the easiest way to construct a query URL. It uses a `path`, the web site and page you want to query, and `query_name`, `query_value` pairs from which the query is built. In general, the `query_value` must be URL-encoded. With this function it is always encoded and spaces are always replaced with `'+'` signs.
- *make_url_extended* (stránka 208) `(...)` – this function is similar to *make_url()* (stránka 208) but gives you more control over the URL components. The components of a URL are

  scheme://**authority**/**path**?**query string**.

  See Uniform Resource Locator[89] on Wikipedia for more detail.

  The function has two variants:

  ```
  make_url_extended(scheme, authority, path, [query_name, query_value]+)
  ```

  and

  ```
  make_url_extended(scheme, authority, path, query_string)
  ```

- *query_string* (stránka 209) `([query_name, query_value, how_to_encode]+)` – returns a URL query string constructed from the `query_name, query_value, how_to_encode` triads. A query string is a series of items where each item looks like `query_name=query_value` where `query_value` is URL-encoded as instructed. The query items are separated by `'&'` (ampersand) characters.
- *encode_for_url* (stránka 208) `(value, use_plus)` – returns the `value` encoded for use in a URL as specified by `use_plus`. The value is first URL-encoded. Next, if `use_plus` is `0` then spaces are replaced by `'+'` (plus) signs. If it is `1` then spaces are replaced by `%20`.

[89] https://en.wikipedia.org/wiki/URL

For example, assume you have a custom column *Translators* (`#translators`) where the names are *Last name, First name*. You might need to convert the name to *First name Last name* when creating the URL. You can use the *make_url* (stránka 208) function to do this:

```
program: make_url('https://en.wikipedia.org/w/index.php', 'search', swap_around_comma(
↪$item_value))
```

If we assume that the translator's name is *Boy-Żeleński, Tadeusz* then the above template produces the link:

```
https://en.wikipedia.org/w/index.php?search=Tadeusz+Boy-%C5%BBele%C5%84ski
```

Note that the person's first name is now first, the space is now a plus, and that the non-English characters in the last name are URL-encoded.

The functions *make_url_extended* (stránka 208), *query_string* (stránka 209), and *encode_for_url* (stránka 208) might be useful depending upon any additional processing complexity.

The calibre URL scheme

Calibre supports several different URLs to navigate your calibre libraries. This section shows how to use templates to construct some of the URLs. See *calibre:// URL schéma* (stránka 251) for details on the URLs available.

- Switch to a specific library. The syntax of this URL is:

  ```
  calibre://switch-library/Library_Name
  ```

  `Library_Name` must be replaced with the name of the calibre library you wish to open. The library name is shown in the title bar of the window. It is a simple name, not the file path to the library. You must spell it as shown in the title bar, including letter case. The character _ (underscore) stands for the current library. If the name contains any spaces or special characters then it must be hex encoded using the *to_hex* (stránka 210) function, as in the following example:

  ```
  program: strcat('calibre://switch-library/_hex_-', to_hex(current_library_name()))
  ```

  The template generates the URL:

  ```
  calibre://switch-library/_hex_-4c6962726172792e746573745f736d616c6c
  ```

  You can replace the `current_library_name()` function with the actual name of the library, as in:

  ```
  program: strcat('calibre://switch-library/_hex_-', to_hex('Library.test_small'))
  ```

- Links to show books. These links select a book in the calibre library. The syntax for this URL is:

  ```
  calibre://show-book/Library_Name/book_id
  ```

  The `book id` is the numeric calibre id for the book, available to templates as `$id`. As above, the library name might need to be hex encoded. Here is an example:

  ```
  program: strcat('calibre://show-book/_hex_-', to_hex(current_library_name()), '/',
  ↪ $id)
  ```

  It produces the URL:

  ```
  calibre://show-book/_hex_-4c6962726172792e746573745f736d616c6c/1353
  ```

- Searching for books. These links search for books in the specified calibre library. The syntax for this URL is:

```
calibre://search/Library_Name?q=query
calibre://search/Library_Name?eq=hex_encoded_query
```

where *query* is any valid calibre search expression. You must hex encode any query containing spaces or special characters, which generally means all of them. For example, the calibre search expression for searching for a hierarchical tag beginning with ‚AA' is `tags:"=.AA"`. This template constructs a search URL for that expression:

```
program: strcat('calibre://search/_hex_-', to_hex(current_library_name()), '?eq=',
↪ to_hex('tags:"=.AA"'))
```

The resulting URL is:

```
calibre://search/_hex_-4c6962726172792e746573745f736d616c6c?
↪eq=746167733a223d2e414122
```

Here is an example of the same URL built using the :ref:`ff_make_url_extended` function instead of *strcat* (stránka 206):

```
program: make_url_extended('calibre', '', 'search/_hex_-' & to_hex(current_
↪library_name()),
                           'eq', to_hex('tags:"=.AA"'))
```

- Open a book details window on a book in some library. The syntax for this URL is:

```
calibre://book-details/Library_Name/book_id
```

An example template is:

```
program: strcat('calibre://book-details/_hex_-', to_hex(current_library_name()),
↪'/', $id)
```

which produces the URL:

```
calibre://book-details/_hex_-4c6962726172792e746573745f736d616c6c/1353
```

- Open the notes associated with an author/series/etc. The syntax of the URL is:

```
calibre://book-details/Library_Name/Field_Name/id_Item_Id
calibre://book-details/Library_Name/Field_Name/hex_Hex_Encoded_Item_Name
```

`Field_Name` is the lookup name of the field. If the field is a custom column then replace the `#` character with an underscore (_). `Item_Id` is the internal numeric ID of the value in the field. There isn't a template function that returns the `Item_Id`, so templates will normally use the second form, `Hex_Encoded_Item_Name`. Here is a sample template that opens the note for the person `Boy-Żeleński, Tadeusz` in the field `#authtest`:

```
program: strcat('calibre://show-note/_hex_-', to_hex(current_library_name()),
                '/_authtest/hex_', to_hex('Boy-Żeleński, Tadeusz'))
```

which produces the URL:

```
calibre://show-note/_hex_-4c6962726172792e746573745f736d616c6c/_authtest/hex_
↪426f792dc5bb656c65c584736b692c205461646575737a
```

10.3.10 Uložené šablony

Both *General Program Mode* (stránka 166) and *Python Template Mode* (stránka 172) support saving templates and calling those templates from another template, much like calling stored functions. You save templates using *Preferences → Advanced → Template functions*. More information is provided in that dialog. You call a template the same way you call a function, passing positional arguments if desired. An argument can be any expression. Examples of calling a template, assuming the stored template is named `foo`:

- `foo()` – call the template passing no arguments.
- `foo(a, b)` call the template passing the values of the two variables `a` and `b`.
- `foo(if field('series') then field('series_index') else 0 fi)` – if the book has a `series` then pass the `series_index`, otherwise pass the value `0`.

In GPM you retrieve the arguments passed in the call to the stored template using the `arguments` function. It both declares and initializes local variables, effectively parameters. The variables are positional; they get the value of the parameter given in the call in the same position. If the corresponding parameter is not provided in the call then `arguments` assigns that variable the provided default value. If there is no default value then the variable is set to the empty string. For example, the following `arguments` function declares 2 variables, `key`, `alternate`:

```
arguments(key, alternate='series')
```

Examples, again assuming the stored template is named `foo`:

- `foo('#myseries')` – argument `key` is assigned the value `'myseries'` and the argument `alternate` is assigned the default value `'series'`.
- `foo('series', '#genre')` the variable `key` is assigned the value `'series'` and the variable `alternate` is assigned the value `'#genre'`.
- `foo()` – the variable `key` is assigned the empty string and the variable `alternate` is assigned the value `'series'`.

In PTM the arguments are passed in the `arguments` parameter, which is a list of strings. There isn't any way to specify default values. You must check the length of the `arguments` list to be sure that the number of arguments is what you expect.

An easy way to test stored templates is using the `Template tester` dialog. For ease of access give it a keyboard shortcut in *Preferences → Advanced → Keyboard shortcuts → Template tester*. Giving the `Stored templates` dialog a shortcut will help switching more rapidly between the tester and editing the stored template's source code.

10.3.11 Providing additional information to templates

A developer can choose to pass additional information to the template processor, such as application-specific book metadata or information about what the processor is being asked to do. A template can access this information and use it during the evaluation.

Developer: how to pass additional information

The additional information is a Python dictionary containing pairs `variable_name: variable_value` where the values must be strings. The template can access the dictionary, creating template local variables named `variable_name` containing the value `variable_value`. The user cannot change the name so it is best to use names that won't collide with other template local variables, for example by prefixing the name with an underscore.

This dictionary is passed to the template processor (the `formatter`) using the named parameter `global_vars=your_dict`. The full method signature is:

```
def safe_format(self, fmt, kwargs, error_value, book,
                column_name=None, template_cache=None,
```

(continues on next page)

(pokračujte na předchozí stránce)

```
                strip_results=True, template_functions=None,
                global_vars={})
```

Template writer: how to access the additional information

You access the additional information (the `globals` dictionary) in a template using the template function:

```
globals(id[=expression] [, id[=expression]]*)
```

where `id` is any legal variable name. This function checks whether the additional information provided by the developer contains the name. If it does then the function assigns the provided value to a template local variable with that name. If the name is not in the additional information and if an `expression` is provided, the `expression` is evaluated and the result is assigned to the local variable. If neither a value nor an expression is provided, the function assigns the empty string (`''`) to the local variable.

A template can set a value in the `globals` dictionary using the template function:

```
set_globals(id[=expression] [, id[=expression]]*)
```

This function sets the `globals` dictionary key:value pair `id:value` where `value` is the value of the template local variable `id`. If that local variable doesn't exist then `value` is set to the result of evaluating `expression`.

10.3.12 Notes on the difference between modes

The three program modes, *Single Function Mode* (stránka 163) (SFM), *Template Program Mode* (stránka 171) (*TPM*), and *General Program Mode* (stránka 166) (*GPM*), work differently. SFM is intended to be ‚simple' so it hides a lot of programming language bits.

Differences:

- In SFM the value of the column is always passed as an ‚invisible' first argument to a function included in the template.
- SFM doesn't support the difference between variables and strings; all values are strings.
- The following SFM template returns either the series name or the string „no series":

  ```
  {series:ifempty(no series)}
  ```

  The equivalent template in *TPM* is

  ```
  {series:'ifempty($, 'no series')'}
  ```

  The equivalent template in *GPM* is:

  ```
  program: ifempty(field('series'), 'no series')
  ```

  The first argument to `ifempty` is the value of the field `series`. The second argument is the string `no series`. In SFM the first argument, the value of the field, is automatically passed (the invisible argument).
- Several template functions, for example `booksize()` and `current_library_name()`, take no arguments. Because of the ‚invisible argument' you cannot use these functions in SFM.
- Nested functions, where a function calls another function to compute an argument, cannot be used in SFM. For example this template, intended to return the first 5 characters of the series value uppercased, won't work in SFM:

  ```
  {series:uppercase(substr(0,5))}
  ```

- *TPM* and *GPM* support nested functions. The above template in *TPM* would be:

```
{series:'uppercase(substr($, 0,5))'}
```

  In *GPM* it would be:

```
program: uppercase(substr(field('series'), 0,5))
```

- As noted in the above *Template Program Mode* (stránka 171) section, using { and } characters in *TPM* string literals can lead to errors or unexpected results because they confuse the template processor. It tries to treat them as template boundaries, not characters. In some but not all cases you can replace a { with [[and a } with]]. Generally, if your program contains { and } characters then you should use *General Program Mode*.

10.3.13 User-defined Python template functions

You can add your own Python functions to the template processor. Such functions can be used in any of the three template programming modes. The functions are added by going to *Preferences → Advanced → Template functions*. Instructions are shown in that dialog. Note that you can use *Python Templates* for a similar purpose. As calling user-defined functions is faster than calling a Python template, user-defined functions might be more efficient depending on the complexity of what the function or template does.

10.3.14 Special notes for using templates in different contexts

In the GUI (*Columns made from other columns* and *Template searches*):

- GPM templates work as before.
- Python templates have full access to the calibre database.

In icon rules:

- icon rule templates have no book data so field-based functions such as *format_date_field* (stránka 191), *list_count_field* (stránka 199), and *check_yes_no* (stránka 205) won't work.

In the Content server:

- Templates have access to the new API but not the old API (LibraryDatabase).
- Because of the above, the following formatter functions are not guaranteed to work in GPM templates (composite columns, icon rules, etc) and should be avoided if you use the content server:
  - *connected_device_name* (stránka 194)
  - *connected_device_uuid* (stránka 194)
  - *current_virtual_library_name* (stránka 195)
  - *is_marked* (stránka 196)
  - *virtual_libraries* (stránka 197)

10.3.15 Special notes for save/send templates

Special processing is applied when a template is used in a *Save to disk* or *Send to device* template. The values of the fields are cleaned, replacing characters that are special to file systems with underscores, including slashes. This means that field text cannot be used to create folders. However, slashes are not changed in prefix or suffix strings, so slashes in these strings will cause folders to be created. Because of this, you can create variable-depth folder structure.

For example, assume we want the folder structure *series/series_index - title*, with the caveat that if series does not exist, then the title should be in the top folder. The template to do this is:

```
{series:||/}{series_index:|| - }{title}
```

The slash and the hyphen appear only if series is not empty.

The lookup function lets us do even fancier processing. For example, assume that if a book has a series, then we want the folder structure *series/series index - title.fmt*. If the book does not have a series then we want the folder structure *genre/author_sort/title.fmt*. If the book has no genre then we want to use ‚Unknown'. We want two completely different paths, depending on the value of series.

To accomplish this, we:

1. Create a composite field (give it lookup name #aa) containing `{series}/{series_index} - {title}`. If the series is not empty, then this template will produce *series/series_index - title*.
2. Create a composite field (give it lookup name #bb) containing `{#genre:ifempty(Unknown)}/{author_sort}/{title}`. This template produces *genre/author_sort/title*, where an empty genre is replaced with *Unknown*.
3. Set the save template to `{series:lookup(.,#aa,#bb)}`. This template chooses composite field `#aa` if series is not empty and composite field `#bb` if series is empty. We therefore have two completely different save paths, depending on whether or not *series* is empty.

10.3.16 Tips

- Use the Template Tester to test templates. Add the tester to the context menu for books in the library and/or give it a keyboard shortcut.
- Templates can use other templates by referencing composite columns built with the desired template. Alternatively, you can use Stored Templates.
- In a plugboard, you can set a field to empty (or whatever is equivalent to empty) by using the special template `{}`. This template will always evaluate to an empty string.
- The technique described above to show numbers even if they have a zero value works with the standard field series_index.

10.3.17 Odkaz na funkci šablony

Reference for all built-in template language functions

Here, we document all the built-in functions available in the calibre template language. Every function is implemented as a class in python and you can click the source links to see the source code, in case the documentation is insufficient. The functions are arranged in logical groups by type.

- *Arithmetic* (stránka 184)
  - *add* (stránka 184)
  - *ceiling* (stránka 184)
  - *divide* (stránka 185)
  - *floor* (stránka 185)
  - *fractional_part* (stránka 185)
  - *mod* (stránka 185)
  - *multiply* (stránka 185)

  - *round* (stránka 185)
  - *subtract* (stránka 185)
- *Boolean* (stránka 185)
  - *and* (stránka 185)
  - *not* (stránka 186)
  - *or* (stránka 186)
- *Case changes* (stránka 186)
  - *capitalize* (stránka 186)
  - *lowercase* (stránka 186)
  - *titlecase* (stránka 186)
  - *uppercase* (stránka 186)
- *Database functions* (stránka 186)
  - *book_count* (stránka 186)
  - *book_values* (stránka 187)
  - *extra_file_modtime* (stránka 187)
  - *extra_file_names* (stránka 187)
  - *extra_file_size* (stránka 188)
  - *get_link* (stránka 188)
  - *get_note* (stránka 188)
  - *has_extra_files* (stránka 189)
  - *has_note* (stránka 189)
- *Date functions* (stránka 189)
  - *date_arithmetic* (stránka 189)
  - *days_between* (stránka 190)
  - *today* (stránka 190)
- *Formatting values* (stránka 190)
  - *finish_formatting* (stránka 190)
  - *format_date* (stránka 190)
  - *format_date_field* (stránka 191)
  - *format_duration* (stránka 192)
  - *format_number* (stránka 193)
  - *human_readable* (stránka 193)
  - *rating_to_stars* (stránka 193)
- *Get values from metadata* (stránka 193)
  - *annotation_count* (stránka 193)

  - *approximate_formats* (stránka 193)
  - *author_links* (stránka 193)
  - *author_sorts* (stránka 194)
  - *booksize* (stránka 194)
  - *connected_device_name* (stránka 194)
  - *connected_device_uuid* (stránka 194)
  - *current_library_name* (stránka 194)
  - *current_library_path* (stránka 194)
  - *current_virtual_library_name* (stránka 195)
  - *field* (stránka 195)
  - *formats_modtimes* (stránka 195)
  - *formats_path_segments* (stránka 195)
  - *formats_paths* (stránka 196)
  - *formats_sizes* (stránka 196)
  - *has_cover* (stránka 196)
  - *is_marked* (stránka 196)
  - *language_codes* (stránka 196)
  - *language_strings* (stránka 196)
  - *ondevice* (stránka 197)
  - *raw_field* (stránka 197)
  - *raw_list* (stránka 197)
  - *series_sort* (stránka 197)
  - *user_categories* (stránka 197)
  - *virtual_libraries* (stránka 197)
- *Iterate over values* (stránka 197)
  - *first_non_empty* (stránka 197)
  - *lookup* (stránka 198)
  - *switch* (stránka 198)
  - *switch_if* (stránka 198)
- *List lookup* (stránka 198)
  - *identifier_in_list* (stránka 198)
  - *list_contains* (stránka 198)
  - *list_item* (stránka 198)
  - *select* (stránka 199)
  - *str_in_list* (stránka 199)

- *List manipulation* (stránka 199)
  - *list_count* (stránka 199)
  - *list_count_field* (stránka 199)
  - *list_count_matching* (stránka 199)
  - *list_difference* (stránka 200)
  - *list_equals* (stránka 200)
  - *list_intersection* (stránka 200)
  - *list_join* (stránka 200)
  - *list_re* (stránka 201)
  - *list_re_group* (stránka 201)
  - *list_remove_duplicates* (stránka 201)
  - *list_sort* (stránka 201)
  - *list_split* (stránka 201)
  - *list_union* (stránka 201)
  - *range* (stránka 202)
  - *subitems* (stránka 202)
  - *sublist* (stránka 203)
- *Ostatní* (stránka 203)
  - *arguments* (stránka 203)
  - *assign* (stránka 203)
  - *globals* (stránka 203)
  - *is_dark_mode* (stránka 203)
  - *print* (stránka 204)
  - *set_globals* (stránka 204)
- *Recursion* (stránka 204)
  - *eval* (stránka 204)
  - *template* (stránka 204)
- *Relational* (stránka 204)
  - *cmp* (stránka 204)
  - *first_matching_cmp* (stránka 204)
  - *strcmp* (stránka 205)
  - *strcmpcase* (stránka 205)
- *String manipulation* (stránka 205)
  - *character* (stránka 205)
  - *check_yes_no* (stránka 205)

    - *contains* (stránka 205)
    - *field_exists* (stránka 206)
    - *ifempty* (stránka 206)
    - *re* (stránka 206)
    - *re_group* (stránka 206)
    - *shorten* (stránka 206)
    - *strcat* (stránka 206)
    - *strcat_max* (stránka 207)
    - *strlen* (stránka 207)
    - *substr* (stránka 207)
    - *swap_around_articles* (stránka 207)
    - *swap_around_comma* (stránka 207)
    - *test* (stránka 207)
    - *transliterate* (stránka 207)
- *URL functions* (stránka 208)
    - *encode_for_url* (stránka 208)
    - *make_url* (stránka 208)
    - *make_url_extended* (stránka 208)
    - *query_string* (stránka 209)
    - *to_hex* (stránka 210)
    - *urls_from_identifiers* (stránka 210)
- *API of the Metadata objects* (stránka 210)

Arithmetic

add

class calibre.utils.formatter_functions.**BuiltinAdd**

`add(x [, y]*)` – returns the sum of its arguments. Throws an exception if an argument is not a number. In most cases you can use the + operator instead of this function.

ceiling

class calibre.utils.formatter_functions.**BuiltinCeiling**

`ceiling(value)` – returns the smallest integer greater than or equal to `value`. Throws an exception if `value` is not a number.

divide

class calibre.utils.formatter_functions.**BuiltinDivide**

`divide(x, y)` – returns `x / y`. Throws an exception if either `x` or `y` are not numbers. This function can usually be replaced by the `/` operator.

floor

class calibre.utils.formatter_functions.**BuiltinFloor**

`floor(value)` – returns the largest integer less than or equal to `value`. Throws an exception if `value` is not a number.

fractional_part

class calibre.utils.formatter_functions.**BuiltinFractionalPart**

`fractional_part(value)` – returns the part of the value after the decimal point. For example, `fractional_part(3.14)` returns `0.14`. Throws an exception if `value` is not a number.

mod

class calibre.utils.formatter_functions.**BuiltinMod**

`mod(value, y)` – returns the `floor` of the remainder of `value / y`. Throws an exception if either `value` or `y` is not a number.

multiply

class calibre.utils.formatter_functions.**BuiltinMultiply**

`multiply(x [, y]*)` – returns the product of its arguments. Throws an exception if any argument is not a number. This function can usually be replaced by the `*` operator.

round

class calibre.utils.formatter_functions.**BuiltinRound**

`round(value)` – returns the nearest integer to `value`. Throws an exception if `value` is not a number.

subtract

class calibre.utils.formatter_functions.**BuiltinSubtract**

`subtract(x, y)` – returns `x - y`. Throws an exception if either `x` or `y` are not numbers. This function can usually be replaced by the `-` operator.

Boolean

and

class calibre.utils.formatter_functions.**BuiltinAnd**

`and(value [, value]*)` – returns the string `'1'` if all values are not empty, otherwise returns the empty string. You can have as many values as you want. In most cases you can use the `&&` operator instead of this function. One reason not to replace `and()` with `&&` is when short-circuiting can change the results because of side effects. For example, `and(a='',b=5)` will always do both assignments, where the `&&` operator won't do the second.

not

class calibre.utils.formatter_functions.**BuiltinNot**

`not(value)` – returns the string `'1'` if the value is empty, otherwise returns the empty string. This function can usually be replaced with the unary not (`!`) operator.

or

class calibre.utils.formatter_functions.**BuiltinOr**

`or(value [, value]*)` – returns the string `'1'` if any value is not empty, otherwise returns the empty string. You can have as many values as you want. This function can usually be replaced by the `||` operator. A reason it cannot be replaced is if short-circuiting will change the results because of side effects.

Case changes

capitalize

class calibre.utils.formatter_functions.**BuiltinCapitalize**

`capitalize(value)` – returns the `value` with the first letter in upper case and the rest lower case.

lowercase

class calibre.utils.formatter_functions.**BuiltinLowercase**

`lowercase(value)` – returns the `value` in lower case.

titlecase

class calibre.utils.formatter_functions.**BuiltinTitlecase**

`titlecase(value)` – returns the `value` in title case.

uppercase

class calibre.utils.formatter_functions.**BuiltinUppercase**

`uppercase(value)` – returns the `value` in upper case.

Database functions

book_count

class calibre.utils.formatter_functions.**BuiltinBookCount**

`book_count(query, use_vl)` – returns the count of books found by searching for `query`. If `use_vl` is `0` (zero) then virtual libraries are ignored. This function and its companion `book_values()` are particularly useful in template searches, supporting searches that combine information from many books such as looking for series with only one book. It cannot be used in composite columns unless the tweak `allow_template_database_functions_in_composites` is set to True. It can be used only in the GUI.

For example this template search uses this function and its companion to find all series with only one book:

- Define a stored template (using *Preferences → Advanced → Template functions*) named `series_only_one_book` (the name is arbitrary). The template is:

```
program:
    vals = globals(vals='');
    if !vals then
        all_series = book_values('series', 'series:true', ',', 0);
        for series in all_series:
            if book_count('series:="' & series & '"', 0) == 1 then
                vals = list_join(',', vals, ',', series, ',')
            fi
        rof;
        set_globals(vals)
    fi;
    str_in_list(vals, ',', $series, 1, '')
```

The first time the template runs (the first book checked) it stores the results of the database lookups in a `global` template variable named `vals`. These results are used to check subsequent books without redoing the lookups.

- Use the stored template in a template search:

```
template:"program: series_only_one_book()#@#:n:1"
```

Using a stored template instead of putting the template into the search eliminates problems caused by the requirement to escape quotes in search expressions.

This function can be used only in the GUI and the content server.

book_values

class calibre.utils.formatter_functions.**BuiltinBookValues**

`book_values(column, query, sep, use_vl)` – returns a list of the unique values contained in the column `column` (a lookup name), separated by `sep`, in the books found by searching for `query`. If `use_vl` is `0` (zero) then virtual libraries are ignored. This function and its companion `book_count()` are particularly useful in template searches, supporting searches that combine information from many books such as looking for series with only one book. It cannot be used in composite columns unless the tweak `allow_template_database_functions_in_composites` is set to True. This function can be used only in the GUI and the content server.

extra_file_modtime

class calibre.utils.formatter_functions.**BuiltinExtraFileModtime**

`extra_file_modtime(file_name, format_string)` – returns the modification time of the extra file `file_name` in the book's `data/` folder if it exists, otherwise `-1`. The modtime is formatted according to `format_string` (see *format_date()* (stránka 190) for details). If `format_string` is the empty string, returns the modtime as the floating point number of seconds since the epoch. See also the functions *has_extra_files()* (stránka 189), *extra_file_names()* (stránka 187) and *extra_file_size()* (stránka 188). The epoch is OS dependent. This function can be used only in the GUI and the content server.

extra_file_names

class calibre.utils.formatter_functions.**BuiltinExtraFileNames**

`extra_file_names(sep [, pattern])` – returns a `sep`-separated list of extra files in the book's `data/` folder. If the optional parameter `pattern`, a regular expression, is supplied then the list is filtered to files that match `pattern`. The pattern match is case insensitive. See also the functions *has_extra_files()* (stránka 189), *extra_file_modtime()* (stránka 187) and *extra_file_size()* (stránka 188). This function can be used only in the GUI and the content server.

extra_file_size

class calibre.utils.formatter_functions.**BuiltinExtraFileSize**

`extra_file_size(file_name)` – returns the size in bytes of the extra file `file_name` in the book's `data/` folder if it exists, otherwise `-1`. See also the functions *has_extra_files()* (stránka 189), *extra_file_names()* (stránka 187) and *extra_file_modtime()* (stránka 187). This function can be used only in the GUI and the content server.

get_link

class calibre.utils.formatter_functions.**BuiltinGetLink**

`get_link(field_name, field_value)` – fetch the link for field `field_name` with value `field_value`. If there is no attached link, return the empty string. Examples:

- The following returns the link attached to the tag `Fiction`:

```
get_link('tags', 'Fiction')
```

- This template makes a list of the links for all the tags associated with a book in the form `value:link, ...`:

```
program:
    ans = '';
    for t in $tags:
        l = get_link('tags', t);
        if l then
            ans = list_join(', ', ans, ',', t & ':' & get_link('tags', t), ',')
        fi
    rof;
ans
```

This function works only in the GUI and the content server.

get_note

class calibre.utils.formatter_functions.**BuiltinGetNote**

`get_note(field_name, field_value, plain_text)` – fetch the note for field `field_name` with value `field_value`. If `plain_text` is empty, return the note's HTML including images. If `plain_text` is `1` (or `'1'`), return the note's plain text. If the note doesn't exist, return the empty string in both cases. Example:

- Return the HTML of the note attached to the tag *Fiction*:

```
program:
    get_note('tags', 'Fiction', '')
```

- Return the plain text of the note attached to the author *Isaac Asimov*:

```
program:
    get_note('authors', 'Isaac Asimov', 1)
```

This function works only in the GUI and the content server.

has_extra_files

class calibre.utils.formatter_functions.**BuiltinHasExtraFiles**

`has_extra_files([pattern])` – returns the count of extra files, otherwise ‚' (the empty string). If the optional parameter `pattern` (a regular expression) is supplied then the list is filtered to files that match `pattern` before the files are counted. The pattern match is case insensitive. See also the functions *extra_file_names()* (stránka 187), *extra_file_size()* (stránka 188) and *extra_file_modtime()* (stránka 187). This function can be used only in the GUI and the content server.

has_note

class calibre.utils.formatter_functions.**BuiltinHasNote**

`has_note(field_name, field_value)`. Check if a field has a note. This function has two variants:

- if `field_value` is not `''` (the empty string) return `'1'` if the value `field_value` in the field `field_name` has a note, otherwise `''`.

  Example: `has_note('tags', 'Fiction')` returns `'1'` if the tag `fiction` has an attached note, otherwise `''`.

- If `field_value` is `''` then return a list of values in `field_name` that have a note. If no item in the field has a note, return `''`. This variant is useful for showing column icons if any value in the field has a note, rather than a specific value.

  Example: `has_note('authors', '')` returns a list of authors that have notes, or `''` if no author has a note.

You can test if all the values in `field_name` have a note by comparing the list length of this function's return value against the list length of the values in `field_name`. Example:

```
list_count(has_note('authors', ''), '&') ==# list_count_field('authors')
```

This function works only in the GUI and the content server.

Date functions

date_arithmetic

class calibre.utils.formatter_functions.**BuiltinDateArithmetic**

`date_arithmetic(value, calc_spec, fmt)` – Calculate a new date from `value` using `calc_spec`. Return the new date formatted according to optional `fmt`: if not supplied then the result will be in ISO format. The `calc_spec` is a string formed by concatenating pairs of `vW` (`valueWhat`) where `v` is a possibly-negative number and W is one of the following letters:

- `s`: add `v` seconds to `date`
- `m`: add `v` minutes to `date`
- `h`: add `v` hours to `date`
- `d`: add `v` days to `date`
- `w`: add `v` weeks to `date`
- `y`: add `v` years to `date`, where a year is 365 days.

Example: `'1s3d-1m'` will add 1 second, add 3 days, and subtract 1 minute from `date`.

days_between

class calibre.utils.formatter_functions.**BuiltinDaysBetween**

`days_between(date1, date2)` – return the number of days between `date1` and `date2`. The number is positive if `date1` is greater than `date2`, otherwise negative. If either `date1` or `date2` are not dates, the function returns the empty string.

today

class calibre.utils.formatter_functions.**BuiltinToday**

`today()` – return a date+time string for today (now). This value is designed for use in `format_date` or `days_between`, but can be manipulated like any other string. The date is in ISO[90] date/time format.

Formatting values

finish_formatting

class calibre.utils.formatter_functions.**BuiltinFinishFormatting**

`finish_formatting(value, format, prefix, suffix)` – apply the `format`, `prefix`, and `suffix` to the `value` in the same way as done in a template like `{series_index:05.2f| - |- }`. This function is provided to ease conversion of complex single-function- or template-program-mode templates to *GPM* Templates. For example, the following program produces the same output as the above template:

```
program: finish_formatting(field("series_index"), "05.2f", " - ", " - ")
```

Another example: for the template:

```
{series:re(([^\s])[^\s]+(\s|$),\1)}{series_index:0>2s| - | - }{title}
```

use:

```
program:
    strcat(
        re(field('series'), '([^\s])[^\s]+(\s|$)', '\1'),
        finish_formatting(field('series_index'), '0>2s', ' - ', ' - '),
        field('title')
    )
```

format_date

class calibre.utils.formatter_functions.**BuiltinFormatDate**

`format_date(value, format_string)` – format the `value`, which must be a date string, using the `format_string`, returning a string. It is best if the date is in ISO format as using other date formats often causes errors because the actual date value cannot be unambiguously determined. Note that the `format_date_field()` function is both faster and more reliable.

The formatting codes are:

- `d` : the day as number without a leading zero (1 to 31)
- `dd` : the day as number with a leading zero (01 to 31)

[90] https://en.wikipedia.org/wiki/ISO_8601

- `ddd` : the abbreviated localized day name (e.g. „Mon“ to „Sun“)
- `dddd` : the long localized day name (e.g. „Monday“ to „Sunday“)
- `M` : the month as number without a leading zero (1 to 12)
- `MM` : the month as number with a leading zero (01 to 12)
- `MMM` : the abbreviated localized month name (e.g. „Jan“ to „Dec“)
- `MMMM` : the long localized month name (e.g. „January“ to „December“)
- `yy` : the year as two digit number (00 to 99)
- `yyyy` : the year as four digit number.
- `h` : the hours without a leading 0 (0 to 11 or 0 to 23, depending on am/pm)
- `hh` : the hours with a leading 0 (00 to 11 or 00 to 23, depending on am/pm)
- `m` : the minutes without a leading 0 (0 to 59)
- `mm` : the minutes with a leading 0 (00 to 59)
- `s` : the seconds without a leading 0 (0 to 59)
- `ss` : the seconds with a leading 0 (00 to 59)
- `ap` : use a 12-hour clock instead of a 24-hour clock, with ‚ap‘ replaced by the lowercase localized string for am or pm
- `AP` : use a 12-hour clock instead of a 24-hour clock, with ‚AP‘ replaced by the uppercase localized string for AM or PM
- `aP` : use a 12-hour clock instead of a 24-hour clock, with ‚aP‘ replaced by the localized string for AM or PM
- `Ap` : use a 12-hour clock instead of a 24-hour clock, with ‚Ap‘ replaced by the localized string for AM or PM
- `iso` : the date with time and timezone. Must be the only format present
- `to_number` : convert the date & time into a floating point number (a *timestamp*)
- `from_number` : convert a floating point number (a *timestamp*) into an ISO-formatted date. If you want a different date format then add the desired formatting string after `from_number` and a colon (`:`). Example:

```
format_date(val, 'from_number:MMM dd yyyy')
```

You might get unexpected results if the date you are formatting contains localized month names, which can happen if you changed the date format to contain `MMMM`. Using `format_date_field()` avoids this problem.

format_date_field

class calibre.utils.formatter_functions.**BuiltinFormatDateField**

`format_date_field(field_name, format_string)` – format the value in the field `field_name`, which must be the lookup name of a date field, either standard or custom. See *format_date()* (stránka 190) for the formatting codes. This function is much faster than format_date() and should be used when you are formatting the value in a field (column). It is also more reliable because it works directly on the underlying date. It can’t be used for computed dates or dates in string variables. Examples:

```
format_date_field('pubdate', 'yyyy.MM.dd')
format_date_field('#date_read', 'MMM dd, yyyy')
```

format_duration

class calibre.utils.formatter_functions.**BuiltinFormatDuration**

`format_duration(value, template, [largest_unit])` – format the value, a number of seconds, into a string showing weeks, days, hours, minutes, and seconds. If the value is a float then it is rounded to the nearest integer. You choose how to format the value using a template consisting of value selectors surrounded by `[` and `]` characters. The selectors are:

- `[w]`: weeks
- `[d]`: days
- `[h]`: hours
- `[m]`: minutes
- `[s]`: seconds

You can put arbitrary text between selectors.

The following examples use a duration of 2 days (172,800 seconds) 1 hour (3,600 seconds) and 20 seconds, which totals to 176,420 seconds.

- `format_duration(176420, '[d][h][m][s]')` will return the value `2d 1h 0m 20s`.
- `format_duration(176420, '[h][m][s]')` will return the value `49h 0m 20s`.
- `format_duration(176420, 'Your reading time is [d][h][m][s]')` returns the value `Your reading time is 49h 0m 20s`.
- `format_duration(176420, '[w][d][h][m][s]')` will return the value `2d 1h 0m 20s`. Note that the zero weeks value is not returned.

If you want to see zero values for items such as weeks in the above example, use an uppercase selector. For example, the following uses `'W'` to show zero weeks:

`format_duration(176420, '[W][d][h][m][s]')` returns `0w 2d 1h 0m 20s`.

By default the text following a value is the selector followed by a space. You can change that to whatever text you want. The format for a selector with your text is the selector followed by a colon followed by text segments separated by `'|'` characters. You must include any space characters you want in the output.

You can provide from one to three text segments.

- If you provide one segment, as in `[w: weeks ]` then that segment is used for all values.
- If you provide two segments, as in `[w: weeks | week ]` then the first segment is used for 0 and more than 1. The second segment is used for 1.
- If you provide three segments, as in `[w: weeks | week | weeks ]` then the first segment is used for 0, the second segment is used for 1, and the third segment is used for more than 1.

The second form is equivalent to the third form in many languages.

For example, the selector:

- `[w: weeks | week | weeks ]` produces `'0 weeks '`, `'1 week '`, or `'2 weeks '`.
- `[w: weeks | week ]` produces `'0 weeks '`, `'1 week '`, or `'2 weeks '`.
- `[w: weeks ]` produces `0 weeks '`, `1 weeks '`, or `2 weeks '`.

The optional `largest_unit` parameter specifies the largest of weeks, days, hours, minutes, and seconds that will be produced by the template. It must be one of the value selectors. This can be useful to truncate a value.

`format_duration(176420, '[h][m][s]', 'd')` will return the value `1h 0m 20s` instead of `49h 0m 20s`.

format_number

class calibre.utils.formatter_functions.**BuiltinFormatNumber**

`format_number(value, template)` – interprets the `value` as a number and formats that number using a Python formatting template such as `{0:5.2f}` or `{0:,d}` or `${0:5,.2f}`. The formatting template must begin with `{0:` and end with `}` as in the above examples. Exception: you can leave off the leading „{0:“ and trailing „}“ if the format template contains only a format. See the Template Language[91] and the Python[92] documentation for more examples. Returns the empty string if formatting fails.

human_readable

class calibre.utils.formatter_functions.**BuiltinHumanReadable**

`human_readable(value)` – expects the `value` to be a number and returns a string representing that number in KB, MB, GB, etc.

rating_to_stars

class calibre.utils.formatter_functions.**BuiltinRatingToStars**

`rating_to_stars(value, use_half_stars)` – Returns the `value` as string of star (★) characters. The value must be a number between `0` and `5`. Set `use_half_stars` to `1` if you want half star characters for fractional numbers available with custom ratings columns.

Get values from metadata

annotation_count

class calibre.utils.formatter_functions.**BuiltinAnnotationCount**

`annotation_count()` – return the total number of annotations of all types attached to the current book. This function works only in the GUI and the content server.

approximate_formats

class calibre.utils.formatter_functions.**BuiltinApproximateFormats**

`approximate_formats()` – return a comma-separated list of formats associated with the book. Because the list comes from calibre's database instead of the file system, there is no guarantee that the list is correct, although it probably is. Note that resulting format names are always uppercase, as in EPUB. The `approximate_formats()` function is much faster than the `formats_...` functions.

This function works only in the GUI. If you want to use these values in save-to-disk or send-to-device templates then you must make a custom „Column built from other columns“, use the function in that column's template, and use that column's value in your save/send templates.

author_links

class calibre.utils.formatter_functions.**BuiltinAuthorLinks**

`author_links(val_separator, pair_separator)` – returns a string containing a list of authors and those authors' link values in the form: `author1 val_separator author1_link pair_separator author2 val_separator author2_link` etc.

[91] https://manual.calibre-ebook.com/template_lang.html
[92] https://docs.python.org/3/library/string.html#formatstrings

An author is separated from its link value by the `val_separator` string with no added spaces. Assuming the `val_separator` is a colon, `author:link value` pairs are separated by the `pair_separator` string argument with no added spaces. It is up to you to choose separators that do not occur in author names or links. An author is included even if the author link is empty.

author_sorts

class calibre.utils.formatter_functions.**BuiltinAuthorSorts**

`author_sorts(val_separator)` – returns a string containing a list of author's sort values for the authors of the book. The sort is the one in the author metadata information, which can be different from the author_sort in books. The returned list has the form `author sort 1 val_separator author sort 2` etc. with no added spaces. The author sort values in this list are in the same order as the authors of the book. If you want spaces around `val_separator` then include them in the `val_separator` string.

booksize

class calibre.utils.formatter_functions.**BuiltinBooksize**

`booksize()` – returns the value of the calibre `size` field. Returns ‚' if the book has no formats.

This function works only in the GUI. If you want to use this value in save-to-disk or send-to-device templates then you must make a custom „Column built from other columns", use the function in that column's template, and use that column's value in your save/send templates

connected_device_name

class calibre.utils.formatter_functions.**BuiltinConnectedDeviceName**

`connected_device_name(storage_location_key)` – if a device is connected then return the device name, otherwise return the empty string. Each storage location on a device has its own device name. The `storage_location_key` names are `'main'`, `'carda'` and `'cardb'`. This function works only in the GUI.

connected_device_uuid

class calibre.utils.formatter_functions.**BuiltinConnectedDeviceUUID**

`connected_device_uuid(storage_location_key)` – if a device is connected then return the device uuid (unique id), otherwise return the empty string. Each storage location on a device has a different uuid. The `storage_location_key` location names are `'main'`, `'carda'` and `'cardb'`. This function works only in the GUI.

current_library_name

class calibre.utils.formatter_functions.**BuiltinCurrentLibraryName**

`current_library_name()` – return the last name on the path to the current calibre library.

current_library_path

class calibre.utils.formatter_functions.**BuiltinCurrentLibraryPath**

`current_library_path()` – return the full path to the current calibre library.

current_virtual_library_name

class calibre.utils.formatter_functions.**BuiltinCurrentVirtualLibraryName**

`current_virtual_library_name()` – return the name of the current virtual library if there is one, otherwise the empty string. Library name case is preserved. Example:

```
program: current_virtual_library_name()
```

This function works only in the GUI.

field

class calibre.utils.formatter_functions.**BuiltinField**

`field(lookup_name)` – returns the value of the metadata field with lookup name `lookup_name`. The `$` prefix can be used instead of the function, as in `$tags`.

formats_modtimes

class calibre.utils.formatter_functions.**BuiltinFormatsModtimes**

`formats_modtimes(date_format_string)` – return a comma-separated list of colon-separated items `FMT:DATE` representing modification times for the formats of a book. The `date_format_string` parameter specifies how the date is to be formatted. See the *format_date()* (stránka 190) function for details. You can use the *select()* (stránka 199) function to get the modification time for a specific format. Note that format names are always uppercase, as in EPUB.

formats_path_segments

class calibre.utils.formatter_functions.**BuiltinFormatsPathSegments**

`formats_path_segments(with_author, with_title, with_format, with_ext, sep)` – return parts of the path to a book format in the calibre library separated by `sep`. The parameter `sep` should usually be a slash (`'/'`). One use is to be sure that paths generated in Save to disk and Send to device templates are shortened consistently. Another is to be sure the paths on the device match the paths in the calibre library.

A book path consists of 3 segments: the author, the title including the calibre database id in parentheses, and the format (author - title). Calibre can shorten any of the three because of file name length limitations. You choose which segments to include by passing `1` for that segment. If you don't want a segment then pass `0` or the empty string for that segment. For example, the following returns just the format name without the extension:

```
formats_path_segments(0, 0, 1, 0, '/')
```

Because there is only one segment the separator is ignored.

If there are multiple formats (multiple extensions) then one of the extensions will be picked at random. If you care about which extension is used then get the path without the extension then add the desired extension to it.

Examples: Assume there is a book in the calibre library with an epub format by Joe Blogs with title ‚Help'. It would have the path

```
Joe Blogs/Help - (calibre_id)/Help - Joe Blogs.epub
```

The following shows what is returned for various parameters:

- `formats_path_segments(0, 0, 1, 0, '/')` returns *Help - Joe Blogs*
- `formats_path_segments(0, 0, 1, 1, '/')` returns *Help - Joe Blogs.epub*

- `formats_path_segments(1, 0, 1, 1, '/')` returns *Joe Blogs/Help - Joe Blogs.epub*
- `formats_path_segments(1, 0, 1, 0, '/')` returns *Joe Blogs/Help - Joe Blogs*
- `formats_path_segments(0, 1, 0, 0, '/')` returns *Help - (calibre_id)*

formats_paths

class calibre.utils.formatter_functions.**BuiltinFormatsPaths**

`formats_paths([separator])` – return a `separator`-separated list of colon-separated items `FMT:PATH` giving the full path to the formats of a book. The `separator` argument is optional. If not supplied then the separator is `', '` (comma space). If the separator is a comma then you can use the `select()` function to get the path for a specific format. Note that format names are always uppercase, as in EPUB.

formats_sizes

class calibre.utils.formatter_functions.**BuiltinFormatsSizes**

`formats_sizes()` – return a comma-separated list of colon-separated `FMT:SIZE` items giving the sizes of the formats of a book in bytes. You can use the `select()` function to get the size for a specific format. Note that format names are always uppercase, as in EPUB.

has_cover

class calibre.utils.formatter_functions.**BuiltinHasCover**

`has_cover()` – return `'Yes'` if the book has a cover, otherwise the empty string.

is_marked

class calibre.utils.formatter_functions.**BuiltinIsMarked**

`is_marked()` – check whether the book is *marked* in calibre. If it is then return the value of the mark, either `'true'` (lower case) or a comma-separated list of named marks. Returns `''` (the empty string) if the book is not marked. This function works only in the GUI.

language_codes

class calibre.utils.formatter_functions.**BuiltinLanguageCodes**

`language_codes(lang_strings)` – return the language codes[93] for the language names passed in `lang_strings`. The strings must be in the language of the current locale. `lang_strings` is a comma-separated list.

language_strings

class calibre.utils.formatter_functions.**BuiltinLanguageStrings**

`language_strings(value, localize)` – return the language names for the language codes (see here for names and codes[94]) passed in `value`. Example: `{languages:language_strings()}`. If `localize` is zero, return the strings in English. If `localize` is not zero, return the strings in the language of the current locale. `lang_codes` is a comma-separated list.

[93] https://www.loc.gov/standards/iso639-2/php/code_list.php
[94] https://www.loc.gov/standards/iso639-2/php/code_list.php

ondevice

class calibre.utils.formatter_functions.**BuiltinOndevice**

`ondevice()` – return the string `'Yes'` if `ondevice` is set, otherwise return the empty string. This function works only in the GUI. If you want to use this value in save-to-disk or send-to-device templates then you must make a custom „Column built from other columns“, use the function in that column's template, and use that column's value in your save/send templates.

raw_field

class calibre.utils.formatter_functions.**BuiltinRawField**

`raw_field(lookup_name [, optional_default])` – returns the metadata field named by `lookup_name` without applying any formatting. It evaluates and returns the optional second argument `optional_default` if the field's value is undefined (`None`). The `$$` prefix can be used instead of the function, as in `$$pubdate`.

raw_list

class calibre.utils.formatter_functions.**BuiltinRawList**

`raw_list(lookup_name, separator)` – returns the metadata list named by `lookup_name` without applying any formatting or sorting, with the items separated by `separator`.

series_sort

class calibre.utils.formatter_functions.**BuiltinSeriesSort**

`series_sort()` – returns the series sort value.

user_categories

class calibre.utils.formatter_functions.**BuiltinUserCategories**

`user_categories()` – return a comma-separated list of the user categories that contain this book. This function works only in the GUI. If you want to use these values in save-to-disk or send-to-device templates then you must make a custom *Column built from other columns*, use the function in that column's template, and use that column's value in your save/send templates

virtual_libraries

class calibre.utils.formatter_functions.**BuiltinVirtualLibraries**

`virtual_libraries()` – return a comma-separated list of Virtual libraries that contain this book. This function works only in the GUI. If you want to use these values in save-to-disk or send-to-device templates then you must make a custom „Column built from other columns“, use the function in that column's template, and use that column's value in your save/send templates.

Iterate over values

first_non_empty

class calibre.utils.formatter_functions.**BuiltinFirstNonEmpty**

`first_non_empty(value [, value]*)` – returns the first `value` that is not empty. If all values are empty, then the empty string is returned. You can have as many values as you want.

lookup

class calibre.utils.formatter_functions.**BuiltinLookup**

`lookup(value, [ pattern, key, ]* else_key)` – The patterns will be checked against the `value` in order. If a `pattern` matches then the value of the field named by `key` is returned. If no pattern matches then the value of the field named by `else_key` is returned. See also the *switch()* (stránka 198) function.

switch

class calibre.utils.formatter_functions.**BuiltinSwitch**

`switch(value, [patternN, valueN,]+ else_value)` – for each `patternN, valueN` pair, checks if the `value` matches the regular expression `patternN` and if so returns the associated `valueN`. If no `patternN` matches, then `else_value` is returned. You can have as many `patternN, valueN` pairs as you wish. The first match is returned.

switch_if

class calibre.utils.formatter_functions.**BuiltinSwitchIf**

`switch_if([test_expression, value_expression,]+ else_expression)` – for each `test_expression, value_expression` pair, checks if `test_expression` is True (non-empty) and if so returns the result of `value_expression`. If no `test_expression` is True then the result of `else_expression` is returned. You can have as many `test_expression, value_expression` pairs as you want.

List lookup

identifier_in_list

class calibre.utils.formatter_functions.**BuiltinIdentifierInList**

`identifier_in_list(val, id_name [, found_val, not_found_val])` – treat `val` as a list of identifiers separated by commas. An identifier has the format `id_name:value`. The `id_name` parameter is the id_name text to search for, either `id_name` or `id_name:regexp`. The first case matches if there is any identifier matching that id_name. The second case matches if id_name matches an identifier and the regexp matches the identifier's value. If `found_val` and `not_found_val` are provided then if there is a match then return `found_val`, otherwise return `not_found_val`. If `found_val` and `not_found_val` are not provided then if there is a match then return the `identifier:value` pair, otherwise the empty string (`''`).

list_contains

class calibre.utils.formatter_functions.**BuiltinInList**

`list_contains(value, separator, [ pattern, found_val, ]* not_found_val)` – interpret the `value` as a list of items separated by `separator`, checking the `pattern` against each item in the list. If the `pattern` matches an item then return `found_val`, otherwise return `not_found_val`. The pair `pattern` and `found_value` can be repeated as many times as desired, permitting returning different values depending on the item's value. The patterns are checked in order, and the first match is returned.

Aliases: in_list(), list_contains()

list_item

class calibre.utils.formatter_functions.**BuiltinListitem**

`list_item(value, index, separator)` – interpret the `value` as a list of items separated by `separator`, returning the ‚index'th item. The first item is number zero. The last item has the index `-1` as in `list_item(-1,separator)`. If the item is not in the list, then the empty string is returned. The separator has the same meaning as in the count function, usually comma but is ampersand for author-like lists.

select

class calibre.utils.formatter_functions.**BuiltinSelect**

`select(value, key)` – interpret the `value` as a comma-separated list of items with each item having the form `id:id_value` (the calibre `identifier` format). The function finds the first pair with the id equal to `key` and returns the corresponding `id_value`. If no id matches then the function returns the empty string.

str_in_list

class calibre.utils.formatter_functions.**BuiltinStrInList**

`str_in_list(value, separator, [ string, found_val, ]+ not_found_val)` – interpret the `value` as a list of items separated by `separator` then compare `string` against each value in the list. The `string` is not a regular expression. If `string` is equal to any item (ignoring case) then return the corresponding `found_val`. If `string` contains `separators` then it is also treated as a list and each subvalue is checked. The `string` and `found_value` pairs can be repeated as many times as desired, permitting returning different values depending on string's value. If none of the strings match then `not_found_value` is returned. The strings are checked in order. The first match is returned.

List manipulation

list_count

class calibre.utils.formatter_functions.**BuiltinCount**

`list_count(value, separator)` – interprets the value as a list of items separated by `separator` and returns the number of items in the list. Most lists use a comma as the separator, but `authors` uses an ampersand (&).

Examples: `{tags:list_count(,)}`, `{authors:list_count(&)}`.

Aliases: `count()`, `list_count()`

list_count_field

class calibre.utils.formatter_functions.**BuiltinFieldListCount**

`list_count_field(lookup_name)` – returns the count of items in the field with the lookup name `lookup_name`. The field must be multi-valued such as `authors` or `tags`, otherwise the function raises an error. This function is much faster than `list_count()` because it operates directly on calibre data without converting it to a string first. Example: `list_count_field('tags')`.

list_count_matching

class calibre.utils.formatter_functions.**BuiltinListCountMatching**

`list_count_matching(value, pattern, separator)` – interprets `value` as a list of items separated by `separator`, returning the number of items in the list that match the regular expression `pattern`.

Aliases: `list_count_matching()`, `count_matching()`

list_difference

class calibre.utils.formatter_functions.**BuiltinListDifference**

`list_difference(list1, list2, separator)` – return a list made by removing from `list1` any item found in `list2` using a case-insensitive comparison. The items in `list1` and `list2` are separated by `separator`, as are the items in the returned list.

list_equals

class calibre.utils.formatter_functions.**BuiltinListEquals**

`list_equals(list1, sep1, list2, sep2, yes_val, no_val)` – return `yes_val` if `list1` and `list2` contain the same items, otherwise return `no_val`. The items are determined by splitting each list using the appropriate separator character (`sep1` or `sep2`). The order of items in the lists is not relevant. The comparison is case-insensitive.

list_intersection

class calibre.utils.formatter_functions.**BuiltinListIntersection**

`list_intersection(list1, list2, separator)` – return a list made by removing from `list1` any item not found in `list2`, using a case-insensitive comparison. The items in `list1` and `list2` are separated by `separator`, as are the items in the returned list.

list_join

class calibre.utils.formatter_functions.**BuiltinListJoin**

`list_join(with_separator, list1, separator1 [, list2, separator2]*)` – return a list made by joining the items in the source lists (`list1` etc) using `with_separator` between the items in the result list. Items in each source `list[123...]` are separated by the associated `separator[123...]`. A list can contain zero values. It can be a field like `publisher` that is single-valued, effectively a one-item list. Duplicates are removed using a case-insensitive comparison. Items are returned in the order they appear in the source lists. If items on lists differ only in letter case then the last is used. All separators can be more than one character.

Example:

```
program:
    list_join('#@#', $authors, '&', $tags, ',')
```

You can use `list_join` on the results of previous calls to `list_join` as follows:

```
program:
    a = list_join('#@#', $authors, '&', $tags, ',');
    b = list_join('#@#', a, '#@#', $#genre, ',', $#people, '&', 'some value', ',')
```

You can use expressions to generate a list. For example, assume you want items for `authors` and `#genre`, but with the genre changed to the word „Genre: „ followed by the first letter of the genre, i.e. the genre „Fiction“ becomes „Genre: F“. The following will do that:

```
program:
    list_join('#@#', $authors, '&', list_re($#genre, ',', '^(.).*$', 'Genre: \1'),  ',
↪')
```

list_re

class calibre.utils.formatter_functions.**BuiltinListRe**

`list_re(src_list, separator, include_re, opt_replace)` – Construct a list by first separating `src_list` into items using the `separator` character. For each item in the list, check if it matches `include_re`. If it does then add it to the list to be returned. If `opt_replace` is not the empty string then apply the replacement before adding the item to the returned list.

list_re_group

class calibre.utils.formatter_functions.**BuiltinListReGroup**

`list_re_group(src_list, separator, include_re, search_re [,template_for_group]*)` – Like list_re except replacements are not optional. It uses `re_group(item, search_re, template ...)` when doing the replacements.

list_remove_duplicates

class calibre.utils.formatter_functions.**BuiltinListRemoveDuplicates**

`list_remove_duplicates(list, separator)` – return a list made by removing duplicate items in `list`. If items differ only in case then the last is returned. The items in `list` are separated by `separator`, as are the items in the returned list.

list_sort

class calibre.utils.formatter_functions.**BuiltinListSort**

`list_sort(value, direction, separator)` – return `value` sorted using a case-insensitive lexical sort. If `direction` is zero (number or character), `value` is sorted ascending, otherwise descending. The list items are separated by `separator`, as are the items in the returned list.

list_split

class calibre.utils.formatter_functions.**BuiltinListSplit**

`list_split(list_val, sep, id_prefix)` – splits `list_val` into separate values using `sep`, then assigns the values to local variables named `id_prefix_N` where N is the position of the value in the list. The first item has position 0 (zero). The function returns the last element in the list.

Example:

```
list_split('one:two:foo', ':', 'var')
```

is equivalent to:

```
var_0 = 'one'
    var_1 = 'two'
    var_2 = 'foo'
```

list_union

class calibre.utils.formatter_functions.**BuiltinListUnion**

`list_union(list1, list2, separator)` – return a list made by merging the items in `list1` and `list2`, removing duplicate items using a case-insensitive comparison. If items differ in case, the one in `list1` is used. The items in `list1` and `list2` are separated by `separator`, as are the items in the returned list. Aliases: `merge_lists()`, `list_union()`

range

class calibre.utils.formatter_functions.**BuiltinRange**

`range(start, stop, step, limit)` – returns a list of numbers generated by looping over the range specified by the parameters `start`, `stop`, and `step`, with a maximum length of `limit`. The first value produced is ‚start'. Subsequent values `next_v = current_v + step`. The loop continues while `next_v < stop` assuming `step` is positive, otherwise while `next_v > stop`. An empty list is produced if `start` fails the test: `start >= stop` if `step` is positive. The `limit` sets the maximum length of the list and has a default of 1000. The parameters `start`, `step`, and `limit` are optional. Calling `range()` with one argument specifies `stop`. Two arguments specify `start` and `stop`. Three arguments specify `start`, `stop`, and `step`. Four arguments specify `start`, `stop`, `step` and `limit`.

Examples:

```
range(5) -> '0, 1, 2, 3, 4'
range(0, 5) -> '0, 1, 2, 3, 4'
range(-1, 5) -> '-1, 0, 1, 2, 3, 4'
range(1, 5) -> '1, 2, 3, 4'
range(1, 5, 2) -> '1, 3'
range(1, 5, 2, 5) -> '1, 3'
range(1, 5, 2, 1) -> error(limit exceeded)
```

subitems

class calibre.utils.formatter_functions.**BuiltinSubitems**

`subitems(value, start_index, end_index)` – This function breaks apart lists of tag-like hierarchical items such as genres. It interprets the `value` as a comma- separated list of tag-like items, where each item is a period-separated list. It returns a new list made by extracting from each item the components from `start_index` to `end_index`, then merging the results back together. Duplicates are removed. The first subitem in a period-separated list has an index of zero. If an index is negative then it counts from the end of the list. As a special case, an `end_index` of zero is assumed to be the length of the list.

Examples:

- Assuming a #genre column containing „A.B.C“:
  - `{#genre:subitems(0,1)}` returns „A“
  - `{#genre:subitems(0,2)}` returns „A.B“
  - `{#genre:subitems(1,0)}` returns „B.C“
- Assuming a #genre column containing „A.B.C, D.E“:
  - `{#genre:subitems(0,1)}` returns „A, D“
  - `{#genre:subitems(0,2)}` returns „A.B, D.E“

sublist

class calibre.utils.formatter_functions.**BuiltinSublist**

`sublist(value, start_index, end_index, separator)` – interpret the `value` as a list of items separated by `separator`, returning a new list made from the items from `start_index` to `end_index`. The first item is number zero. If an index is negative, then it counts from the end of the list. As a special case, an end_index of zero is assumed to be the length of the list.

Examples assuming that the tags column (which is comma-separated) contains „A, B, C“:

- `{tags:sublist(0,1,\,)}` returns „A“
- `{tags:sublist(-1,0,\,)}` returns „C“
- `{tags:sublist(0,-1,\,)}` returns „A, B“

Ostatní

arguments

class calibre.utils.formatter_functions.**BuiltinArguments**

`arguments(id[=expression] [, id[=expression]]*)` – Used in a stored template to retrieve the arguments passed in the call. It both declares and initializes local variables with the supplied names, the `id``s, making them effectively parameters. The variables are positional; they get the value of the argument given in the call in the same position. If the corresponding argument is not provided in the call then ``arguments()` assigns that variable the provided default value. If there is no default value then the variable is set to the empty string.

assign

class calibre.utils.formatter_functions.**BuiltinAssign**

`assign(id, value)` – assigns `value` to `id`, then returns `value`. `id` must be an identifier, not an expression. In most cases you can use the = operator instead of this function.

globals

class calibre.utils.formatter_functions.**BuiltinGlobals**

`globals(id[=expression] [, id[=expression]]*)` – Retrieves „global variables“ that can be passed into the formatter. The name `id` is the name of the global variable. It both declares and initializes local variables with the names of the global variables passed in (the `id` parameters. If the corresponding variable is not provided in the globals then it assigns that variable the provided default value. If there is no default value then the variable is set to the empty string.)

is_dark_mode

class calibre.utils.formatter_functions.**BuiltinIsDarkMode**

`is_dark_mode()` – returns `'1'` if calibre is running in dark mode, `''` (the empty string) otherwise. This function can be used in advanced color and icon rules to choose different colors/icons according to the mode. Example:

```
if is_dark_mode() then 'dark.png' else 'light.png' fi
```

print

class calibre.utils.formatter_functions.**BuiltinPrint**

`print(a [, b]*)` – prints the arguments to standard output. Unless you start calibre from the command line (`calibre-debug -g`), the output will go into a black hole. The `print` function always returns its first argument.

set_globals

class calibre.utils.formatter_functions.**BuiltinSetGlobals**

`set_globals(id[=expression] [, id[=expression]]*)` – Sets *globalvariables* that can be passed into the formatter. The globals are given the name of the `id` passed in. The value of the `id` is used unless an expression is provided.

Recursion

eval

class calibre.utils.formatter_functions.**BuiltinEval**

`eval(string)` – evaluates the string as a program, passing the local variables. This permits using the template processor to construct complex results from local variables. In Template Program Mode[95], because the `{` and `}` characters are interpreted before the template is evaluated you must use `[[` for the `{` character and `]]` for the `}` character. They are converted automatically. Note also that prefixes and suffixes (the `|prefix|suffix` syntax) cannot be used in the argument to this function when using Template Program Mode.

template

class calibre.utils.formatter_functions.**BuiltinTemplate**

`template(x)` – evaluates `x` as a template. The evaluation is done in its own context, meaning that variables are not shared between the caller and the template evaluation. If not using General Program Mode, because the `{` and `}` characters are special, you must use `[[` for the `{` character and `]]` for the } character; they are converted automatically. For example, `template(\'[[title_sort]]\')` will evaluate the template `{title_sort}` and return its value. Note also that prefixes and suffixes (the `|prefix|suffix` syntax) cannot be used in the argument to this function when using template program mode.

Relational

cmp

class calibre.utils.formatter_functions.**BuiltinCmp**

`cmp(value, y, lt, eq, gt)` – compares `value` and `y` after converting both to numbers. Returns `lt` if `value <# y`, `eq` if `value ==# y`, otherwise `gt`. This function can usually be replaced with one of the numeric compare operators (`==#`, `<#`, `>#`, etc).

first_matching_cmp

class calibre.utils.formatter_functions.**BuiltinFirstMatchingCmp**

`first_matching_cmp(val, [ cmp, result, ]* else_result)` – compares `val < cmp` in sequence, returning the associated `result` for the first comparison that succeeds. Returns `else_result` if no comparison succeeds.

Example:

[95] https://manual.calibre-ebook.com/template_lang.html#more-complex-programs-in-template-expressions-template-program-mode

```
i = 10;
first_matching_cmp(i,5,"small",10,"middle",15,"large","giant")
```

returns "large". The same example with a first value of 16 returns "giant".

strcmp

class calibre.utils.formatter_functions.**BuiltinStrcmp**

strcmp(x, y, lt, eq, gt) – does a case-insensitive lexical comparison of x and y. Returns lt if x < y, eq if x == y, otherwise gt. This function can often be replaced by one of the lexical comparison operators (==, >, <, etc.)

strcmpcase

class calibre.utils.formatter_functions.**BuiltinStrcmpcase**

strcmpcase(x, y, lt, eq, gt) – does a case-sensitive lexical comparison of x and y. Returns lt if x < y, eq if x == y, otherwise gt.

Note: This is NOT the default behavior used by calibre, for example, in the lexical comparison operators (==, >, <, etc.). This function could cause unexpected results, preferably use strcmp() whenever possible.

String manipulation

character

class calibre.utils.formatter_functions.**BuiltinCharacter**

character(character_name) – returns the character named by character_name. For example, character('newline') returns a newline character ('\n'). The supported character names are newline, return, tab, and backslash. This function is used to put these characters into the output of templates.

check_yes_no

class calibre.utils.formatter_functions.**BuiltinCheckYesNo**

check_yes_no(field_name, is_undefined, is_false, is_true) – checks if the value of the yes/no field named by the lookup name field_name is one of the values specified by the parameters, returning 'Yes' if a match is found otherwise returning the empty string. Set the parameter is_undefined, is_false, or is_true to 1 (the number) to check that condition, otherwise set it to 0.

Example: check_yes_no("#bool", 1, 0, 1) returns 'Yes' if the yes/no field #bool is either True or undefined (neither True nor False).

More than one of is_undefined, is_false, or is_true can be set to 1.

contains

class calibre.utils.formatter_functions.**BuiltinContains**

contains(value, pattern, text_if_match, text_if_not_match) – checks if the value is matched by the regular expression pattern. Returns text_if_match if the pattern matches the value, otherwise returns text_if_not_match.

field_exists

class calibre.utils.formatter_functions.**BuiltinFieldExists**

`field_exists(lookup_name)` – checks if a field (column) with the lookup name `lookup_name` exists, returning `'1'` if so and the empty string if not.

ifempty

class calibre.utils.formatter_functions.**BuiltinIfempty**

`ifempty(value, text_if_empty)` – if the `value` is not empty then return that `value`, otherwise return `text_if_empty`.

re

class calibre.utils.formatter_functions.**BuiltinRe**

`re(value, pattern, replacement)` – return the `value` after applying the regular expression. All instances of `pattern` in the value are replaced with `replacement`. The template language uses case insensitive Python regular expressions[96].

re_group

class calibre.utils.formatter_functions.**BuiltinReGroup**

`re_group(value, pattern [, template_for_group]*)` – return a string made by applying the regular expression `pattern` to `value` and replacing each matched instance with the value returned by the corresponding template. In Template Program Mode[97], like for the `template` and the `eval` functions, you use `[[` for `{` and `]]` for `}`.

The following example looks for a series with more than one word and uppercases the first word:

```
program: re_group(field('series'), "(\S* )(.*)", "{$:uppercase()}", "{$}")'}
```

shorten

class calibre.utils.formatter_functions.**BuiltinShorten**

`shorten(value, left_chars, middle_text, right_chars)` – Return a shortened version of the `value`, consisting of `left_chars` characters from the beginning of the `value`, followed by `middle_text`, followed by `right_chars` characters from the end of the `value`. `left_chars` and `right_chars` must be non-negative integers.

Example: assume you want to display the title with a length of at most 15 characters in length. One template that does this is `{title:shorten(9,-,5)}`. For a book with the title *Ancient English Laws inthe Times of Ivanhoe* the result will be *Ancient E-anhoe*: the first 9 characters of the title, a `-`, then the last 5 characters. If the value's length is less than `left chars` + `right chars` + the length of `middle text` then the value will be returned unchanged. For example, the title *TheDome* would not be changed.

strcat

class calibre.utils.formatter_functions.**BuiltinStrcat**

`strcat(a [, b]*)` – returns a string formed by concatenating all the arguments. Can take any number of arguments. In most cases you can use the `&` operator instead of this function.

[96] https://docs.python.org/3/library/re.html
[97] https://manual.calibre-ebook.com/template_lang.html#more-complex-programs-in-template-expressions-template-program-mode

strcat_max

class calibre.utils.formatter_functions.**BuiltinStrcatMax**

`strcat_max(max, string1 [, prefix2, string2]*)` – Returns a string formed by concatenating the arguments. The returned value is initialized to `string1`. Strings made from `prefix, string` pairs are added to the end of the value as long as the resulting string length is less than `max`. Prefixes can be empty. Returns `string1` even if `string1` is longer than `max`. You can pass as many `prefix, string` pairs as you wish.

strlen

class calibre.utils.formatter_functions.**BuiltinStrlen**

`strlen(value)` – Returns the length of the string `value`.

substr

class calibre.utils.formatter_functions.**BuiltinSubstr**

`substr(value, start, end)` – returns the `start`'th through the `end`'th characters of `value`. The first character in `value` is the zero'th character. If `end` is negative then it indicates that many characters counting from the right. If `end` is zero, then it indicates the last character. For example, `substr('12345', 1, 0)` returns `'2345'`, and `substr('12345', 1, -1)` returns `'234'`.

swap_around_articles

class calibre.utils.formatter_functions.**BuiltinSwapAroundArticles**

`swap_around_articles(value, separator)` – returns the `value` with articles moved to the end, separated by a semicolon. The `value` can be a list, in which case each item in the list is processed. If the `value` is a list then you must provide the `separator`. If no `separator` is provided or the separator is the empty string then the `value` is treated as being a single value, not a list. The *articles* are those used by calibre to generate the `title_sort`.

swap_around_comma

class calibre.utils.formatter_functions.**BuiltinSwapAroundComma**

`swap_around_comma(value)` – given a `value` of the form `B, A`, return `A B`. This is most useful for converting names in LN, FN format to FN LN. If there is no comma in the `value` then the function returns the value unchanged.

test

class calibre.utils.formatter_functions.**BuiltinTest**

`test(value, text_if_not_empty, text_if_empty)` – return `text_if_not_empty` if the value is not empty, otherwise return `text_if_empty`.

transliterate

class calibre.utils.formatter_functions.**BuiltinTransliterate**

`transliterate(value)` – Return a string in a latin alphabet formed by approximating the sound of the words in `value`. For example, if `value` is `Фёдор Михáйлович Достоевский` this function returns `Fiodor Mikhailovich Dostoievskii`.

URL functions

encode_for_url

class calibre.utils.formatter_functions.**BuiltinEncodeForURL**

`encode_for_url(value, use_plus)` – returns the `value` encoded for use in a URL as specified by `use_plus`. The value is first URL-encoded. Next, if `use_plus` is `0` then spaces are replaced by `'+'` (plus) signs. If it is `1` then spaces are replaced by `%20`.

If you do not want the value to be encoding but to have spaces replaced then use the *re()* (stránka 206) function, as in `re($series, ' ', '%20')`

See also the functions *make_url()* (stránka 208), *make_url_extended()* (stránka 208) and *query_string()* (stránka 209).

make_url

class calibre.utils.formatter_functions.**BuiltinMakeUrl**

`make_url(path, [query_name, query_value]+)` – this function is the easiest way to construct a query URL. It uses a `path`, the web site and page you want to query, and `query_name`, `query_value` pairs from which the query is built. In general, the `query_value` must be URL-encoded. With this function it is always encoded and spaces are always replaced with `'+'` signs.

At least one `query_name, query_value` pair must be provided.

Example: constructing a Wikipedia search URL for the author *Niccolò Machiavelli*:

```
make_url('https://en.wikipedia.org/w/index.php', 'search', 'Niccolò Machiavelli')
```

returns

```
https://en.wikipedia.org/w/index.php?search=Niccol%C3%B2+Machiavelli
```

If you are writing a custom column book details URL template then use `$item_name` or `field('item_name')` to obtain the value of the field that was clicked on. Example: if *Niccolò Machiavelli* was clicked then you can construct the URL using:

```
make_url('https://en.wikipedia.org/w/index.php', 'search', $item_name)
```

See also the functions *make_url_extended()* (stránka 208), *query_string()* (stránka 209) and *encode_for_url()* (stránka 208).

make_url_extended

class calibre.utils.formatter_functions.**BuiltinMakeUrlExtended**

`make_url_extended(...)` – this function is similar to *make_url()* (stránka 208) but gives you more control over the URL components. The components of a URL are

scheme://**authority**/**path**?**query string**.

See Uniform Resource Locator[98] on Wikipedia for more detail.

The function has two variants:

```
make_url_extended(scheme, authority, path, [query_name, query_value]+)
```

[98] https://en.wikipedia.org/wiki/URL

and

```
make_url_extended(scheme, authority, path, query_string)
```

This function returns a URL constructed from the `scheme`, `authority`, `path`, and either the `query_string` or a query string constructed from the query argument pairs. The `authority` can be empty, which is the case for `calibre` scheme URLs. You must supply either a `query_string` or at least one `query_name, query_value` pair. If you supply `query_string` and it is empty then the resulting URL will not have a query string section.

Example 1: constructing a Wikipedia search URL for the author *Niccolò Machiavelli*:

```
make_url_extended('https', 'en.wikipedia.org', '/w/index.php', 'search', 'Niccolò␣
↪Machiavelli')
```

returns

```
https://en.wikipedia.org/w/index.php?search=Niccol%C3%B2+Machiavelli
```

See the *query_string()* (stránka 209) function for an example using `make_url_extended()` with a `query_string`.

If you are writing a custom column book details URL template then use `$item_name` or `field('item_name')` to obtain the value of the field that was clicked on. Example: if *Niccolò Machiavelli* was clicked on then you can construct the URL using :

```
make_url_extended('https', 'en.wikipedia.org', '/w/index.php', 'search', $item_name')
```

See also the functions *make_url()* (stránka 208), *query_string()* (stránka 209) and *encode_for_url()* (stránka 208).

query_string

class calibre.utils.formatter_functions.**BuiltinQueryString**

`query_string([query_name, query_value, how_to_encode]+)` – returns a URL query string constructed from the `query_name, query_value, how_to_encode` triads. A query string is a series of items where each item looks like `query_name=query_value` where `query_value` is URL-encoded as instructed. The query items are separated by `'&'` (ampersand) characters.

If `how_to_encode` is `0` then `query_value` is encoded and spaces are replaced with `'+'` (plus) signs. If `how_to_encode` is `1` then `query_value` is encoded with spaces replaced by `%20`. If `how_to_encode` is `2` then `query_value` is returned unchanged; no encoding is done and spaces are not replaced. If you want `query_value` not to be encoded but spaces to be replaced then use the *re()* (stránka 206) function, as in `re($series, ' ', '%20')`

You use this function if you need specific control over how the parts of the query string are constructed. You could then use the resultingquery string in *make_url_extended()* (stránka 208), as in

```
make_url_extended(
       'https', 'your_host', 'your_path',
       query_string('encoded', 'Hendrik Bäßler', 0, 'unencoded', 'Hendrik Bäßler', 2))
```

giving you

```
https://your_host/your_path?encoded=Hendrik+B%C3%A4%C3%9Fler&unencoded=Hendrik Bäßler
```

You must have at least one `query_name, query_value, how_to_encode` triad, but can have as many as you wish.

The returned value is a URL query string with all the specified items, for example: `name1=val1[&nameN=valN]*`. Note that the `'?'` *path / query string* separator is not included in the returned result.

If you are writing a custom column book details URL template then use `$item_name` or `field('item_name')` to obtain the unencoded value of the field that was clicked. You also have `item_value_quoted` where the value is already encoded with plus signs replacing spaces, and `item_value_no_plus` where the value is already encoded with `%20` replacing spaces.

See also the functions *make_url()* (stránka 208), *make_url_extended()* (stránka 208) and *encode_for_url()* (stránka 208).

to_hex

class calibre.utils.formatter_functions.**BuiltinToHex**

`to_hex(val)` – returns the string `val` encoded into hex. This is useful when constructing calibre URLs.

urls_from_identifiers

class calibre.utils.formatter_functions.**BuiltinUrlsFromIdentifiers**

`urls_from_identifiers(identifiers, sort_results)` – given a comma-separated list of `identifiers`, where an `identifier` is a colon-separated pair of values (`id_name:id_value`), returns a comma-separated list of HTML URLs generated from the identifiers. The list not sorted if `sort_results` is `0` (character or number), otherwise it is sorted alphabetically by the identifier name. The URLs are generated in the same way as the built-in identifiers column when shown in Book Details.

API of the Metadata objects

The python implementation of the template functions is passed in a Metadata object. Knowing it's API is useful if you want to define your own template functions.

class calibre.ebooks.metadata.book.base.**Metadata**(*title*, *authors=('Neznámý',)*, *other=None*, *template_cache=None*, *formatter=None*)

A class representing all the metadata for a book. The various standard metadata fields are available as attributes of this object. You can also stick arbitrary attributes onto this object.

Metadata from custom columns should be accessed via the get() method, passing in the lookup name for the column, for example: „#mytags“.

Use the *is_null()* (stránka 210) method to test if a field is null.

This object also has functions to format fields into strings.

The list of standard metadata fields grows with time is in *STANDARD_METADATA_FIELDS* (stránka 212).

Please keep the method based API of this class to a minimum. Every method becomes a reserved field name.

is_null(*field*)

Return True if the value of field is null in this object. ‚null‘ means it is unknown or evaluates to False. So a title of _(‚Unknown‘) is null or a language of ‚und‘ is null.

Be careful with numeric fields since this will return True for zero as well as None.

Also returns True if the field does not exist.

deepcopy(*class_generator=<function Metadata.<lambda>>*)

Do not use this method unless you know what you are doing, if you want to create a simple clone of this object, use `deepcopy_metadata()` instead. Class_generator must be a function that returns an instance of Metadata or a subclass of it.

get_identifiers()

Return a copy of the identifiers dictionary. The dict is small, and the penalty for using a reference where a copy is needed is large. Also, we don't want any manipulations of the returned dict to show up in the book.

set_identifiers(*identifiers*)

Set all identifiers. Note that if you previously set ISBN, calling this method will delete it.

set_identifier(*typ*, *val*)

If val is empty, deletes identifier of type typ

standard_field_keys()

return a list of all possible keys, even if this book doesn't have them

custom_field_keys()

return a list of the custom fields in this book

all_field_keys()

All field keys known by this instance, even if their value is None

metadata_for_field(*key*)

return metadata describing a standard or custom field.

all_non_none_fields()

Return a dictionary containing all non-None metadata fields, including the custom ones.

get_standard_metadata(*field*, *make_copy*)

return field metadata from the field if it is there. Otherwise return None. field is the key name, not the label. Return a copy if requested, just in case the user wants to change values in the dict.

get_all_standard_metadata(*make_copy*)

return a dict containing all the standard field metadata associated with the book.

get_all_user_metadata(*make_copy*)

return a dict containing all the custom field metadata associated with the book.

get_user_metadata(*field*, *make_copy*)

return field metadata from the object if it is there. Otherwise return None. field is the key name, not the label. Return a copy if requested, just in case the user wants to change values in the dict.

set_all_user_metadata(*metadata*)

store custom field metadata into the object. Field is the key name not the label

set_user_metadata(*field*, *metadata*)

store custom field metadata for one column into the object. Field is the key name not the label

remove_stale_user_metadata(*other_mi*)

Remove user metadata keys (custom column keys) if they don't exist in ‚other_mi', which must be a metadata object

template_to_attribute(*other*, *ops*)

Takes a list [(src,dest), (src,dest)], evaluates the template in the context of other, then copies the result to self[dest]. This is on a best-efforts basis. Some assignments can make no sense.

smart_update(*other*, *replace_metadata=False*)

Merge the information in *other* into self. In case of conflicts, the information in *other* takes precedence, unless the information in *other* is NULL.

format_field(*key*, *series_with_index=True*)

Returns the tuple (display_name, formatted_value)

to_html()

A HTML representation of this object.

calibre.ebooks.metadata.book.base.**STANDARD_METADATA_FIELDS**

The set of standard metadata fields.

```
'''
All fields must have a NULL value represented as None for simple types,
an empty list/dictionary for complex types and (None, None) for cover_data
'''

SOCIAL_METADATA_FIELDS = frozenset((
    'tags',             # Ordered list
    'rating',           # A floating point number between 0 and 10
    'comments',         # A simple HTML enabled string
    'series',           # A simple string
    'series_index',     # A floating point number
    # Of the form { scheme1:value1, scheme2:value2}
    # For example: {'isbn':'123456789', 'doi':'xxxx', ... }
    'identifiers',
))

'''
The list of names that convert to identifiers when in get and set.
'''

TOP_LEVEL_IDENTIFIERS = frozenset((
    'isbn',
))

PUBLICATION_METADATA_FIELDS = frozenset((
    'title',            # title must never be None. Should be _('Unknown')
    # Pseudo field that can be set, but if not set is auto generated
    # from title and languages
    'title_sort',
    'authors',          # Ordered list. Must never be None, can be [_('Unknown')]
    'author_sort_map',  # Map of sort strings for each author
    # Pseudo field that can be set, but if not set is auto generated
    # from authors and languages
    'author_sort',
    'book_producer',
    'timestamp',        # Dates and times must be timezone aware
    'pubdate',
    'last_modified',
    'rights',
    # So far only known publication type is periodical:calibre
    # If None, means book
    'publication_type',
    'uuid',             # A UUID usually of type 4
    'languages',        # ordered list of languages in this publication
```

(continues on next page)

(pokračujte na předchozí stránce)

```
    'publisher',        # Simple string, no special semantics
    # Absolute path to image file encoded in filesystem_encoding
    'cover',
    # Of the form (format, data) where format is, e.g. 'jpeg', 'png', 'gif'...
    'cover_data',
    # Either thumbnail data, or an object with the attribute
    # image_path which is the path to an image file, encoded
    # in filesystem_encoding
    'thumbnail',
))

BOOK_STRUCTURE_FIELDS = frozenset((
    # These are used by code, Null values are None.
    'toc', 'spine', 'guide', 'manifest',
))

USER_METADATA_FIELDS = frozenset((
    # A dict of dicts similar to field_metadata. Each field description dict
    # also contains a value field with the key #value#.
    'user_metadata',
))

DEVICE_METADATA_FIELDS = frozenset((
    'device_collections',   # Ordered list of strings
    'lpath',                # Unicode, / separated
    'size',                 # In bytes
    'mime',                 # Mimetype of the book file being represented
))

CALIBRE_METADATA_FIELDS = frozenset((
    'application_id',   # An application id, currently set to the db_id.
    'db_id',            # the calibre primary key of the item.
    'formats',          # list of formats (extensions) for this book
    # a dict of user category names, where the value is a list of item names
    # from the book that are in that category
    'user_categories',
    # a dict of items to associated hyperlink
    'link_maps',
))

ALL_METADATA_FIELDS =      SOCIAL_METADATA_FIELDS.union(
                           PUBLICATION_METADATA_FIELDS).union(
                           BOOK_STRUCTURE_FIELDS).union(
                           USER_METADATA_FIELDS).union(
                           DEVICE_METADATA_FIELDS).union(
                           CALIBRE_METADATA_FIELDS)

# All fields except custom fields
STANDARD_METADATA_FIELDS = SOCIAL_METADATA_FIELDS.union(
                           PUBLICATION_METADATA_FIELDS).union(
                           BOOK_STRUCTURE_FIELDS).union(
                           DEVICE_METADATA_FIELDS).union(
```

(continues on next page)

(pokračujte na předchozí stránce)

```
                            CALIBRE_METADATA_FIELDS)

# Metadata fields that smart update must do special processing to copy.
SC_FIELDS_NOT_COPIED =      frozenset(('title', 'title_sort', 'authors',
                                       'author_sort', 'author_sort_map',
                                       'cover_data', 'tags', 'languages',
                                       'identifiers'))

# Metadata fields that smart update should copy only if the source is not None
SC_FIELDS_COPY_NOT_NULL =  frozenset(('device_collections', 'lpath', 'size', 'comments
↪', 'thumbnail'))

# Metadata fields that smart update should copy without special handling
SC_COPYABLE_FIELDS =        SOCIAL_METADATA_FIELDS.union(
                            PUBLICATION_METADATA_FIELDS).union(
                            BOOK_STRUCTURE_FIELDS).union(
                            DEVICE_METADATA_FIELDS).union(
                            CALIBRE_METADATA_FIELDS) - \
                            SC_FIELDS_NOT_COPIED.union(
                            SC_FIELDS_COPY_NOT_NULL)

SERIALIZABLE_FIELDS =       SOCIAL_METADATA_FIELDS.union(
                            USER_METADATA_FIELDS).union(
                            PUBLICATION_METADATA_FIELDS).union(
                            CALIBRE_METADATA_FIELDS).union(
                            DEVICE_METADATA_FIELDS) - \
                            frozenset(('device_collections', 'formats',
                                'cover_data'))
# these are rebuilt when needed
```

10.4 Vše o používání regulárních výrazů v Calibre

Regulární výrazy jsou funkce používané na mnoha místech v Calibre pro provádění propracovaného zpracování obsahu e-knihy a metadat. Tento kurz je mírný úvod do problematiky používání regulárních výrazů v Calibre.

Obsah

- *Nejdříve malé varování a malé povzbuzení* (stránka 215)
- *Kde v Calibre můžete použít regulární výrazy?* (stránka 215)
- *Co to u všech všudy* je *regulární příkaz?* (stránka 215)
- *Zajímá vás vysvětlení?* (stránka 216)
- *To nezní tak špatně. Co dál?* (stránka 216)
- *Parádní! Začíná to dávat smysl!* (stránka 216)
- *No, tyto speciální znaky jsou velice elegantní a tak, ale co když chci porovnat tečku nebo otazník?* (stránka 217)
- *Takže, jaké jsou nejužitečnější sady?* (stránka 217)

- *Ale když budu mít několik různých řetězců, které chci porovnat, zkomplikuje se to?* (stránka 217)
- *Zapomněl jsi…* (stránka 218)
- *Na začátku jsi říkal, že existuje způsob, aby regulární výraz nerozlišoval malá a velká písmena?* (stránka 218)
- *Myslím, že teď už začínám ty regulární výrazy chápat… Jak je použiju v Calibre?* (stránka 218)
  - *Převody* (stránka 218)
  - *Přidávání knih* (stránka 219)
  - *Hromadná úprava metadat* (stránka 219)
- *Rychlý odkaz* (stránka 220)
- *Poděkování* (stránka 224)

10.4.1 Nejdříve malé varování a malé povzbuzení

Toto bude nevyhnutelně poněkud technické – ostatně regulární výrazy jsou technickým nástrojem pro to technické záležitosti. Budu muset použít nějaký žargon a pojmy, které se mohou zdát složité a spletité. Budu se snažit vysvětlit tyto pojmy co nejjasněji, ale opravdu není možné obejít se bez jejich použití. Jak již bylo řečeno, nenechte se odradit žádným žargonem, protože se snažím vysvětlit všechno nové. A zatímco samotné regulární výrazy vám můžou připadat jako tajemná, černé magie (nebo, abychom byli prozaičtější, náhodný řetězec nesmyslných písmen a znaků), slibuji, že nejsou vůbec tak složité. Dokonce i ti, kteří pochopili regulární výrazy opravdu dobře, mají problémy se čtením těch složitějších, ale jejich psaní není tak obtížně – vytváříte výraz krok za krokem. Takže vykročte a následujte mě do králičí nory.

10.4.2 Kde v Calibre můžete použít regulární výrazy?

Existuje několik míst, kde Calibre používá regulární výrazy. Je zde *Najít a nahradit* ve volbách převodu, detekce metadat z názvů souborů v nastavení importu a Najít a nahradit při hromadné úpravě metadat knih. Editor knih Calibre může také používat regulární výrazy ve své funkci *Najít a nahradit*. A nakonec můžete regulární výrazy použít při prohledávání seznamu knih Calibre a při vyhledávání v Prohlížeči e-knih Calibre.

10.4.3 Co to u všech všudy *je* regulární příkaz?

Regulární výraz je způsob, jak popsat sady řetězců. Jeden regulární výraz může *odpovídat* velkému počtu různých řetězců. Proto jsou regulární výrazy tak mocné – jsou stručným způsobem popisu potenciálně velkého počtu variací.

Poznámka

Řetězec zde používám ve smyslu, v jakém se používá v programovacích jazycích: řetězec jednoho nebo více znaků, znaků včetně skutečných znaků, čísel, interpunkčních znamének a takzvaných prázdných znaků (konce řádků, tabulátorů atd). Pamatujte, že velká a malá písmena nejsou obecně považována za stejná, tedy „a“ je jiný znak než „A“ a tak dále. V Calibre regulární výrazy rozlišují velikost písmen v Panelu hledání, ale ne ve volbách převodu. Existuje způsob, aby každý regulární výraz rozlišoval velikost písmen, ale to probereme později. Trochu se to komplikuje, protože regulární výrazy umožňují variace v řetězcích, se kterými se porovnávají, takže jeden výraz může odpovídat více řetězcům, což je důvod, proč se je lidé vůbec obtěžují používat. Více si o tom povíme za chvíli.

10.4.4 Zajímá vás vysvětlení?

No, proto jsme tady. Za prvé, toto je nejdůležitější koncept v regulárních výrazech: *Řetězec sám o sobě je regulární výraz, který odpovídá sobě samému*. To znamená, že když chci pomocí regulárního výrazu porovnat řetězec `"Hello, World!"`, použitý regulární výraz by byl `Hello, World!`. Ano, opravdu je to tak jednoduché. Ale všimněte si, že toto odpovídá *pouze* přesnému řetězci `"Hello, World!"`, ne například `"Hello, wOrld!"` nebo `"hello, world!"` nebo jakékoliv jiné takové variaci.

10.4.5 To nezní tak špatně. Co dál?

Dál začínají opravdu dobré věci. Pamatujete si, jak jsem říkal, že regulární výrazy mohou odpovídat více řetězcům? Teď to začne být trochu složitější. Řekněme, jako poněkud praktičtější cvičení, že e-kniha, kterou chcete převést, má ošklivé zápatí počítající stránky, jako „Strana 5 z 423“. Je zřejmé, že číslo stránky bude růst od 1 do 423, takže budete muset porovnat 423 různých řetězců, že? Špatně, ve skutečnosti: regulární výrazy vám umožňují definovat sady znaků, které jsou porovnávány. Abyste definovali sadu, dejte všechny znaky, které chcete mít v sadě, do hranatých závorek. Takže například sada `[abc]` by odpovídala buď znaku „a“, „b“ nebo „c“. *Sady budou vždy porovnávat pouze jeden ze znaků v sadě*. „Rozumějí“ rozsahům znaků, takže pokud chcete porovnat všechna malá písmena, použili byste sadu `[a-z]`, pro malá a velká písmena byste použili `[a-zA-Z]` a tak dále. Chápete to? Takže samozřejmě pomocí výrazu `Strana [0-9] z 423` byste byli schopni porovnat prvních 9 stránek, čímž se sníží počet potřebných výrazů na tři: Druhý výraz `Strana [0-9][0-9] z 423` by porovnal všechna dvojciferná čísla stránek, a jsem si jistý, že umíte odhadnout, jak by vypadal třetí výraz. Ano, do toho. Napište ho.

10.4.6 Parádní! Začíná to dávat smysl!

Doufal jsem, že to řeknete. Ale připravte se, teď to bude ještě lepší! Právě jsme viděli, že pomocí sad jsme mohli porovnat jeden z několika znaků najednou. Ale můžete dokonce opakovat znak nebo sadu, čímž snížíte počet výrazů potřebných pro zpracování výše uvedeného příkladu s čísly stránek na jeden. Ano, JEDEN! Jste nadšení? Měli byste být! Funguje to takhle: Některé takzvané speciální znaky, „+“, „?“ a „*“, *opakují jednotlivý předcházející prvek*. (Prvek znamená buď jeden znak, sadu znaků, řídící sekvenci nebo skupinu (o těch posledních dvou se budeme učit později.) – zkrátka každý jednotlivý prvek v regulárním výrazu). Tyto znaky se nazývají zástupné znaky nebo kvantifikátory. Abychom byli přesnější, „?“ odpovídá *0 nebo 1* výskytů předcházejícího prvku, „*“ odpovídá *0 nebo více* výskytů předcházejícího prvku a „+“ odpovídá *1 nebo více* výskytů předcházejícího prvku. Několik příkladů: Výraz `a?` by odpovídal buď „“ (což je prázdný řetězec, není nezbytně užitečný v tomto případě), nebo „a“, výraz `a*` by odpovídal „“, „a“, „aa“ nebo libovolnému počtu a v řadě, a konečně výraz `a+` by odpovídal „a“, „aa“ nebo libovolnému počtu a v řadě (Poznámka: neodpovídal by prázdnému řetězci!). To samé pro sady: Výraz `[0-9]+` by odpovídal *každému existujícímu celému číslu*! Vím, co si myslíte, a máte pravdu: Když toto použijete ve výše uvedeném případu porovnávání čísel stránek, nebyl by to jediný výraz pro porovnání všech čísel stránek? Ano, výraz `Strana [0-9]+ z 423` by porovnal každé číslo stránky v této knize!

Poznámka

Poznámka o těchto kvantifikátorech: Obvykle se snaží porovnat tolik textu, kolik je možné, takže buďte opatrní při jejich použití. Nazývá se to „chamtivý chování“ – jsem si jistý, že chápete proč. Je to komplikovanější, když, řekněme, chcete porovnat značku. Vezměme například řetězec `"<p class="calibre2">Tady je nadpis</p>"` a řekněme, že byste chtěli porovnat otevírací značku (část mezi první dvojicí hranatých závorek, značky probereme trochu více později). Mysleli byste si, že by této značce odpovídal výraz `<p.*>`, ale ve skutečnosti odpovídá celému řetězci! (Znak „.“ je další speciální znak. Odpovídá čemukoliv *kromě* koncům řádků, takže v podstatě výraz `.*` by odpovídal každičkému řádku, který si dokážete představit.) Místo toho zkuste použít ``<p.*?>``, čímž přestane být kvantifikátor `"*"` chamtivým. Tento výraz bude odpovídat pouze první otevírací značce, jak bylo zamýšleno. Existuje vlastně další způsob, jak toho dosáhnout: Výraz `<p[^>]*>` bude odpovídat té samé otevírací značce – po další části uvidíte proč. Jen vezměte na vědomí, že dost často existuje více než jeden způsob, jak napsat regulární výraz.

10.4.7 No, tyto speciální znaky jsou velice elegantní a tak, ale co když chci porovnat tečku nebo otazník?

Samozřejmě to můžete udělat: Stačí před jakýkoliv speciální znak vložit zpětné lomítko a je interpretován jako prostý znak bez jakéhokoliv speciálního významu. Tento pár zpětného lomítka následovaného jedním znakem se nazývá řídící sekvence, a vložení zpětného lomítka před speciální znak se nazývá uvozování toho znaku. Řídící sekvence je interpretována jako jeden prvek. Existují samozřejmě řídící sekvence, které dělají více než jen uvozování speciálních znaků, například `"\t"` znamená tabulátor. K některým řídícím sekvencím se dostaneme později. Jo a mimochodem, pokud jde o tyto speciální znaky: Uvažujte o kterémkoliv znaku, o kterém se budeme v tomto úvodu bavit, že má nějakou funkci, která je speciální, a tedy musí být uvozen, pokud chcete prostý znak.

10.4.8 Takže, jaké jsou nejužitečnější sady?

Věděl jsem, že se zeptáte. Některé užitečné sady jsou `[0-9]` odpovídající jednomu číslu, `[a-z]` odpovídající jednomu malému písmenu, `[A-Z]` odpovídající jednomu velkému písmenu, `[a-zA-Z]` odpovídající jednomu písmenu a `[a-zA-Z0-9]` odpovídající jednomu písmenu nebo číslu. Můžete také použít řídící sekvenci jako zjednodušení:

\d
: je ekvivalentem k `[0-9]`

\w
: je ekvivalentem k `[a-zA-Z0-9_]`

\s
: je ekvivalentem k jakémukoliv prázdnému znaku

> **Poznámka**
>
> „Prázdný znak“ je termín pro cokoliv, co nebude vytištěno. Tyto znaky zahrnují mezeru, tabulátor, posun řádku, posun strany, návrat na začátek řádku, pevné mezery atd.

> **Poznámka**
>
> Sady velkých a malých písmen mohou odpovídat jak velkým, tak i malým písmenům, pokud je povoleno nastavení pro vyhledávání bez rozlišování velkých a malých písmen. Taková nastavení lze nalézt například v Předvolby->Vyhledávání v samotném Calibre a na panelu Vyhledávání v nástroji Calibre *Prohlížeč e-knih* a také v nástroji Calibre *Upravit knihu*.

Jako poslední poznámka o sadách, sadu můžete také definovat jako jakýkoliv znak *kromě* těch v sadě. Uděláte to zahrnutím znaku `"^"` jako *úplně prvního znaku v sadě*. Takže `[^a]` by odpovídalo libovolnému znaku kromě „a“. Tomu se říká doplňující sada. Zkratky řídící sekvence, které jsme viděli dříve, mohou být také doplněny: `"\D"` znamená jakýkoli nečíselný znak, takže je ekvivalentem `[^0-9]`. Jak už jste asi uhodli, ostatní zkratky mohou být doplněny pomocí příslušného velkého písmena namísto malého. Takže když se vrátíme k příkladu `<p[^>]*>` z předchozí části, můžete nyní vidět, že použitá sada znaků se pokusí porovnat jakýkoliv znak kromě uzavírací úhlové závorky.

10.4.9 Ale když budu mít několik různých řetězců, které chci porovnat, zkomplikuje se to?

Nebojte se, život je pořád jednoduchý. Vezměme si tento příklad: Kniha, kterou převádíte má napsán „Název“ na každé liché stránce a „Autor“ na každé sudé stránce. V tisku to vypadá skvěle, že jo? Ale v e-knize je to otravné. Můžete seskupit celé výrazy do normálních závorek a znak `"|"` vám umožní porovnat *buď* výraz s jeho pravou *nebo* levou stranou. Toto zkombinujte a máte hotovo. Je to na vás příliš? Dobře, nejdříve seskupíme výrazy pro sudé a liché stránky, takže získáme `(Název)(Autor)` jako naše dva potřebné výrazy. Nyní věci zjednodušíme pomocí svislé čáry ho pruhu (znak `"|"` se

nazývá svislá čára): Pokud použijete výraz `(Název|Autor)`, dostanete shodu buď pro „Název“ (na lichých stránkách), nebo pro „Autor“ (na sudých stránkách). No, nebylo to jednoduché?

Svislou čáru můžete samozřejmě použít také bez použití seskupovacích závorek. Pamatujete, když jsem říkal, že kvantifikátory opakují jim předcházející prvek? No, svislá čára funguje trochu jinak: Výraz „Název|Autor“ bude také odpovídat buď řetězci „Název“ nebo řetězci „Autor“, stejně jako výše uvedený příklad používající seskupování. *Svislá čára vybírá mezi celým výrazem, který ji předchází a následuje*. Takže pokud byste chtěli porovnat řetězce „Calibre“ a „calibre“ a chtěli byste vybrat pouze mezi malým a velkým „c“, museli byste použít výraz `(c|C)alibre`, kde seskupení zajišťuje, že bude vybráno pouze „c“. Pokud byste použili `c|Calibre`, získali byste shodu na řetězec „c“ nebo na řetězec „Calibre“, což není to, co jsme chtěli. Stručně řečeno: Pokud si nejste jisti, použijte seskupování společně se svislou čárou.

10.4.10 Zapomněl jsi...

… počkejte chvilku, je tu ještě jedna poslední, opravdu užitečná věc, kterou můžete dělat se skupinami. Pokud máte skupinu, kterou jste předtím porovnal, můžete použít odkazy na tuto skupinu později ve výrazu: Skupiny jsou číslovány od 1, a odkazujete na ně uvozením čísla skupiny, na kterou chcete odkázat, tedy pátá skupina by byla odkázána `\5`. Takže pokud jste hledali `([^ ]+) \1` v řetězci „Test Test“, porovnali byste celý řetězec!

10.4.11 Na začátku jsi říkal, že existuje způsob, aby regulární výraz nerozlišoval malá a velká písmena?

Ano, říkal, díky za věnování pozornosti a připomenutí. Můžete říct Calibre, jak chcete řešit některé věci pomocí tzv. vlajek. Vlajky zahrnete do svého výrazu pomocí speciální konstrukce `(?sem přijdou vlajky)`, kde byste samozřejmě nahradili „sem přijdou vlajky“ konkrétními vlajkami, které chcete. Pro ignorování velikosti písmen je vlajka `i`, takže zahrnete do svého výrazu `(?i)`. Takže `(?i)test` by odpovídal „Test“, „tEst“, „TEst“ a jakémukoliv případu variace si dokážete vymyslet.

Další užitečná vlajka umožňuje, aby tečka odpovídala všem libovolným znakům, *včetně* nového řádku, a to vlajka `s`. Pokud chcete použít ve výrazu více vlajek, stačí je dát do stejného příkazu: `(?is)` by ignorovalo velikost písmen a tečka by odpovídala všemu. Nezáleží na tom, kterou vlajkou uvedete první, `(?si)` by bylo ekvivalentem výše uvedeného.

10.4.12 Myslím, že teď už začínám ty regulární výrazy chápat... Jak je použiju v Calibre?

Převody

Začněme s nastavením převodu, což je opravdu elegantní. Do části *Hledat a nahradit* můžete zadat regulární výraz, který popisuje řetězec, který bude při převodu nahrazen. Elegantní na tom je průvodce. Klikněte na kouzelnickou hůlku a získáte náhled toho, co Calibre „vidí“ během procesu převodu. Přejděte dolů na řetězec, který chcete odebrat, vyberte ho a zkopírujte, vložte ho do pole regulárního výrazu v horní části okna. Pokud jsou zde proměnné části, jako jsou čísla stránek nebo tak, použijte pro jejich pokrytí sady a kvantifikátory, a když už jsme u toho, nezapomeňte uvodit speciální znaky, pokud tu nějaké jsou. Stiskněte tlačítko označené *Testovat* a Calibre zvýrazní části, které budou nahrazeny, pokud použijete regulární výraz. Jakmile jste spokojeni, klikněte na tlačítko OK a proveďte převod. Buďte opatrní, pokud váš zdroj převodu obsahuje značky, jako tento příklad:

```
Maybe, but the cops feel like you do, Anita. What's one more dead vampire?
New laws don't change that. </p>
<p class="calibre4"> <b class="calibre2">Generated by ABC Amber LIT Conv
<a href="http://www.processtext.com/abclit.html" class="calibre3">erter,
http://www.processtext.com/abclit.html</a></b></p>
<p class="calibre4"> It had only been two years since Addison v. Clark.
The court case gave us a revised version of what life was
```

(nestydatě vytrženo z tohoto vlákna[99]). Museli byste také odstranit některé značky. V tomto příkladu bych doporučoval

[99] https://www.mobileread.com/forums/showthread.php?t=75594"

začít značkou `<b class="calibre2">`, nyní budete muset skončit odpovídající uzavírací značkou (otevírací značky jsou `<značka>`, uzavírací značky jsou `</značka>`), což je v tomto případě jednoduše následující `</b>`. (Podívejte se do dobré příručky k HTML nebo se zeptejte ve fóru, pokud si v tomto bodě nejste jisti.) Počáteční značku lze popsat pomocí `<b.*?>`, uzavírací značku pomocí `</b>`, takže bychom mohli odstranit vše mezi těmito značkami pomocí `<b.*?>.*?</b>`. Ale použití tohoto výrazu by byl špatný nápad, protože odstraňuje vše, co je uzavřeno značkami <b> (který mimochodem vykreslí uzavřený text tučným písmem), a vsaďte se, že tímto způsobem odstraníme části knihy. Místo toho zahrňte i začátek uzavřeného řetězce, takže regulární výraz bude `<b.*?>\s*Generated\s+by\s+ABC\s+Amber\s+LIT.*?</b>`. `\s` s kvantifikátory jsou zde zahrnuty namísto výslovného použití mezer, jak je vidět v řetězci, aby byly zachyceny všechny variace řetězce, které by mohly nastat. Nezapomeňte zkontrolovat, co Calibre odstraní, když testuje nový výraz, abyste se ujistili, že neodstraníte žádné části, které chcete zachovat. Pokud zkontrolujete pouze jeden výskyt, může vám uniknout neshoda někde jinde v textu. Také pamatujte, že kdybyste omylem odstranili více nebo méně značek, než jste doopravdy chtěli, Calibre se pokusí opravit poškozený kód po provedení odstranění.

Přidávání knih

Další věc, ke které můžete použít regulární výrazy, je extrahování metadat z názvů souborů. Tuto funkci můžete najít v části nastavení „Přidávání knih“. Je zde speciální funkce: Můžete použít názvy polí pro pole metadat, například `(?P<title>)` by znamenalo, že Calibre použije tuto část řetězce jako název knihy. Povolené názvy polí jsou uvedeny v oknech, společně s dalším pěkným testovacím polem. Příklad: Řekněme, že chcete importovat spoustu souborů s názvem jako je `Klasické texty: Božská komedie, Dante Alighieri.mobi`. (Samozřejmě, že tu už ve své knihovně máte, protože všichni milujeme klasickou italskou poezii) nebo `Sci-fi eposy: Trilogie Nadace, Isaac Asimov.epub`. To je samozřejmě schéma názvů, ze kterého by Calibre nevyextrahovalo žádná smysluplná data – standardní výraz pro extrahování metadat je `(?P<title>.+) - (?P,author>[^_]+)`. Regulární výraz, který by tu fungoval, by byl `[a-zA-Z]+: (?P<title>.+), (?P<author>.+)`. Pamatujte, že uvnitř skupiny pro pole metadat musíte použít výrazy k popisu toho, čemu pole skutečně odpovídá. A také pamatujte, že při použití testovacího pole, které Calibre poskytuje, musíte přidat příponu souboru k testovanému souboru, jinak nezískáte vůbec žádnou shodu, i když použijete fungující výraz.

Hromadná úprava metadat

Poslední část je *Hledat a nahradit* v polích metadat pomocí regulárního výrazu. K tomuto se můžete dostat výběr více knih v knihovně a použitím hromadné úpravy metadat. Buďte velmi opatrní při používání této poslední funkce, protože může způsobit **velice špatné věci** vaší knihovně! Překontrolujte, že vaše výrazy dělají to, co chcete, pomocí testovacích polí, a označte pouze knihy, které opravdu chcete změnit! V režimu hledání regulárními výrazy můžete hledat v jednom poli, nahradit text něčím a dokonce zapsat výsledek do jiného pole. Praktický příklad: Řekněme, že vaše knihovna obsahuje knihy série Duna Franka Herberta pojmenované po vzoru `Duna 1 - Duna`, `Duna 2 - Spasitel Duny` a tak dále. Nyní chcete dostat `Duna` do pole série. Můžete to udělat hledáním `(.*?) \d+ - .*` v poli názvu a nahrazením `\1` v poli série. Vidíte, co jsem udělal? To je odkaz na první skupinu, kterou nahrazujete pole série. Nyní, když máte sérii nastavenou, potřebujete jen provést další vyhledávání `.*? –` v poli názvu a nahrazením `""` (prázdný řetězec) opět v poli názvu, a vaše metadata jsou jsou pěkně čistá a uklizená. Není to skvělé? Mimochodem, namísto nahrazení celého pole můžete také poli připojit nebo předřadit, takže pokud *chcete*, aby byla názvu knihy předřazena informace o sérii, mohli byste to udělat stejně. Jak jste si už nepochybně všimli, je zde zaškrtávací políčko označené *Rozlišovat malá a velká*, takže zde nebudete muset používat vlajky pro výběr chování.

No, je asi na čase uzavřít velice krátký úvod do regulárních výrazů. Doufám, že jsem vám ukázal dost, abyste alespoň mohli začít a umožnil vám pokračovat v samostatném učení – dobrým výchozím bodem je Dokumentace Pythonu pro regulární výrazy[100].

Ale ještě jedna poslední varování: Regulární výrazy jsou výkonné, ale také je velice snadné je pokazit. Calibre poskytuje opravdu skvělé možnosti pro testování, abyste viděli, jestli se vaše výrazy chovat podle očekávání. Používejte je. Pokuste se nestřelit se do nohy. (Bože, tenhle výraz miluju…). Ale kdybyste si i přes varování poranili nohu (nebo jiné části těla), pokuste se z toho poučit.

[100] https://docs.python.org/library/re.html

10.4.13 Rychlý odkaz

Základy vyhledávání prostřednictvím regulárních výrazů

Tento kontrolní seznam shrnuje nejčastěji používané nebo těžko zapamatovatelné části modulu regulárních výrazů, které jsou k dostupné ve většině součástí Calibre.

Obsah

- *Třídy znaků* (stránka 220)
- *Zjednodušené třídy znaků* (stránka 221)
- *Kvantifikátory* (stránka 221)
- *Nenasytnost* (stránka 221)
- *Alternace* (stránka 221)
- *Vyloučení* (stránka 221)
- *Kotvy* (stránka 222)
- *Skupiny* (stránka 222)
- *Prohledávání* (stránka 222)
- *Rekurze* (stránka 223)
- *Speciální znaky* (stránka 223)
- *Metaznaky* (stránka 223)
- *Režimy* (stránka 224)

Třídy znaků

Stručně řečeno jsou třídy znaků užitečné pro zastupování různých skupin znaků.

Příklady:

Zastoupení	Třída
`[a-z]`	Malá písmena. Nezahrnuje znaky s diakritikou a ligatury
`[a-z0-9]`	Malá písmena od a do z nebo čísla od 0 do 9
`[A-Za-z-]`	Velká nebo malá písmena nebo pomlčka. Pro zahrnutí pomlčky do třídy ji musíte umístit na začátek nebo na konec, aby se nepletla se spojovníkem, který určuje rozsah znaků.
`[^0-9]`	Libovolný znak kromě číslice. Stříška (^) umístěná na začátku třídy vylučuje znaky ze třídy (doplňková třída)
`[[a-z]--[ae`	Malá písmena souhlásek. Třída může být zahrnuta do třídy. Znaky `--` vylučují to, co následuje
`[\w--[\d_]]`	Všechna písmena (včetně cizích znaků s diakritikou). Uvnitř třídy lze použít zkrácené třídy

Příklad:

```
<[^<>]+> to select an HTML tag
```

Zjednodušené třídy znaků

Za-stoupe-ní	Třída
`\d`	Číslo (stejné jako `[0-9]`)
`\D`	Libovolný nečíselný znak (stejné jako `[^0-9]`)
`\w`	Alfanumerický znak (`[a-zA-Z0-9]`) včetně znaků s diakritikou a ligatur
`\W`	Libovolný znak, který není „slovo“
`\s`	Mezera, pevná mezera, tabulátor, enter.
`\S`	Libovolný znak, který není „prázdný znak“
`.`	Jakýkoliv znak kromě nového řádku. Použijte zaškrtávací políčko „vytečkovat vše“ nebo modifikátor regulárních výrazů `(?s)` pro zahrnutí nového řádku.

Kvantifikátory

Kvantifikátor	Počet výskytů výrazů předcházejících kvantifikátor
`?`	0 nebo 1 výskytů výrazu. Stejné jako `{0,1}`
`+`	1 nebo více výskytů výrazu. Stejné jako `{1,}`
`*`	0, 1 nebo více výskytů výrazu. Stejné jako `{0,}`
`{n}`	Přesně n výskytů výrazu
`{min,max}`	Počet výskytů mezi minimální a maximální hodnotou včetně
`{min,}`	Počet výskytů mezi minimální hodnotou včetně a nekonečnem
`{,max}`	Počet výskytů mezi 0 a maximální hodnotou včetně

Nenasytnost

Ve výchozím nastavení je modul regulárního výrazu s kvantifikátory nenasytný: rozšiřuje výběr co nejvíce. To zpočátku často způsobuje překvapení. `?` následovaný kvantifikátorem je lenost. Vyhněte se dávání dvou do stejného výrazu, výsledek může být nepředvídatelný.

Dejte si pozor na vnořené kvantifikátory, například vzor `(a*)*`, protože exponenciálně zvyšuje dobu zpracování.

Alternace

Znak `|` v regulárním výrazu je logické `OR`. To znamená, že se může shodovat buď předcházející nebo následující výraz.

Vyloučení

Metoda 1

```
vzor_k_vyloučení(*SKIP)(*FAIL)|vzor_k_vybrání
```

Příklad:

```
"Blabla"(*SKIP)(*FAIL)|Blabla
```

vybere Blabla v řetězcích Blabla nebo „Blabla nebo Blabla“, ale ne v „Blabla“.

Metoda 2

```
vzor_k_vyloučení\K|(vzor_k_vybrání)
```

```
"Blabla"\K|(Blabla)
```

vybere Blabla v řetězcích Blabla nebo „Blabla nebo Blabla“, ale ne v „Blabla“.

Kotvy

Kotva je způsob, jak porovnat logické umístění v řetězci, spíše než znaku. Nejužitečnější kotvy pro zpracování textu jsou:

\b
: Určuje hranici slova, tj. přechod z mezery na nemezerový znak. Například můžete použít `\bdar`, aby odpovídal `mám dar`, ale ne `nezdar`.

^
: Odpovídá začátku řádku (v režimu více řádků, což je výchozí nastavení)

$
: Odpovídá konci řádku (v režimu více řádků, což je výchozí nastavení)

\K
: Obnovuje výchozí pozici výběru na pozici ve vzoru. Některé moduly regulárních výrazů (ale ne Calibre) nedovolí hledat vpřed v proměnlivé délce, obzvláště s kvantifikátory. Když můžete s těmito moduly použít `\K`, také vám to umožní zbavit se tohoto limitu napsáním ekvivalentu pozitivního hledání vpřed v proměnlivé délce.

Skupiny

(výraz)
: Zachytávací skupina: Uloží výběr a může být později vyvolána ve vzorech *search* nebo *replace* pomocí `\n`, kde `n` je pořadové číslo zachytávací skupiny (počínaje od 1 v pořadí čtení)

(?:výraz)
: Skupina, která nezachytí výběr

(?>výraz)
: Atomická skupina: Jakmile je výraz splněn, modul regulárních výrazů ji předá. A pokud zbytek vzoru selže, nebude se snažit vyzkoušet jiné kombinace s výrazem. Atomické skupiny nezachycují.

(?|výraz)
: Skupina obnovení větve: Větve alternativ zahrnutých ve výrazu sdílejí stejná čísla skupin

(?<name>výraz)
: Skupina s názvem „name“: Výběr lze později vyvolat ve vzorci *search* pomocí výrazu `(?P=name)` a ve vzorci *replace* výrazem `\g`. Dvě různé skupiny mohou používat stejný název.

Prohledávání

Prohledávání	Význam
`?=`	Pozitivní hledání vpřed (bude umístěno za výběr)
`?!`	Negativní hledání vpřed (bude umístěno za výběr)
`?<=`	Pozitivní hledání zpět (bude umístěno před výběr)
`?<!`	Negativní hledání zpět (bude umístěno před výběr)

Hledání vpřed a zpět nespotřebovávají znaky, jsou nulové délky a nezachycují. Jsou to atomické skupiny: jakmile je tvrzení splněno, modul regulárních výrazů ho předá, a pokud zbytek vzoru selže, nebude se vracet zpět do prohledávání, aby vyzkoušel jiné kombinace.

Při hledání více shod v řetězci, v počáteční pozici každého pokusu o shodu, může hledání zpět zkontrolovat znaky před aktuální pozicí. Proto v řetězci 123 by měl vzor `(?<=\d)\d` (číslice předcházející číslici) teoreticky vybrat 2 a 3. Na druhou stranu `\d\K\d` může vybrat pouze 2, protože počáteční pozice po prvním výběru je bezprostředně před 3 a pro druhou shodu zde není dostatek číslic. Podobně `\d(\d)` zachytí pouze 2. V modulu regulárních výrazů Calibre se pozitivní hledání zpět chová stejně a vybere pouze 2, na rozdíl od teorie.

Skupiny mohou být umístěny uvnitř prohledávání, ale zachytávání je zřídka užitečné. Pokud je to přesto užitečné, bude muset být při používání kvantifikátoru v hledání zpět velmi opatrní: chamtivost spojená s absencí zpětného dozoru může poskytovat překvapující zachytávání. Z tohoto důvodu použijte spíše `\K` než pozitivní hledání zpět, když máte v zachytávací skupině pozitivního hledání zpět kvantifikátor (nebo ještě hůře několik).

Příklad negativního hledání vpřed:

```
(?![^<>{}]*[>}])
```

Umístění na konci vzoru zabraňuje výběru uvnitř značky nebo stylu vloženého do souboru.

Kdykoliv je to možné, je vždy lepší vyhledávání „ukotvit“, aby se snížil počet kroků nezbytných k dosažení výsledku.

Rekurze

Zastoupení	Význam
`(?R)`	Rekurze celého vzoru
`(?1)`	Rekurze jediného vzoru číslované zachytávací skupiny, zde skupina 1

Rekurze je vyvolání sebe sama. To je užitečné pro vyvážené dotazy, jako jsou řetězce v uvozovkách, které mohou obsahovat vložené řetězce v uvozovkách. Pokud tedy během zpracování řetězce mezi uvozovkami narazíme na začátek nového řetězce mezi uvozovkami, dobře víme, jak to udělat, a vyvoláme sebe sama. Pak máme podobný vzor:

```
start-pattern(?>atomic sub-pattern|(?R))*end-pattern
```

Pro výběr řetězce mezi uvozovkami bez zastavení na vloženém řetězci:

```
“((?>[^“”]+|(?R))*[^“”]+)”
```

Tuto šablonu lze také použít ke změně párů značek, které mohou být vloženy, například značek `<div>`.

Speciální znaky

Zastoupení	Znak
`\t`	tabulátor
`\n`	konec řádku
`\x20`	(měkká) mezera
`\xa0`	tvrdá mezera

Metaznaky

Metaznaky jsou ty, které mají zvláštní význam pro modul regulárních výrazů. Dvanáct z nich musí předcházet řídící znak, zpětné lomítko (`\`), aby ztratily svůj zvláštní význam a stal se z nich opět normální znak:

```
^ . [ ] $ ( ) * + ? | \
```

Sedm dalších metaznaků nemusí předcházet zpětné lomítko (ale může bez jakýchkoliv dalších důsledků):

```
{ } ! < > = :
```

Speciální znaky ztratí svůj status, pokud jsou použity ve třídě (mezi lomenými závorkami `[]`). Uzavírací lomená závorka a pomlčka mají ve třídě zvláštní status. Mimo třídu je pomlčka jednoduchým literálem, uzavírací lomená závorka zůstává metaznakem.

Lomítko (/) a křížek (nebo znak hash) (#) nejsou metaznaky, nemusejí mít řídící znak.

V některých nástrojích, jako je například regex101.com s modulem Python, mají dvojité uvozovky zvláštní status oddělovače a musí mít řídící znak, nebo se možnosti změní. Toto není případ editoru Calibre.

Režimy

(?s)
: Způsobí, že tečka (`.`) bude odpovídat také znakům nového řádku

(?m)
: Učiní, že kotvy `^` a `$` budou odpovídat začátku a konci řádků namísto začátku a konce celého řetězce.

10.4.14 Poděkování

Díky za pomoc s tipy, opravami a tak:

- ldolse
- kovidgoyal
- chaley
- dwanthny
- kacir
- Starson17
- Orpheu

Více informací o regulárních výrazech naleznete v Uživatelské příručce Pythonu[101]. Skutečná knihovna regulárních výrazů používaná Calibre je: regex[102], která podporuje několik užitečných vylepšení oproti standardní knihovně Pythonu.

10.5 Psaní vlastních modulů pro rozšíření funkcí Calibre

Calibre má velice modulární provedení. Téměř všechny funkce jsou ve formě modulů. Moduly jsou použity pro převod, stahování zpráv (ačkoliv ty jsou nazvány předpisy), různé komponenty uživatelského rozhraní, připojení k různým zařízením, zpracování souborů při jejich přidávání do Calibre a tak dále. Kompletní seznam všech vestavěných modulů můžete získat v Calibre v *Předvolby → Rozšířené → Moduly*.

Zde vás naučíme, jak vytvořit vlastní moduly pro přidání nových funkcí do Calibre.

Obsah

- *Anatomie modulu Calibre* (stránka 225)
- *Modul uživatelského rozhraní* (stránka 226)
  - *__init__.py* (stránka 227)
  - *ui.py* (stránka 228)
  - *main.py* (stránka 230)
  - *Získání zdrojů ze souboru ZIP modulu* (stránka 232)

[101] https://docs.python.org/library/re.html
[102] https://bitbucket.org/mrabarnett/mrab-regex/src/hg/

  - *Povolení uživatelské konfigurace vašeho modulu* (stránka 233)
- *Moduly pro úpravu knihy* (stránka 235)
  - *main.py* (stránka 235)
- *Přidávání překladů do vašeho modulu* (stránka 238)
- *Rozhraní API modulu* (stránka 239)
- *Ladění modulů* (stránka 239)
- *Další příklady modulů* (stránka 239)
- *Sdílení vašeho modulu s ostatními* (stránka 239)

Poznámka

Toto platí pouze pro vydání Calibre >= 0.8.60

10.5.1 Anatomie modulu Calibre

Modul Calibre je velice jednoduchý, je to prostě soubor ZIP, který obsahuje nějaký kód Pythonu a jakékoliv jiné zdroje, jako soubory obrázků vyžadované modulem. Bez dalších okolků se pojďme podívat na základní příklad.

Předpokládejme, že máte instalaci Calibre, kterou používáte pro vlastní publikaci různých elektronických dokumentů ve formátech EPUB a MOBI. Chtěli byste, aby všechny soubory generované v Calibre měli vydavatele nastaveného jako „Hello World“, zde je návod, jak to udělat. Vytvořte soubor s názvem `__init__.py` (toto je speciální název a musí být vždy použit pro hlavní soubor vašeho modulu) a zadejte do něj následující kód Pythonu:

```
from calibre.customize import FileTypePlugin


class HelloWorld(FileTypePlugin):

    name                = 'Hello World Plugin'  # Name of the plugin
    description         = 'Set the publisher to Hello World for all new conversions'
    supported_platforms = ['windows', 'osx', 'linux']  # Platforms this plugin will␣
↪run on
    author              = 'Acme Inc.'  # The author of this plugin
    version             = (1, 0, 0)    # The version number of this plugin
    file_types          = {'epub', 'mobi'}  # The file types that this plugin will be␣
↪applied to
    on_postprocess      = True  # Run this plugin after conversion is complete
    minimum_calibre_version = (0, 7, 53)

    def run(self, path_to_ebook):
        from calibre.ebooks.metadata.meta import get_metadata, set_metadata
        with open(path_to_ebook, 'r+b') as file:
            ext  = os.path.splitext(path_to_ebook)[-1][1:].lower()
            mi = get_metadata(file, ext)
            mi.publisher = 'Hello World'
            set_metadata(file, mi, ext)
        return path_to_ebook
```

To je vše. Pro přidání tohoto kódu do Calibre jako modulu stačí spustit následující ve složce, ve které jste vytvořili `__init__.py`:

```
calibre-customize -b .
```

Poznámka

Na macOS jsou nástroje příkazového řádku uvnitř sady Calibre, například pokud jste nainstalovali Calibre do `/Applications`, nástroje příkazového řádku jsou v `/Applications/calibre.app/Contents/MacOS/`.

Modul Hello World můžete stáhnout z helloworld_plugin.zip[103].

Pokaždé, když použijete Calibre, aby převedlo knihu, bude zavolána metoda `run()` modulu a převedená kniha bude mít vydavatele nastaveného na „Hello World“. Toto je jednoduchý modul, pojďme se přesunout na složitější příklad, který skutečně přidá komponentu do uživatelského rozhraní.

10.5.2 Modul uživatelského rozhraní

Tento modul bude rozdělen do několika souborů (aby kód zůstal čistý). Ukáže vám, jak získat zdroje (soubory obrázků nebo dat) ze souboru ZIP modulu, což umožňuje uživatelům konfigurovat váš modul, jak vytvořit prvky v uživatelském rozhraní Calibre a jak přistupovat k a dotazovat se databáze knih v Calibre.

Tento modul můžete stáhnout z interface_demo_plugin.zip[104]

První věc, na kterou pamatujte, je to, že tento soubor ZIP má v sobě mnohem více souborů, vysvětleno níže, věnujte zvláštní pozornost `plugin-import-name-interface_demo.txt`.

plugin-import-name-interface_demo.txt
: Prázdný textový soubor použitý pro umožnění kouzla vícesouborového modulu. Tento soubor musí být přítomen ve všech modulech, které používají více než jeden soubor .py. Měl by být prázdný a jeho název souboru musí být ve formě: plugin-import-name-**some_name**.txt. Přítomnost tohoto souboru vám umožňuje importovat kód ze souborů .py přítomných uvnitř souboru ZIP pomocí příkazu, jako je:

  ```
  from calibre_plugins.some_name.some_module import some_object
  ```

  Předpona `calibre_plugins` musí být vždy přítomná. `some_name` pochází z názvu prázdného textového souboru. `some_module` odkazuje na soubor `some_module.py` uvnitř souboru ZIP. Pamatujte, že toto importování je právě tak výkonné jako obyčejné importy Pythonu. Uvnitř souboru ZIP můžete vytvářet balíčky a podbalíčky modulů .py, jako byste to dělali normálně (definováním __init__.py v každé podsložce) a všechno by mělo „prostě fungovat.

  Název, který použijete pro `some_name`, zadá globální obor názvů sdílený všemi moduly, **takže by měl být co nejjedinečnější**. Ale nezapomeňte, že to musí být platný identifikátor Pythonu (pouze abecední znaky, číslice a podtržítko).

__init__.py
: Jako předtím, soubor, který definuje třídu modulu

main.py
: Tento soubor obsahuje skutečný kód, který dělá něco užitečného

ui.py
: Tento soubor definuje součást rozhraní modulu

[103] https://calibre-ebook.com/downloads/helloworld_plugin.zip
[104] https://calibre-ebook.com/downloads/interface_demo_plugin.zip

images/icon.png
: Ikona pro tento modul

about.txt
: Textový soubor s informacemi o modulu

translations
: Složka obsahující soubory .mo s překlady uživatelského rozhraní vašeho modulu do různých jazyků. Podívejte se níže na podrobnosti.

Nyní se podívejme na kód.

__init__.py

Nejdříve povinný `__init__.py` pro definování metadat modulu:

```
from calibre.customize import InterfaceActionBase


class InterfacePluginDemo(InterfaceActionBase):
    '''
    This class is a simple wrapper that provides information about the actual
    plugin class. The actual interface plugin class is called InterfacePlugin
    and is defined in the ui.py file, as specified in the actual_plugin field
    below.

    The reason for having two classes is that it allows the command line
    calibre utilities to run without needing to load the GUI libraries.
    '''
    name                = 'Interface Plugin Demo'
    description         = 'An advanced plugin demo'
    supported_platforms = ['windows', 'osx', 'linux']
    author              = 'Kovid Goyal'
    version             = (1, 0, 0)
    minimum_calibre_version = (0, 7, 53)

    #: This field defines the GUI plugin class that contains all the code
    #: that actually does something. Its format is module_path:class_name
    #: The specified class must be defined in the specified module.
    actual_plugin       = 'calibre_plugins.interface_demo.ui:InterfacePlugin'

    def is_customizable(self):
        '''
        This method must return True to enable customization via
        Preferences->Plugins
        '''
        return True

    def config_widget(self):
        '''
        Implement this method and :meth:`save_settings` in your plugin to
        use a custom configuration dialog.

        This method, if implemented, must return a QWidget. The widget can have
        an optional method validate() that takes no arguments and is called
```

(continues on next page)

(pokračujte na předchozí stránce)

```
        immediately after the user clicks OK. Changes are applied if and only
        if the method returns True.

        If for some reason you cannot perform the configuration at this time,
        return a tuple of two strings (message, details), these will be
        displayed as a warning dialog to the user and the process will be
        aborted.

        The base class implementation of this method raises NotImplementedError
        so by default no user configuration is possible.
        '''
        # It is important to put this import statement here rather than at the
        # top of the module as importing the config class will also cause the
        # GUI libraries to be loaded, which we do not want when using calibre
        # from the command line
        from calibre_plugins.interface_demo.config import ConfigWidget
        return ConfigWidget()

    def save_settings(self, config_widget):
        '''
        Save the settings specified by the user with config_widget.

        :param config_widget: The widget returned by :meth:`config_widget`.
        '''
        config_widget.save_settings()

        # Apply the changes
        ac = self.actual_plugin_
        if ac is not None:
            ac.apply_settings()
```

Jediná funkce, která stojí za pozornost, je pole `actual_plugin`. Protože Calibre má jak příkazový řádek, tak grafická rozhraní, moduly grafického rozhraní, jako je tento, by v __init__.py neměly načítat žádné knihovny grafického rozhraní. Pole actual_plugin to dělá za vás tím, že řekne Calibre, že skutečný modul je třeba hledat v jiném souboru uvnitř vašeho archivu ZIP, který bude načten pouze v kontextu grafického rozhraní.

Nezapomeňte, že aby toto fungovalo, musíte mít ve svém souboru ZIP soubor plugin-import-name-some_name.txt, jak bylo probráno výše.

Také existuje pár metod pro povolení uživatelské konfigurace modulu. Ty jsou popsány níže.

ui.py

Nyní se podívejme na ui.py, který definuje skutečné grafické rozhraní modulu. Zdrojový kód je silně okomentovaný a měl by být sebevysvětlující:

```
from calibre.gui2.actions import InterfaceAction
from calibre_plugins.interface_demo.main import DemoDialog


class InterfacePlugin(InterfaceAction):

    name = 'Interface Plugin Demo'
```

(continues on next page)

(pokračujte na předchozí stránce)

```
    # Declare the main action associated with this plugin
    # The keyboard shortcut can be None if you don't want to use a keyboard
    # shortcut. Remember that currently calibre has no central management for
    # keyboard shortcuts, so try to use an unusual/unused shortcut.
    action_spec = ('Interface Plugin Demo', None,
            'Run the Interface Plugin Demo', 'Ctrl+Shift+F1')

    def genesis(self):
        # This method is called once per plugin, do initial setup here

        # Set the icon for this interface action
        # The get_icons function is a builtin function defined for all your
        # plugin code. It loads icons from the plugin zip file. It returns
        # QIcon objects, if you want the actual data, use the analogous
        # get_resources builtin function.
        #
        # Note that if you are loading more than one icon, for performance, you
        # should pass a list of names to get_icons. In this case, get_icons
        # will return a dictionary mapping names to QIcons. Names that
        # are not found in the zip file will result in null QIcons.
        icon = get_icons('images/icon.png', 'Interface Demo Plugin')

        # The qaction is automatically created from the action_spec defined
        # above
        self.qaction.setIcon(icon)
        self.qaction.triggered.connect(self.show_dialog)

    def show_dialog(self):
        # The base plugin object defined in __init__.py
        base_plugin_object = self.interface_action_base_plugin
        # Show the config dialog
        # The config dialog can also be shown from within
        # Preferences->Plugins, which is why the do_user_config
        # method is defined on the base plugin class
        do_user_config = base_plugin_object.do_user_config

        # self.gui is the main calibre GUI. It acts as the gateway to access
        # all the elements of the calibre user interface, it should also be the
        # parent of the dialog
        d = DemoDialog(self.gui, self.qaction.icon(), do_user_config)
        d.show()

    def apply_settings(self):
        from calibre_plugins.interface_demo.config import prefs
        # In an actual non trivial plugin, you would probably need to
        # do something based on the settings in prefs
        prefs
```

main.py

Skutečná logika pro implementaci dialogového okna rozhraní ukázkového modulu.

```
from calibre_plugins.interface_demo.config import prefs


class DemoDialog(QDialog):

    def __init__(self, gui, icon, do_user_config):
        QDialog.__init__(self, gui)
        self.gui = gui
        self.do_user_config = do_user_config

        # The current database shown in the GUI
        # db is an instance of the class LibraryDatabase from db/legacy.py
        # This class has many, many methods that allow you to do a lot of
        # things. For most purposes you should use db.new_api, which has
        # a much nicer interface from db/cache.py
        self.db = gui.current_db

        self.l = QVBoxLayout()
        self.setLayout(self.l)

        self.label = QLabel(prefs['hello_world_msg'])
        self.l.addWidget(self.label)

        self.setWindowTitle('Interface Plugin Demo')
        self.setWindowIcon(icon)

        self.about_button = QPushButton('About', self)
        self.about_button.clicked.connect(self.about)
        self.l.addWidget(self.about_button)

        self.marked_button = QPushButton(
            'Show books with only one format in the calibre GUI', self)
        self.marked_button.clicked.connect(self.marked)
        self.l.addWidget(self.marked_button)

        self.view_button = QPushButton(
            'View the most recently added book', self)
        self.view_button.clicked.connect(self.view)
        self.l.addWidget(self.view_button)

        self.update_metadata_button = QPushButton(
            "Update metadata in a book's files", self)
        self.update_metadata_button.clicked.connect(self.update_metadata)
        self.l.addWidget(self.update_metadata_button)

        self.conf_button = QPushButton(
                'Configure this plugin', self)
        self.conf_button.clicked.connect(self.config)
        self.l.addWidget(self.conf_button)
```

(continues on next page)

(pokračujte na předchozí stránce)

```
        self.resize(self.sizeHint())

    def about(self):
        # Get the about text from a file inside the plugin zip file
        # The get_resources function is a builtin function defined for all your
        # plugin code. It loads files from the plugin zip file. It returns
        # the bytes from the specified file.
        #
        # Note that if you are loading more than one file, for performance, you
        # should pass a list of names to get_resources. In this case,
        # get_resources will return a dictionary mapping names to bytes. Names that
        # are not found in the zip file will not be in the returned dictionary.
        text = get_resources('about.txt')
        QMessageBox.about(self, 'About the Interface Plugin Demo',
                text.decode('utf-8'))

    def marked(self):
        ''' Show books with only one format '''
        db = self.db.new_api
        matched_ids = {book_id for book_id in db.all_book_ids() if len(db.
↪formats(book_id)) == 1}
        # Mark the records with the matching ids
        # new_api does not know anything about marked books, so we use the full
        # db object
        self.db.set_marked_ids(matched_ids)

        # Tell the GUI to search for all marked records
        self.gui.search.setEditText('marked:true')
        self.gui.search.do_search()

    def view(self):
        ''' View the most recently added book '''
        most_recent = most_recent_id = None
        db = self.db.new_api
        for book_id, timestamp in db.all_field_for('timestamp', db.all_book_ids()).
↪items():
            if most_recent is None or timestamp > most_recent:
                most_recent = timestamp
                most_recent_id = book_id

        if most_recent_id is not None:
            # Get a reference to the View plugin
            view_plugin = self.gui.iactions['View']
            # Ask the view plugin to launch the viewer for row_number
            view_plugin._view_calibre_books([most_recent_id])

    def update_metadata(self):
        '''
        Set the metadata in the files in the selected book's record to
        match the current metadata in the database.
        '''
```

(continues on next page)

(pokračujte na předchozí stránce)

```
        from calibre.ebooks.metadata.meta import set_metadata
        from calibre.gui2 import error_dialog, info_dialog

        # Get currently selected books
        rows = self.gui.library_view.selectionModel().selectedRows()
        if not rows or len(rows) == 0:
            return error_dialog(self.gui, 'Cannot update metadata',
                                'No books selected', show=True)
        # Map the rows to book ids
        ids = list(map(self.gui.library_view.model().id, rows))
        db = self.db.new_api
        for book_id in ids:
            # Get the current metadata for this book from the db
            mi = db.get_metadata(book_id, get_cover=True, cover_as_data=True)
            fmts = db.formats(book_id)
            if not fmts:
                continue
            for fmt in fmts:
                fmt = fmt.lower()
                # Get a python file object for the format. This will be either
                # an in memory file or a temporary on disk file
                ffile = db.format(book_id, fmt, as_file=True)
                ffile.seek(0)
                # Set metadata in the format
                set_metadata(ffile, mi, fmt)
                ffile.seek(0)
                # Now replace the file in the calibre library with the updated
                # file. We don't use add_format_with_hooks as the hooks were
                # already run when the file was first added to calibre.
                db.add_format(book_id, fmt, ffile, run_hooks=False)

        info_dialog(self, 'Updated files',
                f'Updated the metadata in the files of {len(ids)} book(s)',
                show=True)

    def config(self):
        self.do_user_config(parent=self)
        # Apply the changes
        self.label.setText(prefs['hello_world_msg'])
```

Získání zdrojů ze souboru ZIP modulu

Systém načítání doplňků Calibre definuje pár vestavěných funkcí, které vám umožňují pohodlně získat soubory ze souboru ZIP modulu.

get_resources(name_or_list_of_names)

Tato funkce by měla být volána se seznamem cest k souborům uvnitř souboru ZIP. Například pro přístup k souboru `icon.png` ve složce images v souboru ZIP byste použili: `images/icon.png`. Jako oddělovač cest vždy používejte lomítko, a to i ve Windows. Pokud předáte jeden název, funkce vrátí nezpracované bajty tohoto souboru nebo None, pokud název nebyl v souboru ZIP nalezen. Pokud předáte více než jeden název, vrátí slovník mapující názvy na bajty. Pokud název nebyl nalezen, nebude ve vráceném slovníku přítomen.

get_icons(name_or_list_of_names, plugin_name=“)
Obal pro funkci get_resources(), který vytváří objekty QIcon ze surových bajtů vrácených funkcí get_resources. Pokud v souboru ZIP není nalezen název, bude odpovídající QIcon nulový. Aby bylo možné podporovat změnu ikon motivu, předejte lidsky přívětivý název vašeho modulu jako `nazev_modulu`. Pokud uživatel používá motiv ikon s ikonami vašeho modulu, budou načteny přednostně.

Povolení uživatelské konfigurace vašeho modulu

Abyste povolili uživatelům konfigurovat váš modul, musíte definovat ve své základní třídě modulu tři metody, **is_customizable**, **config_widget** a **save_settings** jak je uvedeno níže:

```
    def is_customizable(self):
        '''
        This method must return True to enable customization via
        Preferences->Plugins
        '''
        return True
```

```
    def config_widget(self):
        '''
        Implement this method and :meth:`save_settings` in your plugin to
        use a custom configuration dialog.

        This method, if implemented, must return a QWidget. The widget can have
        an optional method validate() that takes no arguments and is called
        immediately after the user clicks OK. Changes are applied if and only
        if the method returns True.

        If for some reason you cannot perform the configuration at this time,
        return a tuple of two strings (message, details), these will be
        displayed as a warning dialog to the user and the process will be
        aborted.

        The base class implementation of this method raises NotImplementedError
        so by default no user configuration is possible.
        '''
        # It is important to put this import statement here rather than at the
        # top of the module as importing the config class will also cause the
        # GUI libraries to be loaded, which we do not want when using calibre
        # from the command line
        from calibre_plugins.interface_demo.config import ConfigWidget
        return ConfigWidget()
```

```
    def save_settings(self, config_widget):
        '''
        Save the settings specified by the user with config_widget.

        :param config_widget: The widget returned by :meth:`config_widget`.
        '''
        config_widget.save_settings()

        # Apply the changes
```

(continues on next page)

(pokračujte na předchozí stránce)

```
        ac = self.actual_plugin_
        if ac is not None:
            ac.apply_settings()
```

Calibre má mnoho různých způsobů, jak ukládat konfigurační data (starší verze jeho dlouhé historie). Doporučený způsob je použít třídu **JSONConfig**, která ukládá vaše konfigurační informace v souboru .json.

Kód pro správu konfiguračních dat v ukázkovém modulu je v config.py:

```
from calibre.utils.config import JSONConfig

# This is where all preferences for this plugin will be stored
# Remember that this name (i.e. plugins/interface_demo) is also
# in a global namespace, so make it as unique as possible.
# You should always prefix your config file name with plugins/,
# so as to ensure you don't accidentally clobber a calibre config file
prefs = JSONConfig('plugins/interface_demo')

# Set defaults
prefs.defaults['hello_world_msg'] = 'Hello, World!'


class ConfigWidget(QWidget):

    def __init__(self):
        QWidget.__init__(self)
        self.l = QHBoxLayout()
        self.setLayout(self.l)

        self.label = QLabel('Hello world &message:')
        self.l.addWidget(self.label)

        self.msg = QLineEdit(self)
        self.msg.setText(prefs['hello_world_msg'])
        self.l.addWidget(self.msg)
        self.label.setBuddy(self.msg)

    def save_settings(self):
        prefs['hello_world_msg'] = self.msg.text()
```

Object `prefs` je nyní dostupný v celém kódu modulu jednoduchým:

```
from calibre_plugins.interface_demo.config import prefs
```

Můžete vidět objekt `prefs` použitý v main.py:

```
    def config(self):
        self.do_user_config(parent=self)
        # Apply the changes
        self.label.setText(prefs['hello_world_msg'])
```

10.5.3 Moduly pro úpravu knihy

Nyní pojďme na chvíli přeřadit a podívejme se na vytvoření modulu pro přidání nástrojů do editoru knih Calibre. Modul je dostupný zde: editor_demo_plugin.zip[105].

Prvním krokem, stejně jako pro všechny moduly, je vytvořit název importu prázdného souboru txt, jak je popsáno *výše* (stránka 226). Pojmenujeme soubor `plugin-import-name-editor_plugin_demo.txt`.

Nyní vytvoříme povinný soubor `__init__.py`, který obsahuje metadata o modulu – jeho název, autora, verzi atd.

```
class DemoPlugin(EditBookToolPlugin):

    name = 'Edit Book plugin demo'
    version = (1, 0, 0)
    author = 'Kovid Goyal'
    supported_platforms = ['windows', 'osx', 'linux']
    description = 'A demonstration of the plugin interface for the ebook editor'
    minimum_calibre_version = (1, 46, 0)
```

Jeden modul editoru může poskytnout více nástrojů, každý nástroj odpovídá jednomu tlačítku na panelu nástrojů a položce v nabídce :guilabel:Moduly` v editoru. Ty mohou mít podnabídky v případě, že nástroj má více souvisejících akcí.

Nástroje musí být všechny definovány v souboru `main.py` ve vašem modulu. Každý nástroj je třída, která dědí ze třídy *calibre.gui2.tweak_book.plugin.Tool* (stránka 368). Pojďme se podívat na `main.py` z ukázkového modulu, zdrojový kód je silně komentován a měl by být samovysvětlující. Pro více informací si přečtěte dokumenty API ke třídě *calibre.gui2.tweak_book.plugin.Tool* (stránka 368).

main.py

Zde uvidíme definici jednoho nástroje, který se bude násobit všechny velikosti písem v knize číslem zadaným uživatelem. Tento nástroj ukazuje různé důležité pojmy, které budete potřebovat při vývoji svých vlastních modulů, takže byste si měli pečlivě přečíst (silně komentovaný) zdrojový kód.

```
from css_parser.css import CSSRule
from qt.core import QAction, QInputDialog

from calibre import force_unicode
from calibre.ebooks.oeb.polish.container import OEB_DOCS, OEB_STYLES, serialize
from calibre.gui2 import error_dialog

# The base class that all tools must inherit from
from calibre.gui2.tweak_book.plugin import Tool


class DemoTool(Tool):

    #: Set this to a unique name it will be used as a key
    name = 'demo-tool'

    #: If True the user can choose to place this tool in the plugins toolbar
    allowed_in_toolbar = True
```

(continues on next page)

[105] https://calibre-ebook.com/downloads/editor_demo_plugin.zip

(pokračujte na předchozí stránce)

```
    #: If True the user can choose to place this tool in the plugins menu
    allowed_in_menu = True

    def create_action(self, for_toolbar=True):
        # Create an action, this will be added to the plugins toolbar and
        # the plugins menu
        ac = QAction(get_icons('images/icon.png'), 'Magnify fonts', self.gui)  #␣
↪noqa: F821
        if not for_toolbar:
            # Register a keyboard shortcut for this toolbar action. We only
            # register it for the action created for the menu, not the toolbar,
            # to avoid a double trigger
            self.register_shortcut(ac, 'magnify-fonts-tool', default_keys=(
↪'Ctrl+Shift+Alt+D',))
        ac.triggered.connect(self.ask_user)
        return ac

    def ask_user(self):
        # Ask the user for a factor by which to multiply all font sizes
        factor, ok = QInputDialog.getDouble(
            self.gui, 'Enter a magnification factor', 'Allow font sizes in the book␣
↪will be multiplied by the specified factor',
            value=2, min=0.1, max=4
        )
        if ok:
            # Ensure any in progress editing the user is doing is present in the␣
↪container
            self.boss.commit_all_editors_to_container()
            try:
                self.magnify_fonts(factor)
            except Exception:
                # Something bad happened report the error to the user
                import traceback
                error_dialog(self.gui, _('Failed to magnify fonts'), _(
                    'Failed to magnify fonts, click "Show details" for more info'),
                    det_msg=traceback.format_exc(), show=True)
                # Revert to the saved restore point
                self.boss.revert_requested(self.boss.global_undo.previous_container)
            else:
                # Show the user what changes we have made, allowing her to
                # revert them if necessary
                self.boss.show_current_diff()
                # Update the editor UI to take into account all the changes we
                # have made
                self.boss.apply_container_update_to_gui()

    def magnify_fonts(self, factor):
        # Magnify all font sizes defined in the book by the specified factor
        # First we create a restore point so that the user can undo all changes
        # we make.
        self.boss.add_savepoint('Before: Magnify fonts')
```

(continues on next page)

(pokračujte na předchozí stránce)

```
        container = self.current_container  # The book being edited as a container
↪object

        # Iterate over all style declarations in the book, this means css
        # stylesheets, <style> tags and style="" attributes
        for name, media_type in container.mime_map.items():
            if media_type in OEB_STYLES:
                # A stylesheet. Parsed stylesheets are css_parser CSSStylesheet
                # objects.
                self.magnify_stylesheet(container.parsed(name), factor)
                container.dirty(name)  # Tell the container that we have changed the
↪stylesheet
            elif media_type in OEB_DOCS:
                # A HTML file. Parsed HTML files are lxml elements

                for style_tag in container.parsed(name).xpath('//*[local-name="style"]
↪'):
                    if style_tag.text and style_tag.get('type', None) in {None, 'text/
↪css'}:
                        # We have an inline CSS <style> tag, parse it into a
                        # stylesheet object
                        sheet = container.parse_css(style_tag.text)
                        self.magnify_stylesheet(sheet, factor)
                        style_tag.text = serialize(sheet, 'text/css', pretty_
↪print=True)
                        container.dirty(name)  # Tell the container that we have
↪changed the stylesheet
                for elem in container.parsed(name).xpath('//*[@style]'):
                    # Process inline style attributes
                    block = container.parse_css(elem.get('style'), is_
↪declaration=True)
                    self.magnify_declaration(block, factor)
                    elem.set('style', force_unicode(block.getCssText(separator=' '),
↪'utf-8'))

    def magnify_stylesheet(self, sheet, factor):
        # Magnify all fonts in the specified stylesheet by the specified
        # factor.
        for rule in sheet.cssRules.rulesOfType(CSSRule.STYLE_RULE):
            self.magnify_declaration(rule.style, factor)

    def magnify_declaration(self, style, factor):
        # Magnify all fonts in the specified style declaration by the specified
        # factor
        val = style.getPropertyValue('font-size')
        if not val:
            return
        # see if the font-size contains a number
        num = re.search(r'[0-9.]+', val)
        if num is not None:
            num = num.group()
```

(continues on next page)

(pokračujte na předchozí stránce)

```
            val = val.replace(num, f'{float(num)*factor:f}')
            style.setProperty('font-size', val)
        # We should also be dealing with the font shorthand property and
        # font sizes specified as non numbers, but those are left as exercises
        # for the reader
```

Pojďme si rozebrat `main.py`. Vidíme, že definuje jeden nástroj s názvem *Zvětšit písma*. Tento nástroj požádá uživatele o číslo a vynásobí všechny velikosti písem v knize tímto číslem.

První důležitá věc je název nástroje, který musíte nastavit na nějaký relativně jedinečný řetězec, protože bude použit jako klíč pro tento nástroj.

Dalším důležitým vstupním bodem je *calibre.gui2.tweak_book.plugin.Tool.create_action()* (stránka 369). Tato metoda vytvoří objekty QAction, které se zobrazí v panelu nástrojů modulů a nabídce modulu. Také to volitelně přiřadí klávesovou zkratku, kterou si uživatel může přizpůsobit. Spuštěný signál z QAction je spojený s metodou ask_user(), která se zeptá uživatele na násobitel velikosti písma, a pak spustí kód zvětšení.

Kód zvětšení kód je dobře okomentován a poměrně jednoduchý. Hlavními věcmi, na které pamatujte, je to, že jste získali odkaz na okno editoru jako `self.gui` a *Boss* editoru jako `self.boss`. *Boss* je objekt, který řídí uživatelské rozhraní editoru. Má mnoho užitečných metod, které jsou popsány ve třídě *calibre.gui2.tweak_book.boss.Boss* (stránka 370).

Nakonec je zde `self.current_container`, což je odkaz na knihu upravovanou jako objekt *calibre.ebooks.oeb.polish.container.Container* (stránka 362). Znázorňuje knihu jako kolekci jejích základních souborů HTML, CSS nebo obrázků a má pohodlné metody pro provádění mnoha užitečných věcí. Objekt kontejneru a různé užitečné funkce nástrojů, které mohou být znovu použity v kódu vašeho modulu, jsou popsány v *Dokumentace API pro nástroje pro úpravu e-knih* (stránka 361).

10.5.4 Přidávání překladů do vašeho modulu

Všechny řetězce uživatelského rozhraní ve svém modulu můžete mít přeloženy a zobrazeny v jakémkoliv jazyce, který je nastaven pro hlavní uživatelské rozhraní Calibre.

Prvním krokem je projít zdrojový kód svého modulu a označit všechny řetězce viditelné uživatelem jako přeložitelné jejich obklopením _(). Například:

```
action_spec = (_('My plugin'), None, _('My plugin is cool'), None)
```

Pak použijte nějaký program pro vygenerování souborů .po ze zdrojového kódu vašeho doplňku. Měl by být jeden soubor .po pro každý jazyk, do kterého chcete překládat. Například: cs.po pro češtinu, de.po pro němčinu, fr.po pro francouzštinu a tak dále. K tomu můžete použít program Poedit[106].

Tyto soubory .po pošlete svým překladatelům. Jakmile je dostanete zpátky, zkompilovat je do souborů .mo. K tomu můžete znovu použít Poedit, nebo prostě udělejte:

```
calibre-debug -c "from calibre.translations.msgfmt import main; main()" filename.po
```

Soubory .mo umístěte do složky `translations` ve svém modulu.

Posledním krokem je jednoduše zavolat funkci *load_translations()* na začátku souborů .py vašeho modulu. Z výkonnostních důvodů byste měli volat tuto funkci pouze v těch souborech .py, které skutečně mají přeložitelné řetězce. Takže v typickém modulu uživatelského rozhraní byste ji volali na začátku `ui.py` ale ne `__init__.py`.

Překlady svých modulů můžete otestovat změnou jazyka uživatelského rozhraní v Calibre pod *Předvolby → Rozhraní → Vzhled a chování* nebo spuštěním Calibre s nastavenou proměnnou prostředí `CALIBRE_OVERRIDE_LANG`. Například:

[106] https://poedit.net/

```
CALIBRE_OVERRIDE_LANG=de
```

Nahraďte `de` kódem jazyka, který chcete otestovat.

Pro překlady s množnými čísly použijte funkci `ngettext()` namísto `_()`. Například:

```
ngettext('Delete a book', 'Delete {} books', num_books).format(num_books)
```

10.5.5 Rozhraní API modulu

Jak jste si možná všimli výše, modul Calibre je třída. V Calibre existují různé třídy pro různé typy modulů. Podrobnosti o každé třídě, včetně základní třídy všech modulů, lze nalézt v *moduly* (stránka 256).

Váš modul téměř určitě bude používat kód z Calibre. Abyste zjistili, jak najít různé kousky funkcí v základním kódu Calibre, přečtěte si část o *Rozložení kódu* (stránka 344) Calibre.

10.5.6 Ladění modulů

Prvním a nejdůležitějším krokem je spuštění Calibre v režimu ladění. To můžete udělat z příkazového řádku pomocí:

```
calibre-debug -g
```

Nebo z Calibre kliknutím pravým tlačítkem myši na tlačítko *Předvolby* nebo pomocí klávesové zkratky `Ctrl+Shift+R`.

Při spuštění z příkazového řádku, bude výstup ladění vypsán do konzole, při spuštění z Calibre, půjde výstup do textového souboru.

Do kódu svého modulu můžete kamkoliv vložit příkazy tisku, budou mít výstup v režimu ladění. Nezapomeňte, že je to Python, neměli byste opravdu potřebovat nic víc než příkazy tisku pro ladění ;) Celé Celibre jsem vytvořil pouze pomocí této techniky ladění.

Změny ve svém modulu můžete rychle testovat pomocí následujícího příkazového řádku:

```
calibre-debug -s; calibre-customize -b /path/to/your/plugin/folder; calibre
```

Toto vypne spuštěné Calibre, počkejte až bude vypnutí dokončeno, a pak aktualizujte svůj modul v Calibre a znovu spusťte Calibre.

10.5.7 Další příklady modulů

Seznam mnoha propracovaných modulů Calibre můžete nalézt zde[107].

10.5.8 Sdílení vašeho modulu s ostatními

Pokud byste chtěli sdílet moduly, které jste vytvořili, s ostatními uživateli Calibre, zveřejněte svůj modul v novém vláknu ve fóru modulů Calibre[108].

10.6 Sazba matematiky v e-knihách

Prohlížeč e-knih Calibre má schopnost zobrazit matematiku vloženou do e-knih (soubory EPUB a HTML). Můžete sázet matematiku přímo pomocí TeX, MathML nebo AsciiMath. Prohlížeč e-knih Calibre k tomuto používá vynikající knihovnu MathJax[109]. Toto je stručný kurz vytváření e-knih s matematikou, které fungují dobře s Prohlížečem e-knih Calibre.

[107] https://www.mobileread.com/forums/showthread.php?t=118764
[108] https://www.mobileread.com/forums/forumdisplay.php?f=237
[109] https://www.mathjax.org

10.6.1 Zjednodušený soubor HTML s matematikou

Můžete zapsat matematiku vloženou uvnitř souboru zjednodušeného HTML a Prohlížeč e-knih Calibre ji vykreslí se správnou sazbou matematiky. V příkladu níže používáme pro matematiku zápis TeX. Uvidíte, že můžete použít normální příkazy TeX, s malým upozorněním, že ampersandy a znaky menší než a větší než musí být zapsány jako & < a > v uvedeném pořadí.

První krok je říct Calibre, že toto bude obsahovat matematiku. To uděláte přidáním následujícího fragmentu kódu do oddílu <head> souboru HTML:

```
<script type="text/x-mathjax-config"></script>
```

To je vše, nyní můžete psát matematiku, jak byste to udělali v souboru .tex. Například zde jsou Lorentzovy rovnice:

```
<h2>The Lorenz Equations</h2>

<p>
\begin{align}
\dot{x} &amp;= \sigma(y-x) \\
\dot{y} &amp;= \rho x - y - xz \\
\dot{z} &amp;= -\beta z + xy
\end{align}
</p>
```

Tento fragment vypadá jako následující snímek obrazovky v Prohlížeči e-knih Calibre.

$$\dot{x} = \sigma(y - x)$$
$$\dot{y} = \rho x - y - xz$$
$$\dot{z} = -\beta z + xy$$

Obr. 1: *Lorentzovy rovnice*

Kompletní soubor HTML s více rovnicemi a vloženou matematikou je reprodukován níže. Tento soubor HTML můžete převést do EPUB v Calibre, abyste získali e-knihu, kterou můžete snadno distribuovat ostatním lidem.

```
<!DOCTYPE html>
<html>
<!-- Copyright (c) 2012 Design Science, Inc. -->
<head>
<title>Math Test Page</title>
<meta http-equiv="content-type" content="text/html; charset=UTF-8" />

<!-- This script tag is needed to make calibre's ebook-viewer recpgnize that this␣
↪file needs math typesetting -->
<script type="text/x-mathjax-config">
    // This line adds numbers to all equations automatically, unless explicitly␣
↪suppressed.
    MathJax.tex = {tags: 'all'};
</script>
```

(continues on next page)

(pokračujte na předchozí stránce)

```
<style>
h1 {text-align:center}
h2 {
  font-weight: bold;
  background-color: #DDDDDD;
  padding: .2em .5em;
  margin-top: 1.5em;
  border-top: 3px solid #666666;
  border-bottom: 2px solid #999999;
}
</style>
</head>
<body>

<h1>Sample Equations</h1>

<h2>The Lorenz Equations</h2>

<p>
\begin{align}
\dot{x} &amp; = \sigma(y-x) \label{lorenz}\\
\dot{y} &amp; = \rho x - y - xz \\
\dot{z} &amp; = -\beta z + xy
\end{align}
</p>

<h2>The Cauchy-Schwarz Inequality</h2>

<p>\[
\left( \sum_{k=1}^n a_k b_k \right)^{\!\!2} \leq
 \left( \sum_{k=1}^n a_k^2 \right) \left( \sum_{k=1}^n b_k^2 \right)
\]</p>

<h2>A Cross Product Formula</h2>

<p>\[
  \mathbf{V}_1 \times \mathbf{V}_2 =
   \begin{vmatrix}
    \mathbf{i} &amp; \mathbf{j} &amp; \mathbf{k} \\
    \frac{\partial X}{\partial u} &amp; \frac{\partial Y}{\partial u} &amp; 0 \\
    \frac{\partial X}{\partial v} &amp; \frac{\partial Y}{\partial v} &amp; 0 \\
   \end{vmatrix}
\]</p>

<h2>The probability of getting \(k\) heads when flipping \(n\) coins is:</h2>

<p>\[P(E) = {n \choose k} p^k (1-p)^{ n-k} \]</p>

<h2>An Identity of Ramanujan</h2>

<p>\[
```

(continues on next page)

(pokračujte na předchozí stránce)

```
   \frac{1}{(\sqrt{\phi \sqrt{5}}-\phi) e^{\frac25 \pi}} =
     1+\frac{e^{-2\pi}} {1+\frac{e^{-4\pi}} {1+\frac{e^{-6\pi}}
      {1+\frac{e^{-8\pi}} {1+\ldots} } } }
\]</p>

<h2>A Rogers-Ramanujan Identity</h2>

<p>\[
  1 +  \frac{q^2}{(1-q)}+\frac{q^6}{(1-q)(1-q^2)}+\cdots =
    \prod_{j=0}^{\infty}\frac{1}{(1-q^{5j+2})(1-q^{5j+3})},
     \quad\quad \text{for $|q|&lt;1$}.
\]</p>

<h2>Maxwell's Equations</h2>

<p>
\begin{align}
  \nabla \times \vec{\mathbf{B}} -\, \frac1c\, \frac{\partial\vec{\mathbf{E}}}{\
↪partial t} &amp; = \frac{4\pi}{c}\vec{\mathbf{j}} \\
  \nabla \cdot \vec{\mathbf{E}} &amp; = 4 \pi \rho \\
  \nabla \times \vec{\mathbf{E}}\, +\, \frac1c\, \frac{\partial\vec{\mathbf{B}}}{\
↪partial t} &amp; = \vec{\mathbf{0}} \\
  \nabla \cdot \vec{\mathbf{B}} &amp; = 0
\end{align}
</p>

<h2>Inline Mathematics</h2>

<p>While display equations look good for a page of samples, the
ability to mix math and text in a paragraph is also important.  This
expression \(\sqrt{3x-1}+(1+x)^2\) is an example of an inline equation.  As
you see, equations can be used this way as well, without unduly
disturbing the spacing between lines.</p>

<h2>References to equations</h2>

<p>Here is a reference to the Lorenz Equations (\ref{lorenz}). Clicking on the␣
↪equation number will take you back to the equation.</p>

</body>
</html>
```

Poznámka

The calibre E-book viewer supports MathML as well as TeX, but you must include the `<script type="text/x-mathjax-config"></script>` line in your HTML file otherwise the MathML will not render.

10.6.2 Více informací

Protože Prohlížeč e-knih Calibre používá pro vykreslení matematiky knihovnu MathJax, nejlepší místo pro zjištění více o matematice v e-knihách a získání nápovědy je Webová stránka MathJax[110].

10.7 Vytváření katalogů AZW3 • EPUB • MOBI

Funkce calibre Vytvořit katalog vám umožňuje vytvořit z vaší knihovny katalog v různých formátech. Tento soubor nápovědy popisuje volby zařazení do katalogu při generování katalogu ve formátech AZW3, EPUB a MOBI.

- *Výběr knih do katalogu* (stránka 243)
- *Zahrnuté oddíly* (stránka 244)
- *Předpony* (stránka 244)
- *Vyloučené knihy* (stránka 245)
- *Vyloučené žánry* (stránka 245)
- *Další volby* (stránka 246)
- *Vlastní obálky katalogu* (stránka 247)
- *Dodatečné zdroje nápovědy* (stránka 247)

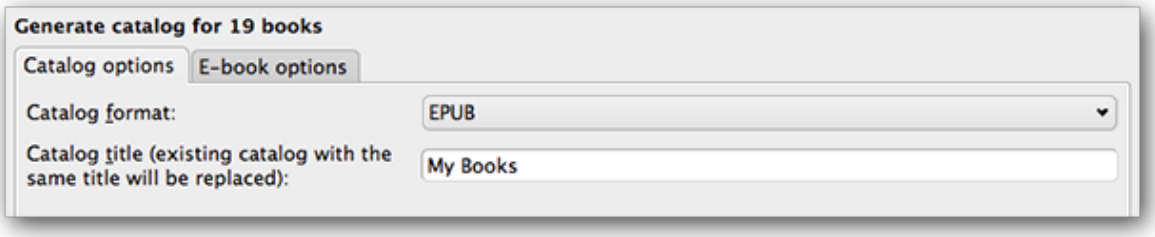
10.7.1 Výběr knih do katalogu

Pokud chcete katalogizovat *celou* svou knihovnu, odeberte v hlavním okně jakákoliv kritéria hledání nebo filtrování. Pokud máte vybránu jednu knihu, všechny knihy ve vaší knihovně budou kandidáti na zahrnutí do generovaného katalogu. Jednotlivé knihy mohou být vyloučeny různými kritérii, pro více informací se podívejte na oddíl *Vyloučené žánry* (stránka 245) níže.

Pokud chcete katalogizovat pouze *něco* z vaší knihovny, máte dvě volby:

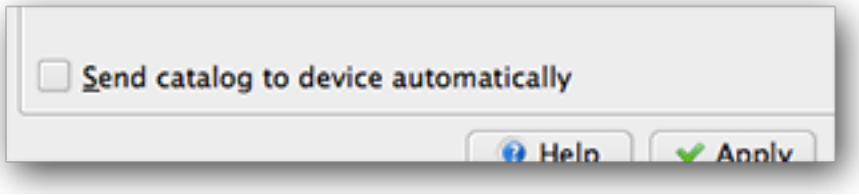
- Vytvořte výběr více knih, které chcete katalogizovat. Pokud je v hlavním okně calibre vybrána více než jedna kniha, budou katalogizovány pouze vybrané knihy.
- Použijte pole Hledat nebo Prohlížeč štítků pro filtrování zobrazených knih. Budou katalogizovány pouze vybrané knihy.

Pro zahájení generování katalogu vyberte položku nabídky *Převést knihy* > *Vytvořit katalog knih ve vaší knihovně calibre*. Můžete také přidat tlačítko *Vytvořit katalog* na panel nástrojů v *Předvolby* > *Rozhraní* > *Panely nástrojů a nabídky* pro snadnější přístup k dialogovému oknu Vytvořit katalog.



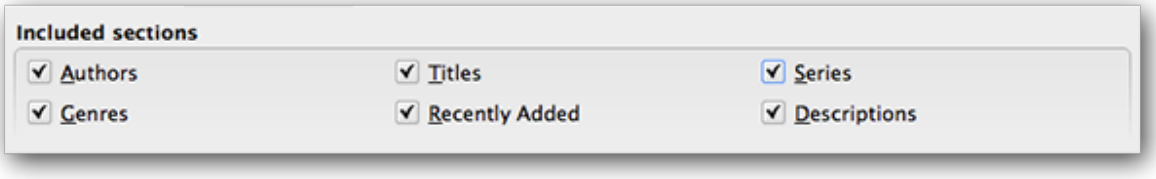
[110] https://www.mathjax.org

Ve *Volby katalogu* vyberte jako formát katalogu **AZW3, EPUB nebo MOBI**. V poli *Název katalogu* zadejte název, který bude použit pro vygenerovaný katalog. Pokud již existuje katalog se stejným názvem a formátem, bude nahrazen nově vytvořeným katalogem.



Povolení *Odeslat katalog do zařízení automaticky* stáhne po dokončení vygenerovaný katalog do připojeného zařízení.

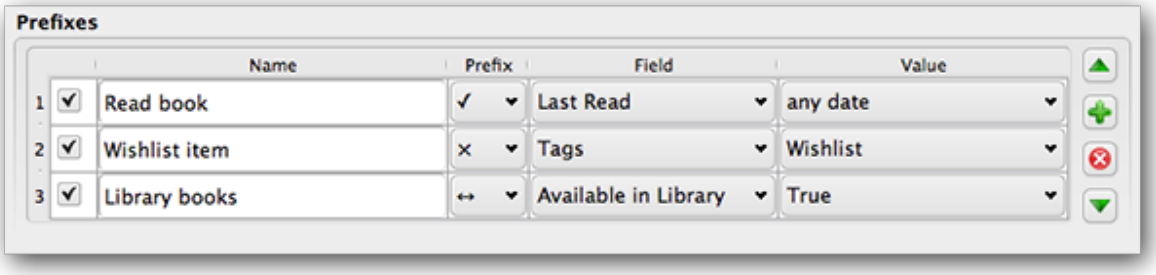
10.7.2 Zahrnuté oddíly



Oddíly povolené zatržítkem budou zahrnuty do generovaného katalogu:

- *Autoři* – všechny knihy seřazené podle autora zobrazené ve formátu seznamu. Knihy bez sérií jsou vypsány před knihami v sérii.
- *Názvy* – všechny knihy seřazené podle názvu zobrazené ve formátu seznamu.
- *Série* – všechny knihy, které jsou součástí série, seřazené podle názvu zobrazené ve formátu seznamu.
- *Žánry* – jednotlivé žánry zobrazené v seznamu, seřazené podle autora a série.
- *Nedávno přidané* – všechny knihy seřazené v obráceném chronologickém pořadí. Seznam obsahuje knihy přidané za posledních 30 dní, pak výpis přidaných knih po měsících.
- *Popisy* – podrobná stránka popisu pro každou knihu včetně náhledu obálky a komentářů. Seřazeno podle autora s knihami bez sérií vypsanými před knihami v sérii.

10.7.3 Předpony



Pravidla předpony vám umožňují přidat předponu do výpisů knih, když jsou splněna určitá kritéria. Například možná chcete označit přečtené knihy zatržítkem, nebo knihy na seznamu přání označit X.

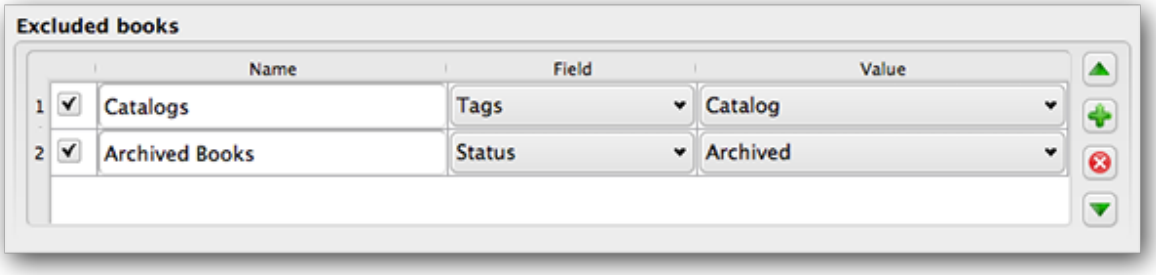
Zatržítko v prvním sloupci povolí pravidlo. *Název* je název pravidla, který zadáte. *Pole* je buď *Štítky*, nebo vlastní sloupec z vaší knihovny. *Hodnota* je obsah *Pole*, který porovnat. Když je pravidlo předpony splněno, kniha bude označena vybranou *Předponou*.

Vy výše uvedeném příkladu byla zadána tři pravidla předpony:

1. *Přečtená kniha* určuje, že kniha s jakýmkoliv datem ve vlastním sloupci pojmenovaném *Přečteno* bude označena předponou se symbolem zatržítka.
2. *Položka seznamu přání* určuje, že jakákoliv kniha se štítkem *Seznam přání* bude označena předponou se symbolem X.
3. *Knihy v knihovně* určuje, že jakákoliv kniha s hodnotou Pravda (nebo Ano) ve vlastním sloupci *Dostupná v knihovně* bude označena předponou se symbolem obousměrné šipky.

První odpovídající pravidlo určí předponu. Zakázaná nebo neúplná pravidla jsou ignorována.

10.7.4 Vyloučené knihy



Pravidla vyloučení vám umožňují zadat knihy, které nebudou katalogizovány.

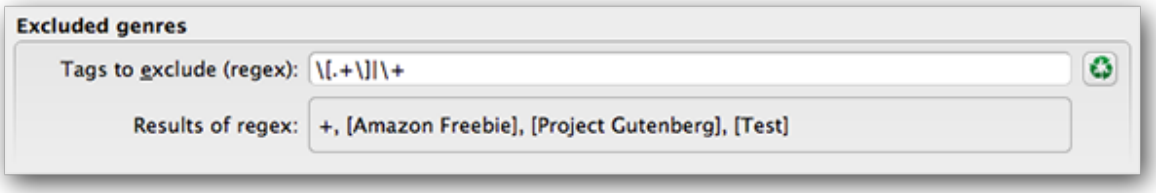
Zatržítko v prvním sloupci povolí pravidlo. *Název* je název pravidla, který zadáte. *Pole* je buď *Štítky*, nebo vlastní sloupec z vaší knihovny. *Hodnota* je obsah *Pole*, který porovnat. Když je pravidlo předpony splněno, kniha bude vyloučena z generovaného katalogu.

Vy výše uvedeném příkladu byla zadána dvě pravidla vyloučení:

1. Pravidlo *Katalogy* určuje, že jakákoliv kniha se štítkem *Katalog* bude vyloučena z generovaného katalogu.
2. Pravidlo *Archivované knihy* určuje, že jakákoliv kniha s hodnotou *Archivováno* ve vlastním sloupci guilabel:*Stav* bude vyloučena z generovaného katalogu.

Jsou vyhodnocována všechna pravidla pro každou knihu. Zakázaná nebo neúplná pravidla jsou ignorována.

10.7.5 Vyloučené žánry



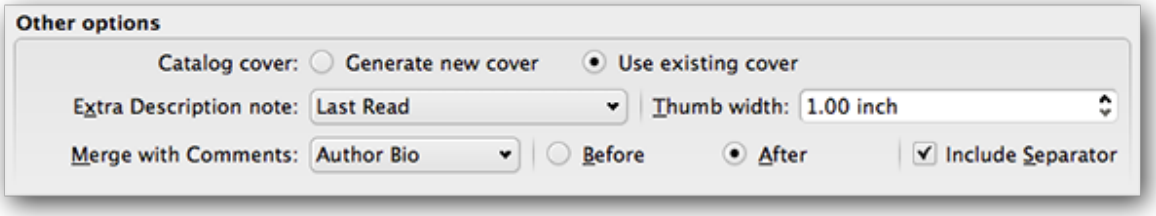
Když je generován katalog, jsou štítky ve vaší databázi použity jako žánry. Například můžete použít štítky `Román` a `Literatura faktu`. Tyto štítky se stanou žánry ve vygenerovaném katalogu s knihami vypsanými pod jejich příslušnými seznamy žánrů na základě svých přiřazených štítků.

Možná používáte určité štítky pro jiné účely, možná + označení přečtené knihy, nebo štítek v hranatých závorkách, jako je `[Amazon Freebie]`, pro označení zdroje knihy. Regulární výraz *Vyloučené žánry* vám umožňuje zadat štítky, které nechcete použít jako žánry v generovaném katalogu. Výchozí regulární výraz vyloučení `\[.+\]\+` vyloučí jakékoliv štítky ve formě `[štítek]`, stejně jako vyloučí `+`, výchozí štítek pro přečtené knihy, aby nebyly použity jako žánry v generovaném katalogu.

V regulárním výrazu můžete také použít přesný název štítku. Například `[Amazon Freebie]` nebo `[Projekt Gutenberg]`. Pokud chcete vypsat více přesných štítků k vyloučení, vložte mezi ně znak svislé čáry: `[Amazon Freebie]|[Projekt Gutenberg]`.

Výsledky regulárního výrazu zobrazí, které štítky budou vyloučeny při sestavování katalogu na základě štítků ve vaší databázi a regulárního výrazu, který jste zadali. Výsledky jsou aktualizovány, jak měníte regulární výraz.

10.7.6 Další volby



Obálka katalogu určuje, zda generovat novou obálku nebo použít existující obálku. Je možné vytvořit vlastní obálku pro vaše katalogy – pro více informací se podívejte na *Vlastní obálky katalogu* (stránka 247). Pokud jste vytvořili vlastní obálku, kterou chcete znovu použít, vyberte *Použít existující obálku*. Jinak vyberte *Generovat novou obálku*.

Dodatečná poznámka popisu určuje obsah vlastního sloupce, který vložit do stránky Popis vedle náhledu obálky. Například možná chcete zobrazit datum, kdy jste naposledy četli knihu, pomocí vlastního slupce *Naposledy čteno*. Pro pokročilé používání funkce poznámky Popisu se podívejte na tento příspěvek ve fóru calibre <https://www.mobileread.com/forums/showpost.php?p=1335767&postcount=395>`_.

Šířka náhledu určuje předvolbu šířky pro náhledy obálek obsažené na stránkách Popis. Náhledy jsou ukládány do mezipaměti pro zlepšení výkonu. Pro experimentování s různými šířkami zkuste generovat katalog jen s pár knihami, doku neurčíte svou upřednostňovanou šířku, pak vygenerujte úplný katalog. Když je katalog generován poprvé s novou šířkou náhledu, výkon bude nižší, ale následující sestavení katalogu využijí náhledy v mezipaměti.

Sloučit s komentáři určuje vlastní sloupec, jehož obsah bude nedestruktivně sloučen s metadaty komentářů během generování katalogu. Například můžete mít vlastní sloupec *Biografie autora*, který byste chtěli připojit k metadatům komentářů. Můžete zvolit vložit obsah vlastního sloupce *před nebo za* oddíl komentářů a volitelně oddělit připojený obsah vodorovnou čárou. Vhodné typy vlastních sloupců zahrnují `text, komentáře a složený`.

10.7.7 Vlastní obálky katalogu

S nainstalovaným modulem Generovat obálku[111] můžete vytvořit vlastní obálky pro své katalogy. Pro nainstalování modulu přejděte na *Předvolby* > *Pokročilé* > *Moduly* > *Získat nové moduly*.

10.7.8 Dodatečné zdroje nápovědy

Pro více informací o funkci Katalog calibre navštivte fórum MobileRead Creating Catalogs: Start here[112], kde můžete najít informace o tom, jak přizpůsobit šablony katalogu, a jak poslat chybové hlášení.

Pokud máte otázky nebo chcete diskutovat o funkci Katalog Calibre s ostatními uživateli, navštivte fórum MobileRead Library Management[113].

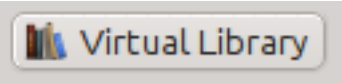
10.8 Virtuální knihovny

Virtuální knihovna je způsob, jak říct Calibre, aby otevřelo pouze podmnožinu normální knihovny. Můžete například chtít pracovat pouze s knihami od určitých autorů nebo s knihami, které mají pouze určitý štítek. Používání Virtuálních knihoven je upřednostňovaným způsobem rozdělení velké kolekce knih do menších podkolekcí. Je lepší rozdělit knihovnu do více menších knihoven, protože pokud chcete prohledávat celou kolekci, můžete se jednoduše vrátit zpět do plné knihovny. Neexistuje způsob, jak v Calibre prohledávat více samostatných knihoven současně.

Virtuální knihovna se liší od jednoduchého vyhledávání. Hledání je omezeno pouze na seznam knih zobrazených v seznamu knih. Virtuální knihovna dělá to samé a navíc také omezuje položky zobrazené v *Prohlížeči štítků* vlevo. Prohlížeč štítků zobrazí pouze štítky, autory, série, vydavatele atd., kteří pocházejí z knih ve Virtuální knihovně. Virtuální knihovna se tak chová, jako kdyby skutečná knihovna obsahovala pouze omezenou sadu knih.

[111] https://www.mobileread.com/forums/showthread.php?t=124219
[112] https://www.mobileread.com/forums/showthread.php?t=118556
[113] https://www.mobileread.com/forums/forumdisplay.php?f=236

10.8.1 Vytváření Virtuálních knihoven



Pro použití Virtuální knihovny klikněte na tlačítko *Virtuální knihovna* umístěné na levé straně panelu Hledat a vyberte možnost *Vytvořit Virtuální knihovnu*. Jako první příklad vytvořme Virtuální knihovnu, která nám zobrazí pouze knihy od konkrétního autora. Klikněte na odkaz *Autoři*, jak je znázorněno na obrázku níže, zvolte autora, kterého chcete použít, a klikněte na OK.

Dialogové okno Vytvořit Virtuální knihovnu bylo pro vás vyplněno. Klikněte na OK a uvidíte, že byla vytvořena nová Virtuální knihovna a automaticky na ni bylo přepnuto, takže ukazuje pouze knihy od vybraného autora. Pokud jde o Calibre, je to, jako kdyby vaše knihovna obsahovala pouze knihy od vybraného autora.

Na plnou knihovnu můžete kdykoliv přepnout zpět opětovným kliknutím na tlačítko *Virtuální knihovna* a výběrem položky pojmenované *<Žádná>*.

Virtuální knihovny jsou založeny na *hledání*. Jako základ Virtuální knihovny můžete použít jakékoliv hledání. Virtuální knihovna bude obsahovat pouze knihy odpovídající tomuto hledání. Do Panelu hledání nejdříve vepište hledání, které chcete použít, nebo sestavte hledání pomocí *Prohlížeče štítků*. Až budete spokojení s vrácenými výsledky, klikněte na tlačítko *Virtuální knihovna*, zvolte *Vytvořit knihovnu* a zadejte název pro novou Virtuální knihovnu. Virtuální knihovna pak bude vytvořena na základě hledání, které jste právě zadali. Hledání jsou velice výkonná, pro příklady druhů věcí, které s nimi můžete dělat, se podívejte na *Rozhraní hledání* (stránka 12).

Příklady užitečných Virtuálních knihoven

- **Knihy přidané do Calibre za poslední den:**
  date:>1daysago
- **Knihy přidané do Calibre za poslední měsíc:**
  date:>30daysago
- **Knihy s hodnocením 5 hvězdiček:**
  rating:5
- **Knihy s hodnocením alespoň 4 hvězdičky:**
  rating:>=4
- **Knihy bez hodnocení:**
  rating:false

- **Periodika stahovaná funkcí Načíst zprávy v Calibre:**
  tags:=Zprávy and author:=Calibre
- **Knihy bez štítků:**
  tags:false
- **Knihy bez obálek:**
  cover:false

10.8.2 Práce s Virtuálními knihovnami

Upravit nebo odebrat dříve vytvořenou Virtuální knihovnu můžete kliknutím na *Virtuální knihovna* a zvolením příslušné akce.

Můžete říct Calibre, že při otevření aktuální knihovny chcete vždy použít určitou Virtuální knihovnu, a to na adrese *Předvolby → Rozhraní → Chování*.

Aktuální hledání můžete rychle použít jako dočasnou Virtuální knihovnu kliknutím na tlačítko *Virtuální knihovna* a zvolením položky **aktuální hledání*.

Všechny Virtuální knihovny můžete zobrazit jako karty nad seznamem knih. To je obzvlášť šikovné, pokud chcete velice často přepínat mezi Virtuálními knihovnami. Klikněte na tlačítko *Virtuální knihovna* a vyberte *Zobrazit Virtuální knihovny jako karty*. Karty můžete přeuspořádat přetažením a zavřením těch, které nechcete vidět. Zavřené karty mohou být obnoveny kliknutím pravým tlačítkem myši na panel karet.

10.8.3 Používání Virtuálních knihoven v hledáních

Knihy, které jsou ve Virtuální knihovně, můžete hledat pomocí předpony `vl:`. Například `vl:Přečíst` najde všechny knihy ve Virtuální knihovně *Přečíst*. Hledání `vl:Přečíst a `vl:"Sci-Fi"` najdete všechny knihy, které jsou v obou Virtuálních knihovnách *Přečíst* a * Sci-Fi*.

Hodnota následující `vl:` musí být název Virtuální knihovny. Pokud název virtuální knihovny obsahuje mezery, uzavřete ho uvozovkami.

Jedno použití pro hledání ve Virtuální knihovně je v Serveru s obsahem. V *Předvolby → Sdílení po síti>Vyžadovat uživatelské jméno a heslo* můžete omezit knihovny Calibre viditelné uživateli. Pro každou viditelnou knihovnu můžete zadat vyhledávací výraz pro další omezení, které knihy jsou vidět. Použijte `vl:"Název virtuální knihovny"` pro omezení knih na ty, které jsou ve Virtuální knihovně.

10.8.4 Používání dodatečných omezení

Knihy zobrazené ve virtuální knihovně můžete dále omezit pomocí *Dodatečných omezení*. Dodatečné omezení je vámi dříve vytvořené uložené hledání, které může být použito na aktuální virtuální knihovnu pro další omezení zobrazených knih ve Virtuální knihovně. Například řekněme, že máte virtuální knihovnu pro knihy se štítkem *Historický román* a uložené hledání, které vám ukazuje nepřečtené knihy. Můžete kliknout na tlačítko *Virtuální knihovna* a zvolit volbu *Dodatečná omezení* pro zobrazení pouze nepřečtených historických románů. Abyste se naučili o uložených hledáních, podívejte se na *Ukládání hledání* (stránka 17).

KAPITOLA 11

calibre:// URL schéma

calibre se registruje jako handler program pro calibre:// URL. Takže je můžete použít pro vykonávaní akcí jako jsou otevírání knih, vyhládávání knih, atd. z jiných programů/dokumentů nebo přes příkazový řádek. Například spuštěním následujícího příkazu:

```
calibre calibre://switch-library/Some_Library
```

Otevře se calibre s knihovnou pojmenovanou „Some Library“. Názvy knihoven jsou názvy adresářů složky knihovny, kde jsou mezery nahrazeny podtržítky. Speciální hodnota „_“ znamená současnou knihovnu. Níže jsou zdokumentovány rozličné typy URL.

Dokonce můžete vložit odkazy do HTML souborů nebo Word dokumentů a operační systém automaticky otevře calibre pro provedení specifikované akce.

- *Přepnout na konkrétní knihovnu* (stránka 251)
- *Zobrazit určitou knihu v calibre* (stránka 252)
- *Otevřít konkrétní knihu v Prohlížeči e-knih na konkrétní pozici* (stránka 252)
- *Hledání knih* (stránka 252)
- *Otevřít okno s podrobnostmi o knize na knize v některé knihovně* (stránka 253)
- *Otevřít poznámky přidružené k autorovi, sérii atd.* (stránka 253)
- *Hex kódování URL parametrů* (stránka 253)

11.1 Přepnout na konkrétní knihovnu

Syntaxe URL je:

```
calibre://switch-library/Library_Name
```

Názvy knihoven jsou názvy adresářů složky knihovny, kde jsou mezery nahrazeny podtržítky. Speciální hodnota „_“ znamená současnou knihovnu. Pro názvy knihoven můžete rovněž použít :ref:‘hex enconding<hex_encoding>‘, to je užitečné v případě, že názvy knihoven obsahují speciální znaky, který by jinak vyžadovaly URL kódování. Hex kódované názvy vypadají takto:

```
_hex_-AD23F4BC
```

Kde část za „_hex_“ předponou je název knihovny zakódovaný v UTF-8 a každý byte jej reprezentován dvěma hexadecimálními znaky.

11.2 Zobrazit určitou knihu v calibre

Syntaxe URL je:

```
calibre://show-book/Library_Name/book_id
```

Toto v Calibre zobrazí knihu s `book_id` (číslo). Tyto identifikátory knih lze v rozhraní Calibre zobrazit přejetím myši nad odkazem *Kliknout pro otevření* na panelu *Podrobnosti o knize*, je to to číslo v závorce na konci cesty ke složce knihy.

Odkaz na aktuální knihu zobrazenou v Calibre můžete zkopírovat kliknutím pravým tlačítkem myši na panel *Podrobnosti o knize* a výběrem *Zkopírovat odkaz na knihu*.

Pokud je vyhledávání aktivní a kniha neodpovídá vyhledávání, pak se vyhledávání vymaže.

Pokud je vybrána virtuální knihovna, Calibre ji použije při zobrazení knihy. Pokud kniha není nalezena v této virtuální knihovně, je virtuální knihovna vymazána.

Pokud chcete přepnout na určitou Virtuální knihovnu, když zobrazuje knihu, použijte:

```
calibre://show-book/Library_Name/book_id?virtual_library=Library%20Name
or
calibre://show-book/Library_Name/book_id?encoded_virtual_library=hex_encoded_virtual_
↪library_name
```

nahrazení mezer v názvu virtuální knihovny znakem `%20`. Pokud kniha není nalezena v této virtuální knihovně, pak je virtuální knihovna ignorována.

11.3 Otevřít konkrétní knihu v Prohlížeči e-knih na konkrétní pozici

Syntaxe URL je:

```
calibre://view-book/Library_Name/book_id/book_format?open_at=location
```

Tady je „book_format“ formát knihy, například „EPUB“ nebo „MOBI“ a „location“ je volitelná pozice v knize. Nejednodušší způsob získání takových odkazů je otevřít knihu v prohlížeči, pak přes ovládání prohlížeče vyberte :guilabel:‘Přejít na->Umístění‘, kde se tento odkaz pro kopírování/vložení jinam zobrazí.

11.4 Hledání knih

Syntaxe URL je:

```
calibre://search/Library_Name?q=query
calibre://search/Library_Name?eq=hex_encoded_query
```

Zde je dotaz jakýkoliv platný *vyhledávací výraz* (stránka 12). Pokud je vyledávací výraz komplikovaný, *zakódujte jej jako hexadecimální řetězec* (stránka 253) a namísto toho použijte `eq`. Vynechání dotazu způsobí vymazání aktuálního vyhledávání.

Ve výchozím nastavení, pokud je vybrána Virtuální knihovna, Calibre ji před provedením vyhledávání vymaže, aby se zajistilo, že budou nalezeny všechny knihy. Pokud chcete zachovat Virtuální knihovnu, použijte:

```
calibre://search/Library_Name?q=query&virtual_library=_
```

Pokud chcete přepnout na určitou Virtuální knihovnu, použijte:

```
calibre://search/Library_Name?virtual_library=Library%20Name
or
calibre://search/Library_Name?encoded_virtual_library=hex_encoded_virtual_library_name
```

nahrazování mezer v názvu Virtuální knihovně hodnotou `%20`.

Pokud v Calibre provádíte vyhledávání a chcete pro něj vygenerovat odkaz, můžete tak učinit kliknutím pravým tlačítkem myši na panel hledání a výběrem *Kopírovat hledání jako URL*.

11.5 Otevřít okno s podrobnostmi o knize na knize v některé knihovně

Syntaxe URL je:

```
calibre://book-details/Library_Name/book_id
```

Toto otevře okno s podrobnostmi o knize na zadané knize ze zadané knihovny, aniž by se změnila aktuální knihovna nebo vybraná kniha.

11.6 Otevřít poznámky přidružené k autorovi, sérii atd.

Syntaxe URL je:

```
calibre://book-details/Library_Name/Field_Name/id_Item_Id
```

Otevře se okno s poznámkami k zadané položce. Nejjednodušší způsob, jak takové adresy URL vytvořit, je zobrazit požadované poznámky v Calibre a kliknutím na tlačítko *Kopírovat URL* zkopírovat adresu URL do schránky a vložit ji, kam potřebujete.

Zde `Field_Name` je název sloupce, například `authors` nebo `tags`. U sloupců vytvořených uživatelem nahraďte počáteční `#` v názvu pole podtržítkem, takže `#mytags` se změní na `_mytags`.

Kromě zadávání položek podle id pomocí `Item_Id` je můžete zadat také podle názvu pomocí `val_Item_Name` nebo `hex_Hex_Encoded_Item_Name`. Například:

```
calibre://book-details/Library_Name/authors/val_John%20Doe
```

11.7 Hex kódování URL parametrů

Hex kódování URL parametrů se provádí tak, že se parametry nejprve zakódují jako UTF-8 byty a pak se každý byte nahradí dvěma hexadecimálními znaky reprezentujícími tento byte. Například řetězec „abc“ jsou byty „0x61 0x62 a 0x63“ v UTF-8 a zakódovaná verze tedy bude mít podobu řetězce: „616263“.

KAPITOLA 12

Přizpůsobení Calibre

Calibre má vysoce modulární návrh. Různé jeho části mohou být přizpůsobeny. Zde se dozvíte,:

- jak používat proměnné prostředí a *vylepšení* k přizpůsobení chování Calibre,
- jak zadat vlastní statické zdroje, jako jsou ikony a šablony, pro přepsání výchozího nastavení,
- jak používat *moduly* pro přidání funkcí do Calibre,
- jak sdílet motivy ikon a moduly s ostatními uživateli Calibre.
- pro zjištění, jak vytvořit *předpisy* pro přidávání nových zdrojů online obsahu do Calibre, navštivte oddíl *Přidávání oblíbených webových stránek se zprávami* (stránka 33).

Poznámka

Pamatujte, že ačkoliv jsou motivy ikon a moduly indexovány a lze je stáhnout prostřednictvím vestavěného nástroje aktualizací Calibre, nejsou součástí Calibre a jejich kanonické umístění pro podporu a zdrojový kód je na fóru Mobileread forums[114] v jejich vláknech podpory.

- *Proměnné prostředí* (stránka 287)
- *Vylepšení* (stránka 288)
- *Přepsání ikon, šablon atd.* (stránka 300)
- *Vytvoření vlastního motivu ikon pro Calibre* (stránka 301)
- *Přizpůsobení Calibre pomocí modulů* (stránka 301)

[114] https://www.mobileread.com/forums/forumdisplay.php?f=166

12.1 Dokumentace API pro moduly

Definuje různé abstraktní základní třídy, které mohou být podtřídami pro vytvoření výkonných modulů. Užitečné třídy jsou:

- *Modul* (stránka 256)
- *ModulTypuSouboru* (stránka 258)
- *Moduly metadat* (stránka 260)
- *Moduly katalogu* (stránka 261)
- *Moduly stahování metadat* (stránka 262)
- *Moduly převodu* (stránka 265)
- *Ovladače zařízení* (stránka 268)
- *User interface actions* (stránka 281)
- *Moduly předvoleb* (stránka 285)

12.1.1 Modul

class calibre.customize.**Plugin**(*plugin_path*)

A calibre plugin. Useful members include:

- `self.installation_type`: Stores how the plugin was installed.
- **`self.plugin_path`: Stores path to the ZIP file that contains** this plugin or None if it is a builtin plugin
- **`self.site_customization`: Stores a customization string entered** uživatelem.

Methods that should be overridden in sub classes:

- *`initialize()`* (stránka 257)
- *`customization_help()`* (stránka 258)

Useful methods:

- *`temporary_file()`* (stránka 258)
- `__enter__()`
- *`load_resources()`* (stránka 257)

`supported_platforms = []`

List of platforms this plugin works on. For example: `['windows', 'osx', 'linux']`

`name = 'Trivial Plugin'`

The name of this plugin. You must set it something other than Trivial Plugin for it to work.

`version = (1, 0, 0)`

The version of this plugin as a 3-tuple (major, minor, revision)

`description = 'Nedělá vůbec nic'`

A short string describing what this plugin does

author = 'Neznámý'

Autor doplňku

priority = 1

When more than one plugin exists for a filetype, the plugins are run in order of decreasing priority. Plugins with higher priority will be run first. The highest possible priority is `sys.maxsize`. Default priority is 1.

minimum_calibre_version = (0, 4, 118)

The earliest version of calibre this plugin requires

installation_type = None

The way this plugin is installed

can_be_disabled = True

If False, the user will not be able to disable this plugin. Use with care.

type = 'Základ'

The type of this plugin. Used for categorizing plugins in the GUI

initialize()

Called once when calibre plugins are initialized. Plugins are re-initialized every time a new plugin is added. Also note that if the plugin is run in a worker process, such as for adding books, then the plugin will be initialized for every new worker process.

Perform any plugin specific initialization here, such as extracting resources from the plugin ZIP file. The path to the ZIP file is available as `self.plugin_path`.

Note that `self.site_customization` is **not** available at this point.

config_widget()

Implement this method and *save_settings()* (stránka 257) in your plugin to use a custom configuration dialog, rather then relying on the simple string based default customization.

This method, if implemented, must return a QWidget. The widget can have an optional method validate() that takes no arguments and is called immediately after the user clicks OK. Changes are applied if and only if the method returns True.

If for some reason you cannot perform the configuration at this time, return a tuple of two strings (message, details), these will be displayed as a warning dialog to the user and the process will be aborted.

save_settings(*config_widget*)

Save the settings specified by the user with config_widget.

Parametry

config_widget – The widget returned by *config_widget()* (stránka 257).

do_user_config(*parent=None*)

This method shows a configuration dialog for this plugin. It returns True if the user clicks OK, False otherwise. The changes are automatically applied.

load_resources(*names*)

If this plugin comes in a ZIP file (user added plugin), this method will allow you to load resources from the ZIP file.

For example to load an image:

```
pixmap = QPixmap()
pixmap.loadFromData(self.load_resources(['images/icon.png'])['images/icon.png
↪'])
icon = QIcon(pixmap)
```

Parametry
names – List of paths to resources in the ZIP file using / as separator

Vrací
A dictionary of the form `{name: file_contents}`. Any names that were not found in the ZIP file will not be present in the dictionary.

customization_help(*gui=False*)

Return a string giving help on how to customize this plugin. By default raise a `NotImplementedError`, which indicates that the plugin does not require customization.

If you re-implement this method in your subclass, the user will be asked to enter a string as customization for this plugin. The customization string will be available as `self.site_customization`.

Site customization could be anything, for example, the path to a needed binary on the user's computer.

Parametry
gui – If True return HTML help, otherwise return plain text help.

temporary_file(*suffix*)

Return a file-like object that is a temporary file on the file system. This file will remain available even after being closed and will only be removed on interpreter shutdown. Use the `name` member of the returned object to access the full path to the created temporary file.

Parametry
suffix – The suffix that the temporary file will have.

cli_main(*args*)

This method is the main entry point for your plugins command line interface. It is called when the user does: calibre-debug -r „Plugin Name“. Any arguments passed are present in the args variable.

12.1.2 ModulTypuSouboru

class calibre.customize.**FileTypePlugin**(*plugin_path*)

Bases: *Plugin* (stránka 256)

A plugin that is associated with a particular set of file types.

file_types = {}

Set of file types for which this plugin should be run. Use ‚*‘ for all file types. For example: `{'lit', 'mobi', 'prc'}`

on_import = False

If True, this plugin is run when books are added to the database

on_postimport = False

If True, this plugin is run after books are added to the database. In this case the postimport and postadd methods of the plugin are called.

on_postconvert = False

If True, this plugin is run after a book is converted. In this case the postconvert method of the plugin is called.

on_postdelete = False

If True, this plugin is run after a book file is deleted from the database. In this case the postdelete method of the plugin is called.

on_preprocess = False

If True, this plugin is run just before a conversion

on_postprocess = False

If True, this plugin is run after conversion on the final file produced by the conversion output plugin.

type = 'Typ souboru'

The type of this plugin. Used for categorizing plugins in the GUI

run(*path_to_ebook*)

Run the plugin. Must be implemented in subclasses. It should perform whatever modifications are required on the e-book and return the absolute path to the modified e-book. If no modifications are needed, it should return the path to the original e-book. If an error is encountered it should raise an Exception. The default implementation simply return the path to the original e-book. Note that the path to the original file (before any file type plugins are run, is available as self.original_path_to_file).

The modified e-book file should be created with the `temporary_file()` method.

Parametry

path_to_ebook – Absolute path to the e-book.

Vrací

Absolute path to the modified e-book.

postimport(*book_id*, *book_format*, *db*)

Called post import, i.e., after the book file has been added to the database. Note that this is different from *postadd()* (stránka 259) which is called when the book record is created for the first time. This method is called whenever a new file is added to a book record. It is useful for modifying the book record based on the contents of the newly added file.

Parametry

- **book_id** – Database id of the added book.
- **book_format** – The file type of the book that was added.
- **db** – Library database.

postconvert(*book_id*, *book_format*, *db*)

Called post conversion, i.e., after the conversion output book file has been added to the database. Note that it is run after a conversion only, not after a book is added. It is useful for modifying the book record based on the contents of the newly added file.

Parametry

- **book_id** – Database id of the added book.
- **book_format** – The file type of the book that was added.
- **db** – Library database.

postdelete(*book_id*, *book_format*, *db*)

Called post deletion, i.e., after the book file has been deleted from the database. Note that it is not run when a book record is deleted, only when one or more formats from the book are deleted. It is useful for modifying the book record based on the format of the deleted file.

Parametry

- **book_id** – Database id of the added book.
- **book_format** – The file type of the book that was added.
- **db** – Library database.

postadd(*book_id*, *fmt_map*, *db*)

Called post add, i.e. after a book has been added to the db. Note that this is different from *postimport()* (stránka 259), which is called after a single book file has been added to a book. postadd() is called only when an entire book record with possibly more than one book file has been created for the first time. This is useful if you wish to modify the book record in the database when the book is first added to calibre.

Parametry

- **book_id** – Database id of the added book.
- **fmt_map** – Map of file format to path from which the file format was added. Note that this might or might not point to an actual existing file, as sometimes files are added as streams. In which case it might be a dummy value or a non-existent path.
- **db** – Library database

12.1.3 Moduly metadat

class calibre.customize.**MetadataReaderPlugin**(**args*, ***kwargs*)

Bases: *Plugin* (stránka 256)

A plugin that implements reading metadata from a set of file types.

file_types = {}

Set of file types for which this plugin should be run. For example: `set(['lit', 'mobi', 'prc'])`

supported_platforms = ['windows', 'osx', 'linux']

List of platforms this plugin works on. For example: `['windows', 'osx', 'linux']`

version = (8, 7, 0)

The version of this plugin as a 3-tuple (major, minor, revision)

author = 'Kovid Goyal'

Autor doplňku

type = 'Čtení metadat'

The type of this plugin. Used for categorizing plugins in the GUI

get_metadata(*stream*, *type*)

Return metadata for the file represented by stream (a file like object that supports reading). Raise an exception when there is an error with the input data.

Parametry

type – The type of file. Guaranteed to be one of the entries in *file_types* (stránka 260).

Vrací

A `calibre.ebooks.metadata.book.Metadata` object

class calibre.customize.**MetadataWriterPlugin**(**args*, ***kwargs*)

Bases: *Plugin* (stránka 256)

A plugin that implements reading metadata from a set of file types.

file_types = {}

Set of file types for which this plugin should be run. For example: `set(['lit', 'mobi', 'prc'])`

supported_platforms = ['windows', 'osx', 'linux']

List of platforms this plugin works on. For example: `['windows', 'osx', 'linux']`

version = (8, 7, 0)

The version of this plugin as a 3-tuple (major, minor, revision)

author = 'Kovid Goyal'

Autor doplňku

type = 'Zápis metadat'

The type of this plugin. Used for categorizing plugins in the GUI

set_metadata(*stream*, *mi*, *type*)

Set metadata for the file represented by stream (a file like object that supports reading). Raise an exception when there is an error with the input data.

Parametry

- **type** – The type of file. Guaranteed to be one of the entries in *file_types* (stránka 260).
- **mi** – A `calibre.ebooks.metadata.book.Metadata` object

12.1.4 Moduly katalogu

class `calibre.customize.`**CatalogPlugin**(*plugin_path*)

Bases: *Plugin* (stránka 256)

A plugin that implements a catalog generator.

file_types = {}

Output file type for which this plugin should be run. For example: ‚epub' or ‚xml'

type = 'Generátor katalogu'

The type of this plugin. Used for categorizing plugins in the GUI

cli_options = []

CLI parser options specific to this plugin, declared as *namedtuple Option*:

from collections import namedtuple Option = namedtuple(‚Option', ‚option, default, dest, help') cli_options = [Option('–catalog-title', default = ‚My Catalog', dest = ‚catalog_title', help = (_(‚Title of generated catalog. nDefault:') + „ ‚" + ‚%default' + „"'))] cli_options parsed in calibre.db.cli.cmd_catalog:option_parser()

initialize()

If plugin is not a built-in, copy the plugin's .ui and .py files from the ZIP file to $TMPDIR. Tab will be dynamically generated and added to the Catalog Options dialog in calibre.gui2.dialogs.catalog.py:Catalog

run(*path_to_output*, *opts*, *db*, *ids*, *notification=None*)

Run the plugin. Must be implemented in subclasses. It should generate the catalog in the format specified in file_types, returning the absolute path to the generated catalog file. If an error is encountered it should raise an Exception.

The generated catalog file should be created with the `temporary_file()` method.

Parametry

- **path_to_output** – Absolute path to the generated catalog file.
- **opts** – A dictionary of keyword arguments
- **db** – A LibraryDatabase2 object

12.1.5 Moduly stahování metadat

class calibre.ebooks.metadata.sources.base.**Source**(**args*, ***kwargs*)

Bases: *Plugin* (stránka 256)

type = 'Zdroj metadat'

The type of this plugin. Used for categorizing plugins in the GUI

author = 'Kovid Goyal'

Autor doplňku

supported_platforms = ['windows', 'osx', 'linux']

List of platforms this plugin works on. For example: `['windows', 'osx', 'linux']`

capabilities = frozenset({})

Set of capabilities supported by this plugin. Useful capabilities are: ‚identify', ‚cover'

touched_fields = frozenset({})

List of metadata fields that can potentially be download by this plugin during the identify phase

has_html_comments = False

Set this to True if your plugin returns HTML formatted comments

supports_gzip_transfer_encoding = False

Setting this to True means that the browser object will indicate that it supports gzip transfer encoding. This can speedup downloads but make sure that the source actually supports gzip transfer encoding correctly first

ignore_ssl_errors = False

Set this to True to ignore HTTPS certificate errors when connecting to this source.

cached_cover_url_is_reliable = True

Cached cover URLs can sometimes be unreliable (i.e. the download could fail or the returned image could be bogus). If that is often the case with this source, set to False

options = ()

A list of `Option` objects. They will be used to automatically construct the configuration widget for this plugin

config_help_message = None

A string that is displayed at the top of the config widget for this plugin

can_get_multiple_covers = False

If True this source can return multiple covers for a given query

auto_trim_covers = False

If set to True covers downloaded by this plugin are automatically trimmed.

prefer_results_with_isbn = True

If set to True, and this source returns multiple results for a query, some of which have ISBNs and some of which do not, the results without ISBNs will be ignored

is_configured()

Return False if your plugin needs to be configured before it can be used. For example, it might need a username/password/API key.

customization_help()

Return a string giving help on how to customize this plugin. By default raise a `NotImplementedError`, which indicates that the plugin does not require customization.

If you re-implement this method in your subclass, the user will be asked to enter a string as customization for this plugin. The customization string will be available as `self.site_customization`.

Site customization could be anything, for example, the path to a needed binary on the user's computer.

Parametry
gui – If True return HTML help, otherwise return plain text help.

config_widget()

Implement this method and *save_settings()* (stránka 263) in your plugin to use a custom configuration dialog, rather then relying on the simple string based default customization.

This method, if implemented, must return a QWidget. The widget can have an optional method validate() that takes no arguments and is called immediately after the user clicks OK. Changes are applied if and only if the method returns True.

If for some reason you cannot perform the configuration at this time, return a tuple of two strings (message, details), these will be displayed as a warning dialog to the user and the process will be aborted.

save_settings(*config_widget*)

Save the settings specified by the user with config_widget.

Parametry
config_widget – The widget returned by *config_widget()* (stránka 263).

get_author_tokens(*authors*, *only_first_author=True*)

Take a list of authors and return a list of tokens useful for an AND search query. This function tries to return tokens in first name middle names last name order, by assuming that if a comma is in the author name, the name is in lastname, other names form.

get_title_tokens(*title*, *strip_joiners=True*, *strip_subtitle=False*)

Take a title and return a list of tokens useful for an AND search query. Excludes connectives(optionally) and punctuation.

split_jobs(*jobs*, *num*)

Split a list of jobs into at most num groups, as evenly as possible

test_fields(*mi*)

Return the first field from self.touched_fields that is null on the mi object

clean_downloaded_metadata(*mi*)

Call this method in your plugin's identify method to normalize metadata before putting the Metadata object into result_queue. You can of course, use a custom algorithm suited to your metadata source.

get_book_url(*identifiers*)

Return a 3-tuple or None. The 3-tuple is of the form: (identifier_type, identifier_value, URL). The URL is the URL for the book identified by identifiers at this source. identifier_type, identifier_value specify the identifier corresponding to the URL. This URL must be browsable to by a human using a browser. It is meant to provide a clickable link for the user to easily visit the books page at this source. If no URL is found, return None. This method must be quick, and consistent, so only implement it if it is possible to construct the URL from a known scheme given identifiers.

get_book_url_name(*idtype*, *idval*, *url*)

Return a human readable name from the return value of get_book_url().

get_book_urls(*identifiers*)

Override this method if you would like to return multiple URLs for this book. Return a list of 3-tuples. By default this method simply calls *get_book_url()* (stránka 263).

get_cached_cover_url(*identifiers*)

Return cached cover URL for the book identified by the identifiers dictionary or None if no such URL exists.

Note that this method must only return validated URLs, i.e. not URLS that could result in a generic cover image or a not found error.

id_from_url(*url*)

Parse a URL and return a tuple of the form: (identifier_type, identifier_value). If the URL does not match the pattern for the metadata source, return None.

identify_results_keygen(*title=None*, *authors=None*, *identifiers={}*)

Return a function that is used to generate a key that can sort Metadata objects by their relevance given a search query (title, authors, identifiers).

These keys are used to sort the results of a call to *identify()* (stránka 264).

For details on the default algorithm see *InternalMetadataCompareKeyGen* (stránka 265). Re-implement this function in your plugin if the default algorithm is not suitable.

identify(*log*, *result_queue*, *abort*, *title=None*, *authors=None*, *identifiers={}*, *timeout=30*)

Identify a book by its Title/Author/ISBN/etc.

If identifiers(s) are specified and no match is found and this metadata source does not store all related identifiers (for example, all ISBNs of a book), this method should retry with just the title and author (assuming they were specified).

If this metadata source also provides covers, the URL to the cover should be cached so that a subsequent call to the get covers API with the same ISBN/special identifier does not need to get the cover URL again. Use the caching API for this.

Every Metadata object put into result_queue by this method must have a *source_relevance* attribute that is an integer indicating the order in which the results were returned by the metadata source for this query. This integer will be used by `compare_identify_results()`. If the order is unimportant, set it to zero for every result.

Make sure that any cover/ISBN mapping information is cached before the Metadata object is put into result_queue.

Parametry

- **log** – A log object, use it to output debugging information/errors
- **result_queue** – A result Queue, results should be put into it. Each result is a Metadata object
- **abort** – If abort.is_set() returns True, abort further processing and return as soon as possible
- **title** – The title of the book, can be None
- **authors** – A list of authors of the book, can be None
- **identifiers** – A dictionary of other identifiers, most commonly {‚isbn‘:‘1234…‘}
- **timeout** – Timeout in seconds, no network request should hang for longer than timeout.

Vrací

None if no errors occurred, otherwise a unicode representation of the error suitable for showing to the user

download_cover(*log*, *result_queue*, *abort*, *title=None*, *authors=None*, *identifiers={}*, *timeout=30*, *get_best_cover=False*)

Download a cover and put it into result_queue. The parameters all have the same meaning as for *identify()* (stránka 264). Put (self, cover_data) into result_queue.

This method should use cached cover URLs for efficiency whenever possible. When cached data is not present, most plugins simply call identify and use its results.

If the parameter get_best_cover is True and this plugin can get multiple covers, it should only get the „best“ one.

class calibre.ebooks.metadata.sources.base.**InternalMetadataCompareKeyGen**(*mi*, *source_plugin*, *title*, *authors*, *identifiers*)

Generate a sort key for comparison of the relevance of Metadata objects, given a search query. This is used only to compare results from the same metadata source, not across different sources.

The sort key ensures that an ascending order sort is a sort by order of decreasing relevance.

The algorithm is:

- Prefer results that have at least one identifier the same as for the query
- Prefer results with a cached cover URL
- Prefer results with all available fields filled in
- Prefer results with the same language as the current user interface language
- Prefer results that are an exact title match to the query
- Prefer results with longer comments (greater than 10% longer)
- **Use the relevance of the result as reported by the metadata source's search** engine

12.1.6 Moduly převodu

class calibre.customize.conversion.**InputFormatPlugin**(**args*)

Bases: *Plugin* (stránka 256)

InputFormatPlugins are responsible for converting a document into HTML+OPF+CSS+etc. The results of the conversion *must* be encoded in UTF-8. The main action happens in *convert()* (stránka 266).

type = 'Vstup převodu'

The type of this plugin. Used for categorizing plugins in the GUI

can_be_disabled = False

If False, the user will not be able to disable this plugin. Use with care.

supported_platforms = ['windows', 'osx', 'linux']

List of platforms this plugin works on. For example: `['windows', 'osx', 'linux']`

file_types = {}

Set of file types for which this plugin should be run For example: `set(['azw', 'mobi', 'prc'])`

is_image_collection = False

If True, this input plugin generates a collection of images, one per HTML file. This can be set dynamically, in the convert method if the input files can be both image collections and non-image collections. If you set this to True, you must implement the get_images() method that returns a list of images.

core_usage = 1

Number of CPU cores used by this plugin. A value of -1 means that it uses all available cores

for_viewer = False

If set to True, the input plugin will perform special processing to make its output suitable for viewing

output_encoding = 'utf-8'

The encoding that this input plugin creates files in. A value of None means that the encoding is undefined and must be detected individually

common_options = {<calibre.customize.conversion.OptionRecommendation object>}

Options shared by all Input format plugins. Do not override in sub-classes. Use *options* (stránka 266) instead. Every option must be an instance of `OptionRecommendation`.

options = {}

Options to customize the behavior of this plugin. Every option must be an instance of `OptionRecommendation`.

recommendations = {}

A set of 3-tuples of the form (option_name, recommended_value, recommendation_level)

get_images()

Return a list of absolute paths to the images, if this input plugin represents an image collection. The list of images is in the same order as the spine and the TOC.

convert(*stream*, *options*, *file_ext*, *log*, *accelerators*)

This method must be implemented in sub-classes. It must return the path to the created OPF file or an `OEBBook` instance. All output should be contained in the current folder. If this plugin creates files outside the current folder they must be deleted/marked for deletion before this method returns.

Parametry

- **stream** – A file like object that contains the input file.
- **options** – Options to customize the conversion process. Guaranteed to have attributes corresponding to all the options declared by this plugin. In addition, it will have a verbose attribute that takes integral values from zero upwards. Higher numbers mean be more verbose. Another useful attribute is `input_profile` that is an instance of `calibre.customize.profiles.InputProfile`.
- **file_ext** – The extension (without the .) of the input file. It is guaranteed to be one of the *file_types* supported by this plugin.
- **log** – A `calibre.utils.logging.Log` object. All output should use this object.
- **accelerators** – A dictionary of various information that the input plugin can get easily that would speed up the subsequent stages of the conversion.

postprocess_book(*oeb*, *opts*, *log*)

Called to allow the input plugin to perform postprocessing after the book has been parsed.

specialize(*oeb*, *opts*, *log*, *output_fmt*)

Called to allow the input plugin to specialize the parsed book for a particular output format. Called after postprocess_book and before any transforms are performed on the parsed book.

gui_configuration_widget(*parent*, *get_option_by_name*, *get_option_help*, *db*, *book_id=None*)

Called to create the widget used for configuring this plugin in the calibre GUI. The widget must be an instance of the PluginWidget class. See the builtin input plugins for examples.

class calibre.customize.conversion.**OutputFormatPlugin**(**args*)

Bases: *Plugin* (stránka 256)

OutputFormatPlugins are responsible for converting an OEB document (OPF+HTML) into an output e-book.

The OEB document can be assumed to be encoded in UTF-8. The main action happens in *convert()* (stránka 267).

type = 'Výstup převodu'

The type of this plugin. Used for categorizing plugins in the GUI

can_be_disabled = False

If False, the user will not be able to disable this plugin. Use with care.

supported_platforms = ['windows', 'osx', 'linux']

List of platforms this plugin works on. For example: `['windows', 'osx', 'linux']`

file_type = None

The file type (extension without leading period) that this plugin outputs

common_options = {<calibre.customize.conversion.OptionRecommendation object>}

Options shared by all Input format plugins. Do not override in sub-classes. Use *options* (stránka 267) instead. Every option must be an instance of `OptionRecommendation`.

options = {}

Options to customize the behavior of this plugin. Every option must be an instance of `OptionRecommendation`.

recommendations = {}

A set of 3-tuples of the form (option_name, recommended_value, recommendation_level)

property description

str(object='') -> str str(bytes_or_buffer[, encoding[, errors]]) -> str

Create a new string object from the given object. If encoding or errors is specified, then the object must expose a data buffer that will be decoded using the given encoding and error handler. Otherwise, returns the result of object.__str__() (if defined) or repr(object). encoding defaults to ‚utf-8'. errors defaults to ‚strict'.

convert(*oeb_book*, *output*, *input_plugin*, *opts*, *log*)

Render the contents of *oeb_book* (which is an instance of `calibre.ebooks.oeb.OEBBook`) to the file specified by output.

Parametry

- **output** – Either a file like object or a string. If it is a string it is the path to a folder that may or may not exist. The output plugin should write its output into that folder. If it is a file like object, the output plugin should write its output into the file.
- **input_plugin** – The input plugin that was used at the beginning of the conversion pipeline.
- **opts** – Conversion options. Guaranteed to have attributes corresponding to the OptionRecommendations of this plugin.
- **log** – The logger. Print debug/info messages etc. using this.

specialize_options(*log*, *opts*, *input_fmt*)

Can be used to change the values of conversion options, as used by the conversion pipeline.

specialize_css_for_output(*log*, *opts*, *item*, *stylizer*)

Can be used to make changes to the CSS during the CSS flattening process.

Parametry

- **item** – The item (HTML file) being processed
- **stylizer** – A Stylizer object containing the flattened styles for item. You can get the style for any element by stylizer.style(element).

gui_configuration_widget(*parent*, *get_option_by_name*, *get_option_help*, *db*, *book_id=None*)

Called to create the widget used for configuring this plugin in the calibre GUI. The widget must be an instance of the PluginWidget class. See the builtin output plugins for examples.

12.1.7 Ovladače zařízení

The base class for all device drivers is *DevicePlugin* (stránka 268). However, if your device exposes itself as a USBMS drive to the operating system, you should use the USBMS class instead as it implements all the logic needed to support these kinds of devices.

class calibre.devices.interface.**DevicePlugin**(*plugin_path*)

Bases: *Plugin* (stránka 256)

Defines the interface that should be implemented by backends that communicate with an e-book reader.

type = 'Rozhraní zařízení'

The type of this plugin. Used for categorizing plugins in the GUI

FORMATS = ['lrf', 'rtf', 'pdf', 'txt']

Ordered list of supported formats

VENDOR_ID = 0

VENDOR_ID can be either an integer, a list of integers or a dictionary If it is a dictionary, it must be a dictionary of dictionaries, of the form:

```
{
 integer_vendor_id : { product_id : [list of BCDs], ... },
 ...
}
```

PRODUCT_ID = 0

An integer or a list of integers

BCD = None

BCD can be either None to not distinguish between devices based on BCD, or it can be a list of the BCD numbers of all devices supported by this driver.

THUMBNAIL_HEIGHT = 68

Height for thumbnails on the device

THUMBNAIL_COMPRESSION_QUALITY = 75

Compression quality for thumbnails. Set this closer to 100 to have better quality thumbnails with fewer compression artifacts. Of course, the thumbnails get larger as well.

WANTS_UPDATED_THUMBNAILS = False

Set this to True if the device supports updating cover thumbnails during sync_booklists. Setting it to true will ask device.py to refresh the cover thumbnails during book matching

CAN_SET_METADATA = ['title', 'authors', 'collections']

Whether the metadata on books can be set via the GUI.

CAN_DO_DEVICE_DB_PLUGBOARD = False

Whether the device can handle device_db metadata plugboards

path_sep = '/'

Path separator for paths to books on device

icon = 'reader.png'

Icon for this device

UserAnnotation

alias of `Annotation`

OPEN_FEEDBACK_MESSAGE = None

GUI displays this as a message if not None in the status bar. Useful if opening can take a long time

VIRTUAL_BOOK_EXTENSIONS = frozenset({})

Set of extensions that are „virtual books“ on the device and therefore cannot be viewed/saved/added to library. For example: `frozenset(['kobo'])`

VIRTUAL_BOOK_EXTENSION_MESSAGE = None

Message to display to user for virtual book extensions.

NUKE_COMMENTS = None

Whether to nuke comments in the copy of the book sent to the device. If not None this should be short string that the comments will be replaced by.

MANAGES_DEVICE_PRESENCE = False

If True indicates that this driver completely manages device detection, ejecting and so forth. If you set this to True, you *must* implement the detect_managed_devices and debug_managed_device_detection methods. A driver with this set to true is responsible for detection of devices, managing a blacklist of devices, a list of ejected devices and so forth. calibre will periodically call the detect_managed_devices() method and if it returns a detected device, calibre will call open(). open() will be called every time a device is returned even if previous calls to open() failed, therefore the driver must maintain its own blacklist of failed devices. Similarly, when ejecting, calibre will call eject() and then assuming the next call to detect_managed_devices() returns None, it will call post_yank_cleanup().

SLOW_DRIVEINFO = False

If set the True, calibre will call the *get_driveinfo()* (stránka 271) method after the books lists have been loaded to get the driveinfo.

ASK_TO_ALLOW_CONNECT = False

If set to True, calibre will ask the user if they want to manage the device with calibre, the first time it is detected. If you set this to True you must implement *get_device_uid()* (stránka 274) and *ignore_connected_device()* (stránka 274) and *get_user_blacklisted_devices()* (stránka 274) and *set_user_blacklisted_devices()* (stránka 274)

user_feedback_after_callback = None

Set this to a dictionary of the form {‚title‘:title, ‚msg‘:msg, ‚det_msg‘:detailed_msg} to have calibre popup a message to the user after some callbacks are run (currently only upload_books). Be careful to not spam the user with too many messages. This variable is checked after *every* callback, so only set it when you really need to.

classmethod get_open_popup_message()

GUI displays this as a non-modal popup. Should be an instance of OpenPopupMessage

classmethod model_metadata() → tuple[ModelMetadata, ...]

Metadata about all the actual device models this driver supports

is_usb_connected(*devices_on_system*, *debug=False*, *only_presence=False*)

Return True, device_info if a device handled by this plugin is currently connected.

Parametry

devices_on_system – List of devices currently connected

detect_managed_devices(*devices_on_system*, *force_refresh=False*)

Called only if MANAGES_DEVICE_PRESENCE is True.

Scan for devices that this driver can handle. Should return a device object if a device is found. This object will be passed to the open() method as the connected_device. If no device is found, return None. The returned object can be anything, calibre does not use it, it is only passed to open().

This method is called periodically by the GUI, so make sure it is not too resource intensive. Use a cache to avoid repeatedly scanning the system.

Parametry

- **devices_on_system** – Set of USB devices found on the system.
- **force_refresh** – If True and the driver uses a cache to prevent repeated scanning, the cache must be flushed.

debug_managed_device_detection(*devices_on_system*, *output*)

Called only if MANAGES_DEVICE_PRESENCE is True.

Should write information about the devices detected on the system to output, which is a file like object.

Should return True if a device was detected and successfully opened, otherwise False.

reset(*key='-1'*, *log_packets=False*, *report_progress=None*, *detected_device=None*)

Parametry

- **key** – The key to unlock the device
- **log_packets** – If true the packet stream to/from the device is logged
- **report_progress** – Function that is called with a % progress (number between 0 and 100) for various tasks. If it is called with -1 that means that the task does not have any progress information
- **detected_device** – Device information from the device scanner

can_handle_windows(*usbdevice*, *debug=False*)

Optional method to perform further checks on a device to see if this driver is capable of handling it. If it is not it should return False. This method is only called after the vendor, product ids and the bcd have matched, so it can do some relatively time intensive checks. The default implementation returns True. This method is called only on Windows. See also *can_handle()* (stránka 270).

Note that for devices based on USBMS this method by default delegates to *can_handle()* (stránka 270). So you only need to override *can_handle()* (stránka 270) in your subclass of USBMS.

Parametry

usbdevice – A usbdevice as returned by `calibre.devices.winusb.scan_usb_devices()`

can_handle(*device_info*, *debug=False*)

Unix version of *can_handle_windows()* (stránka 270).

Parametry

device_info – Is a tuple of (vid, pid, bcd, manufacturer, product, serial number)

open(*connected_device*, *library_uuid*)

Perform any device specific initialization. Called after the device is detected but before any other functions that communicate with the device. For example: For devices that present themselves as USB Mass storage devices, this method would be responsible for mounting the device or if the device has been automounted,

for finding out where it has been mounted. The method *calibre.devices.usbms.device.Device.open()* (stránka 278) has an implementation of this function that should serve as a good example for USB Mass storage devices.

This method can raise an OpenFeedback exception to display a message to the user.

Parametry

- **connected_device** – The device that we are trying to open. It is a tuple of (vendor id, product id, bcd, manufacturer name, product name, device serial number). However, some devices have no serial number and on Windows only the first three fields are present, the rest are None.
- **library_uuid** – The UUID of the current calibre library. Can be None if there is no library (for example when used from the command line).

eject()

Un-mount / eject the device from the OS. This does not check if there are pending GUI jobs that need to communicate with the device.

NOTE: That this method may not be called on the same thread as the rest of the device methods.

post_yank_cleanup()

Called if the user yanks the device without ejecting it first.

set_progress_reporter(*report_progress*)

Set a function to report progress information.

Parametry

report_progress – Function that is called with a % progress (number between 0 and 100) for various tasks. If it is called with -1 that means that the task does not have any progress information

get_device_information(*end_session=True*)

Ask device for device information. See L{DeviceInfoQuery}.

Vrací

(device name, device version, software version on device, MIME type) The tuple can optionally have a fifth element, which is a drive information dictionary. See usbms.driver for an example.

get_driveinfo()

Return the driveinfo dictionary. Usually called from get_device_information(), but if loading the driveinfo is slow for this driver, then it should set SLOW_DRIVEINFO. In this case, this method will be called by calibre after the book lists have been loaded. Note that it is not called on the device thread, so the driver should cache the drive info in the books() method and this function should return the cached data.

card_prefix(*end_session=True*)

Return a 2 element list of the prefix to paths on the cards. If no card is present None is set for the card's prefix. E.G. (‚/place', ‚/place2') (None, ‚place2') (‚place', None) (None, None)

total_space(*end_session=True*)

Get total space available on the mountpoints:

1. Hlavní paměť
2. Memory Card A
3. Memory Card B

Vrací
A 3 element list with total space in bytes of (1, 2, 3). If a particular device doesn't have any of these locations it should return 0.

free_space(*end_session=True*)

Get free space available on the mountpoints:

1. Hlavní paměť
2. Karta A
3. Karta B

Vrací
A 3 element list with free space in bytes of (1, 2, 3). If a particular device doesn't have any of these locations it should return -1.

books(*oncard=None*, *end_session=True*)
Return a list of e-books on the device.

Parametry
oncard – If ‚carda' or ‚cardb' return a list of e-books on the specific storage card, otherwise return list of e-books in main memory of device. If a card is specified and no books are on the card return empty list.

Vrací
A BookList.

upload_books(*files*, *names*, *on_card=None*, *end_session=True*, *metadata=None*)
Upload a list of books to the device. If a file already exists on the device, it should be replaced. This method should raise a `FreeSpaceError` if there is not enough free space on the device. The text of the FreeSpaceError must contain the word „card" if `on_card` is not None otherwise it must contain the word „memory".

Parametry

- **files** – A list of paths
- **names** – A list of file names that the books should have once uploaded to the device. len(names) == len(files)
- **metadata** – If not None, it is a list of `Metadata` objects. The idea is to use the metadata to determine where on the device to put the book. len(metadata) == len(files). Apart from the regular cover (path to cover), there may also be a thumbnail attribute, which should be used in preference. The thumbnail attribute is of the form (width, height, cover_data as jpeg).

Vrací
A list of 3-element tuples. The list is meant to be passed to *add_books_to_metadata()* (stránka 272).

classmethod add_books_to_metadata(*locations*, *metadata*, *booklists*)
Add locations to the booklists. This function must not communicate with the device.

Parametry

- **locations** – Result of a call to L{upload_books}
- **metadata** – List of `Metadata` objects, same as for *upload_books()* (stránka 272).
- **booklists** – A tuple containing the result of calls to (`books(oncard=None)()`, `books(oncard='carda')()`, :meth`books(oncard='cardb')`).

delete_books(*paths*, *end_session=True*)

Delete books at paths on device.

classmethod remove_books_from_metadata(*paths*, *booklists*)

Remove books from the metadata list. This function must not communicate with the device.

Parametry

- **paths** – paths to books on the device.
- **booklists** – A tuple containing the result of calls to (`books(oncard=None)()`, `books(oncard='carda')()`, :meth`books(oncard='cardb')`).

sync_booklists(*booklists*, *end_session=True*)

Update metadata on device.

Parametry

booklists – A tuple containing the result of calls to (`books(oncard=None)()`, `books(oncard='carda')()`, :meth`books(oncard='cardb')`).

get_file(*path*, *outfile*, *end_session=True*)

Read the file at `path` on the device and write it to outfile.

Parametry

outfile – file object like `sys.stdout` or the result of an *open()* (stránka 270) call.

classmethod config_widget()

Should return a QWidget. The QWidget contains the settings for the device interface

classmethod save_settings(*settings_widget*)

Should save settings to disk. Takes the widget created in *config_widget()* (stránka 273) and saves all settings to disk.

classmethod settings()

Should return an opts object. The opts object should have at least one attribute *format_map* which is an ordered list of formats for the device.

set_plugboards(*plugboards*, *pb_func*)

provide the driver the current set of plugboards and a function to select a specific plugboard. This method is called immediately before add_books and sync_booklists.

pb_func is a callable with the following signature::

def pb_func(device_name, format, plugboards)

You give it the current device name (either the class name or DEVICE_PLUGBOARD_NAME), the format you are interested in (a ‚real' format or ‚device_db'), and the plugboards (you were given those by set_plugboards, the same place you got this method).

Vrací

None or a single plugboard instance.

set_driveinfo_name(*location_code*, *name*)

Set the device name in the driveinfo file to ‚name'. This setting will persist until the file is re-created or the name is changed again.

Non-disk devices should implement this method based on the location codes returned by the get_device_information() method.

prepare_addable_books(*paths*)

Given a list of paths, returns another list of paths. These paths point to addable versions of the books.

If there is an error preparing a book, then instead of a path, the position in the returned list for that book should be a three tuple: (original_path, the exception instance, traceback)

startup()

Called when calibre is starting the device. Do any initialization required. Note that multiple instances of the class can be instantiated, and thus __init__ can be called multiple times, but only one instance will have this method called. This method is called on the device thread, not the GUI thread.

shutdown()

Called when calibre is shutting down, either for good or in preparation to restart. Do any cleanup required. This method is called on the device thread, not the GUI thread.

get_device_uid()

Must return a unique id for the currently connected device (this is called immediately after a successful call to open()). You must implement this method if you set ASK_TO_ALLOW_CONNECT = True

ignore_connected_device(*uid*)

Should ignore the device identified by uid (the result of a call to get_device_uid()) in the future. You must implement this method if you set ASK_TO_ALLOW_CONNECT = True. Note that this function is called immediately after open(), so if open() caches some state, the driver should reset that state.

get_user_blacklisted_devices()

Return map of device uid to friendly name for all devices that the user has asked to be ignored.

set_user_blacklisted_devices(*devices*)

Set the list of device uids that should be ignored by this driver.

specialize_global_preferences(*device_prefs*)

Implement this method if your device wants to override a particular preference. You must ensure that all call sites that want a preference that can be overridden use device_prefs[‚something‘] instead of prefs[‚something‘]. Your method should call device_prefs.set_overrides(pref=val, pref=val, …). Currently used for: metadata management (prefs[‚manage_device_metadata‘])

set_library_info(*library_name*, *library_uuid*, *field_metadata*)

Implement this method if you want information about the current calibre library. This method is called at startup and when the calibre library changes while connected.

is_dynamically_controllable()

Called by the device manager when starting plugins. If this method returns a string, then a) it supports the device manager's dynamic control interface, and b) that name is to be used when talking to the plugin.

This method can be called on the GUI thread. A driver that implements this method must be thread safe.

start_plugin()

This method is called to start the plugin. The plugin should begin to accept device connections however it does that. If the plugin is already accepting connections, then do nothing.

This method can be called on the GUI thread. A driver that implements this method must be thread safe.

stop_plugin()

This method is called to stop the plugin. The plugin should no longer accept connections, and should cleanup behind itself. It is likely that this method should call shutdown. If the plugin is already not accepting connections, then do nothing.

This method can be called on the GUI thread. A driver that implements this method must be thread safe.

get_option(*opt_string*, *default=None*)

Return the value of the option indicated by opt_string. This method can be called when the plugin is not started. Return None if the option does not exist.

This method can be called on the GUI thread. A driver that implements this method must be thread safe.

set_option(*opt_string*, *opt_value*)

Set the value of the option indicated by opt_string. This method can be called when the plugin is not started.

This method can be called on the GUI thread. A driver that implements this method must be thread safe.

is_running()

Return True if the plugin is started, otherwise false

This method can be called on the GUI thread. A driver that implements this method must be thread safe.

synchronize_with_db(*db*, *book_id*, *book_metadata*, *first_call*)

Called during book matching when a book on the device is matched with a book in calibre's db. The method is responsible for synchronizing data from the device to calibre's db (if needed).

The method must return a two-value tuple. The first value is a set of calibre book ids changed if calibre's database was changed or None if the database was not changed. If the first value is an empty set then the metadata for the book on the device is updated with calibre's metadata and given back to the device, but no GUI refresh of that book is done. This is useful when the calibre data is correct but must be sent to the device.

The second value is itself a 2-value tuple. The first value in the tuple specifies whether a book format should be sent to the device. The intent is to permit verifying that the book on the device is the same as the book in calibre. This value must be None if no book is to be sent, otherwise return the base file name on the device (a string like foobar.epub). Be sure to include the extension in the name. The device subsystem will construct a send_books job for all books with not- None returned values. Note: other than to later retrieve the extension, the name is ignored in cases where the device uses a template to generate the file name, which most do. The second value in the returned tuple indicated whether the format is future-dated. Return True if it is, otherwise return False. calibre will display a dialog to the user listing all future dated books.

Extremely important: this method is called on the GUI thread. It must be threadsafe with respect to the device manager's thread.

book_id: the calibre id for the book in the database. book_metadata: the Metadata object for the book coming from the device. first_call: True if this is the first call during a sync, False otherwise

class calibre.devices.interface.**BookList**(*oncard*, *prefix*, *settings*)

Bases: `list`

A list of books. Each Book object must have the fields

1. title
2. authors
3. size (file size of the book)
4. datetime (a UTC time tuple)
5. path (path on the device to the book)
6. thumbnail (can be None) thumbnail is either a str/bytes object with the image data or it should have an attribute image_path that stores an absolute (platform native) path to the image
7. tags (a list of strings, can be empty).

supports_collections()

Return True if the device supports collections for this book list.

add_book(*book*, *replace_metadata*)

Add the book to the booklist. Intent is to maintain any device-internal metadata. Return True if booklists must be sync'ed

remove_book(*book*)

Remove a book from the booklist. Correct any device metadata at the same time

get_collections(*collection_attributes*)

Return a dictionary of collections created from collection_attributes. Each entry in the dictionary is of the form collection name:[list of books]

The list of books is sorted by book title, except for collections created from series, in which case series_index is used.

Parametry

collection_attributes – A list of attributes of the Book object

Zařízení založená na Velkokapacitním paměťovém zařízení USB

The base class for such devices is *calibre.devices.usbms.driver.USBMS* (stránka 279). This class in turn inherits some of its functionality from its bases, documented below. A typical basic USBMS based driver looks like this:

```
from calibre.devices.usbms.driver import USBMS

class PDNOVEL(USBMS):
    name = 'Pandigital Novel device interface'
    gui_name = 'PD Novel'
    description = _('Communicate with the Pandigital Novel')
    author = 'Kovid Goyal'
    supported_platforms = ['windows', 'linux', 'osx']
    FORMATS = ['epub', 'pdf']

    VENDOR_ID   = [0x18d1]
    PRODUCT_ID  = [0xb004]
    BCD         = [0x224]

    THUMBNAIL_HEIGHT = 144

    EBOOK_DIR_MAIN = 'eBooks'
    SUPPORTS_SUB_DIRS = False

    def upload_cover(self, path, filename, metadata):
        coverdata = getattr(metadata, 'thumbnail', None)
        if coverdata and coverdata[2]:
            with open('%s.jpg' % os.path.join(path, filename), 'wb') as coverfile:
                coverfile.write(coverdata[2])
```

class calibre.devices.usbms.device.**Device**(*plugin_path*)

Bases: DeviceConfig, *DevicePlugin* (stránka 268)

This class provides logic common to all drivers for devices that export themselves as USB Mass Storage devices. Provides implementations for mounting/ejecting of USBMS devices on all platforms.

VENDOR_ID = 0

VENDOR_ID can be either an integer, a list of integers or a dictionary If it is a dictionary, it must be a dictionary of dictionaries, of the form:

```
{
 integer_vendor_id : { product_id : [list of BCDs], ... },
 ...
}
```

PRODUCT_ID = 0

An integer or a list of integers

BCD = None

BCD can be either None to not distinguish between devices based on BCD, or it can be a list of the BCD numbers of all devices supported by this driver.

WINDOWS_MAIN_MEM = None

String identifying the main memory of the device in the Windows PnP id strings This can be None, string, list of strings or compiled regex

WINDOWS_CARD_A_MEM = None

String identifying the first card of the device in the Windows PnP id strings This can be None, string, list of strings or compiled regex

WINDOWS_CARD_B_MEM = None

String identifying the second card of the device in the Windows PnP id strings This can be None, string, list of strings or compiled regex

OSX_MAIN_MEM_VOL_PAT = None

Used by the new driver detection to disambiguate main memory from storage cards. Should be a regular expression that matches the main memory mount point assigned by macOS

MAX_PATH_LEN = 250

The maximum length of paths created on the device

NEWS_IN_FOLDER = True

Put news in its own folder

classmethod model_metadata() → tuple[ModelMetadata, ...]

Metadata about all the actual device models this driver supports

reset(*key='-1'*, *log_packets=False*, *report_progress=None*, *detected_device=None*)

Parametry

- **key** – The key to unlock the device
- **log_packets** – If true the packet stream to/from the device is logged
- **report_progress** – Function that is called with a % progress (number between 0 and 100) for various tasks. If it is called with -1 that means that the task does not have any progress information
- **detected_device** – Device information from the device scanner

set_progress_reporter(*report_progress*)

Set a function to report progress information.

Parametry

report_progress – Function that is called with a % progress (number between 0 and 100) for various tasks. If it is called with -1 that means that the task does not have any progress information

card_prefix(*end_session=True*)

Return a 2 element list of the prefix to paths on the cards. If no card is present None is set for the card's prefix. E.G. (‚/place', ‚/place2') (None, ‚place2') (‚place', None) (None, None)

total_space(*end_session=True*)

Get total space available on the mountpoints:

1. Hlavní paměť
2. Memory Card A
3. Memory Card B

Vrací

A 3 element list with total space in bytes of (1, 2, 3). If a particular device doesn't have any of these locations it should return 0.

free_space(*end_session=True*)

Get free space available on the mountpoints:

1. Hlavní paměť
2. Karta A
3. Karta B

Vrací

A 3 element list with free space in bytes of (1, 2, 3). If a particular device doesn't have any of these locations it should return -1.

windows_sort_drives(*drives*)

Called to disambiguate main memory and storage card for devices that do not distinguish between them on the basis of *WINDOWS_CARD_NAME*. For example: The EB600

can_handle_windows(*usbdevice*, *debug=False*)

Optional method to perform further checks on a device to see if this driver is capable of handling it. If it is not it should return False. This method is only called after the vendor, product ids and the bcd have matched, so it can do some relatively time intensive checks. The default implementation returns True. This method is called only on Windows. See also `can_handle()`.

Note that for devices based on USBMS this method by default delegates to `can_handle()`. So you only need to override `can_handle()` in your subclass of USBMS.

Parametry

usbdevice – A usbdevice as returned by `calibre.devices.winusb.scan_usb_devices()`

open(*connected_device*, *library_uuid*)

Perform any device specific initialization. Called after the device is detected but before any other functions that communicate with the device. For example: For devices that present themselves as USB Mass storage devices, this method would be responsible for mounting the device or if the device has been automounted, for finding out where it has been mounted. The method *`calibre.devices.usbms.device.Device.open()`* (stránka 278) has an implementation of this function that should serve as a good example for USB Mass storage devices.

This method can raise an OpenFeedback exception to display a message to the user.

Parametry

- **connected_device** – The device that we are trying to open. It is a tuple of (vendor id, product id, bcd, manufacturer name, product name, device serial number). However, some devices have no serial number and on Windows only the first three fields are present, the rest are None.
- **library_uuid** – The UUID of the current calibre library. Can be None if there is no library (for example when used from the command line).

eject()

Un-mount / eject the device from the OS. This does not check if there are pending GUI jobs that need to communicate with the device.

NOTE: That this method may not be called on the same thread as the rest of the device methods.

post_yank_cleanup()

Called if the user yanks the device without ejecting it first.

sanitize_callback(*path*)

Callback to allow individual device drivers to override the path sanitization used by `create_upload_path()`.

filename_callback(*default*, *mi*)

Callback to allow drivers to change the default file name set by `create_upload_path()`.

sanitize_path_components(*components*)

Perform any device specific sanitization on the path components for files to be uploaded to the device

get_annotations(*path_map*)

Resolve path_map to annotation_map of files found on the device

add_annotation_to_library(*db*, *db_id*, *annotation*)

Add an annotation to the calibre library

class calibre.devices.usbms.cli.**CLI**

class calibre.devices.usbms.driver.**USBMS**(*plugin_path*)

Bases: *CLI* (stránka 279), *Device* (stránka 276)

The base class for all USBMS devices. Implements the logic for sending/getting/updating metadata/caching metadata/etc.

description = 'Komunikace se čtečkou e-knih.'

A short string describing what this plugin does

author = 'John Schember'

Autor doplňku

supported_platforms = ['windows', 'osx', 'linux']

List of platforms this plugin works on. For example: `['windows', 'osx', 'linux']`

booklist_class

alias of `BookList`

book_class

alias of `Book`

FORMATS = []

Ordered list of supported formats

CAN_SET_METADATA = []

Whether the metadata on books can be set via the GUI.

get_device_information(*end_session=True*)

Ask device for device information. See L{DeviceInfoQuery}.

Vrací

(device name, device version, software version on device, MIME type) The tuple can optionally have a fifth element, which is a drive information dictionary. See usbms.driver for an example.

set_driveinfo_name(*location_code*, *name*)

Set the device name in the driveinfo file to ‚name'. This setting will persist until the file is re-created or the name is changed again.

Non-disk devices should implement this method based on the location codes returned by the get_device_information() method.

books(*oncard=None*, *end_session=True*)

Return a list of e-books on the device.

Parametry

oncard – If ‚carda' or ‚cardb' return a list of e-books on the specific storage card, otherwise return list of e-books in main memory of device. If a card is specified and no books are on the card return empty list.

Vrací

A BookList.

upload_books(*files*, *names*, *on_card=None*, *end_session=True*, *metadata=None*)

Upload a list of books to the device. If a file already exists on the device, it should be replaced. This method should raise a `FreeSpaceError` if there is not enough free space on the device. The text of the FreeSpaceError must contain the word „card" if `on_card` is not None otherwise it must contain the word „memory".

Parametry

- **files** – A list of paths
- **names** – A list of file names that the books should have once uploaded to the device. len(names) == len(files)
- **metadata** – If not None, it is a list of `Metadata` objects. The idea is to use the metadata to determine where on the device to put the book. len(metadata) == len(files). Apart from the regular cover (path to cover), there may also be a thumbnail attribute, which should be used in preference. The thumbnail attribute is of the form (width, height, cover_data as jpeg).

Vrací

A list of 3-element tuples. The list is meant to be passed to *add_books_to_metadata()* (stránka 280).

upload_cover(*path*, *filename*, *metadata*, *filepath*)

Upload book cover to the device. Default implementation does nothing.

Parametry

- **path** – The full path to the folder where the associated book is located.
- **filename** – The name of the book file without the extension.
- **metadata** – metadata belonging to the book. Use metadata.thumbnail for cover
- **filepath** – The full path to the e-book file

add_books_to_metadata(*locations*, *metadata*, *booklists*)

Add locations to the booklists. This function must not communicate with the device.

Parametry

- **locations** – Result of a call to L{upload_books}
- **metadata** – List of `Metadata` objects, same as for *upload_books()* (stránka 280).
- **booklists** – A tuple containing the result of calls to (`books(oncard=None)()`, `books(oncard='carda')()`, :meth`books(oncard='cardb')`).

delete_books(*paths*, *end_session=True*)

Delete books at paths on device.

remove_books_from_metadata(*paths*, *booklists*)

Remove books from the metadata list. This function must not communicate with the device.

Parametry

- **paths** – paths to books on the device.
- **booklists** – A tuple containing the result of calls to (`books(oncard=None)()`, `books(oncard='carda')()`, :meth`books(oncard='cardb')`).

sync_booklists(*booklists*, *end_session=True*)

Update metadata on device.

Parametry

booklists – A tuple containing the result of calls to (`books(oncard=None)()`, `books(oncard='carda')()`, :meth`books(oncard='cardb')`).

classmethod normalize_path(*path*)

Return path with platform native path separators

12.1.8 User interface actions

If you are adding your own plugin in a ZIP file, you should subclass both InterfaceActionBase and InterfaceAction. The `load_actual_plugin()` method of your InterfaceActionBase subclass must return an instantiated object of your InterfaceBase subclass.

class `calibre.gui2.actions.`**InterfaceAction**(*parent*, *site_customization*)

Bases: `QObject`

A plugin representing an „action“ that can be taken in the graphical user interface. All the items in the toolbar and context menus are implemented by these plugins.

Note that this class is the base class for these plugins, however, to integrate the plugin with calibre's plugin system, you have to make a wrapper class that references the actual plugin. See the `calibre.customize.builtins` module for examples.

If two *InterfaceAction* (stránka 281) objects have the same name, the one with higher priority takes precedence.

Sub-classes should implement the *genesis()* (stránka 284), *library_changed()* (stránka 284), *location_selected()* (stránka 284), *shutting_down()* (stránka 284), *initialization_complete()* (stránka 284) and *tag_browser_context_action()* (stránka 284) methods.

Once initialized, this plugin has access to the main calibre GUI via the `gui` member. You can access other plugins by name, for example:

```
self.gui.iactions['Save To Disk']
```

To access the actual plugin, use the `interface_action_base_plugin` attribute, this attribute only becomes available after the plugin has been initialized. Useful if you want to use methods from the plugin class like do_user_config().

The QAction specified by *action_spec* (stránka 282) is automatically create and made available as `self.qaction`.

name = 'Implement me'

The plugin name. If two plugins with the same name are present, the one with higher priority takes precedence.

priority = 1

The plugin priority. If two plugins with the same name are present, the one with higher priority takes precedence.

popup_type = 1

The menu popup type for when this plugin is added to a toolbar

auto_repeat = False

Whether this action should be auto repeated when its shortcut key is held down.

action_spec = ('text', 'icon', None, None)

Of the form: (text, icon_path, tooltip, keyboard shortcut). icon, tooltip and keyboard shortcut can be None. keyboard shortcut must be either a string, None or tuple of shortcuts. If None, a keyboard shortcut corresponding to the action is not registered. If you pass an empty tuple, then the shortcut is registered with no default key binding.

action_shortcut_name = None

If not None, used for the name displayed to the user when customizing the keyboard shortcuts for the above action spec instead of action_spec[0]

action_add_menu = False

If True, a menu is automatically created and added to self.qaction

action_menu_clone_qaction = False

If True, a clone of self.qaction is added to the menu of self.qaction If you want the text of this action to be different from that of self.qaction, set this variable to the new text

dont_add_to = frozenset({})

Set of locations to which this action must not be added. See `all_locations` for a list of possible locations

dont_remove_from = frozenset({})

Set of locations from which this action must not be removed. See `all_locations` for a list of possible locations

action_type = 'global'

Type of action ‚current' means acts on the current view ‚global' means an action that does not act on the current view, but rather on calibre as a whole

accepts_drops = False

If True, then this InterfaceAction will have the opportunity to interact with drag and drop events. See the methods, *accept_enter_event()* (stránka 282), :meth`:accept_drag_move_event`, *drop_event()* (stránka 283) for details.

accept_enter_event(*event*, *mime_data*)

This method should return True iff this interface action is capable of handling the drag event. Do not call accept/ignore on the event, that will be taken care of by the calibre UI.

accept_drag_move_event(*event*, *mime_data*)

This method should return True iff this interface action is capable of handling the drag event. Do not call accept/ignore on the event, that will be taken care of by the calibre UI.

drop_event(*event*, *mime_data*)

This method should perform some useful action and return True iff this interface action is capable of handling the drop event. Do not call accept/ignore on the event, that will be taken care of by the calibre UI. You should not perform blocking/long operations in this function. Instead emit a signal or use QTimer.singleShot and return quickly. See the builtin actions for examples.

create_menu_action(*menu*, *unique_name*, *text*, *icon=None*, *shortcut=None*, *description=None*, *triggered=None*, *shortcut_name=None*, *persist_shortcut=False*)

Convenience method to easily add actions to a QMenu. Returns the created QAction. This action has one extra attribute calibre_shortcut_unique_name which if not None refers to the unique name under which this action is registered with the keyboard manager.

Parametry

- **menu** – The QMenu the newly created action will be added to
- **unique_name** – A unique name for this action, this must be globally unique, so make it as descriptive as possible. If in doubt, add an UUID to it.
- **text** – The text of the action.
- **icon** – Either a QIcon or a file name. The file name is passed to the QIcon.ic() builtin, so you do not need to pass the full path to the images folder.
- **shortcut** – A string, a list of strings, None or False. If False, no keyboard shortcut is registered for this action. If None, a keyboard shortcut with no default keybinding is registered. String and list of strings register a shortcut with default keybinding as specified.
- **description** – A description for this action. Used to set tooltips.
- **triggered** – A callable which is connected to the triggered signal of the created action.
- **shortcut_name** – The text displayed to the user when customizing the keyboard shortcuts for this action. By default it is set to the value of `text`.
- **persist_shortcut** – Shortcuts for actions that don't always appear, or are library dependent, may disappear when other keyboard shortcuts are edited unless `` `persist_shortcut` `` is set True.

load_resources(*names*)

If this plugin comes in a ZIP file (user added plugin), this method will allow you to load resources from the ZIP file.

For example to load an image:

```
pixmap = QPixmap()
pixmap.loadFromData(tuple(self.load_resources(['images/icon.png']).
↪values())[0])
icon = QIcon(pixmap)
```

Parametry

names – List of paths to resources in the ZIP file using / as separator

Vrací

A dictionary of the form `{name : file_contents}`. Any names that were not found in the ZIP file will not be present in the dictionary.

genesis()

Setup this plugin. Only called once during initialization. self.gui is available. The action specified by *action_spec* (stránka 282) is available as `self.qaction`.

location_selected(*loc*)

Called whenever the book list being displayed in calibre changes. Currently values for loc are: `library, main, card and cardb`.

This method should enable/disable this action and its sub actions as appropriate for the location.

library_about_to_change(*olddb*, *db*)

Called whenever the current library is changed.

Parametry

- **olddb** – The LibraryDatabase corresponding to the previous library.
- **db** – The LibraryDatabase corresponding to the new library.

library_changed(*db*)

Called whenever the current library is changed.

Parametry

db – The LibraryDatabase corresponding to the current library.

gui_layout_complete()

Called once per action when the layout of the main GUI is completed. If your action needs to make changes to the layout, they should be done here, rather than in *initialization_complete()* (stránka 284).

initialization_complete()

Called once per action when the initialization of the main GUI is completed.

tag_browser_context_action(*index*)

Called when displaying the context menu in the Tag browser. `index` is the QModelIndex that points to the Tag browser item that was right clicked. Test it for validity with index.valid() and get the underlying TagTreeItem object with index.data(Qt.ItemDataRole.UserRole). Any action objects yielded by this method will be added to the context menu.

shutting_down()

Called once per plugin when the main GUI is in the process of shutting down. Release any used resources, but try not to block the shutdown for long periods of time.

class calibre.customize.**InterfaceActionBase**(**args*, ***kwargs*)

Bases: *Plugin* (stránka 256)

supported_platforms = ['windows', 'osx', 'linux']

List of platforms this plugin works on. For example: `['windows', 'osx', 'linux']`

author = 'Kovid Goyal'

Autor doplňku

type = 'Akce uživatelského rozhraní'

The type of this plugin. Used for categorizing plugins in the GUI

can_be_disabled = False

If False, the user will not be able to disable this plugin. Use with care.

load_actual_plugin(*gui*)

This method must return the actual interface action plugin object.

12.1.9 Moduly předvoleb

class calibre.customize.**PreferencesPlugin**(*plugin_path*)

Bases: *Plugin* (stránka 256)

A plugin representing a widget displayed in the Preferences dialog.

This plugin has only one important method *create_widget()* (stránka 285). The various fields of the plugin control how it is categorized in the UI.

supported_platforms = ['windows', 'osx', 'linux']

List of platforms this plugin works on. For example: `['windows', 'osx', 'linux']`

author = 'Kovid Goyal'

Autor doplňku

type = 'Předvolby'

The type of this plugin. Used for categorizing plugins in the GUI

can_be_disabled = False

If False, the user will not be able to disable this plugin. Use with care.

config_widget = None

Import path to module that contains a class named ConfigWidget which implements the ConfigWidgetInterface. Used by *create_widget()* (stránka 285).

category_order = 100

Where in the list of categories the *category* (stránka 285) of this plugin should be.

name_order = 100

Where in the list of names in a category, the *gui_name* (stránka 285) of this plugin should be

category = None

The category this plugin should be in

gui_category = None

The category name displayed to the user for this plugin

gui_name = None

The name displayed to the user for this plugin

icon = None

The icon for this plugin, should be an absolute path

description = None

The description used for tooltips and the like

create_widget(*parent=None*)

Create and return the actual Qt widget used for setting this group of preferences. The widget must implement the *calibre.gui2.preferences.ConfigWidgetInterface* (stránka 285).

The default implementation uses *config_widget* (stránka 285) to instantiate the widget.

class calibre.gui2.preferences.**ConfigWidgetInterface**

This class defines the interface that all widgets displayed in the Preferences dialog must implement. See *ConfigWidgetBase* (stránka 286) for a base class that implements this interface and defines various convenience methods as well.

changed_signal = None

This signal must be emitted whenever the user changes a value in this widget

supports_restoring_to_defaults = True

Set to True iff the `restore_to_defaults()` method is implemented.

restore_defaults_desc = 'Obnovit nastavení na výchozí hodnoty. Musíte kliknout na Použít pro skutečné uložení výchozího nastavení.'

The tooltip for the „Restore defaults“ button

restart_critical = False

If True the Preferences dialog will not allow the user to set any more preferences. Only has effect if *commit()* (stránka 286) returns True.

genesis(*gui*)

Called once before the widget is displayed, should perform any necessary setup.

Parametry

gui – The main calibre graphical user interface

initialize()

Should set all config values to their initial values (the values stored in the config files). A „return“ statement is optional. Return False if the dialog is not to be shown.

restore_defaults()

Should set all config values to their defaults.

commit()

Save any changed settings. Return True if the changes require a restart, False otherwise. Raise an `AbortCommit` exception to indicate that an error occurred. You are responsible for giving the user feedback about what the error is and how to correct it.

refresh_gui(*gui*)

Called once after this widget is committed. Responsible for causing the gui to reread any changed settings. Note that by default the GUI re-initializes various elements anyway, so most widgets won't need to use this method.

initial_tab_changed()

Called if the initially displayed tab is changed before the widget is shown, but after it is initialized.

class calibre.gui2.preferences.**ConfigWidgetBase**(*parent=None*)

Base class that contains code to easily add standard config widgets like checkboxes, combo boxes, text fields and so on. See the *register()* (stránka 286) method.

This class automatically handles change notification, resetting to default, translation between gui objects and config objects, etc. for registered settings.

If your config widget inherits from this class but includes setting that are not registered, you should override the *ConfigWidgetInterface* (stránka 285) methods and call the base class methods inside the overrides.

changed_signal

This signal must be emitted whenever the user changes a value in this widget

supports_restoring_to_defaults = True

Set to True iff the `restore_to_defaults()` method is implemented.

restart_critical = False

If True the Preferences dialog will not allow the user to set any more preferences. Only has effect if *commit()* (stránka 287) returns True.

register(*name*, *config_obj*, *gui_name=None*, *choices=None*, *restart_required=False*, *empty_string_is_None=True*, *setting=<class 'calibre.gui2.preferences.Setting'>*)

Register a setting.

Parametry

- **name** – The setting name
- **config_obj** – The config object that reads/writes the setting
- **gui_name** – The name of the GUI object that presents an interface to change the setting. By default it is assumed to be `'opt_' + name`.
- **choices** – If this setting is a multiple choice (combobox) based setting, the list of choices. The list is a list of two element tuples of the form: `[(gui name, value), ...]`
- **setting** – The class responsible for managing this setting. The default class handles almost all cases, so this param is rarely used.

initialize()

Should set all config values to their initial values (the values stored in the config files). A „return“ statement is optional. Return False if the dialog is not to be shown.

commit(**args*)

Save any changed settings. Return True if the changes require a restart, False otherwise. Raise an `AbortCommit` exception to indicate that an error occurred. You are responsible for giving the user feedback about what the error is and how to correct it.

restore_defaults(**args*)

Should set all config values to their defaults.

12.2 Proměnné prostředí

- `CALIBRE_CONFIG_DIRECTORY` – Nastaví složku, kde jsou uložené konfigurační soubory.
- `CALIBRE_TEMP_DIR` – Nastaví dočasnou složku používanou Calibre.
- `CALIBRE_CACHE_DIRECTORY` – Nastaví složku, kterou Calibre používá pro ukládání trvalých dat do mezipaměti mezi relacemi
- `CALIBRE_OVERRIDE_DATABASE_PATH` – Umožní vám zadat úplnou cestu k metadata.db. Pomocí této proměnné můžete mít metadata.db v jiném umístění než ve složce knihovny. Užitečné, pokud je vaše složka knihovny na síťové jednotce, která nepodporuje uzamykání souboru.
- `CALIBRE_DEVELOP_FROM` – použito pro spuštění z vývojového prostředí Calibre. Podívejte se na *Nastavení vývojového prostředí Calibre* (stránka 343).
- `CALIBRE_OVERRIDE_LANG` – použito pro vynucení jazyka použitého rozhraním (kód jazyka ISO 639).
- `CALIBRE_TEST_TRANSLATION` - použito pro testování souboru překladu .po (měla by to být cesta k souboru .po)
- `CALIBRE_NO_NATIVE_FILEDIALOGS` – způsobí, že Calibre nepoužívá nativní dialogová okna souboru pro výběr souborů nebo složek.
- `CALIBRE_NO_NATIVE_MENUBAR` – způsobí, aby Calibre nevytvářelo nativní (globální) nabídku v Ubuntu Unity a podobných desktopových prostředí Linux. Nabídka je namísto toho umístěna uvnitř okna, jak je obvyklé.
- `CALIBRE_USE_SYSTEM_THEME` - ve výchozím nastavení na Linuxu používá Calibre svůj vlastní vestavěný styl Qt. Tím je zabráněno pádům a zablokováním způsobeným nekompatibilitou mezi verzí Qt Calibre a systému. Nevýhodou je, že Calibre nemusí odpovídat vzhledu a chování systému. Pokud nastavíte tuto proměnnou prostředí na Linuxu, způsobí to, že Calibre bude používat motiv systému - dejte pozor na pády a zablokování.

- `CALIBRE_SHOW_DEPRECATION_WARNINGS` – způsobí, že Calibre vytiskne varování o neschválení do stdout. Užitečné pro vývojáře Calibre.
- `CALIBRE_NO_DEFAULT_PROGRAMS` - zabrání Calibre, aby automaticky registrovalo typy souborů, se kterými je schopné pracovat pomocí Windows.
- `CALIBRE_USE_SYSTEM_CERTIFICATES` - make calibre use the system certificate store for SSL certificate verification instead of its own certificate store on Windows and macOS.
- `CALIBRE_NO_ICONS_IN_MENUS` - Disable icons in menus
- `QT_QPA_PLATFORM` - Na Linuxu toto nastavte na `wayland` pro vynucení, aby Calibre používalo Wayland a `xcb` pro vynucení použití X11.
- `SYSFS_PATH` – Použijte, pokud je sysfs připojeno někde jinde než /sys.
- `http_proxy`, `https_proxy` – použito v Linuxu pro zadání proxy HTTP.

Podívejte se na Jak nastavit proměnné prostředí ve Windows[115]. Pokud jste na macOS, můžete nastavit proměnné prostředí vytvořením `~/Library/Preferences/calibre/macos-env.txt` a uložením proměnných prostředí do něj, každou na jeden řádek, na příklad:

```
CALIBRE_DEVELOP_FROM=$HOME/calibre-src/src
CALIBRE_NO_NATIVE_FILEDIALOGS=1
CALIBRE_CONFIG_DIRECTORY=~/.config/calibre
```

12.3 Vylepšení

Vylepšení jsou malé změny, které můžete zadat pro určení různých aspektů chování Calibre. Můžete je změnit v Předvolby->Rozšířené->Vylepšení. Výchozí hodnoty pro vylepšení jsou zopakovány níže

```
#!/usr/bin/env python
# vim:fileencoding=UTF-8:ts=4:sw=4:sta:et:sts=4:ai
# License: GPLv3 Copyright: 2010, Kovid Goyal <kovid at kovidgoyal.net>


# Contains various tweaks that affect calibre behavior. Only edit this file if
# you know what you are doing. If you delete this file, it will be recreated from
# defaults.

#: Auto increment series index
# The algorithm used to assign a book added to an existing series a series number.
# New series numbers assigned using this tweak are always integer values, except
# if a constant non-integer is specified.
# Possible values are:
#   next - First available integer larger than the largest existing number
#   first_free - First available integer larger than 0
#   next_free - First available integer larger than the smallest existing number
#   last_free - First available integer smaller than the largest existing number.␣
↪Return largest existing + 1 if no free number is found
#   const - Assign the number 1 always
#   no_change - Do not change the series index
#   a number - Assign that number always. The number is not in quotes. Note that 0.0␣
↪can be used here.
```

(continues on next page)

[115] https://www.computerhope.com/issues/ch000549.htm

(pokračujte na předchozí stránce)

```
# Examples:
#   series_index_auto_increment = 'next'
#   series_index_auto_increment = 'next_free'
#   series_index_auto_increment = 16.5
#
# Set the use_series_auto_increment_tweak_when_importing tweak to True to
# use the above values when importing/adding books. If this tweak is set to
# False (the default) then the series number will be set to 1 if it is not
# explicitly set during the import. If set to True, then the
# series index will be set according to the series_index_auto_increment setting.
# Note that the use_series_auto_increment_tweak_when_importing tweak is used
# only when a value is not provided during import. If the importing regular
# expression produces a value for series_index, or if you are reading metadata
# from books and the import plugin produces a value, then that value will
# be used irrespective of the setting of the tweak.
series_index_auto_increment = 'next'
use_series_auto_increment_tweak_when_importing = False

#: Add separator after completing an author name
# Set this if the completion separator should be appended to the end of the
# completed text to automatically begin a new completion operation for authors.
# It can be either True or False
authors_completer_append_separator = False

#: Author sort name algorithm
# The algorithm used to copy author to author_sort.
# Possible values are:
#  invert: use "fn ln" -> "ln, fn"
#  copy  : copy author to author_sort without modification
#  comma : use 'copy' if there is a ',' in the name, otherwise use 'invert'
#  nocomma : "fn ln" -> "ln fn" (without the comma)
# When this tweak is changed, the author_sort values stored with each author
# must be recomputed by right-clicking on an author in the left-hand tags
# panel, selecting 'Manage authors', and pressing
# 'Recalculate all author sort values'.
#
# The author_name_suffixes are words that are ignored when they occur at the
# end of an author name. The case of the suffix is ignored and trailing
# periods are automatically handled.
#
# The same is true for author_name_prefixes.
#
# The author_name_copywords are a set of words which, if they occur in an
# author name, cause the automatically generated author sort string to be
# identical to the author's name. This means that the sort for a string like
# "Acme Inc." will be "Acme Inc." instead of "Inc., Acme".
#
# If author_use_surname_prefixes is enabled, any of the words in
# author_surname_prefixes will be treated as a prefix to the surname, if they
# occur before the surname. So for example, "John von Neumann" would be sorted
# as "von Neumann, John" and not "Neumann, John von".
author_sort_copy_method = 'comma'
```

(continues on next page)

(pokračujte na předchozí stránce)

```
author_name_suffixes = ('Jr', 'Sr', 'Inc', 'Ph.D', 'Phd',
                        'MD', 'M.D', 'I', 'II', 'III', 'IV',
                        'Junior', 'Senior')
author_name_prefixes = ('Mr', 'Mrs', 'Ms', 'Dr', 'Prof')
author_name_copywords = (
    'Agency', 'Corporation', 'Company', 'Co.', 'Council',
    'Committee', 'Inc.', 'Institute', 'National', 'Society', 'Club', 'Team',
    'Software', 'Games', 'Entertainment', 'Media', 'Studios',
)
author_use_surname_prefixes = False
author_surname_prefixes = ('da', 'de', 'di', 'la', 'le', 'van', 'von')

#: Splitting multiple author names
# By default, calibre splits a string containing multiple author names on
# ampersands and the words "and" and "with". You can customize the splitting
# by changing the regular expression below. Strings are split on whatever the
# specified regular expression matches, in addition to ampersands.
# Default: r'(?i),?\s+(and|with)\s+'
authors_split_regex = r'(?i),?\s+(and|with)\s+'

#: Use author_sort and/or series_sort for names in Tag browser
# Set which author or series field is used to display as the item name in the
# Tag browser (the list of authors, series, publishers etc on the left hand
# side). The choices for author are 'author' and 'author_sort'. The choices for
# series are 'series' and 'series_sort'. This tweak affects only what is
# displayed under the category in the Tag browser and Content server. Please
# note that if you set this to …_sort, it is possible to see duplicate names in
# the list because although it is guaranteed that author and series names are
# unique, there is no such guarantee for their sort values. Showing duplicates
# won't break anything but it could lead to some confusion. The tooltip for an
# item will show the item's name no matter how you set this tweak.
# Examples:
#   categories_use_field_for_author_name = 'author'
#   categories_use_field_for_author_name = 'author_sort'
#   categories_use_field_for_series_name = 'series'
#   categories_use_field_for_series_name = 'series_sort'
categories_use_field_for_author_name = 'author'
categories_use_field_for_series_name = 'series'

#: Control partitioning of Tag browser
# When partitioning the Tag browser, the format of the subcategory label is
# controlled by a template: categories_collapsed_name_template if sorting by
# name, categories_collapsed_rating_template if sorting by average rating, and
# categories_collapsed_popularity_template if sorting by popularity. There are
# two variables available to the template: first and last. The variable 'first'
# is the initial item in the subcategory, and the variable 'last' is the final
# item in the subcategory. Both variables are 'objects'; they each have multiple
# values that are obtained by using a suffix. For example, first.name for an
# author category will be the name of the author. The sub-values available are:
#  name: the printable name of the item
#  count: the number of books that references this item
#  avg_rating: the average rating of all the books referencing this item
```

(continues on next page)

(pokračujte na předchozí stránce)

```
#  sort: the sort value. For authors, this is the author_sort for that author
#  category: the category (e.g., authors, series) that the item is in.
# Note that the "r'" in front of the { is necessary if there are backslashes
# (\ characters) in the template. It doesn't hurt anything to leave it there
# even if there aren't any backslashes.
categories_collapsed_name_template = r'{first.sort:shorten(4,,0)} - {last.
↪sort:shorten(4,,0)}'
categories_collapsed_rating_template = r'{first.avg_rating:4.2f:ifempty(0)} - {last.
↪avg_rating:4.2f:ifempty(0)}'
categories_collapsed_popularity_template = r'{first.count:d} - {last.count:d}'

#: Specify columns to sort the booklist by on startup
# Provide a set of columns to be sorted on when calibre starts.
# The argument is None if saved sort history is to be used
# otherwise it is a list of column,order pairs. Column is the
# lookup/search name, found using the tooltip for the column
# Order is 0 for ascending, 1 for descending.
# For example, set it to [('authors',0),('title',0)] to sort by
# title within authors.
sort_columns_at_startup = None

#: Control how dates are displayed
# Format to be used for publication date and the timestamp (date).
#  A string controlling how the publication date is displayed in the GUI
#  d     the day as number without a leading zero (1 to 31)
#  dd    the day as number with a leading zero (01 to 31)
#  ddd   the abbreviated localized day name (e.g. 'Mon' to 'Sun')
#  dddd  the long localized day name (e.g. 'Monday' to 'Sunday')
#  M     the month as number without a leading zero (1-12)
#  MM    the month as number with a leading zero (01-12)
#  MMM   the abbreviated localized month name (e.g. 'Jan' to 'Dec')
#  MMMM  the long localized month name (e.g. 'January' to 'December')
#  yy    the year as two digit number (00-99)
#  yyyy  the year as four digit number
#  h     the hours without a leading 0 (0 to 11 or 0 to 23, depending on am/pm)
#  hh    the hours with a leading 0 (00 to 11 or 00 to 23, depending on am/pm)
#  m     the minutes without a leading 0 (0 to 59)
#  mm    the minutes with a leading 0 (00 to 59)
#  s     the seconds without a leading 0 (0 to 59)
#  ss    the seconds with a leading 0 (00 to 59)
#  ap    use a 12-hour clock instead of a 24-hour clock, with "ap" replaced by the␣
↪lowercase localized string for am or pm
#  AP    use a 12-hour clock instead of a 24-hour clock, with "AP" replaced by the␣
↪uppercase localized string for AM or PM
#  aP    use a 12-hour clock instead of a 24-hour clock, with "aP" replaced by the␣
↪localized string for am or pm
#  Ap    use a 12-hour clock instead of a 24-hour clock, with "Ap" replaced by the␣
↪localized string for AM or PM
#  iso   the date with time and timezone. Must be the only format present
#  For example, given the date of 9 Jan 2010, the following formats show
#  MMM yyyy ==> Jan 2010    yyyy ==> 2010       dd MMM yyyy ==> 09 Jan 2010
#  MM/yyyy ==> 01/2010      d/M/yy ==> 9/1/10   yy ==> 10
```

(continues on next page)

(pokračujte na předchozí stránce)

```
#
# publication default if not set: MMM yyyy
# timestamp default if not set: dd MMM yyyy
# last_modified_display_format if not set: dd MMM yyyy
gui_pubdate_display_format = 'MMM yyyy'
gui_timestamp_display_format = 'dd MMM yyyy'
gui_last_modified_display_format = 'dd MMM yyyy'

#: Control sorting of titles and series in the library display
# Control title and series sorting in the library view. If set to
# 'library_order', the title sort field will be used instead of the title.
# Unless you have manually edited the title sort field, leading articles such as
# The and A will be ignored. If set to 'strictly_alphabetic', the titles will be
# sorted as-is (sort by title instead of title sort). For example, with
# library_order, The Client will sort under 'C'. With strictly_alphabetic, the
# book will sort under 'T'.
# This flag affects calibre's library display. It has no effect on devices. In
# addition, titles for books added before changing the flag will retain their
# order until the title is edited. Editing a title and hitting Enter
# without changing anything is sufficient to change the sort. Or you can use
# the 'Update title sort' action in the Bulk metadata edit dialog to update
# it for many books at once.
title_series_sorting = 'library_order'

#: Control formatting of title and series when used in templates
# Control how title and series names are formatted when saving to disk/sending
# to device. The behavior depends on the field being processed. If processing
# title, then if this tweak is set to 'library_order', the title will be
# replaced with title_sort. If it is set to 'strictly_alphabetic', then the
# title will not be changed. If processing series, then if set to
# 'library_order', articles such as 'The' and 'An' will be moved to the end. If
# set to 'strictly_alphabetic', the series will be sent without change.
# For example, if the tweak is set to library_order, "The Lord of the Rings"
# will become "Lord of the Rings, The". If the tweak is set to
# strictly_alphabetic, it would remain "The Lord of the Rings". Note that the
# formatter function raw_field will return the base value for title and
# series regardless of the setting of this tweak.
save_template_title_series_sorting = 'library_order'

#: Set the list of words considered to be "articles" for sort strings
# Set the list of words that are to be considered 'articles' when computing the
# title sort strings. The articles differ by language. By default, calibre uses
# a combination of articles from English and whatever language the calibre user
# interface is set to. In addition, in some contexts where the book language is
# available, the language of the book is used. You can change the list of
# articles for a given language or add a new language by editing
# per_language_title_sort_articles. To tell calibre to use a language other
# than the user interface language, set, default_language_for_title_sort. For
# example, to use German, set it to 'deu'. A value of None means the user
# interface language is used. The setting title_sort_articles is ignored
# (present only for legacy reasons).
per_language_title_sort_articles = {
```

(continues on next page)

(pokračujte na předchozí stránce)

```
        # English
        'eng': (r'A\s+', r'The\s+', r'An\s+'),
        # Esperanto
        'epo': (r'La\s+', r"L'", 'L´'),
        # Spanish
        'spa': (r'El\s+', r'La\s+', r'Lo\s+', r'Los\s+', r'Las\s+', r'Un\s+',
                r'Una\s+', r'Unos\s+', r'Unas\s+'),
        # French
        'fra': (r'Le\s+', r'La\s+', r"L'", r'L´', r'L’', r'Les\s+', r'Un\s+', r'Une\s+
↪',
                r'Des\s+', r'De\s+La\s+', r'De\s+', r"D'", r'D´', r'D’'),
        # Polish
        'pol': (),
        # Italian
        'ita': (r'Lo\s+', r'Il\s+', r"L'", r'L´', r'La\s+', r'Gli\s+',
                r'I\s+', r'Le\s+', r'Uno\s+', r'Un\s+', r'Una\s+', "rUn'",
                r'Un´', r'Dei\s+', r'Degli\s+', r'Delle\s+', r'Del\s+',
                r'Della\s+', r'Dello\s+', r"Dell'", r'Dell´'),
        # Portuguese
        'por': (r'A\s+', r'O\s+', r'Os\s+', r'As\s+', r'Um\s+', r'Uns\s+',
                r'Uma\s+', r'Umas\s+'),
        # Romanian
        'ron': (r'Un\s+', r'O\s+', r'Nişte\s+'),
        # German
        'deu': (r'Der\s+', r'Die\s+', r'Das\s+', r'Den\s+', r'Ein\s+',
                r'Eine\s+', r'Einen\s+', r'Dem\s+', r'Des\s+', r'Einem\s+',
                r'Eines\s+'),
        # Dutch
        'nld': (r'De\s+', r'Het\s+', r'Een\s+', r"'n\s+", r"'s\s+", r'Ene\s+',
                r'Ener\s+', r'Enes\s+', r'Den\s+', r'Der\s+', r'Des\s+',
                r"'t\s+"),
        # Swedish
        'swe': (r'En\s+', r'Ett\s+', r'Det\s+', r'Den\s+', r'De\s+'),
        # Turkish
        'tur': (r'Bir\s+',),
        # Afrikaans
        'afr': (r"'n\s+", r'Die\s+'),
        # Greek
        'ell': (r'O\s+', r'I\s+', r'To\s+', r'Ta\s+', r'Tus\s+', r'Tis\s+',
                r"'Enas\s+", r"'Mia\s+", r"'Ena\s+", r"'Enan\s+"),
        # Hungarian
        'hun': (r'A\s+', r'Az\s+', r'Egy\s+'),
}
default_language_for_title_sort = None
title_sort_articles = r'^(A|The|An)\s+'

#: Specify a folder calibre should connect to at startup
# Specify a folder that calibre should connect to at startup using
# connect_to_folder. This must be a full path to the folder. If the folder does
# not exist when calibre starts, it is ignored.
# Example for Windows:
#     auto_connect_to_folder = 'C:/Users/someone/Desktop/testlib'
```

(continues on next page)

(pokračujte na předchozí stránce)

```
# Example for other operating systems:
#     auto_connect_to_folder = '/home/dropbox/My Dropbox/someone/library'
auto_connect_to_folder = ''

#: Specify renaming rules for SONY collections
# Specify renaming rules for SONY collections. This tweak is only applicable if
# metadata management is set to automatic. Collections on SONYs are named
# depending upon whether the field is standard or custom. A collection derived
# from a standard field is named for the value in that field.
#
# For example, if the standard 'series' column contains the value 'Darkover', then the
# collection name is 'Darkover'. A collection derived from a custom field will
# have the name of the field added to the value. For example, if a custom series
# column named 'My Series' contains the name 'Darkover', then the collection
# will by default be named 'Darkover (My Series)'. For purposes of this
# documentation, 'Darkover' is called the value and 'My Series' is called the
# category. If two books have fields that generate the same collection name,
# then both books will be in that collection.
#
# This set of tweaks lets you specify for a standard or custom field how
# the collections are to be named. You can use it to add a description to a
# standard field, for example 'Foo (Tag)' instead of the 'Foo'. You can also use
# it to force multiple fields to end up in the same collection.
#
# For example, you could force the values in 'series', '#my_series_1', and
# '#my_series_2' to appear in collections named 'some_value (Series)', thereby
# merging all of the fields into one set of collections.
#
# There are two related tweaks. The first determines the category name to use
# for a metadata field.  The second is a template, used to determines how the
# value and category are combined to create the collection name.
# The syntax of the first tweak, sony_collection_renaming_rules, is:
# {'field_lookup_name':'category_name_to_use', 'lookup_name':'name', ...}
#
# The second tweak, sony_collection_name_template, is a template. It uses the
# same template language as plugboards and save templates. This tweak controls
# how the value and category are combined together to make the collection name.
# The only two fields available are {category} and {value}. The {value} field is
# never empty. The {category} field can be empty. The default is to put the
# value first, then the category enclosed in parentheses, it isn't empty:
# '{value} {category:|(|)}'
#
# Examples: The first three examples assume that the second tweak
# has not been changed.
#
# 1) I want three series columns to be merged into one set of collections. The
# column lookup names are 'series', '#series_1' and '#series_2'. I want nothing
# in the parenthesis. The value to use in the tweak value would be:
#    sony_collection_renaming_rules={'series':'', '#series_1':'', '#series_2':''}
#
# 2) I want the word '(Series)' to appear on collections made from series, and
# the word '(Tag)' to appear on collections made from tags. Use:
```

(continues on next page)

(pokračujte na předchozí stránce)

```
#   sony_collection_renaming_rules={'series':'Series', 'tags':'Tag'}
#
# 3) I want 'series' and '#myseries' to be merged, and for the collection name
# to have '(Series)' appended. The renaming rule is:
#   sony_collection_renaming_rules={'series':'Series', '#myseries':'Series'}
#
# 4) Same as example 2, but instead of having the category name in parentheses
# and appended to the value, I want it prepended and separated by a colon, such
# as in Series: Darkover. I must change the template used to format the category name
#
# The resulting two tweaks are:
#    sony_collection_renaming_rules={'series':'Series', 'tags':'Tag'}
#    sony_collection_name_template='{category:||: }{value}'
sony_collection_renaming_rules = {}
sony_collection_name_template = '{value}{category:| (|)}'

#: Specify how SONY collections are sorted
# Specify how SONY collections are sorted. This tweak is only applicable if
# metadata management is set to automatic. You can indicate which metadata is to
# be used to sort on a collection-by-collection basis. The format of the tweak
# is a list of metadata fields from which collections are made, followed by the
# name of the metadata field containing the sort value.
# Example: The following indicates that collections built from pubdate and tags
# are to be sorted by the value in the custom column '#mydate', that collections
# built from 'series' are to be sorted by 'series_index', and that all other
# collections are to be sorted by title. If a collection metadata field is not
# named, then if it is a series- based collection it is sorted by series order,
# otherwise it is sorted by title order.
# [(['pubdate', 'tags'],'#mydate'), (['series'],'series_index'), (['*'], 'title')]
# Note that the bracketing and parentheses are required. The syntax is
# [ ( [list of fields], sort field ) , ( [ list of fields ] , sort field ) ]
# Default: empty (no rules), so no collection attributes are named.
sony_collection_sorting_rules = []

#: Control how tags are applied when copying books to another library
# Set this to True to ensure that tags in 'Tags to add when adding
# a book' are added when copying books to another library
add_new_book_tags_when_importing_books = False

#: Set the maximum number of sort 'levels'
# Set the maximum number of sort 'levels' that calibre will use to resort the
# library after certain operations such as searches or device insertion. Each
# sort level adds a performance penalty. If the database is large (thousands of
# books) the penalty might be noticeable. If you are not concerned about multi-
# level sorts, and if you are seeing a slowdown, reduce the value of this tweak.
maximum_resort_levels = 5

#: Choose whether dates are sorted using visible fields
# Date values contain both a date and a time. When sorted, all the fields are
# used, regardless of what is displayed. Set this tweak to True to use only
# the fields that are being displayed.
sort_dates_using_visible_fields = False
```

(continues on next page)

(pokračujte na předchozí stránce)

```
#: Fuzz value for trimming covers
# The value used for the fuzz distance when trimming a cover.
# Colors within this distance are considered equal.
# The distance is in absolute intensity units.
cover_trim_fuzz_value = 10

#: Control behavior of the book list
# You can control the behavior of double clicks and pressing Enter on the books
# list. Choices: open_viewer, do_nothing, show_book_details,
# show_locked_book_details, edit_cell, edit_metadata. Selecting anything other
# than open_viewer, show_book_details, or show_locked_book_details has the side
# effect of disabling editing a field using a single click.
# Default: open_viewer.
# Example: doubleclick_on_library_view = 'do_nothing'
# You can also control whether the book list scrolls per item or
# per pixel. Default is per item.
doubleclick_on_library_view = 'open_viewer'
enter_key_behavior = 'do_nothing'
horizontal_scrolling_per_column = False
vertical_scrolling_per_row = False

#: Language to use when sorting
# Setting this tweak will force sorting to use the
# collating order for the specified language. This might be useful if you run
# calibre in English but want sorting to work in the language where you live.
# Set the tweak to the desired ISO 639-1 language code, in lower case.
# You can find the list of supported locales at
# https://en.wikipedia.org/wiki/List_of_ISO_639-1_codes
# Default: locale_for_sorting = '' -- use the language calibre displays in
# Example: locale_for_sorting = 'fr' -- sort using French rules.
# Example: locale_for_sorting = 'nb' -- sort using Norwegian rules.
locale_for_sorting = ''

#: The number of seconds to wait before sending emails
# The number of seconds to wait before sending emails when using a
# public email server like GMX/Hotmail/Gmail. Default is: 5 minutes
# Setting it to lower may cause the server's SPAM controls to kick in,
# making email sending fail. Changes will take effect only after a restart of
# calibre. You can also change the list of hosts that calibre considers
# to be public relays here. Any relay host ending with one of the suffixes
# in the list below will be considered a public email server.
public_smtp_relay_delay = 301
public_smtp_relay_host_suffixes = ['gmail.com', 'live.com', 'gmx.com', 'outlook.com']

#: The maximum width and height for covers saved in the calibre library
# All covers in the calibre library will be resized, preserving aspect ratio,
# to fit within this size. This is to prevent slowdowns caused by extremely
# large covers
maximum_cover_size = (1650, 2200)

#: Where to send downloaded news
```

(continues on next page)

(pokračujte na předchozí stránce)

```
# When automatically sending downloaded news to a connected device, calibre
# will by default send it to the main memory. By changing this tweak, you can
# control where it is sent. Valid values are "main", "carda", "cardb". Note
# that if there isn't enough free space available on the location you choose,
# the files will be sent to the location with the most free space.
send_news_to_device_location = 'main'

#: Unified toolbar on macOS
# If you enable this option and restart calibre, the toolbar will be 'unified'
# with the titlebar as is normal for macOS applications. However, doing this has
# various bugs, for instance the minimum width of the toolbar becomes twice
# what it should be and it causes other random bugs on some systems, so turn it
# on at your own risk!
unified_title_toolbar_on_osx = False

#: Save original file when converting/polishing from same format to same format
# When calibre does a conversion from the same format to the same format, for
# example, from EPUB to EPUB, the original file is saved, so that in case the
# conversion is poor, you can tweak the settings and run it again. By setting
# this to False you can prevent calibre from saving the original file.
# Similarly, by setting save_original_format_when_polishing to False you can
# prevent calibre from saving the original file when polishing.
save_original_format = True
save_original_format_when_polishing = True

#: Number of recently viewed books to show
# Right-clicking the "View" button shows a list of recently viewed books. Control
# how many should be shown, here.
gui_view_history_size = 15

#: Change the font size of the Book details panel in the interface
# Change the font size at which book details are rendered in the side panel and
# comments are rendered in the metadata edit dialog. Set it to a positive or
# negative number to increase or decrease the font size.
change_book_details_font_size_by = 0

#: What format to default to when using the "Unpack book" feature
# The "Unpack book" feature of calibre allows direct editing of a book format.
# If multiple formats are available, calibre will offer you a choice
# of formats, defaulting to your preferred output format if it is available.
# Set this tweak to a specific value of 'EPUB' or 'AZW3' to always default
# to that format rather than your output format preference.
# Set to a value of 'remember' to use whichever format you chose last time you
# used the "Unpack book" feature.
# Examples:
#   default_tweak_format = None       (Use output format)
#   default_tweak_format = 'EPUB'
#   default_tweak_format = 'remember'
default_tweak_format = None

#: Do not preselect a completion when editing authors/tags/series/etc.
# This means that you can make changes and press Enter and your changes will
```

(continues on next page)

(pokračujte na předchozí stránce)

```
# not be overwritten by a matching completion. However, if you wish to use the
# completions you will now have to press Tab to select one before pressing
# Enter. Which technique you prefer will depend on the state of metadata in
# your library and your personal editing style.
#
# If preselect_first_completion is False and you want Tab to accept what you
# typed instead of the first completion then set tab_accepts_uncompleted_text
# to True. If you do this then to select from the completions you must press
# the Down or Up arrow keys. The tweak tab_accepts_uncompleted_text is ignored
# if preselect_first_completion is True
preselect_first_completion = False
tab_accepts_uncompleted_text = False

#: Completion mode when editing authors/tags/series/etc.
# By default, when completing items, calibre will show you all the candidates
# that start with the text you have already typed. You can instead have it show
# all candidates that contain the text you have already typed. To do this, set
# completion_mode to 'contains'. For example, if you type asi it will match both
# Asimov and Quasimodo, whereas the default behavior would match only Asimov.
completion_mode = 'prefix'

#: Sort the list of libraries alphabetically
# The list of libraries in the Copy to library and Quick switch menus are
# normally sorted by most used. However, if there are more than a certain
# number of such libraries, the sorting becomes alphabetic. You can set that
# number here. The default is ten libraries.
many_libraries = 10

#: Choose available output formats for conversion
# Restrict the list of available output formats in the conversion dialogs.
# For example, if you only want to convert to EPUB and AZW3, change this to
# restrict_output_formats = ['EPUB', 'AZW3']. The default value of None causes
# all available output formats to be present.
restrict_output_formats = None

#: Set the thumbnail image quality used by the Content server
# The quality of a thumbnail is largely controlled by the compression quality
# used when creating it. Set this to a larger number to improve the quality.
# Note that the thumbnails get much larger with larger compression quality
# numbers.
# The value can be between 50 and 99
content_server_thumbnail_compression_quality = 75

#: Image file types to treat as e-books when dropping onto the "Book details" panel
# Normally, if you drop any image file in a format known to calibre onto the
# "Book details" panel, it will be used to set the cover. If you want to store
# some image types as e-books instead, you can set this tweak.
# Examples:
#    cover_drop_exclude = {'tiff', 'webp'}
cover_drop_exclude = ()

#: Exclude fields when copy/pasting metadata
```

(continues on next page)

(pokračujte na předchozí stránce)

```
# You can ask calibre to not paste some metadata fields when using the
# Edit metadata->Copy metadata/Paste metadata actions. For example,
# exclude_fields_on_paste = ['cover', 'timestamp', '#mycolumn']
# to prevent pasting of the cover, Date and custom column, mycolumn.
# You can also add a shortcut in Preferences->Shortcuts->Edit metadata
# to paste metadata ignoring this tweak.
exclude_fields_on_paste = []

#: Skip internet connected check
# Skip checking whether the internet is available before downloading news.
# Useful if for some reason your operating systems network checking
# facilities are not reliable (for example NetworkManager on Linux).
skip_network_check = False

#: Tab stop width in the template editor
# Sets the width of the tab stop in the template editor in "average characters".
# For example, a value of 1 results in a space with the width of one average␣
↪character.
template_editor_tab_stop_width = 4

#: Value for undefined numbers when sorting
# Sets the value to use for undefined numbers when sorting.
# For example, the value -10 sorts undefined numbers as if they were set to -10.
# Use 'maximum' for the largest possible number. Use 'minimum' for the smallest
# possible number. Quotes are optional if entering a number.
# Examples:
#   value_for_undefined_numbers_when_sorting = -100
#   value_for_undefined_numbers_when_sorting = '2'
#   value_for_undefined_numbers_when_sorting = -0.01
#   value_for_undefined_numbers_when_sorting = 'minimum'
#   value_for_undefined_numbers_when_sorting = 'maximum'
value_for_undefined_numbers_when_sorting = 0

#: Allow template database functions in composite columns
# If True then the template database functions book_values() and book_count()
# can be used in composite custom columns. Note: setting this tweak to True and
# using these functions in composites can be very slow.
# Default: False
allow_template_database_functions_in_composites = False


#: Change the programs that are run when opening files/URLs
# By default, calibre passes URLs to the operating system to open using
# whatever default programs are configured there. Here you can override
# that by specifying the program to use, per URL type. For local files,
# the type is "file" and for web links it is "http*". For example:
# openers_by_scheme = { "http*": "firefox %u" } will make calibre run Firefox
# for https://whatever URLs. %u is replaced by the URL to be opened. The scheme
# takes a glob pattern allowing a single entry to match multiple URL types.
openers_by_scheme = {}

#: Set the first day of the week for calendar popups
```

(continues on next page)

(pokračujte na předchozí stránce)

```
# It must be one of the values Default, Sunday, Monday, Tuesday, Wednesday,
# Thursday, Friday, or Saturday, all in English, spelled exactly as shown.
calendar_start_day_of_week = 'Default'

#: East Asian language to use for transliteration
# Setting this tweak will make calibre use the specified language as the "base"
# language when transliterating East Asian languages to English. This might be
# useful if you run calibre in English but want text transliterated to
# Japanese instead of Chinese. The valid values are:
#   'ja' for Japanese
#   'kr' for Korean
#   'vn' for Vietnamese
#   'zh' for Chinese
# Any other value will use the language set in calibre preferences as the base
# language. A base language other than those in the above list causes transliteration
# with a base language of Chinese.
# Example: east_asian_base_language = 'ja'
east_asian_base_language = ''
```

12.4 Přepsání ikon, šablon atd.

Poznámka

Calibre má přímou podporu pro motivy ikon, je pro něj dostupných několik motivů ikon, které můžete použít přejitím na *Předvolby → Rozhraní → Vzhled a chování → Změnit motiv ikon*. Je vhodnější používat motivy ikon než přepisovat jednotlivé ikony.

Calibre vám umožňuje přepsat statické zdroje, jako jsou ikony, JavaScript a šablony, pro přebal s metadaty, katalogy atd. upravenými verzemi, které se vám líbí. Všechny statické zdroje jsou uložené v podsložce resources instalačního umístění Calibre. Ve Windows je to obvykle `C:\Program Files\Calibre2\app\resources`. V macOS `/Applications/calibre.app/Contents/Resources/resources/`. V Linuxu, pokud používáte binární instalační balíček z webové stránky Calibre, to bude `/opt/calibre/resources`. Tyto cesty se můžou měnit v závislosti na tom, kam si zvolíte nainstalovat Calibre.

Soubory v této složce zdrojů byste neměli měnit, protože vaše změny budou přepsaány při příští aktualizaci Calibre. Namísto toho přejděte do *Předvolby → Rozšířené → Různé* a klikněte na *Otevřít konfigurační složku Calibre*. V této konfigurační složce vytvořte podsložku nazvanou resources a umístěte do ní soubory, které chcete přepsat. Umístěte soubory do příslušných podsložek, například obrázky umístěte do `resources/images` atd. Calibre automaticky použije při příštím spuštění váš vlastní soubor přednostně před vestavěným.

Například pokud chcete změnit ikonu pro akci *Odebrat knihy*, nejdříve byste se podívali do vestavěné složky resources a viděli byste, že odpovídající soubor je `resources/images/remove_books.png`. Pokud máte alternativní ikonu ve formátu PNG nazvanou `my_remove_books.png`, uložili byste ji do konfigurační složky jako `resources/images/remove_books.png`. Všechny ikony použité uživatelským rozhraním Calibre jsou v `resources/images` a jeho podsložkách. Umístění přepisujícího souboru sem bude mít ještě vyšší prioritu než vlastní motiv ikony.

12.5 Vytvoření vlastního motivu ikon pro Calibre

Pokud jste vytvořili krásnou sadu ikon a chcete ji sdílet s ostatními uživateli Calibre prostřednictvím vestavěného podpory motivů ikon Calibre, můžete snadno zabalit vaše ikony do motivu. K tomu přejděte na *Předvolby → Různé → Vytvořit motiv ikon*, vyberte složku, kam jste vložili své ikony. Pak vyplňte metadata motivu a klikněte na OK. Vznikne tak soubor ZIP obsahující ikony motivu. Ten můžete nahrát na fórum Calibre na Mobileread[116] a já pak zpřístupním váš motiv prostřednictvím vestavěného systému motivu ikon Calibre. Ve výchozím nastavení se právě vytvořené ikony motivu nainstalují jako aktuální motiv v Calibre. Pokud svůj motiv testujete, nezapomeňte odebrat obrázky ze složky `resources/images`, aby se použily ikony z motivu.

Od Calibre 6 můžete mít vlastní ikony pro světlý a tmavý režim. Jednoduše vytvořte dvě verze ikony a pojmenujte soubory s příponou `-for-dark-theme` a `-for-light-theme`. Například `modified-for-dark-theme.png` a `modified-for-light-theme.png`. Calibre pak automaticky použije příslušnou ikonu na základě aktuálního motivu.

12.6 Přizpůsobení Calibre pomocí modulů

Calibre má velice modulární provedení. Téměř všechny funkce jsou ve formě modulů. Moduly jsou použity pro převod, stahování zpráv (ačkoliv ty jsou nazvány předpisy), různé komponenty uživatelského rozhraní, připojení k různým zařízením, zpracování souborů při jejich přidávání do Calibre a tak dále. Kompletní seznam všech vestavěných modulů můžete získat v Calibre v *Předvolby → Rozšířené → Moduly*.

Pro přizpůsobení a rozšíření chování Calibre můžete napsat své vlastní moduly. Architektura modulu v Calibre je velice jednoduchá, podívejte se na kurz *Psaní vlastních modulů pro rozšíření funkcí Calibre* (stránka 224).

Jakmile napíšete modul, můžete jej nahrát na fórum modulů Calibre na Mobileread[117] a bude dostupný prostřednictvím vestavěného nástroje pro aktualizaci modulů Calibre.

[116] https://www.mobileread.com/forums/forumdisplay.php?f=166
[117] https://www.mobileread.com/forums/forumdisplay.php?f=237

KAPITOLA 13

Rozhraní příkazového řádku

Poznámka

On macOS, the command line tools are inside the calibre bundle, for example, if you installed calibre in `/Applications` the command line tools are in `/Applications/calibre.app/Contents/MacOS/`. So, for example, to run `ebook-convert` you would use: `/Applications/calibre.app/Contents/MacOS/ebook-convert`.

13.1 Dokumentované příkazy

13.1.1 `calibre`

```
calibre [volby] [cesta_k_eknize nebo url calibre...]
```

Spustí hlavní grafické uživatelské rozhraní Calibre a volitelně přidejte e-knihu v cestě_k_eknize do databáze. Můžete také zadat adresy URL Calibre pro provádění jednotlivých různých akcí, než jen přidávání knih. Například:

calibre://view-book/testovací_knihovna/1842/epub

Ro otevře knihu s ID 1842 ve formátu EPUB z knihovny „testovací_knihovna“ v Prohlížeči e-knih Calibre. Názvy knihoven jsou názvy složek knihoven s mezerami nahrazenými podtržítky. Úplný popis různých akcí založených na URL naleznete v Uživatelské příručce.

Kdykoliv předáváte parametry pro **calibre**, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

[volby]

--detach

Odpojit od řídícího terminálu, pokud nějaký je (pouze pro Linux)

--help, -h

zobrazit tuto zprávu nápovědy a ukončit

--ignore-plugins

Ignorovat vlastní moduly. Užitečné, pokud jste nainstalovali modul, který brání Calibre ve spuštění.

--no-update-check

Nekontrolovat aktualizace

--shutdown-running-calibre, -s

Způsobí vypnutí spuštěných instancí Calibre. Pamatujte, že pokud máte spuštěné úlohy, budou bez upozornění přerušeny, takže to používejte opatrně.

--start-in-tray

Spustit minimalizované do hlavního panelu systému.

--verbose, -v

Ignorováno, nepoužívejte. Přítomno pouze z historických důvodů

--version

zobrazit číslo verze programu a ukončit

--with-library

Použít knihovnu umístěnou v zadané cestě.

13.1.2 calibre-customize

```
calibre-customize options
```

Přizpůsobit Calibre načtením externích modulů.

Kdykoliv předáváte parametry pro **calibre-customize**, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

[volby]

--add-plugin, -a

Přidat modul zadáním cesty k souboru ZIP, který ho obsahuje.

--build-plugin, -b

Pro vývojáře doplňků: Cesta ke složce, ve které vyvíjíte doplněk. Tento příkaz automaticky zabalí doplněk a aktualizuje ho v Calibre.

--customize-plugin

Přizpůsobení modulu. Zadejte název modulu a řetězec přizpůsobení oddělený čárkou. Řetězec přizpůsobení je stejný, jaký byste zadali při přizpůsobování modulu v hlavním grafickém rozhraní Calibre.

--disable-plugin

Zakázat uvedený modul

--enable-plugin

Povolit uvedený modul

--help, -h

zobrazit tuto zprávu nápovědy a ukončit

--list-plugins, -l

Vypsat všechny nainstalované moduly

--remove-plugin, -r

Odebrat vlastní modul podle jména. Nemá žádný vliv na vestavěné moduly.

--version

zobrazit číslo verze programu a ukončit

13.1.3 calibre-debug

```
calibre-debug [options]
```

Různá rozhraní příkazového řádku užitečná pro ladění Calibre. Bez voleb spouští tento příkaz vložený překladač Pythonu. Můžete také spustit hlavní grafické rozhraní Calibre, Prohlížeč e-knih Calibre a editor Calibre v režimu ladění.

Také obsahuje rozhraní pro různé kusy Calibre, které neobsahují vyhrazené nástroje příkazového řádku, jako jsou částečné vkládání písma, nástroj Rozdílů e-knih a tak dále.

Můžete také použít **calibre-debug** ke spuštění samostatných skriptů. K tomu ho použijte takto:

calibre-debug -e myscript.py -- --option1 --option2 file1 file2 …

Všechno za -- je předáno skriptu. Ladění Calibre můžete také použít jako shebang ve skriptech, například takto:

#!/usr/bin/env -S **calibre-debug** -e -- –

Kdykoliv předáváte parametry pro **calibre-debug**, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

[volby]

--add-simple-plugin

Přidat jednoduchý doplněk (např. doplněk obsahující pouze soubor .py) zadáním cesty k souboru py obsahujícímu kód doplňku.

--command, -c

Spustit kód Pythonu.

--debug-device-driver, -d

Ladit rozpoznávání zařízení

--default-programs

(Od)registrovat Calibre ve výchozích programech Windows. *--default-programs* (stránka 305) = (register|unregister)

--diff

Spustit rozdílový nástroj Calibre. Například: calibre-debug *--diff* (stránka 305) soubor1 soubor2

--edit-book

Spustit nástroj calibre Upravit knihu v režimu ladění.

--exec-file, -e

Spustit kód Pythonu v souboru.

--explode-book, -x

Rozložit knihu do zadané složky. Použití: -x soubor.epub vystupni_adresar Exportuje knihu jako kolekci souborů HTML a metadat, které můžete upravovat pomocí standardních nástrojů pro úpravu HTML. Funguje se soubory EPUB, AZW3, HTMLZ a DOCX.

--export-all-calibre-data

Export všech dat Calibre (knihy, nastavení, moduly). Obvykle budete vyzváni k zadání složky pro export a knihoven, které chcete exportovat. Můžete je také zadat jako parametry příkazového řádku a tyto otázky přeskočit. Pro složku pro export a knihovny použijte absolutní cesty. Pro export všech knihoven lze použít speciální klíčové slovo "all". Příklady: calibre-debug *--export-all-calibre-data* (stránka 306) # pro interaktivní použití calibre-debug *--export-all-calibre-data* (stránka 306) /cesta/k/prazdne/slozce/pro/export /cesta/ke/slozce/knihovny1 /cesta/ke/slozce/knihovny2 calibre-debug *--export-all-calibre-data* (stránka 306) /slozks/pro/export all # export všech známých knihoven

--gui, -g

Spustit grafické rozhraní s povoleným laděním. Výstup ladění je vytištěn do stdout a stderr.

--gui-debug

Spustit grafické rozhraní s konzolí ladění, protokolování do zadané cesty. Pouze pro vnitřní použití, pro spuštění grafického rozhraní v režimu ladění použijte volbu -g

--help, -h

zobrazit tuto zprávu nápovědy a ukončit

--implode-book, -i

Složit dříve rozloženou knihu. Použití: -i vystupni_adresar soubor.epub Importuje knihu ze souborů ve vystupni_adresar, který musí být vytvořen předchozím vyvoláním *--explode-book* (stránka 305). Ujistěte se, že zadáváte stejný typ souboru, jaký byl použit při rozložení.

--import-calibre-data

Importovat dříve exportovaná data Calibre

--inspect-mobi, -m

Zkontrolovat soubory MOBI v zadaných cestách

--kepubify

Převést zadaný soubor EPUB do formátu KEPUB bez provedení úplného převodu. Toto dělá ovladač Kobo při odesílání souborů do zařízení.

--paths

Výstup cest potřebných pro nastavení prostředí Calibre

--run-plugin, -r

Spustit modul poskytující rozhraní příkazového řádku. Například: calibre-debug -r „Název modulu“ -- soubor1 --volba1 Všechno za -- bude předáno modulu jako parametry.

--run-test, -t

Spusťte pojmenované testy. Pro spuštění všech testů použijte speciální hodnotu "all". Pokud název testu začíná tečkou, je považován za název modulu. Pokud název testu začíná @, je považován za název kategorie.

--run-without-debug

Nespouštějte s nastaveným příznakem DEBUG

--shutdown-running-calibre, -s

Způsobí vypnutí spuštěných instancí Calibre. Pamatujte, že pokud máte spuštěné úlohy, budou bez upozornění přerušeny, takže to používejte opatrně.

--subset-font, -f

Částečně vložit zadané písmo. Použijte -- po této volbě pro předání volby programu pro částečné vkládání písem.

--test-build

Testovat binární moduly v sestavení

--un-kepubify

Převést zadaný soubor KEPUB do formátu EPUB bez provedení úplného převodu. Toto dělá ovladač Kobo při importu souborů ze zařízení.

--version

zobrazit číslo verze programu a ukončit

--viewer, -w

Spustit Prohlížeč e-knih v režimu ladění

13.1.4 calibre-server

```
calibre-server [options] [path to library folder ...]
```

Spustit Server s obsahem Calibre. Server s obsahem Calibre vystaví vaše knihovny Calibre na internet. Můžete zadat cestu ke složkám knihoven jako parametry do **calibre-server**. Pokud nezadáte žádné cesty, budou použity všechny knihovny, o kterých hlavní program Calibre ví.

Kdykoliv předáváte parametry pro **calibre-server**, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

[volby]

--access-log

Cesta k souboru protokolů přístupu. Tento protokol obsahuje informace o připojení klientů k serveru a vytváření požadavků. Ve výchozím nastavení není prováděno žádné protokolování přístupu.

--ajax-timeout

Čas (v sekundách) čekání na odezvu ze serveru při dotazování.

--auth-mode

Zvolte typ použitého ověřování. Nastavte režim ověřování HTTP používaný serverem. Nastavte na "základní", pokud je tento server za SSL proxy. Jinak ponechte jako "automaticky", což použije "základní", pokud je nakonfigurováno SSL, jinak použije "digest".

--auto-reload

Automaticky znovu načíst server, když se změní zdrojový kód. Užitečné pro vývoj. Měli byste také zadat malou hodnotu pro časový limit vypnutí.

--ban-after

Počet chybných přihlášení pro zakázání. Počet chybných přihlášení, po kterých je IP adresa zakázána

--ban-for

Zakázat IP adresy, které mají opakovaná chybná přihlášení. Dočasně zakáže přístup pro IP adresy, které mají opakovaná chybná přihlášení, na zadaný počet minut. Užitečné zabránění pokusům o odhadnutí hesel. Pokud je nastaveno na nulu, je zakazování vypnuto.

--book-list-mode

Zvolte výchozí režim seznamu knih. Nastavte výchozí režim seznamu knih, který bude použit pro nové uživatele. Jednotliví uživatelé mohou přepsat výchozí nastavení ve svém vlastním nastavení. Výchozí je použití mřížky obálek.

--compress-min-size

Minimální velikost, pro kterou odpovědi použijí kompresi dat (v bajtech).

--custom-list-template

Cesta k souboru JSON obsahujícímu šablonu pro režim vlastního seznamu knih. Nejsnadnější způsob vytvoření takového souboru šablony je přejít v Calibre do Předvolby->Sdílení po síti->Šablona seznamu knih, vytvořit šablonu a exportovat ji.

--daemonize

Spustit proces na pozadí jako démon (pouze Linux).

--displayed-fields

Omezit zobrazená uživatelem definovaná pole. Čárkami oddělený seznam uživatelem definovaných polí metadat, která budou zobrazena Serverem s obsahem v zobrazeních /opds a /mobilní. Pokud zadáte tuto volbu, nebudou zobrazena žádná pole, která nejsou v tomto seznamu. Například: my_rating,my_tags

--enable-allow-socket-preallocation, --disable-allow-socket-preallocation

Předpřidělení soketu, například při aktivaci soketu systemd. Ve výchozím nastavení je tato volba povolena.

--enable-auth, --disable-auth

Ověřování na základě hesla pro přístup k serveru. Obyčejně je server neomezený, umožňuje přístup každému. Pomocí této volby můžete omezit přístup na předdefinované uživatele. Ve výchozím nastavení je tato volba zakázána.

--enable-fallback-to-detected-interface, --disable-fallback-to-detected-interface

Záloha pro automatické rozpoznání rozhraní. Pokud se server není z nějakého důvodu schopen připojit k rozhraní zadanému ve volbě listen_on, pak se bude snažit rozpoznat rozhraní, které se připojuje k vnějšímu světu a připojit se k němu. Ve výchozím nastavení je tato volba povolena.

--enable-local-write, --disable-local-write

Povolit neověřeným místním připojením provádět změny. Obyčejně, pokud nezapnete ověřování, server pracuje v režimu pouze pro čtení, aby neumožnil anonymním uživatelům provádět změny ve vašich knihovnách Calibre. Tato volba umožňuje provádět změny komukoliv připojenému ze stejného počítače, na kterém je spuštěn server. To je užitečné, pokud chcete spustit server bez ověřování, ale stále používáte calibredb k provádění změn ve vašich knihovnách Calibre. Pamatujte, že zapnutí této volby znamená, že jakýkoliv program spuštěný v počítači může provádět změny ve vašich knihovnách Calibre. Ve výchozím nastavení je tato volba zakázána.

--enable-log-not-found, --disable-log-not-found

Protokolovat požadavky HTTP 404 (nenalezeno). Obyčejně server zaznamenává všechny požadavky HTTP na zdroje, které nejsou nalezeny. To může generovat velké množství protokolového spamu, pokud je váš server cílem botů. Použijte tuto volbu pro vypnutí. Ve výchozím nastavení je tato volba povolena.

--enable-use-bonjour, --disable-use-bonjour

Inzerovat informační kanály OPDS prostřednictvím BonJour. Inzerovat informační kanály OPDS prostřednictvím služby BonJour, aby aplikace založené na čtení OPDS mohly automaticky rozpoznat a připojit k serveru. Ve výchozím nastavení je tato volba povolena.

--enable-use-sendfile, --disable-use-sendfile

Přenosy nulových kopií souborů pro zvýšení výkonu. Toto použije přenosy nulových kopií v jádru při odesílání souborů přes síť, zvýšení výkonu. Může to však způsobit poškozené přenosy souborů u některých poškozených systémů souborů. Pokud narazíte na poškozené přenosy souborů, vypněte to. Ve výchozím nastavení je tato volba povolena.

--help, -h

zobrazit tuto zprávu nápovědy a ukončit

--ignored-fields

Ignorovaná uživatelem definovaná pole metadat. Čárkami oddělený seznam uživatelem definovaných polí metadat, která nebudou zobrazena Serverem s obsahem v zobrazeních /opds a /mobilní. Například: my_rating,my_tags

--listen-on

Rozhraní, na kterém naslouchat pro připojení. The default is to listen on all available IPv6 and IPv4 interfaces. You can change this to, for example, "127.0.0.1" to only listen for IPv4 connections from the local machine, or to "0.0.0.0" to listen to all incoming IPv4 connections.

--log

Cesta k souboru protokolu pro protokol serveru. Tento protokol obsahuje informace a chyby, ne protokoly přístupu. Ve výchozím nastavení je zapsán do stdout.

--manage-users

Spravovat databázi uživatelů, kteří mají povoleno připojovat se k tomuto serveru. V automatizovaném režimu to můžete použít přidáním –. Podrobnosti naleznete na calibre-server *--manage-users* (stránka 309) -- help. Podívejte se také na volbu *--userdb* (stránka 310).

--max-header-line-size

Max. velikost jednoduchého záhlaví HTTP (v KB).

--max-job-time

Maximální čas pro pracovní procesy. Maximální množství času, který mají pracovní procesy povoleno být spuštěny (v minutách). Nastavte na nulu pro žádné omezení.

--max-jobs

Maximální počet pracovních procesů. Pracovní procesy jsou spouštěny podle potřeby a použity pro velké úlohy, jako jsou příprava knihy pro zobrazení, přidávání knih, převod atd. Obyčejně je max. počet takových procesů založen na počtu jader procesoru. Můžete to ovládat tímto nastavením.

--max-log-size

Max. velikost souboru protokolu (v MB). Maximální velikost souborů protokolu generovaných serverem. Když bude protokol větší než tato velikost, je automaticky vyměněn. Nastavte na nulu pro zakázání výměny protokolu.

--max-opds-items

Maximální počet knih v informačních kanálech OPDS. Maximální počet knih, které server vrátí v jednom získaném informačním kanálu OPDS.

--max-opds-ungrouped-items

Maximální počet neseskupených položek v informačních kanálech OPDS. Seskupit položky do kategorií jako jsou autor nebo štítek podle prvního písmene, když je více než tento počet položek. Nastavte na nula pro zakázání.

--max-request-body-size

Max. povolená velikost pro soubory odesílané na server (v MB).

--num-per-page

Počet zobrazených knih na jedné stránce. Počet zobrazených knih na jedné stránce v prohlížeči:

--pidfile

Zapsat kód PID procesu do zadaného souboru

--port

Port, na kterém naslouchat pro připojení.

--search-the-net-urls

Cesta k souboru JSON obsahujícímu adresy URL pro funkci „Vyhledat na internetu“. Nejsnadnější způsob vytvoření takového souboru je přejít v Calibre do Předvolby->Sdílení po síti->Vyhledat na internetu, vytvořit adresy URL a exportovat je.

--shutdown-timeout

Celkový čas v sekundách, který čekat na čisté vypnutí.

--ssl-certfile

Cesta k souboru certifikátu SSL.

--ssl-keyfile

Cesta k souboru soukromého klíče SSL.

--timeout

Čas (v sekundách), po kterém je nečinné připojení uzavřeno.

--trusted-ips

Povolit neověřeným připojením z konkrétní adresy IP provádět změny. Obyčejně, pokud nezapnete ověřování, server pracuje v režimu pouze pro čtení, aby neumožnil anonymním uživatelům provádět změny ve vašich knihovnách Calibre. Tato volba umožňuje provádět změny komukoliv připojenému z konkrétní adresy IP. Musí to být čárkami oddělený seznam adres nebo specifikací sítě. To je užitečné, pokud chcete spustit server bez ověřování, ale stále používáte calibredb k provádění změn ve vašich knihovnách Calibre. Pamatujte, že zapnutí této volby znamená, že kdokoliv připojený ze zadaných adres IP může provádět změny ve vašich knihovnách Calibre.

--url-prefix

Předpona, kterou předřadit všem adresám URL. Užitečné, pokud si přejete spustit tento server za reverzní proxy. Například použijte /calibre jako předponu URL.

--userdb

Cesta k uživatelské databázi použité pro ověřování. Databáze je soubor SQLite. Pro jeho vytvoření použijte *--manage-users* (stránka 309). Více o správě uživatelů si můžete přečíst v: https://manual.calibre-ebook.com/cs/server.html#managing-user-accounts-from-the-command-line-only

--version

zobrazit číslo verze programu a ukončit

--worker-count

Počet pracovních vláken použitých pro zpracování požadavků.

13.1.5 calibre-smtp

```
calibre-smtp [options] [from to text]
```

Odeslat e-mail pomocí protokolu SMTP. **calibre-smtp** má dva režimy. V režimu psaní zadáte from to a text, a ty jsou použity k sestavení a odeslání e-mailové zprávy. V režimu filtrování čte **calibre-smtp** kompletní e-mailovou zprávu ze vstupu STDIN a odešle ji.

text je tělo e-mailové zprávy. Pokud text není zadán, kompletní e-mailová zpráva je čtena ze vstupu STDIN. from je e-mailová adresa odesílatele a to je e-mailová adresa příjemce. Když je kompletní e-mail čten ze vstupu STDIN, from a to jsou použity pouze při připojení vstupu STDIN, hlavičky zpráv nejsou měněny.

Kdykoliv předáváte parametry pro **calibre-smtp**, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

[volby]

--fork, -f

Rozvětvit a doručit zprávu na pozadí. Pokud použijete tuto volbu, měli byste také použít *--outbox* (stránka 311) pro vyřešení selhání doručení.

--help, -h

zobrazit tuto zprávu nápovědy a ukončit

--localhost, -l

Název hostitele localhost. Použito při připojování k serveru SMTP.

--outbox, -o

Cesta ke složce maildir pro e-mailové zprávy, které se nepodařilo odeslat.

--timeout, -t

Časový limit pro připojení

--verbose, -v

Více podrobností

--version

zobrazit číslo verze programu a ukončit

COMPOSE MAIL

Volby pro napsání e-mailu. Ignorovat, pokud text není zadán

--attachment, -a

Soubor připojený k e-mailu

--subject, -s

Předmět e-mailu

SMTP RELAY

Volby pro použití serveru přenosu SMTP pro odesílání pošty. Calibre se pokusí odeslat e-mail přímo, pokud není zadáno –relay.

--cafile

Cesta ke souboru zřetězených certifikátů CA ve formátu PEM používaných při ověřování certifikátu serveru při použití TLS. Ve výchozím nastavení se používají certifikáty systému CA.

--dont-verify-server-certificate

Neověřovat certifikát serveru při připojení pomocí TLS. Toto bylo výchozí chování ve verzích Calibre před 3.27. Pokud používáte přenos s certifikátem s vlastním podpisem nebo jinak neplatným certifikátem, můžete použít tuto volbu k obnovení chování před 3.27

--encryption-method, -e

Metoda šifrování použitá při připojování k přenosu. Možnosti jsou TLS, SSL a Žádné. Výchozí je TLS. VAROVÁNÍ: Výběr volby Žádné je vysoce nezabezpečený.

--password, -p

Heslo pro přenos

--port

Port pro připojení k serveru přenosu. Výchozí je použít 465, pokud je metoda šifrování SSL, jinak 25.

--relay, -r

Server přenosu SMTP použitý pro odeslání e-mailu.

--username, -u

Uživatelské jméno pro přenos

13.1.6 calibredb

```
calibredb command [options] [arguments]
```

calibredb je rozhraní příkazového řádku k databázi Calibre. Má několik níže zdokumentovaných podpříkazů.

calibredb calibredb` lze použít buď pro zacházení s databází Calibre zadanou cestou nebo se *Serverem s obsahem* Calibre běžícím buď na místním počítači nebo přes internet. *Server s obsahem* Calibre můžete spustit buď pomocí programu **calibre-server** nebo v hlavním programu Calibre klikněte na *Připojit nebo sdílet → Spustit Server s obsahem*. Protože **calibredb** může ve vašich knihovnách Calibre provádět změny, musíte nejprve na serveru nastavit ověřování. Jsou dva způsoby, jak toho dosáhnout:

- Pokud se plánujete připojovat pouze k serveru běžícímu na

stejném počítači, můžete jednoduše použít volbu Serveru s obsahem `--enable-local-write`, aby umožnil jakémukoliv programu, včetně calibredb, běžícímu na místním počítači provádět změny v datech Calibre. Při spuštění serveru z hlavního programu Calibru je tato možnost v *Předvolby → Sdílení po síti → Rozšířené*.

- Pokud chcete povolit přístup přes internet, měli byste na serveru nastavit

uživatelské účty a použít volby *--username* (stránka 313) a *--password* (stránka 313) pro **calibredb**, aby jim udělil přístup. Ověřování uživatele pro **calibre-server** můžete nastavit pomocí volby `--enable-auth` a pomocí `--manage-users` vytvořit uživatelské účty. Pokud spouštíte server z hlavního programu Calibre, použijte *Předvolby → Sdílení po síti → Vyžadovat uživatelské jméno a heslo*.

Pro připojení k běžícímu serveru s obsahem předejte adresu URL serveru volbě *--with-library* (stránka 313), viz dokumentace této volby pro podrobnosti a příklady.

- *GLOBÁLNÍ VOLBY* (stránka 313)
- *list* (stránka 313)
- *add* (stránka 314)
  - *PŘIDÁVÁNÍ ZE SLOŽEK* (stránka 315)
- *remove* (stránka 316)
- *add_format* (stránka 316)
- *remove_format* (stránka 316)
- *show_metadata* (stránka 316)
- *set_metadata* (stránka 317)
- *export* (stránka 317)
- *catalog* (stránka 318)
  - *VOLBY EPUB* (stránka 318)
- *saved_searches* (stránka 320)
- *add_custom_column* (stránka 320)
- *custom_columns* (stránka 321)
- *remove_custom_column* (stránka 321)
- *set_custom* (stránka 321)
- *restore_database* (stránka 321)

- *check_library* (stránka 322)
- *list_categories* (stránka 322)
- *backup_metadata* (stránka 322)
- *clone* (stránka 323)
- *embed_metadata* (stránka 323)
- *search* (stránka 323)
- *fts_index* (stránka 324)
- *fts_search* (stránka 324)

GLOBÁLNÍ VOLBY

--help, -h

zobrazit tuto zprávu nápovědy a ukončit

--library-path, --with-library

Cesta ke knihovně Calibre. Výchozí je použít cestu uloženou v nastaveních. Pro provádění akce na vzdálených knihovnách se můžete také připojit k Serveru s obsahem Calibre. Abyste to udělali, použijte URL ve formě: http://hostitel:port/#id_knihovny, například http://localhost:8080/#mojeknihovna. id_knihovny je identifikátor knihovny, ke které se chcete připojit na Serveru s obsahem. Pomocí speciální hodnoty id_knihovny - můžete získat seznam identifikátorů knihoven dostupných na serveru. Pro podrobnosti o nastavení přístupu prostřednictvím Serveru s obsahem navštivte https://manual.calibre-ebook.com/cs/generated/cs/calibredb.html.

--password

Heslo pro připojení k Serveru s obsahem Calibre. Pro přečtení hesla ze standardního vstupu použijte speciální hodnotu: <stdin>. Pro přečtení hesla ze souboru použijte: <f:/path/to/file> (tj. <f: následované úplnou cestou k souboru a koncovou >). Výše uvedené lomené závorky jsou vyžadovány, nezapomeňte je uvodit nebo použijte uvozovky pro váš shell.

--timeout

Časový limit v sekundách u připojení ke knihovně Calibre přes síť. Výchozí je dvě minuty.

--username

Uživatelské jméno pro připojení k Serveru s obsahem Calibre

--version

zobrazit číslo verze programu a ukončit

list

```
calibredb list [options]
```

Vypsat knihy dostupné v databázi Calibre.

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--ascending

Seřadit výsledky ve vzestupném pořadí

--fields, -f

Zobrazená pole při výpisu knih v databázi. Měl by to být čárkami oddělený seznam polí. Dostupná pole: author_sort, authors, comments, cover, formats, identifiers, isbn, languages, last_modified, pubdate, publisher, rating, series, series_index, size, tags, template, timestamp, title, uuid Výchozí: title,authors. Pro výběr všech polí může být použito speciální pole „all“. Kromě výše uvedených polí jsou také dostupná vlastní pole, jako například *field_name, pro vlastní pole #rating použijte název *rating

--for-machine

Generovat výstup ve formátu JSON, který je vhodnější pro strojní analýzu. Způsobí, že šířka řádku a volby oddělovače budou ignorovány.

--limit

Maximální počet výsledků k zobrazení. Výchozí: vše

--line-width, -w

Maximální šířka jednoho řádku ve výstupu. Výchozí nastavení pro rozpoznání velikosti obrazovky.

--prefix

Předpona pro všechny cesty k souborům. Výchozí je absolutní cesta do složky knihovny.

--search, -s

Filtrovat výsledky podle vyhledávacího dotazu. Pro formát vyhledávacího dotazu se podívejte do dokumentace související s hledáním v uživatelské příručce. Výchozí je neprovádět žádné filtrování.

--separator

Řetězec použitý pro oddělení polí. Výchozí je mezera.

--sort-by

Pole, podle kterého řadit výsledky. Více polí můžete zadat jejich oddělením čárkami. Dostupná pole: author_sort, authors, comments, cover, formats, identifiers, isbn, languages, last_modified, pubdate, publisher, rating, series, series_index, size, tags, template, timestamp, title, uuid Výchozí: id

--template

Šablona, která se spustí, pokud je „template“ v seznamu polí. Pamatujte, že šablony jsou při připojování k serveru Calibre ignorovány. Výchozí: Žádná

--template_file, -t

Cesta k souboru obsahujícímu šablonu, kterou spustit, pokud je „template“ v seznamu polí. Výchozí: Žádná

--template_heading

Záhlaví sloupce šablony. Výchozí: %dvýchozí. Tato volba je ignorována, pokud je nastavena volba *--for-machine* (stránka 314)

add

```
calibredb add [options] file1 file2 file3...
```

Přidat zadané soubory jako knihy do databáze. Můžete také zadat složky, podívejte se na volby související s následující složkou.

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--authors, -a

Nastavit autory přidaných knih

--automerge, -m

Pokud jsou nalezeny knihy s podobnými názvy a autory, automaticky sloučit příchozí formáty (soubory) do existujících záznamů knih. Hodnota „ignore“ znamená, že duplicitní formáty jsou zahozeny. Hodnota „overwrite“ znamená, že duplicitní formáty v knihovně jsou přepsány nově přidanými soubory. Hodnota „new_record“ znamená, že duplicitní formáty jsou umístěny do nového záznamu knihy.

--cover, -c

Nastavit použitou obálku pro přidanou knihu

--duplicates, -d

Přidat knihy do databáze, i když již existují. Srovnání je prováděno na základě názvů knih. Pamatujte, že volby *--automerge* (stránka 314) mají přednost.

--empty, -e

Přidat prázdnou knihu (kniha bez formátů)

--identifier, -I

Nastavit identifikátory pro tuto knihu, např. -I asin:XXX -I isbn:YYY

--isbn, -i

Nastavit ISBN přidaných knih

--languages, -l

Čárkami oddělený seznam jazyků (nejlepší je použít kódy jazyka ISO639, i když mohou být rozpoznány také některé názvy jazyků)

--series, -s

Nastavit série přidaných knih

--series-index, -S

Nastavit číslo série přidaných knih

--tags, -T

Nastavit štítky přidaných knih

--title, -t

Nastavit název přidaných knih

PŘIDÁVÁNÍ ZE SLOŽEK

Volby pro ovládání přidávání knih ze složek. Ve výchozím nastavení jsou přidávány pouze soubory, které mají příponu známých typů souborů e-knih.

--add

Vzor názvu souboru (glob), soubory odpovídající tomuto vzoru budou přidány při prohledávání souborů ve složkách, dokonce i když nejsou známého typu souboru e-knihy. Může být zadáno vícekrát pro více vzorů.

--ignore

Vzor názvu souboru (glob), soubory odpovídající tomuto vzoru budou ignorovány při prohledávání souborů ve složkách. Může být zadáno vícekrát pro více vzorů: Například: *.pdf bude ignorovat všechny soubory PDF

--one-book-per-directory, -1

Předpokládat, že každá složka má pouze jednu logickou knihu, a že všechny soubory v ní jsou různé formáty e-knih této knihy

--recurse, -r

Zpracuj složky rekurzivně

remove

```
calibredb remove ids
```

Odebrat knihy identifikované identifikátory z databáze. Identifikátory by měly být čárkami oddělený seznam identifikačních čísel (identifikační čísla můžete získat pomocí příkazu search). Například 23,34,57-85 (když zadáváte rozsah, poslední číslo rozsahu není zahrnuto).

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--permanent

Nepoužívat Koš

add_format

```
calibredb add_format [options] id ebook_file
```

Přidat e-knihu v ebook_file do dostupných formátů pro logickou knihu identifikovanou identifikátorem. Identifikátor můžete získat pomocí příkazu search. Pokud formát již existuje, je nahrazen, pokud není zadána volba nenahrazovat.

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--as-extra-data-file

Přidat soubor jako dodatečný datový soubor do knihy, ne jako formát e-knihy

--dont-replace

Nenahrazovat formát, pokud již existuje

remove_format

```
calibredb remove_format [options] id fmt
```

Odebrat formát fmt z logické knihy identifikované identifikátorem. Identifikátor můžete získat pomocí příkazu search. fmt by měla být přípona souboru, jako je LRF, TXT nebo EPUB. Pokud logická kniha nemá dostupné fmt, nedělat nic.

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

show_metadata

```
calibredb show_metadata [options] id
```

Zobrazit metadata uložená v databázi Calibre pro knihu identifikovanou identifikátorem. Identifikátor je identifikační číslo z příkazu search.

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--as-opf

Vypsat metadata v podobě OPF (XML)

set_metadata

```
calibredb set_metadata [options] book_id [/path/to/metadata.opf]
```

Nastavit metadata uložená v databázi Calibre pro knihu identifikovanou pomocí book_id ze souboru OPF metadata.opf. book_id je identifikační číslo knihy z příkazu search. Můžete získat rychlé chování pro formát OPF pomocí přepínače –as-opf příkazu show_metadata. Můžete také nastavit metadata jednotlivých polí pomocí volby –field. Pokud použijete volbu –field, není potřeba zadávat soubor OPF.

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--field, -f

Pole k nastavení. Formát je field_name:value, například : *--field* (stránka 317) tags:tag1,tag2. Použijte *--list-fields* (stránka 317) pro získání seznamu všech názvů polí. Tuto volbu můžete zadat vícekrát pro nastavení více polí. Poznámka: Pro jazyky musíte použít kódy jazyka ISO639 (např. cs pro češtinu, en pro angličtinu, fr pro francouzštinu atd.). Pro identifikátory je syntaxe *--field* (stránka 317) identifiers:isbn:XXXX,doi:YYYYY. Pro pole logických hodnot (ano/ne) použijte pravda a nepravda nebo ano a ne.

--list-fields, -l

Vypsat názvy polí metadat, která mohou být použita s volbou *--field* (stránka 317)

export

```
calibredb export [options] ids
```

Exportovat knihy zadané identifikátory (čárkami oddělený seznam) do systému souborů. Operace exportu ukládá všechny formáty knihy, její obálku a metadata (v souboru OPF). Jakákoliv dodatečná data přidružená ke knize jsou také uložena Identifikační čísla můžete získat z příkazu search.

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--all

Exportovat všechny knihy v databázi, ignorovat seznam identifikátorů.

--dont-asciiize

Nechat Calibre převést všechny neanglické znaky na anglické ekvivalenty pro názvy souborů. To je užitečné, pokud ukládáte do starší verze systému souborů bez plné podpory názvů souborů Unicode. Zadání tohoto přepínače vypne toto chování.

--dont-save-cover

Obyčejně Calibre uloží obálku do samostatného souboru společně se skutečnými soubory e-knih. Zadání tohoto přepínače vypne toto chování.

--dont-save-extra-files

Při ukládání knihy uložit všechny datové soubory přidružené ke knize Zadání tohoto přepínače vypne toto chování.

--dont-update-metadata

Obyčejně Calibre aktualizuje metadata v uložených souborech z toho, co je v knihovně Calibre. Zpomalí ukládání na disk. Zadání tohoto přepínače vypne toto chování.

--dont-write-opf

Obyčejně Calibre zapíše metadata do samostatného souboru OPF společně se skutečnými soubory e-knih. Zadání tohoto přepínače vypne toto chování.

--formats

Čárkami oddělený seznam formátů pro uložení pro každou knihu. Ve výchozím nastavení jsou ukládány všechny dostupné formáty.

--progress

Průběh hlášení

--replace-whitespace

Nahradit prázdné znaky podtržítky.

--single-dir

Exportovat všechny knihy do jedné složky

--template

Šablona pro ovládání názvu souboru a struktury složekukládaných souborů. Výchozí je "{author_sort}/{title}/{title} - {authors}", což uloží knihy do podsložky podle autora s názvy souborů obsahujícími název a autora. Dostupné ovládací prvky jsou: {author_sort, authors, id, isbn, languages, last_modified, pubdate, publisher, rating, series, series_index, tags, timestamp, title}

--timefmt

Formát, ve kterém zobrazovat data. %d - den, %b - měsíc, %m - číslo měsíce, %Y - rok. Výchozí je: %b, %Y

--to-dir

Exportovat knihy do zadané složky. Výchozí je .

--to-lowercase

Převést cesty na malá písmena.

catalog

```
calibredb catalog /path/to/destination.(csv|epub|mobi|xml...) [options]
```

Exportovat katalog ve formátu zadaném rozšířením path/to/destination. Volby určují, jak jsou položky v generovaném výstupním katalogu zobrazeny. Pamatujte, že různé formáty katalogu podporují různé sady voleb. Pro zobrazení různých voleb zadejte název výstupního soubor a pak volbu –help.

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--ids, -i

Čárkami oddělený seznam identifikátorů databáze pro katalog. Pokud je deklarováno, *--search* (stránka 318) je ignorováno. Výchozí: vše

--search, -s

Filtrovat výsledky podle vyhledávacího dotazu. Pro formát vyhledávacího dotazu se podívejte do dokumentace související s hledáním v uživatelské příručce. Výchozí: žádné filtrování

--verbose, -v

Zobrazit podrobné výstupní informace. Užitečné pro ladění

VOLBY EPUB

--catalog-title

Název generovaného katalogu použitý jako název v metadatech. Výchozí: 'My Books' Použije se na: výstupní formáty AZW3, EPUB, MOBI

--cross-reference-authors

Vytvořit křížové odkazy v oddílu Autoři pro knihy s více autory. Výchozí: 'False' Použije se na: výstupní formáty AZW3, EPUB, MOBI

--debug-pipeline

Uložit výstup z různých stavů převodního řetězce do zadané složky. Užitečné, pokud si nejste jisti, ve kterém stavu procesu převodu se objevuje chyba. Výchozí: 'None' Použije se na: výstupní formáty AZW3, EPUB, MOBI

--exclude-genre

Regulární výraz popisující štítky, které vyloučit jako žánry. Výchozí: '[.+]|^+$' vylučuje štítky v hranatých závorkách, např. '[Project Gutenberg]' a '+', výchozí štítek pro přečtené knihy. Použije se na: výstupní formáty AZW3, EPUB, MOBI

--exclusion-rules

Určuje pravidla použitá pro vyloučení knih z generovaného katalogu. Model pro pravidlo vyloučení je buď ('<název pravidla>','Tags','<čárkami oddělený seznam štítků>') nebo ('<název pravidla>','<vlastní sloupec>','<vzor>'). Například: (('Archivované knihy','#stav','Archivováno'),) vyloučí knihu s hodnotou 'Archivováno' ve vlastním sloupci 'stav'. Pokud je definováno více pravidel, budou použita všechna pravidla. Výchozí: "(('Catalogs','Tags','Catalog'),)" Použije se na výstupní formáty AZW3, EPUB, MOBI

--generate-authors

Zahrnout oddíl 'Autoři' do katalogu. Výchozí: 'False' Použije se na: výstupní formáty AZW3, EPUB, MOBI

--generate-descriptions

Zahrnout oddíl 'Popisy' do katalogu. Výchozí: 'False' Použije se na: výstupní formáty AZW3, EPUB, MOBI

--generate-genres

Zahrnout oddíl 'Žánry' do katalogu. Výchozí: 'False' Použije se na: výstupní formáty AZW3, EPUB, MOBI

--generate-recently-added

Zahrnout oddíl 'Nedávno přidané' do katalogu. Výchozí: 'False' Použije se na: výstupní formáty AZW3, EPUB, MOBI

--generate-series

Zahrnout oddíl 'Série' do katalogu. Výchozí: 'False' Použije se na: výstupní formáty AZW3, EPUB, MOBI

--generate-titles

Zahrnout oddíl 'Názvy' do katalogu. Výchozí: 'False' Použije se na: výstupní formáty AZW3, EPUB, MOBI

--genre-source-field

Zdrojové pole pro oddíl ‚Žánry. Výchozí: 'Štítky' Použije se na: výstupní formáty AZW3, EPUB, MOBI

--header-note-source-field

Vlastní pole obsahující text poznámky k vložení do záhlaví Popisu. Výchozí: '' Použije se na: výstupní formáty AZW3, EPUB, MOBI

--merge-comments-rule

#<custom field>:[before|lafter]:[True|False] určující: <custom field>Vlastní pole obsahující poznámky ke sloučení s komentáři [before|lafter] Umístění poznámek vzhledem ke kometářům [True|False] - Mezi poznámky a komentáře je vložena vodorovná čára Výchozí: '::' Použije se na: výstupní formáty AZW3, EPUB, MOBI

--output-profile

Určuje výstupní profil. V některých případech je vyžadován výstupní profil pro optimalizaci katalogu pro zařízení. Například 'kindle' nebo 'kindle_dx' vytvoří strukturovaný obsah s Oddíly a Články. Výchozí: 'None' Použije se na: výstupní formáty AZW3, EPUB, MOBI

--prefix-rules

Určuje pravidla použitá pro zahrnutí předpon označujících přečtené knihy, seznam přání a jiné předpony zadané uživatelem. Model pro pravidlo předpony je ('<název pravidla>','<zdrojové pole>','<vzor>','<předpona>'). Pokud je definováno více pravidel, bude použito první odpovídající pravidlo. Výchozí: '(('Read books','tags','+','✓'),('Wishlist item','tags','Wishlist','×'))' Použije se na výstupní formáty AZW3, EPUB, MOBI

--preset

Použít pojmenovanou předvolbu vytvořenou Nástrojem pro sestavení katalogu v grafickém rozhraní. Předvolba určuje všechna nastavení pro sestavení katalogu. Výchozí: 'None' Použije se na výstupní formáty AZW3, EPUB, MOBI

--thumb-width

Nápověda velikosti (v palcích) pro obálky knih v katalogu. Rozsah: 1,0 - 2,0 Výchozí: '1.0' Použije se na výstupní formáty AZW3, EPUB, MOBI

--use-existing-cover

Nahradit existující obálku při generování katalogu. Výchozí: 'False' Použije se na: výstupní formáty AZW3, EPUB, MOBI

saved_searches

```
calibredb saved_searches [options] (list|add|remove)
```

Spravovat uložená hledání umístěná v této databázi. Pokud se pokusíte přidat dotaz s názvem, který již existuje, bude nahrazen.

Syntaxe pro přidání:

calibredb **saved_searches** add search_name search_expression

Syntaxe pro odebrání:

calibredb **saved_searches** remove search_name

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

add_custom_column

```
calibredb add_custom_column [options] label name datatype
```

Vytvořit vlastní sloupec. label je název sloupce vhodný pro stroje. Neměl by obsahovat mezery nebo dvojtečky. name je název sloupce vhodný pro lidi. datatype je jeden z: bool, comments, composite, datetime, enumeration, float, int, rating, series, text

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--display

Slovník voleb pro přizpůsobení, jak budou data v tomto sloupci interpretována. Toto je řetězec JSON. Pro sloupce výčtu použijte *--display* (stránka 320)"{\ "enum_values\ ":[\ "val1\ ", \ "val2\ "]}" Je mnoho voleb, které mohou přijít do proměnné display. Volby podle typu sloupce jsou: složené: composite_template, composite_sort, make_category,contains_html, use_decorations datum a čas: date_format výčet: enum_values, enum_colors, use_decorations celá část, plovoucí: number_format text: is_names, use_decorations Nejlepší způsob jak najít zákonité kombinace, je vytvořit v grafickém rozhraní vlastní sloupec vhodného typu, pak vyhledat v záložním OPF knihu (ujistěte se, že byl vytvořen nový OPF od přidání sloupce). Uvidíte JSON pro "display" pro nový sloupec v OPF.

--is-multiple

Tento sloupec ukládá štítky jako data (tj. více čárkami oddělených hodnot). Použije se, pouze pokud je datový typ text.

custom_columns

```
calibredb custom_columns [options]
```

Vypsat dostupné vlastní sloupce. Zobrazí popisky sloupců a identifikátory.

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--details, -d

Zobrazit podrobnosti pro každý sloupec.

remove_custom_column

```
calibredb remove_custom_column [options] label
```

Odebrat vlastní sloupec identifikovaný popiskem. Dostupné sloupce můžete zobrazit příkazem custom_columns.

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--force, -f

Nepožadovat potvrzení

set_custom

```
calibredb set_custom [options] column id value
```

Nastavit hodnotu vlastního sloupce pro knihu identifikovanou identifikátorem. Seznam identifikátorů můžete získat pomocí příkazu search. Seznam názvů vlastních sloupců můžete získat pomocí příkazu custom_columns.

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--append, -a

Pokud sloupec ukládá více hodnot, přidat zadané hodnoty k existujícím, namísto jejich nahrazení.

restore_database

```
calibredb restore_database [options]
```

Obnovit tuto databázi z metadat uložených v souborech OPF v každé složce knihovny Calibre. To je užitečné, pokud byl váš soubor meatadata.db poškozen.

VAROVÁNÍ: Tento příkaz znovu úplně vygeneruje vaši databázi. Ztratíte všechna uložená hledání, uživatelské katalogy, zásuvné panely, uložená nastavení převodu pro knihu a vlastní předpisy. Obnovená metadata budou pouze tak přesná, jako to, co je nalezeno v souborech OPF.

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--really-do-it, -r

Opravdu provést obnovení. Příkaz nebude spuštěn, pokud není zadána tato volba.

check_library

```
calibredb check_library [options]
```

Provést některé kontroly souborového systému představujícího knihovnu. Hlášení jsou invalid_titles, extra_titles, invalid_authors, extra_authors, missing_formats, extra_formats, extra_files, missing_covers, extra_covers, failed_folders

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--csv, -c

Výstup v CSV

--ignore_extensions, -e

Čárkami oddělený seznam ignorovaných přípon. Výchozí: vše

--ignore_names, -n

Čárkami oddělený seznam ignorovaných názvů. Výchozí: vše

--report, -r

Čárkami oddělený seznam hlášení. Výchozí: vše

--vacuum-fts-db

Vyčistit databázi fulltextového vyhledávání. V závislosti na velikosti databáze to může být velmi pomalé a náročné na paměť.

list_categories

```
calibredb list_categories [options]
```

Vytvořit hlášení informací o kategoriích v databázi. Informace je ekvivalentem toho, co je zobrazeno v Prohlížeči štítků.

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--categories, -r

Čárkami oddělený seznam kategorií názvů vyhledávání. Výchozí: vše

--csv, -c

Výstup v CSV

--dialect

Typ souboru CSV, který vytvářet. Možnosti: excel, excel-tab, unix

--item_count, -i

Výstupem je pouze počet položek v kategorii namísto počtů na položku v kategorii.

--width, -w

Maximální šířka jednoho řádku ve výstupu. Výchozí nastavení pro rozpoznání velikosti obrazovky.

backup_metadata

```
calibredb backup_metadata [options]
```

Zálohovat metadata uložená v databázi do jednotlivých souborů OPF v každé složce knih. Toto se normálně děje automaticky, ale můžete tento příkaz spustit s volbou –all, abyste vynutili opětovné vygenerování souborů OPF.

Pamatujte, že normálně není vůbec potřeba toto dělat, protože soubory OPF jsou zálohovány automaticky pokaždé, když se změní metadata.

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--all

Normálně tento příkaz pracuje pouze nad knihami, které mají zastaralé soubory OPF. Pomocí této volby pracuje nad všemi knihami.

clone

```
calibredb clone path/to/new/library
```

Vytvořit klon aktuální knihovny. Toto vytvoří novou, prázdnou knihovnu, která má stejné vlastní sloupce, Virtuální knihovny a jiná nastavení jako aktuální knihovna.

Klonovaná knihovna nebude obsahovat žádné knihy. Pokud chcete vytvořit úplný duplikát včetně všech knih, pak jednoduše použijte nástroje vašeho systému souborů ke zkopírování složky knihovny.

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

embed_metadata

```
calibredb embed_metadata [options] book_id
```

Aktualizovat metadata ve skutečných souborech knih uložená v knihovně Calibre z metadat v databázi Calibre. Normálně jsou metadata aktualizována pouze při exportování souborů z Calibre, tento příkaz je užitečný, pokud chcete, aby byla metadata aktualizována na místě. Pamatujte, že různé formáty souborů podporují různá množství metadat. K aktualizaci metadat ve všech knihách můžete pro book_id použít speciální hodnotu ‚all'. Můžete také zadat mnoho identifikátorů knih oddělených mezerami a rozsahů identifikátorů oddělených spojovníky. Například: calibredb **embed_metadata** 1 2 10-15 23

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--only-formats, -f

Aktualizovat metadata pouze v souborech zadaného formátu. Zadejte vícekrát pro více formátů. Ve výchozím nastavení jsou aktualizovány všechny formáty.

search

```
calibredb search [options] search expression
```

Vyhledat v knihovně zadaný hledaný výraz vracející čárkami oddělený seznam identifikátorů knih odpovídající hledanému výrazu.Výstupní formát je užitečný pro zásobení jiných příkazů, které přijímají seznam identifikátorů jako vstup.

Hledaný výraz může být cokoliv z výkonného jazyka vyhledávacích dotazů Calibre, například: calibredb **search** author:asimov ‚title:"i robot"'

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--limit, -l

Maximální počet výsledků k navrácení. Výchozí je všechny výsledky.

fts_index

```
calibredb fts_index [volby] enable/disable/status/reindex
```

Ovládání procesu indexování fulltextového vyhledávání.

enable
Zapne indexování FTV pro tuto knihovnu.

disable
Vypne indexování FTV pro tuto knihovnu.

status
Zobrazí aktuální stav indexování

reindex
Může být použito ke znovu indexování buď konkrétních knih, nebo celé knihovny. Pro znovu indexování konkrétních knih zadejte identifikátory knih jako další argumenty za příkazem reindex. Pokud nejsou zadány žádné identifikátory knih, bude znovu indexována celá knihovna.

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--indexing-speed
Rychlost indexování. Použijte rychlé pro rychlé indexování využívající všechny zdroje vašeho počítače a pomalé pro indexování méně náročné na zdroje. Pamatujte, že rychlost se po každém vyvolání obnoví na pomalou.

--wait-for-completion
Čekat, dokud nejsou všechny knihy indexovány, pravidelně zobrazovat průběh indexování

fts_search

```
calibredb fts_search [volby] vyhledávací výraz
```

Provede fulltextové vyhledávání v celé knihovně nebo její podmnožině.

Kdykoliv předáváte parametry pro calibredb, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

--do-not-match-on-related-words
Porovnávat pouze přesná slova, nikoliv související slova. Takže oprava se nebude shodovat s opravou.

--include-snippets
Zahrnout úryvky textu obklopující každou shodu. Pamatujte, že to výrazně zpomaluje vyhledávání.

--indexing-threshold
Kolik v procentech z knihovny musí být indexováno, než je povoleno vyhledávání. Výchozí nastavení je 90

--match-end-marker
Značka použitá k označení konce shodného slova uvnitř úryvku

--match-start-marker
Značka použitá k označení začátku shodného slova uvnitř úryvku

--output-format
Formát, ve kterém vypisovat výsledky vyhledávání. Buď „text“ pro prostý text, nebo „json“ pro výstup JSON.

--restrict-to
Omezit vyhledávané knihy buď pomocí vyhledávacího výrazu, nebo identifikátorů. Například: ids:1,2,3 pro omezení podle identifikátorů nebo search:tag:foo pro omezení na knihy se značkou foo.

13.1.7 ebook-convert

```
ebook-convert input_file output_file [options]
```

Převeďte e-knihu z jednoho formátu do jiného.

input_file je vstup a output_file je výstup. Oba musí být zadány jako první dva parametry příkazu.

Výstupní formát e-knihy je odhadnut z příponou output_file. output_file může být také speciálního formátu .EXT, kde EXT je přípona výstupního souboru. V tomto případě je název výstupního souboru odvozen od názvu vstupního souboru. Pamatujte, že názvy souborů nesmí začínat pomlčkou. A konečně, pokud output_file nemá žádnou příponu, pak je s ním zacházeno jako se složkou a do této složky je zapsána „otevřená e-kniha“ (OEB) sestávající ze souborů HTML. Tyto soubory jsou soubory, které by obyčejně byly předány výstupnímu modulu.

Po zadání vstupního a výstupního souboru můžete přizpůsobit převod zadáním různých voleb. Dostupné volby závisí na typech vstupních a výstupních souborů. Pro získání nápovědy k nim zadejte vstupní a výstupní soubor a použijte volbu -h.

Úplnou dokumentaci systému převodu najdete na *Převod e-knihy* (stránka 61)

Kdykoliv předáváte parametry pro **ebook-convert**, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

Volby a výchozí hodnoty pro změnu voleb závisí na vstupním i výstupním formátu, takže byste je měli vždy zkontrolovat pomocí:

ebook-convert myfile.input_format myfile.output_format -h

Níže jsou volby, které jsou společné pro všechny převody, následované volbami konkrétními pro každý vstupní a výstupní formát.

- *VOLBY VSTUPU* (stránka 326)
- *VOLBY VÝSTUPU* (stránka 326)
- *VZHLED* (stránka 327)
- *HEURISTICKÉ ZPRACOVÁNÍ* (stránka 330)
- *NAJÍT A NAHRADIT* (stránka 330)
- *ROZPOZNÁVÁNÍ STRUKTURY* (stránka 331)
- *OBSAH* (stránka 332)
- *METADATA* (stránka 332)
- *LADĚNÍ* (stránka 333)

--help, -h
: zobrazit tuto zprávu nápovědy a ukončit

--input-profile
: Zadejte vstupní profil. Vstupní profil dává převodnímu systému informaci, jak interpretovat různé informace ve vstupním dokumentu. Například délky závislé na rozlišení (t.j. délky v pixelech). Možnosti jsou: cybookg3, cybook_opus, default, hanlinv3, hanlinv5, illiad, irexdr1000, irexdr800, kindle, msreader, mobipocket, nook, sony, sony300, sony900

--list-recipes
: Seznam vestavěných názvů předpisů. Z vestavěného předpisu můžete vytvořit e-knihu takto: ebook-convert "Nazev Predpisu.recipe" output.epub

--output-profile

Zadejte výstupní profil. Výstupní profil říká převodnímu systému, jak optimalizovat vytvořený dokument pro zadané zařízení. V některých případech může být výstupní profil použit pro optimalizaci výstupu pro konkrétní zařízení, ale je to zřídka potřeba. Možnosti jsou:cybookg3, cybook_opus, default, hanlinv3, hanlinv5, illiad, ipad, ipad3, irexdr1000, irexdr800, jetbook5, kindle, kindle_dx, kindle_fire, kindle_oasis, kindle_pw, kindle_pw3, kindle_scribe, kindle_voyage, kobo, msreader, mobipocket, nook, nook_color, nook_hd_plus, pocketbook_inkpad3, pocketbook_lux, pocketbook_hd, pocketbook_900, pocketbook_pro_912, galaxy, sony, sony300, sony900, sony-landscape, sonyt3, tablet, generic_eink, generic_eink_hd, generic_eink_large

--version

zobrazit číslo verze programu a ukončit

VOLBY VSTUPU

Volby pro ovládání zpracování vstupního souboru mobi

--input-encoding

Zadejte kódování znaků vstupního dokumentu. Pokud je tato volba nastavena, přepíše jakékoliv kódování deklarované samotným dokumentem. Zvláště užitečné pro dokumenty, které nedeklarují kódování nebo mají nesprávně deklarované kódování.

VOLBY VÝSTUPU

Volby pro ovládání zpracování výstupního souboru epub

--dont-split-on-page-breaks

Vypnout rozdělování na koncích stránek. Vstupní soubory jsou obyčejně automaticky rozděleny na každém konci stránky na dva soubory. Tak vznikne e-kniha, která může být analyzována rychleji a za použití méně prostředků. Rozdělování je však pomalé, a pokud váš zdrojový soubor obsahuje velmi velké množství konců stránek, měli byste rozdělování na koncích stránek vypnout.

--epub-flatten

Tato volba je nutná, pouze pokud máte v úmyslu použít EPUB v FBReaderJ. Urovná to souborový systém uvnitř EPUB, všechny soubory dá na nejvyšší úroveň.

--epub-inline-toc

Vložit vložený obsah, který se objeví jako část hlavního obsahu knihy.

--epub-max-image-size

Maximální velikost obrázku (šířka x výška). Hodnota none znamená použití velikosti obrazovky z výstupního profilu. Hodnota profile znamená, že není zadána žádná maximální velikost. Například hodnota 100x200 způsobí, že se velikost všech obrázků změní tak, aby jejich šířka nebyla větší než 100 pixelů a jejich výška nebyla větší než 200 pixelů. Pamatujte, že to ovlivňuje pouze velikost samotných souborů obrázků. Každý obrázek může být vykreslen v jiné velikosti v závislosti na stylování, které je na něj v dokumentu použito.

--epub-toc-at-end

Dát vložený obsah na konec knihy místo na začátek.

--epub-version

Verze souboru EPUB, který chcete vygenerovat. EPUB 2 je nejkompatibilnější, EPUB 3 použijte pouze tehdy, pokud víte, že ho skutečně potřebujete.

--extract-to

Extract the contents of the generated book to the specified folder. The contents of the folder are first deleted, so be careful.

--flow-size

Rozdělit všechny soubory HTML větší než tato velikost (v KB). Je to nezbytné, protože většina čteček EPUB neumí zpracovat velké soubory. Výchozí hodnota 260KB je velikost požadovaná pro Adobe Digital Editions. Nastavte na 0 pro zakázání rozdělování na základě velikosti.

--no-default-epub-cover

Pokud vstupní soubor nemá žádnou obálku a vy jste žádnou nezadali, je obyčejně vygenerována výchozí obálka s názvem, autorem atd. Tato volba zakáže generování této obálky.

--no-svg-cover

Nepoužívat SVG pro obálku knihy. Tuto volbu použijte, pokud vaše soubory EPUB budou používány v zařízení, které nepodporuje SVG, jako je iPhone nebo JetBook Lite. Bez této volby budou taková zařízení zobrazovat obálku jako prázdnou stránku.

--preserve-cover-aspect-ratio

Při použití obálky SVG tato volba způsobí změnu velikosti obálky, aby pokryla dostupné oblasti obrazovky, ale přesto zachová její poměr stran (poměr šířky k výšce). To znamená, že mohou být vidět bílé okraje po stranách nebo v horní a dolní části obrázku, ale obrázek nebude nikdy zdeformovaný. Bez této volby může být obrázek mírně deformovaný, ale nebudou vidět žádné okraje.

--pretty-print

Pokud je zadáno, výstupní modul se pokusí vytvořit výstup, který je co nejvíce čitelný pro člověka. Nemusí to mít žádný vliv na některé výstupní moduly.

--toc-title

Název pro jakýkoliv vygenerovaný vložený obsah.

VZHLED

Volby pro ovládání vzheldu a chování výstupu

--asciiize

Překódovat znaky Unicode do ASCII. Používejte opatrně, protože to nahradí znaky Unicode za ASCII. Například to nahradí "Pelé" za "Pele". Také pamatujte, že v případech, kdy je více zastoupení znaku (například znaky společné pro čínštinu a japonštinu), bude použito zastoupení podle aktuálního jazyku rozhraní Calibre.

--base-font-size

Základní velikost písma v bodech. Všechny velikosti písma ve vytvořené knize budou mít změněnou velikost na základě této hodnoty. Výběrem větší velikosti můžete písmo ve výstupu zvětšit a naopak. Ve výchozím nastavení, když je hodnota nula, je základní velikost písma zvolena na základě vámi zvoleného výstupního profilu.

--change-justification

Změnit zarovnání textu. Hodnota "doleva" převede všechen text zarovnaný do bloku ve zdroji na text zarovnaný vlevo (tj. nezarovnaný). Hodnota "do bloku" převede všechen nezarovnaný text na zarovnaný. Hodnota "původní" (výchozí) nezmění zarovnání ve zdrojovém souboru. Pamatujte, že pouze některé výstupní formáty podporují zarovnání.

--disable-font-rescaling

Zakázat všechny úpravy měřítka velikostí písma.

--embed-all-fonts

Vložit každé písmo, na které se odkazuje ve vstupním dokumentu, ale ještě není vloženo. Vyhledá písma ve vašem systému, a pokud je nalezen, budou vložena. Vkládání bude fungovat, pouze pokud formát, do kterého převádíte, podporuje vložená písma, jako jsou EPUB, AZW3, DOCX nebo PDF. Ujistěte se, že máte správnou licenci pro vkládání písem použitých v tomto dokumentu.

--embed-font-family

Vložit zadanou rodinu písem do knihy. To určuje "základní" písmo použité pro knihu. Pokud vstupní dokument určuje svá vlastní písma, mohou přepsat toto základní písmo. Můžete použít volbu filtrování stylu informací pro odebrání písem ze vstupního dokumentu. Pamatujte, že vkládání písem funguje pouze u některých výstupních formátů, hlavně EPUB a AZW3 a DOCX.

--expand-css

Ve výchozím nastavení použije calibre zjednodušený formulář pro různé vlastnosti CSS, jako jsou okraje, odsazení, rámeček atd. Tato volba způsobí, že se místo toho použije rozšířený formulář. Pamatujte, že CSS je vždy rozšířené při generování souborů EPUB s výstupním profilem nastaveným na jeden z profilů Nook, protože Nook neovládá zjednodušené CSS.

--extra-css

Buď cesta k šabloně stylů CSS nebo nezpracované CSS. Toto CSS bude připojeno k pravidlům stylu ze zdrojového souboru, takže může být použito k přepsání těchto pravidel.

--filter-css

Čárkami oddělený seznam vlastností CSS, které budou odebrány ze všech pravidel stylů CSS. To je užitečné, pokud přítomnost některých informací o stylu zabraňuje přepsání ve vašem zařízení. Například: font-family,color,margin-left,margin-right

--font-size-mapping

Mapování z názvů písem CSS na velikosti písem v bodech. Příklad nastavení je 12,12,14,16,18,20,22,24. Toto jsou mapování pro velikosti od xx-small do xx-large, s poslední velikostí pro velice velká písma. Algoritmus úpravy měřítka písma používá tyto velikosti pro chytrou úpravu měřítka písem. Výchozí je použití mapování na základě vámi zvoleného výstupního profilu.

--insert-blank-line

Vložit prázdný řádek mezi odstavce. Nebude fungovat, pokud zdrojový soubor nepoužívá odstavce (značky <p> nebo).

--insert-blank-line-size

Nastavit výšku vložených prázdných řádků (v em). Výška řádků mezi odstavci bude dvojnásobek zde nastavené hodnoty.

--keep-ligatures

Zachovat ligatury přítomné ve vstupním dokumentu. Ligatura je kombinovaný znak páru znaků, jako ff, fi, fl atd. Většina čteček nemá podporu pro ligatury ve svých základních písmech, takže je nepravděpodobné, že je správně vykreslí. Ve výchozím nastavení Calibre změní ligaturu na odpovídající pár normálních znaků. Tato volba je namísto toho ponechá. Pamatujte na to, že ligatury zde znamenají pouze ligatury v jednom kódu, nikoliv ligatury vytvořené pomocí CSS nebo stylů písma.

--line-height

Výška řádku v bodech. Určuje mezery mezi za sebou následujícími řádky textu. Platí pouze pro prvky, které nedefinují svou vlastní výšku řádku. Ve většině případů je užitečnější volba minimální výška řádku. Ve výchozím nastavení se neprovádí žádné zpracování výšky řádku.

--linearize-tables

Některé chybně navržené dokumenty používají tabulky pro ovládání rozložení textu na stránce. Při převodu mají tyto dokumenty často text, který vybíhá mimo stránku a jiné artefakty. Tato volba extrahuje obsah z tabulek a zobrazí ho v linární tvaru.

--margin-bottom

Nastavit dolní okraj v bodech. Výchozí je 5.0. Nastavení na hodnotu menší než nula způsobí, že se nenastaví žádné okraje (bude zachováno nastavení okrajů v původním dokumentu). Poznámka: Formáty orientované na stránky, jako je PDF a DOCX, mají vlastní nastavení okrajů, které má přednost.

--margin-left

Nastavit levý okraj v bodech. Výchozí je 5.0. Nastavení na hodnotu menší než nula způsobí, že se nenastaví žádné okraje (bude zachováno nastavení okrajů v původním dokumentu). Poznámka: Formáty orientované na stránky, jako je PDF a DOCX, mají vlastní nastavení okrajů, které má přednost.

--margin-right

Nastavit pravý okraj v bodech. Výchozí je 5.0. Nastavení na hodnotu menší než nula způsobí, že se nenastaví žádné okraje (bude zachováno nastavení okrajů v původním dokumentu). Poznámka: Formáty orientované na stránky, jako je PDF a DOCX, mají vlastní nastavení okrajů, které má přednost.

--margin-top

Nastavit horní okraj v bodech. Výchozí je 5.0. Nastavení na hodnotu menší než nula způsobí, že se nenastaví žádné okraje (bude zachováno nastavení okrajů v původním dokumentu). Poznámka: Formáty orientované na stránky, jako je PDF a DOCX, mají vlastní nastavení okrajů, které má přednost.

--minimum-line-height

Minimální výška řádku jako procento vypočítané velikosti písma prvku. Calibre zajistí, že každý prvek bude mít výšku řádku alespoň podle tohoto nastavení bez ohledu na to, co určuje vstupní dokument. Nastavte na nulu pro zakázání. Výchozí je 120 %. Použijte toto nastavení přednostně pro přímé určení výšky řádku, pokud nevíte, co děláte. Například můžete dosáhnout textu s dvojitým řádkováním nastavením na 240.

--remove-paragraph-spacing

Odebrat mezery mezi odstavci. Také nastaví odsazení odstavce na 1,5 em. Odebrání mezer nebude fungovat, pokud zdrojový soubor nepoužívá odstavce (značky <p> nebo).

--remove-paragraph-spacing-indent-size

Když Calibre odebírá prázdné řádky mezi odstavci, automaticky nastaví odsazení odstavce, aby bylo zajištěno snadné rozlišení odstavců. Tato volba určuje šířku tohoto odsazení (v em). Pokud nastavíte zápornou hodnotu, pak bude použito odsazení zadané ve vstupním dokumentu, to znamená, že Calibre nezmění odsazení.

--smarten-punctuation

Převést jednoduché uvozovky, pomlčky a trojtečky na jejich typograficky správném ekvivalenty. Pro podrobnosti přejděte na https://daringfireball.net/projects/smartypants.

--subset-embedded-fonts

Podmnožina všech vložených písem. Každé vložené písmo je zmenšeno, aby obsahovalo pouze glyfy použité v tomto dokumentu. To zmenšuje velikost souborů písem. Užitečné, pokud vkládáte obzvlášť velké písmo se spoustou nevyužitých glyfů.

--transform-css-rules

Cesta k souboru obsahujícímu pravidla pro převod stylů CSS v této knize. Nejsnadnější způsob, jak vytvořit takový soubor, je použít průvodce pro vytváření pravidel v grafickém rozhraní Calibre. Přístup k němu získáte v oddílu "Vzhled a chování->Převod stylů" v dialogovém okně převodu. Jakmile vytvoříte pravidla, můžete použít tlačítko Exportovat pro jejich uložení do souboru.

--transform-html-rules

Cesta k souboru obsahujícímu pravidla pro převod HTML v této knize. Nejsnadnější způsob, jak vytvořit takový soubor, je použít průvodce pro vytváření pravidel v grafickém rozhraní Calibre. Přístup k němu získáte v oddílu „Vzhled a chování->Převod HTML“ v dialogovém okně převodu. Jakmile vytvoříte pravidla, můžete použít tlačítko Exportovat pro jejich uložení do souboru.

--unsmarten-punctuation

Převést ozdobné uvozovky, pomlčky a tři tečky na jejich prosté ekvivalenty.

HEURISTICKÉ ZPRACOVÁNÍ

Změnit text a strukturu dokumentu pomocí běžných vzorů. Ve výchozím nastavení zakázáno. Použijte –enable-heuristics pro povolení. Jednotlivé akce mohou být zakázány pomocí voleb –disable-*.

--disable-dehyphenate

Analyzovat rozdělená slova v celém dokumentu. Samotný dokument je použitý jako slovník pro určení, zda by měly být pomlčky zachovány nebo odebrány.

--disable-delete-blank-paragraphs

Odebrat z dokumentu prázdné odstavce, pokud se vyskytují mezi všemi ostatními odstavci

--disable-fix-indents

Změnit odsazení vytvořená z více entit pevné mezery na odsazení CSS.

--disable-format-scene-breaks

Vlevo zarovnané značky přerušení příběhu jsou zarovnané na střed. Nahradit měkká přerušení příběhu, která používají více prázdných řádků, vodorovnými čárami.

--disable-italicize-common-cases

Hledat běžná slova a vzory, které označují kurzívu a převést je na kurzívu.

--disable-markup-chapter-headings

Rozpoznat neformátované nadpisy a podnadpisy kapitol. Změnit je na značky h2 a h3. Toto nastavení nevytvoří obsah, ale může být použito společně s rozpoznáním struktury pro jeho vytvoření.

--disable-renumber-headings

Hledá výskyty po sobě následujících značek <h1> nebo <h2>. Značky jsou přečíslovány, aby se zabránilo rozdělení uprostřed nadpisů kapitol.

--disable-unwrap-lines

Zrušit zalomení řádků pomocí interpunkce a jiných vodítek formátování.

--enable-heuristics

Povolit heuristické zpracování. Tato volba musí být nastavena, aby se provedlo jakékoliv heuristické zpracování.

--html-unwrap-factor

Měřítko použité k určení délky, při které by mělo být řádku zrušeno zalomení. Platné hodnoty jsou desetinná čísla mezi 0 a 1. Výchozí je 0,4, těsně pod mediánem délky řádku. Pokud by pouze pár řádků v dokumentu vyžadovalo zrušení zalomení, měla by být tato hodnota zmenšena.

--replace-scene-breaks

Nahradit přerušení příběhu zadaným textem. Ve výchozím nastavení je použit text ze vstupního dokumentu.

NAJÍT A NAHRADIT

Změnit text a strukturu dokumentu pomocí vzorů definovaných uživatelem.

--search-replace

Cesta k souboru obsahující regulární výrazy hledávání a nahrazování. Soubor musí obsahovat střídavě řádky regulárního výrazu následované vzorem nahrazení (což může být prázdný řádek). Regulární výraz musí mít syntaxi regulárního výrazu Pythonu a soubor musí mít kódování UTF-8.

--sr1-replace

Nahrazení, které má nahradit text nalezený pomocí sr1-search.

--sr1-search

Hledaný vzor (regulární výraz), který má být nahrazen pomocí sr1-replace.

--sr2-replace

Nahrazení, které má nahradit text nalezený pomocí sr2-search.

--sr2-search

Hledaný vzor (regulární výraz), který má být nahrazen pomocí sr2-replace.

--sr3-replace

Nahrazení, které má nahradit text nalezený pomocí sr2-search.

--sr3-search

Hledaný vzor (regulární výraz), který má být nahrazen pomocí sr3-replace.

ROZPOZNÁVÁNÍ STRUKTURY

Nastavení automatického rozpoznávání struktury dokumentu.

--add-alt-text-to-img

When an <img> tag has no alt attribute, check the associated image file for metadata that specifies alternate text, and use it to fill in the alt attribute. The alt attribute improves accessibility by providing text descriptions for users who cannot see or fully interpret visual content.

--chapter

Výraz XPath pro rozpoznání nadpisů kapitol. Výchozí nastavení je považovat tagy <h1> nebo <h2> obsahující slova "chapter" (kapitola), "book" (kniha), "section" (oddíl), "prologue" (prolog), "epilogue" (epilog) nebo "part" (část) za nadpisy kapitol, stejně jako jakékoliv tagy s class="chapter" (třída kapitola). Použitý výraz musí vyhodnotit seznam prvků. Pro zakázání rozpoznávání kapitol použijte výraz "/". Viz XPath Tutorial v ráži uživatelské příručce pro další pomoc pro použití této funkce.

--chapter-mark

Zadejte, jak označit rozpoznané kapitoly. Hodnota "konecstránky" vloží před kapitoly konce stránek. Hodnota "čára" vloží před kapitoly čáru. Hodnota "žádné" zakáže označování kapitol a hodnota "obojí" použije pro označení kapitol jak konce stránek, tak čáry.

--disable-remove-fake-margins

Některé dokumenty zadávají okraje stránek zadáním levého a pravého okraje každému jednotlivému odstavci. Calibre se pokusí rozpoznat a odebrat tyto okraje. Někdy to může způsobit odebírání okrajů, které neměly být odebrány. V tomto případě můžete zakázat odebírání.

--insert-metadata

Vložit metadata knihy na její začátek. Užitečné v případě, že vaše čtečka e-knih nepodporuje zobrazení nebo vyhledávání metadat přímo.

--page-breaks-before

Výraz XPath. Konce stránek jsou vloženy před zadané prvky. Pro zakázání použijte výraz: /

--prefer-metadata-cover

Použít obálku rozpoznanou ve zdrojovém souboru přednostně před zadanou obálkou.

--remove-first-image

Odebrat první obrázek ze vstupní e-knihy. Užitečné, pokud má vstupní dokument obrázek obálky, který není identifikován jako obálka. Pokud v tomto případě nastavíte obálku v Calibre, výstupní dokument bude mít dva obrázky obálky, pokud nezadáte tuto volbu.

--start-reading-at

Výraz XPath pro rozpoznání místa v dokumentu, na kterém začít číst. Některé programy pro čtení e-knih (hlavně Kindle) používají toto místo jako pozici, na které se otevře kniha. Další nápovědu k používání této funkce naleznete v kurzu XPath v uživatelské příručce Calibre.

OBSAH

Nastavení automatického generování obsahu. Pokud má zdrojový soubor obsah, bude ve výchozím nastavení použit přednostně před automaticky generovaným.

--duplicate-links-in-toc

Při vytváření obsah z odkazů ve vstupním dokumentu povolit duplicitní položky, tj. povolit více než jednu položku se stejným textem, za předpokladu že směřují na rozdílná umístění.

--level1-toc

Výraz XPath, který určuje všechny značky, které by měly být přidány do obsahu na první úroveň. Pokud je toto zadáno, má to přednost před jinými formami automatického rozpoznávání. Příklady naleznete v kurzu XPath v uživatelské příručce Calibre.

--level2-toc

Výraz XPath, který určuje všechny značky, které by měly být přidány do obsahu na druhou úroveň. Každá položka je přidána pod předchozí položku první úrovně. Příklady naleznete v kurzu XPath v uživatelské příručce Calibre.

--level3-toc

Výraz XPath, který určuje všechny značky, které by měly být přidány do obsahu na třetí úroveň. Každá položka je přidána pod předchozí položku druhé úrovně. Příklady naleznete v kurzu XPath v uživatelské příručce Calibre.

--max-toc-links

Maximální počet odkazů vložených do obsahu. Nastavte 0 pro zakázání. Výchozí je: 50. Odkazy jsou přidány do obsahu, pouze pokud bylo rozpoznáno méně kapitol, než je mezní hodnota.

--no-chapters-in-toc

Nepřidávat automaticky rozpoznané kapitoly do obsahu.

--toc-filter

Odebrat z obsahu položky, jejichž názvy odpovídají zadanému regulárnímu výrazu. Odpovídající položky a všechny jejich podřízené položky budou odebrány.

--toc-threshold

Pokud je rozpoznáno méně než tento počet kapitol, budou odkazy přidány do obsahu. Výchozí: 6

--use-auto-toc

Pokud má již zdrojový soubor obsah, je obyčejně použit přednostně před automaticky generovaným. Pomocí této volby je vždy použit ten automaticky generovaný.

METADATA

Volby pro nastavení metadat ve výstupu

--author-sort

Řetězec použitý při řazení podle autora.

--authors

Nastavit autory. Více autorů by mělo být odděleno znaky `"&"`.

--book-producer

Nastavit nakladatele knihy.

--comments

Nastavit popis e-knihy.

--cover

Nastavit obálku na zadaný soubor nebo URL

--isbn

Nastavit ISBN knihy.

--language

Nastavit jazyk.

--pubdate

Nastavit datum vydání (předpokládá se, že je v místním časovém pásmu, pokud není časové pásmo výslovně zadáno)

--publisher

Nastavit vydavatele e-knihy.

--rating

Nastavit hodnocení. Mělo by to být číslo mezi 1 a 5.

--read-metadata-from-opf, --from-opf, -m

Číst metadata ze zadaného souboru OPF. Metadata načtená z tohoto souboru přepíší jakákoliv metadata ve zdrojovém souboru.

--series

Nastavit sérii, do které tato e-kniha patří.

--series-index

Nastavit pořadí knihy v této sérii.

--tags

Nastavit štítky pro knihu. Měl by to být seznam oddělený čárkami.

--timestamp

Nastavit časové razítko knihy (nikde se již nepoužívá)

--title

Nastavit název.

--title-sort

Verze názvu použitá při řazení.

LADĚNÍ

Volby pro pomoc s laděním převodu

--debug-pipeline, -d

Uložit výstup z rozdílných fází převodního řetězce do zadané složky. Užitečné, pokud si nejste jisti, v jaké fázi zpracování převodu se objevuje chyba.

--verbose, -v

Úroveň podrobnosti. Zadejte vícekrát pro větší podrobnost. Zadání dvakrát bude mít za následek úplnou podrobnost, jednou střední podrobnost a nulakrát nejmenší podrobnost.

13.1.8 ebook-edit

```
ebook-edit [opts] [path_to_ebook] [name_of_file_inside_book ...]
```

Spustit nástroj Calibre Upravit knihu. Volitelně můžete také zadat názvy souborů uvnitř knihy, které budou automaticky otevřeny pro úpravu.

Kdykoliv předáváte parametry pro **ebook-edit**, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

[volby]

--detach

Odpojit od řídícího terminálu, pokud nějaký je (pouze pro Linux)

--help, -h

zobrazit tuto zprávu nápovědy a ukončit

--select-text

Text, který v knize vybrat při otevírání pro úpravu

--version

zobrazit číslo verze programu a ukončit

13.1.9 ebook-meta

```
ebook-meta ebook_file [options]
```

Číst/Zapisovat metadata z/do souborů e-knih.

Podporované formáty pro čtení metadat: azw, azw1, azw3, azw4, cb7, cbc, cbr, cbz, chm, docx, epub, fb2, fbz, html, htmlz, imp, kepub, lit, lrf, lrx, mobi, odt, oebzip, opf, pdb, pdf, pml, pmlz, pobi, prc, rar, rb, rtf, snb, tpz, txt, txtz, updb, zip

Podporované formáty pro zápis metadat: azw, azw1, azw3, azw4, docx, epub, fb2, fbz, htmlz, kepub, lrf, mobi, odt, pdb, pdf, prc, rtf, tpz, txtz

Různé typy souborů podporují různé druhy metadat. Pokud se pokusíte nastavit nějaká metadata k typu souboru, který je nepodporuje, metadata budou bez upozornění ignorována.

Kdykoliv předáváte parametry pro **ebook-meta**, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

[volby]

--author-sort

Řetězec použitý při řazení podle autora. Pokud není zadán a je zadán autor, bude automaticky generován z autora.

--authors, -a

Nastavit autory. Více autorů by mělo být odděleno znakem &. Jména autorů by měla být v pořadí křestní jméno, příjmení.

--book-producer, -k

Nastavit nakladatele knihy.

--category

Nastavit kategorii knihy.

--comments, -c

Nastavit popis e-knihy.

--cover

Nastavit obálku zadaného souboru.

--date, -d

Nastavit datum vydání.

--disallow-rendered-cover

For formats like EPUB that use a "default cover" of the first page rendered, disallow such default covers

--from-opf

Načíst metadata z vybraného souboru OPF a použít je k nastavení metadat v e-knize. Metadata uvedená v příkazové řádce přepíšou metadata načtená ze souboru OPF.

--get-cover

Získat obálku z e-knihy a uložit ji jako zadaný soubor.

--help, -h

zobrazit tuto zprávu nápovědy a ukončit

--identifier

Nastavit identifikátory pro knihu, může být zadáno vícekrát. Například: *--identifier* (stránka 335) uri:https://acme.com *--identifier* (stránka 335) isbn:12345. Pro odebrání identifikátoru zadejte prázdnou hodnotu, *--identifier* (stránka 335) isbn:. Pamatujte, že identifikátor označený jako identifikátor balíčku nemůže být u souborů EPUB odebrán.

--index, -i

Nastavit pořadí knihy v této sérii.

--isbn

Nastavit ISBN knihy.

--language, -l

Nastavit jazyk.

--lrf-bookid

Nastavit BookID v souborech LRF

--publisher, -p

Nastavit vydavatele e-knihy.

--rating, -r

Nastavit hodnocení. Mělo by to být číslo mezi 1 a 5.

--series, -s

Nastavit sérii, do které tato e-kniha patří.

--tags

Nastavit štítky pro knihu. Měl by to být seznam oddělený čárkami.

--title, -t

Nastavit název.

--title-sort

Verze názvu použitá pro řazení. Pokud není zadán a je zadán název, bude automaticky generován z názvu.

--to-opf

Zadejte název souboru OPF. Metadata budou zapsána do souboru OPF.

--version

zobrazit číslo verze programu a ukončit

13.1.10 ebook-polish

```
ebook-polish [options] input_file [output_file]
```

Vylepšení knih je celé o dotažení vašich pečlivě vytvářených e-knih k dokonalosti.

Vylepšení se pokouší minimalizovat změny vnitřního kódu vaší e-knihy. Na rozdíl od převodu nedělá následující: neuhlazuje CSS, nepřejmenovává soubory, nemění velikosti písma, neupravuje okraje atd. Každá akce provádí pouze minimální sadu změn potřebnou pro požadovaný efekt.

Tento nástroj byste měli použít jako poslední krok při vytváření e-knihy.

Pamatujte, že vylepšení funguje pouze u souborů ve formátech AZW3 nebo EPUB nebo KEPUB.

Kdykoliv předáváte parametry pro **ebook-polish**, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

[volby]

--add-soft-hyphens, -H

Přidat pohyblivé spojovníky do všech slov v knize. To umožní knihu lépe vykreslit, když je text zarovnán do bloku, ve čtečkách, které nepodporují dělení slov.

--compress-images, -i

Bezztrátově komprimovat obrázky v knize, aby se zmenšila velikost souboru bez ovlivnění kvality obrázku.

--cover, -c

Cesta k obrázku obálky. Mění obálku zadanou v e-knize. Pokud není přítomna žádná obálka, nebo pokud není obálka správně identifikována, vloží novou obálku.

--download-external-resources, -d

Download external resources such as images, stylesheets, etc. that point to URLs instead of files in the book. All such resources will be downloaded and added to the book so that the book no longer references any external resources.

--embed-fonts, -e

Vložit všechna písma, na která je odkazováno v dokumentu a nejsou již vložena. To vyhledá písma ve vašem počítači, a pokud jsou nalezena, budou vložena do dokumentu. Ujistěte se, že máte řádnou licenci pro vkládání písem použitých v tomto dokumentu.

--help, -h

zobrazit tuto zprávu nápovědy a ukončit

--jacket, -j

Vložit na začátek knihy stránku "přebalu knihy", která obsahuje všechna metadata knihy, jako jsou název, štítky, autoři, série, komentáře atd. Jakýkoliv předchozí přebal knihy bude nahrazen.

--opf, -o

Cesta k souboru OPF. Metadata v knize jsou aktualizována ze souboru OPF.

--remove-jacket

Odebrat dříve vloženou stránku přebalu knihy.

--remove-soft-hyphens

Odebrat pohyblivé spojovníky z celého textu v knize.

--remove-unused-css, -u

Odebrat všechna nepoužívaná pravidla CSS ze šablon stylů a značek <style>. Některé knihy vytvořené pomocí výrobních šablon mohou mít velký počet dodatečných pravidel CSS, která neodpovídají skutečnému obsahu. Tato dodatečná pravidla mohou zpomalovat čtečky, které je musí všechny analyzovat.

--smarten-punctuation, -p

Převést pomlčky, tři tečky, uvozovky, vícenásobné pomlčky atd. v prostém textu na jejich typograficky správné ekvivalenty. Pamatujte, že algoritmus může někdy generovat nesprávné výsledky, obzvláště pokud se to týká jednoduchých uvozovek na začátku zkrácených tvarů.

--subset-fonts, -f

Částečné vkládání písem znamená zmenšení vloženého písma, aby obsahovalo pouze znaky použité z tohoto písma v knize. To značně zmenšuje velikost souborů písem (poloviční velikosti souborů písem jsou běžné). Například pokud kniha používá konkrétní písmo pro nadpisy, pak částečné vkládání zmenší toto písmo, aby obsahovalo pouze znaky přítomné v nadpisech v knize. Nebo pokud kniha vloží verze písma pro tučné písmo a kurzívu, ale texty tučným písmem a kurzívou jsou poměrně vzácné nebo úplně chybí, pak může být tučné písmo a kurzíva buď zmenšeno, nebo úplně odebráno. Jediná nevýhoda částečného vkládání písem je ta, že pokud se později rozhodnete přidat další text do svých knih, nově přidaný text nemusí být pokryt podmnožinou písma.

--upgrade-book, -U

Upgrade vnitřní struktury knihy, pokud je to možné. Například upgraduje knihy EPUB 2 na knihy EPUB 3.

--verbose

Vytváří podrobnější výstup, užitečné pro ladění.

--version

zobrazit číslo verze programu a ukončit

13.1.11 ebook-viewer

```
ebook-viewer [options] file
```

Zobrazit e-knihu.

Kdykoliv předáváte parametry pro **ebook-viewer**, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

[volby]

--continue

Pokračovat ve čtení naposledy otevřené knihy

--detach

Odpojit od řídícího terminálu, pokud nějaký je (pouze pro Linux)

--force-reload

Vynutit nové načtení všech otevřených knih

--full-screen, --fullscreen, -f

Pokud je zadáno, okno Prohlížeče e-knih se pokusí při spuštění otevřít na celou obrazovku.

--help, -h

zobrazit tuto zprávu nápovědy a ukončit

--new-instance

Otevřít nové okno prohlížeče, i když je nastavena možnost použít pouze jedno okno prohlížeče.

--open-at

Pozice, na které otevřít zadanou knihu. Pozice je umístění nebo pozice, které můžete získat pomocí akce Přejít na->Umístění v ovládacích prvcích prohlížeče. Případně můžete použít formu toc:něco, což se otevře na umístění první položky obsahu, která obsahuje řetězec "něco". Forma toc-href:něco bude odpovídat href (interní cíl odkazu)

uzlům obsahu. Shoda je přesná. Pokud chcete porovnat podřetězec, použijte formu toc-href-contains:něco. Forma ref:něco použije odkazy Režimu odkazu. Forma regex:něco bude po otevření knihy hledat regulární výraz něco.

--raise-window

Pokud je zadáno, okno Prohlížeče e-knih se pokusí při spuštění zobrazit v popředí.

--version

zobrazit číslo verze programu a ukončit

13.1.12 fetch-ebook-metadata

```
fetch-ebook-metadata [options]
```

Načíst metadata knihy z online zdrojů. Musíte zadat alespoň název, autory nebo ISBN.

Kdykoliv předáváte parametry pro **fetch-ebook-metadata**, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

[volby]

--allowed-plugin, -p

Zadejte název modulu pro stahování metadat, který chcete použít. Ve výchozím nastavení budou použity všechny moduly metadat. Může být zadáno vícekrát pro více modulů. Všechny názvy modulů: Google, Google Images, Amazon.com, Edelweiss, Open Library, Big Book Search

--authors, -a

Autoři knihy

--cover, -c

Zadejte název souboru. Pokud existuje obálka, bude do něj uložena. Bez této volby nebude stažena žádná obálka.

--help, -h

zobrazit tuto zprávu nápovědy a ukončit

--identifier, -I

Identifikátory, jako jsou ASIN nebo ID Goodreads atd. Mohou být zadány vícekrát pro více identifikátorů. Například:*--identifier* (stránka 338) asin:B0082BAJA0

--isbn, -i

ISBN knihy

--opf, -o

Výstup metadat ve formátu OPF namísto textu čitelného pro člověka.

--timeout, -d

Časový limit v sekundách. Výchozí je 30

--title, -t

Název knihy

--verbose, -v

Vypsat protokol do konzole (stderr)

--version

zobrazit číslo verze programu a ukončit

13.1.13 lrf2lrs

```
lrf2lrs book.lrf
```

Převést soubor LRF na soubor LRS (XML s kódováním UTF-8)

Kdykoliv předáváte parametry pro **lrf2lrs**, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

[volby]

--dont-output-resources
: Neukládat vložené soubory obrázků a písem na disk

--help, -h
: zobrazit tuto zprávu nápovědy a ukončit

--output, -o
: Výstupní soubor LRS

--verbose
: Více podrobností

--version
: zobrazit číslo verze programu a ukončit

13.1.14 lrfviewer

```
lrfviewer [options] book.lrf
```

Číst e-knihu LRF book.lrf

Kdykoliv předáváte parametry pro **lrfviewer**, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

[volby]

--disable-hyphenation
: Zakázat dělení slov. Mělo by to výrazně zrychlit vykreslování.

--help, -h
: zobrazit tuto zprávu nápovědy a ukončit

--profile
: Profilovat nástroj pro vykreslování LRF

--verbose
: Vypsat více informací o procesu vykreslování

--version
: zobrazit číslo verze programu a ukončit

--visual-debug
: Zapnout vizuální pomůcky pro ladění modulu vykreslování

--white-background
: Ve výchozím nastavení je pozadí skoro bílé, protože je to snadnější pro oči. Použijte tuto volbu, aby bylo pozadí čistě bílé.

13.1.15 lrs2lrf

```
lrs2lrf [options] file.lrs
```

Kompilovat soubor LRS na soubor LRF.

Kdykoliv předáváte parametry pro **lrs2lrf**, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

[volby]

--help, -h

zobrazit tuto zprávu nápovědy a ukončit

--lrs

Převést LRS na LRS, užitečné pro ladění.

--output, -o

Cesta k výstupnímu souboru

--verbose

Podrobné zpracování

--version

zobrazit číslo verze programu a ukončit

13.1.16 web2disk

```
web2disk URL
```

Kde URL je například https://google.com

Kdykoliv předáváte parametry pro **web2disk**, který mají mezery, uzavřete parametry uvozovkami. Například: „/some path/with spaces

[volby]

--base-dir, -d

Základní složka, do které je uložena URL. Výchozí je .

--delay

Minimální interval mezi za sebou následujícími načteními v sekundách. Výchozí je 0 s

--dont-download-stylesheets

Nestahovat šablony stylů CSS.

--encoding

Kódování znaků webových stránek, které se pokoušíte stáhnout. Výchozí je pokusit se odhadnout kódování.

--filter-regexp

Jakýkoliv odkaz, který odpovídá tomuto regulárnímu výrazu, bude ignorován. Tato volba může být zadána vícekrát, v tom případě dokud kterýkoliv regulární výraz odpovídá odkazu, bude ignorován. Pokud je zadáno filtrování regulárních výrazů a porovnání regulárních výrazů, pak filtrování regulárních výrazů je použito první.

--help, -h

zobrazit tuto zprávu nápovědy a ukončit

--match-regexp

Pouze odkazy, které odpovídají tomuto regulárnímu výrazu, budou následovány. Tato volba může být zadána vícekrát, v tom případě bude odkaz následován, dokud odpovídá kterémukoliv regulárnímu výrazu. Ve výchozím nastavení jsou následovány všechny odkazy.

--max-files, -n

Maximální počet stahovaných souborů. Toto se použije pouze na soubory v <a href>štítcích. Výchozí je 9223372036854775807

--max-recursions, -r

Maximální počet úrovní rekurze, tj. hloubka sledovaných odkazů. Výchozí 1

--timeout, -t

Časový limit čekání na odezvu ze serveru v sekundách. Výchozí: 10.0 s

--verbose

Zobrazit podrobné výstupní informace. Užitečné pro ladění

--version

zobrazit číslo verze programu a ukončit

13.2 Nedokumentované příkazy

- ebook-device
- markdown-calibre

Použití pro nedokumentované příkazy můžete vidět jejich provedením bez parametrů v terminálu.

KAPITOLA 14

Nastavení vývojového prostředí Calibre

Calibre je celé open source pod licencí GNU GPL v3[118]. To znamená, že můžete volně stáhnout a upravit program podle své libosti. V této části se naučíte, jak nastavit vývojové prostředí Calibre na vámi zvoleném operačním systému. Calibre je napsáno převážně v Pythonu[119] s nějakým kódem C/C++ kvůli rychlosti a systémovému rozhraní. Pamatujte, že Calibre vyžaduje alespoň Python 3.8.

Obsah

- *Filozofie návrhu* (stránka 344)
  - *Rozložení kódu* (stránka 344)
- *Získání kódu* (stránka 345)
  - *Odesílání vašich změn, které mají být zahrnuty* (stránka 345)
- *Vývojové prostředí Windows* (stránka 346)
- *Vývojové prostředí macOS* (stránka 347)
- *Vývojové prostředí Linux* (stránka 348)
- *Mít samostatnou „normální“ a „vývojovou“ instalaci Calibre na stejném počítači* (stránka 348)
- *Tipy k ladění* (stránka 349)
  - *Používání příkazů tisku* (stránka 349)
  - *Používání interaktivního překladače Pythonu* (stránka 349)
  - *Používání ladicího programu Pythonu jako vzdáleného ladicího programu* (stránka 349)
  - *Používání ladicího programu ve vašem oblíbeném IDE Pythonu* (stránka 350)
  - *Provádění libovolných skriptů uvnitř prostření Pythonu Calibre* (stránka 350)

[118] https://www.gnu.org/licenses/gpl.html
[119] https://www.python.org

- *Používání Calibre ve vašich projektech* (stránka 350)
  - *Binární instalace Calibre* (stránka 351)
  - *Zdrojová instalace na Linuxu* (stránka 351)
- *Dokumentace API pro různé části Calibre* (stránka 351)

14.1 Filozofie návrhu

Calibre má své kořeny ve světě Unixu, což znamená, že jeho design je vysoce modulární. Moduly se navzájem ovlivňují prostřednictvím dobře definovaných rozhraní. To umožňuje velice snadné přidávání nových funkcí a opravování chyb v Calibre, což má za následek zběsilé tempo vývoje. Vzhledem ke svým kořenům má Calibre komplexní rozhraní příkazového řádku pro všechny své funkce, zdokumentované v *Rozhraní příkazového řádku* (stránka 303).

Modulární návrh Calibre je vyjádřen prostřednictvím `Modulů`. K dispozici je *kurz* (stránka 255) psaní modulů Calibre . Například přidání podpory pro nové zařízení do Calibre typicky zahrnuje napsání méně než 100 řádků kódu v podobě modulu ovladače zařízení. Vestavěné ovladače[120] můžete procházet. Podobně přidání podpory pro nové formáty převodu zahrnuje napsání modulu vstupně-výstupního formátu. Dalším příkladem modulárního návrhu je *systém předpisů* (stránka 33) pro načítání zpráv. Pro více příkladů modulů navržených pro přidání funkcí do Calibre se podívejte na Rejstřík modulů[121]..

14.1.1 Rozložení kódu

Všechen kód Pythonu Calibre je v balíčku `calibre`. Tento balíček obsahuje následující hlavní podbalíčky

- zařízení – Všechny ovladače zařízení. Prohlédněte si některé z vestavěných ovladačů pro získání představy o tom, jak fungují.
  - Pro podrobnosti se podívejte na `devices.interface`, které definuje rozhraní podporované ovladači zařízení, a `devices.usbms`, které definuje obecný ovladač, který se připojuje k zařízení USBMS. Všechny ovladače v Calibre založené na USBMS se z něj dědí.
- e-knihy – Všechen kód převodu a metadat e-knih. Vhodným výchozím bodem je balíček `calibre.ebooks.conversion.cli`, což je modul poháněný příkazem **`ebook-convert`**. Zpracování převodu je řízeno prostřednictvím `conversion.plumber`. Kód nezávislý na formátu je všechen `ebooks.oeb` a kód závislý na formátu je v `ebooks.format_name`.
  - Čtení, zápis a stahování jsou v `ebooks.metadata`
  - Převod probíhá v řetězci, pro strukturu řetězce se podívejte na *Úvod* (stránka 63). Řetězec se skládá ze vstupního modulu, různých transformací a výstupního modulu. Kód, který vytvoří a řídí řetězec se nachází v `plumber.py`. Řetězec funguje na základě zobrazení e-knihy, které je jako rozbalené EPUB, s manifestem, páteří, obsahem, průvodcem, obsahem html atd. Třída, která spravuje toto zobrazení je OEBBook v ``ebooks.oeb.base`. Různé transformace, které jsou použity na knihu během převodu jsou v `oeb/transforms/*.py`. A vstupní a výstupní moduly jsou v `conversion/plugins/*.py`.
  - Úprava e-knihy se odehrává pomocí různých objektů kontejneru. To je zdokumentováno v *Dokumentace API pro nástroje pro úpravu e-knih* (stránka 361).
- db – Serverová část databáze. Pro rozhraní do knihovny Calibre se podívejte na *Dokumentace API pro rozhraní databáze* (stránka 351).
- Server s obsahem: `srv` je Server s obsahem Calibre.

[120] https://github.com/kovidgoyal/calibre/tree/master/src/calibre/devices
[121] https://www.mobileread.com/forums/showthread.php?p=1362767#post1362767

- gui2 – Grafické uživatelské rozhraní. Inicializace grafického rozhraní se odehrává v `gui2.main` a `gui2.ui`. Prohlížeč e-knih je v `gui2.viewer`. Editor e-knih je v ``gui2.tweak_book`.

Pokud chcete najít vstupní body pro všechny jednotlivé spustitelné soubory Calibre, podívejte se na strukturu `entry_points` v linux.py[122].

Pokud potřebujete pomoct pochopit kód, odešlete příspěvek do vývojářského fóra[123] a s největší pravděpodobností získáte pomoc od jednoho z mnoha vývojářů Calibre.

14.2 Získání kódu

Zdrojový kód Calibre můžete získat dvěma způsoby, pomocí systému řízení verzí nebo přímo stažením tarball[124].

Calibre používá Git[125], distribuovaný systém řízení verzí. Git je dostupný na všech platformách, které Calibre podporuje. Po nainstalování Gitu, můžete získat zdrojový kód Calibre pomocí příkazu:

```
git clone https://github.com/kovidgoyal/calibre.git
```

Na Windows budete potřebovat úplný název cesty, který bude něco jako `C:\Program Files\Git\git.exe`.

Calibre je velice rozsáhlý projekt s velmi dlouhou historií řízení zdroje, takže výše uvedené může nějakou dobu trvat (10 minut až hodinu v závislosti na vaší rychlosti internetu).

Pokud chcete získat kód rychleji, zdrojový kód pro nejnovější vydanou verzi je vždy dostupný jako archive[126].

Pro aktualizaci větve na nejnovější kód použijte příkaz:

```
git pull --no-edit
```

Také můžete procházet kód na GitHub[127].

14.2.1 Odesílání vašich změn, které mají být zahrnuty

Pokud plánujete pouze provést několik malých změn, můžete provést změny a vytvořit „direktivu sloučení“, kterou pak můžete připojit k oznámení v přehledu chyb[128] Calibre. Abyste to udělali, proveďte změny, pak spusťte:

```
git commit -am "Comment describing your changes"
git format-patch origin/master --stdout > my-changes
```

Tím se vytvoří soubor `my-changes` v aktuální složce, jednoduše ho připojte k oznámení na přehledu chyb[129] Calibre. Pamatujte, že to bude zahrnovat *všechna* potvrzení změn, které jste provedli. Pokud chcete odeslat pouze některá potvrzení, musíte změnit výše uvedený `origin/master`. Pro odeslání pouze posledního potvrzení použijte:

```
git format-patch HEAD~1 --stdout > my-changes
```

Pro odeslání posledních *n* potvrzení nahraďte *1* pomocí *n*, například pro poslední 3 potvrzení:

```
git format-patch HEAD~3 --stdout > my-changes
```

[122] https://github.com/kovidgoyal/calibre/blob/master/src/calibre/linux.py
[123] https://www.mobileread.com/forums/forumdisplay.php?f=240
[124] https://calibre-ebook.com/dist/src
[125] https://www.git-scm.com/
[126] https://calibre-ebook.com/dist/src
[127] https://github.com/kovidgoyal/calibre
[128] https://bugs.launchpad.net/calibre
[129] https://bugs.launchpad.net/calibre

Buďte opatrní, abyste nezahrnuli sloučení, když použijete `HEAD~n`.

Pokud máte v plánu udělat hodně vyvíjet Calibre, pak je nejlepší metodou vytvořit si účet na GitHub[130]. Níže je základní průvodce pro nastavení vaší lastní větve Calibre takovým způsobem, který vám umožní předkládat požadavky o zahrnutí do hlavního úložiště Calibre:

- Nastavte git ve svém počítači, jak je popsáno v tomto článku: Set Up Git[131]
- Nastavte klíče ssh pro ověřování na GitHub, jak je popsáno zde: Generating an SSH key[132]
- Přejděte na https://github.com/kovidgoyal/calibre a klikněte na tlačítko *Fork*.
- V terminálu zadejte:

```
git clone git@github.com:<username>/calibre.git
git remote add upstream https://github.com/kovidgoyal/calibre.git
```

  Nahraďte výše uvedené <username> svým uživatelským jménem na GitHub. Tím místně odsouhlasíte svůj fork.

- Můžete provádět změny a potvrzovat je, kdykoliv budete chtít. Až budete připraveni sloučit svou práci, zadejte:

```
git push
```

  a přejděte na `https://github.com/<username>/calibre` a klikněte na tlačítko *Pull Request* pro vygenerování požadavku, který může být sloučen.

- Svou lokální kopii můžete kdykoliv aktualizovat kódem z hlavního úložiště zadáním:

```
git pull upstream
```

Také byste měli dávat pozor na vývojové fórum[133] Calibre. Před provedením zásadních změn byste měli o nich diskutovat na fóru nebo se přímo obrátit na Kovida (jeho e-mailová adresa je v celém zdrojovém kódu).

14.3 Vývojové prostředí Windows

Poznámka

Musíte také získat zdrojový kód Calibre samostatně, jak je popsáno výše.

Nainstalujte normálně Calibre pomocí instalátoru Windows[134]. Pak otevřete příkazový řádek a přejděte do dříve odsouhlasené složky s kódem Calibre. Například:

```
cd C:\Users\kovid\work\calibre
```

calibre je složka, která obsahuje podsložky src a resources.

Dalším krokem je nastavení proměnné prostředí `CALIBRE_DEVELOP_FROM` na absolutní cestu složky src. Takže v návaznosti na výše uvedený příklad by to bylo `C:\Users\kovid\work\calibre\src`. Zde je krátký průvodce[135] k nastavení proměnných prostředí v systému Windows.

Jakmile nastavíte proměnnou prostředí, otevřete nový příkazový řádek a zkontrolujte, že je správně nastavena pomocí příkazu:

[130] https://github.com
[131] https://help.github.com/articles/set-up-git
[132] https://help.github.com/articles/generating-ssh-keys
[133] https://www.mobileread.com/forums/forumdisplay.php?f=240
[134] https://calibre-ebook.com/download_windows
[135] https://docs.python.org/using/windows.html#excursus-setting-environment-variables

```
echo %CALIBRE_DEVELOP_FROM%
```

Nastavení této proměnné prostředí znamená, že Calibre bude nyní načítat všechen svůj kód Pythonu ze zadaného umístění.

A je to! Nyní jste připraveni upravovat kód Calibre. Například otevřete soubor `src\calibre\__init__.py` ve svém oblíbeném editoru a přidejte řádek:

```
print("Hello, world!")
```

na začátek souboru. Nyní spusťte příkaz **calibredb**. Úplně první řádek výstupu by měl být `Hello, world!`.

Pokud chcete, můžete také nastavit vývojové prostředí Calibre uvnitř bezplatného programu Microsoft Visual Studio podle pokynů zde[136].

14.4 Vývojové prostředí macOS

Poznámka

Musíte také získat zdrojový kód Calibre samostatně, jak je popsáno výše.

Nainstalujte normálně Calibre pomocí poskytnutého .dmg[137]. Pak otevřete terminál a přejděte do dříve odsouhlasené složky s kódem Calibre, například:

```
cd /Users/kovid/work/calibre
```

calibre je složka, která obsahuje podsložky src a resources. Nástroje příkazového řádku Calibre se nacházejí uvnitř balíčku aplikací Calibre, v `/Applications/calibre.app/Contents/MacOS` byste měli tuto složku přidat do proměnné prostředí PATH, pokud chcete snadno spouštět nástroje příkazového řádku.

Dalším krokem je vytvoření bashový skript, který nastaví proměnnou prostředí `CALIBRE_DEVELOP_FROM` na absolutní cestu složky src při spuštění Calibre v režimu ladění.

Vytvořte textový soubor:

```
#!/bin/sh
export CALIBRE_DEVELOP_FROM="/Users/kovid/work/calibre/src"
calibre-debug -g
```

Uložte tento soubor jako `/usr/bin/calibre-develop`, pak nastavte jeho oprávnění tak, aby mohl být spuštěn:

```
chmod +x /usr/local/bin/calibre-develop
```

Jakmile toto uděláte, spusťte:

```
calibre-develop
```

Měli byste vidět nějaké diagnostické informace v okně terminálu při spuštění Calibre a měli byste vidět hvězdičku za číslem verze v okně grafického rozhraní, která znamená, že spouštíte ze zdrojového kódu.

[136] https://www.mobileread.com/forums/showthread.php?t=251201
[137] https://calibre-ebook.com/download_osx

14.5 Vývojové prostředí Linux

Poznámka

Musíte také získat zdrojový kód Calibre samostatně, jak je popsáno výše.

Calibre je vyvíjeno především na Linuxu. Při nastavování vývojového prostředí máte dvě možnosti. Můžete nainstalovat binární kód Calibre jako normálně a použít ho jako běhové prostředí pro váš vývoj. Tento přístup je podobný tomu použitému ve Windows a macOS. Případně můžete nainstalovat Calibre ze zdrojového kódu. Pokyny pro nastavení vývojového prostředí ze zdrojového kódu jsou v souboru INSTALL ve stromu zdrojového kódu. Zde budeme adresovat pomocí binárního kódu jako běhového prostředí, což je doporučená metoda.

Nainstalujte Calibre pomocí instalátoru binárního kódu[138]. Pak otevřete terminál a přejděte do dříve odsouhlasené složky s kódem Calibre, například:

```
cd /home/kovid/work/calibre
```

calibre je složka, která obsahuje podsložky src a resources.

Dalším krokem je nastavení proměnné prostředí `CALIBRE_DEVELOP_FROM` na absolutní cestu složky src. Takže v návaznosti na výše uvedený příklad by to bylo `/home/kovid/work/calibre/src`. Jak nastavit proměnné prostředí závisí na vaší distribuci Linuxu a jaké prostředí používáte.

Poznámka

It is recommended to use the binary installer provided from upstream. Should you insist on using a package provided by your distribution, use the `CALIBRE_PYTHON_PATH` and `CALIBRE_RESOURCES_PATH` variables instead. You can get these by running `calibre-debug --paths`. But note that distribution calibre packages are often terminally broken and are completely unsupported.

Jakmile nastavíte proměnnou prostředí, otevřete nový terminál a zkontrolujte, že je správně nastavena pomocí příkazu:

```
echo $CALIBRE_DEVELOP_FROM
```

Nastavení této proměnné prostředí znamená, že Calibre bude nyní načítat všechen svůj kód Pythonu ze zadaného umístění.

A je to! Nyní jste připraveni upravovat kód Calibre. Například otevřete soubor `src/calibre/__init__.py` ve svém oblíbeném editoru a přidejte řádek:

```
print("Hello, world!")
```

na začátek souboru. Nyní spusťte příkaz **`calibredb`**. Úplně první řádek výstupu by měl být `Hello, world!`.

14.6 Mít samostatnou „normální“ a „vývojovou“ instalaci Calibre na stejném počítači

Zdrojový strom Calibre je velice stabilní a zřídka kdy spadne, ale pokud cítíte potřebu spouštět ze zdroje na samostatné zkušební knihovně a spouštět vydanou verzi Calibre se svou každodenní knihovnou, můžete toho snadno dosáhnout pomocí souborů .bat nebo skriptů prostředí pro spuštění Calibre. Níže uvedený příklad ukazuje, jak to provést v systému Windows pomocí souborů .bat (pokyny pro ostatní platformy jsou stejné, stačí použít skript prostředí namísto souboru .bat)

[138] https://calibre-ebook.com/download_linux

Pro spuštění vydané verze Calibre s vaší každodenní knihovnou:

calibre-normal.bat:

```
calibre.exe "--with-library=C:\path\to\everyday\library folder"
```

calibre-dev.bat:

```
set CALIBRE_DEVELOP_FROM=C:\path\to\calibre\checkout\src
calibre.exe "--with-library=C:\path\to\test\library folder"
```

14.7 Tipy k ladění

Python je dynamicky napsaný jazyk s vynikajícím vybavením pro introspekci. Kovid napsal kód jádra Calibre, aniž by jedinkrát použil ladicí program. Existuje mnoho strategií ladění kódu Calibre:

14.7.1 Používání příkazů tisku

Toto je Kovidův oblíbený způsob ladění. Jednoduše vložte příkazy tisku do nejdůležitějších bodů a spusťte program v terminálu. Grafické rozhraní můžete spustit z terminálu například takto:

```
calibre-debug -g
```

Podobně můžete spustit Prohlížeč e-knih takto:

```
calibre-debug -w /path/to/file/to/be/viewed
```

Editor e-knih může být spuštěn takto:

```
calibre-debug --edit-book /path/to/be/edited
```

14.7.2 Používání interaktivního překladače Pythonu

Můžete vložit následující dva řádky kódu pro spuštění interaktivní relace Pythonu v tomto bodě:

```
from calibre import ipython
ipython(locals())
```

Při spuštění z příkazového řádku toto spustí interaktivní překladač Pythonu s přístupem ke všem místně definovaným proměnným (proměnné v místním rozsahu). Interaktivní výzva má dokonce dokončení `Tab` pro vlastnosti objektů a můžete použít různé vybavení Pythonu pro introspekci, například `dir()`, `type()`, `repr()`, atd.

14.7.3 Používání ladicího programu Pythonu jako vzdáleného ladicího programu

Můžete použít vestavěný ladicí program Pythonu (pdb) jako vzdálený ladicí program z příkazového řádku. Nejdříve takto spusťte vzdálený ladicí program v místě v kódu Calibre, které vás zajímá:

```
from calibre.rpdb import set_trace
set_trace()
```

Pak spusťte Calibre, a to buď jako normálně, nebo pomocí jednoho z ladicích příkazů Calibre popsaných v předchozím oddílu. Jakmile je dosažen výše uvedený bod v kódu, Calibre se zablokuje a počká na připojení ladicího programu.

Nyní otevřete terminálu nebo příkazový řádek a použijte následující příkaz pro spuštění relace ladění:

```
calibre-debug -c "from calibre.rpdb import cli; cli()"
```

Jak používat ladicí program Pythonu, si můžete přečíst v dokumentaci stdlib Pythonu pro modul pdb[139].

Poznámka

Ve výchozím nastavení se vzdálený ladicí program pokusí připojit na portu 4444. Můžete jej změnit předáním parametru portu výše uvedeným funkcím set_trace() a cli() takto : `set_trace(port=1234)` a `cli(port=1234)`.

Poznámka

Ladicí program Pythonu nemůže zpracovat více vláken, takže musíte volat set_trace pro každé vlákno, pokaždé s jiným číslem portu.

14.7.4 Používání ladicího programu ve vašem oblíbeném IDE Pythonu

Je možné použít vestavěný ladicí program ve vašem oblíbeném IDE Pythonu, pokud podporuje vzdálené ladění. Prvním krokem je přidání rezervační složky src Calibre do `PYTHONPATH` ve vašem IDE. Jinými slovy, složka nastavená výše jako `CALIBRE_DEVELOP_FROM`, musí být také v `PYTHONPATH` vašeho IDE.

Pak umístěte modul vzdáleného ladicího programu IDE do podsložky `src` rezervační složky zdrojového kódu Calibre. Přidejte jakýkoliv potřebný kód pro spuštění vzdáleného ladicího programu do Calibre v důležitém místě, například do hlavní funkce. Pak spusťte Calibre jako normálně. Váš IDE by nyní měl být schopný se připojit ke vzdálenému ladicímu programu běžícímu uvnitř Calibre.

14.7.5 Provádění libovolných skriptů uvnitř prostření Pythonu Calibre

Příkaz **`calibre-debug`** poskytuje několik šikovných přepínačů pro provádění vašeho kódu s přístupem k modulům Calibre:

```
calibre-debug -c "some Python code"
```

je skvělý pro testování malého fragmentu kódu na příkazovém řádku. Funguje to stejným způsobem jako přepínač -c pro překladač Pythonu:

```
calibre-debug myscript.py
```

může být použit ke spuštění vašeho skriptu Python. Funguje to stejným způsobem jako předání skriptu překladači Pythonu, kromě toho, že prostředí Calibre je plně inicializované, takže můžete ve svém skriptu použít celý kód Calibre. Pro použití parametrů příkazového řádku s vaším skriptem použijte formulář:

```
calibre-debug myscript.py -- --option1 arg1
```

`--` způsobí, že všechny následující parametry budou předány do vašeho skriptu.

14.8 Používání Calibre ve vašich projektech

Je možné přímo použít funkce nebo kód Calibre ve vašem projektu v Pythonu. Existují dva způsoby, jak to udělat:

[139] https://docs.python.org/library/pdb.html#debugger-commands

14.8.1 Binární instalace Calibre

Pokud máte binární instalaci Calibre, můžete použít překladač Pythonu nainstalovaný s Calibre takto:

```
calibre-debug /path/to/your/python/script.py -- arguments to your script
```

14.8.2 Zdrojová instalace na Linuxu

Kromě použití výše uvedené techniky, pokud provedete zdrojovou instalaci na Linuxu, můžete také přímo importovat Calibre následujícím způsobem:

```
import init_calibre
import calibre

print(calibre.__version__)
```

Je nezbytné, abyste importovali modul init_calibre před jakýmikoliv jinými moduly nebo balíčky Calibre, protože nastavuje překladač ke spuštění kódu Calibre.

14.9 Dokumentace API pro různé části Calibre

14.9.1 Dokumentace API pro rozhraní databáze

Toto rozhraní API je bezpečné z hlediska podprocesů (používá schéma vícenásobné čtečky, zamykání jednoduchého zapisovače). K tomuto rozhraní API můžete přistupovat takto:

```
from calibre.library import db
db = db('Path to calibre library folder').new_api
```

Pokud jste v modulu Calibre, který je součástí hlavního grafického rozhraní Calibre, získáte k němu namísto toho přístup takto:

```
db = self.gui.current_db.new_api
```

class calibre.db.cache.**Cache**(*backend*, *library_database_instance=None*)

An in-memory cache of the metadata.db file from a calibre library. This class also serves as a threadsafe API for accessing the database. The in-memory cache is maintained in normal form for maximum performance.

SQLITE is simply used as a way to read and write from metadata.db robustly. All table reading/sorting/searching/caching logic is re-implemented. This was necessary for maximum performance and flexibility.

class EventType(**values*)

book_created = 4

When a new book record is created in the database, with the book id as the only argument

book_edited = 8

When a book format is edited, with arguments: (book_id, fmt)

books_removed = 5

When books are removed from the database with the list of book ids as the only argument

format_added = 2

When a format is added to a book, with arguments: (book_id, format)

formats_removed = 3

When formats are removed from a book, with arguments: (mapping of book id to set of formats removed from the book)

indexing_progress_changed = 9

When the indexing progress changes

items_removed = 7

When items such as tags or authors are removed from some books. Arguments: (field_name, affected book ids, ids of removed items)

items_renamed = 6

When items such as tags or authors are renamed in some or all books. Arguments: (field_name, affected book ids, map of old item id to new item id)

links_changed = 11

When the links associated with items(s) are changed, with arguments: (field_name, item_ids)

metadata_changed = 1

When some metadata is changed for some books, with arguments: (name of changed field, set of affected book ids)

notes_changed = 10

When the notes associated with item(s) are changed, with arguments: (field_name, item_ids)

add_books(*books*, *add_duplicates=True*, *apply_import_tags=True*, *preserve_uuid=False*, *run_hooks=True*, *dbapi=None*)

Add the specified books to the library. Books should be an iterable of 2-tuples, each 2-tuple of the form `(mi, format_map)` where mi is a Metadata object and format_map is a dictionary of the form `{fmt: path_or_stream}`, for example: `{'EPUB': '/path/to/file.epub'}`.

Returns a pair of lists: `ids, duplicates`. `ids` contains the book ids for all newly created books in the database. `duplicates` contains the `(mi, format_map)` for all books that already exist in the database as per the simple duplicate detection heuristic used by *has_book()* (stránka 357).

add_custom_book_data(*name*, *val_map*, *delete_first=False*)

Add data for name where val_map is a map of book_ids to values. If delete_first is True, all previously stored data for name will be removed.

add_extra_files(*book_id*, *map_of_relpath_to_stream_or_path*, *replace=True*, *auto_rename=False*)

Add extra data files

add_format(*book_id*, *fmt*, *stream_or_path*, *replace=True*, *run_hooks=True*, *dbapi=None*)

Add a format to the specified book. Return True if the format was added successfully.

Parametry

- **replace** – If True replace existing format, otherwise if the format already exists, return False.
- **run_hooks** – If True, file type plugins are run on the format before and after being added.
- **dbapi** – Pouze interní použití.

add_listener(*event_callback_function*, *check_already_added=False*)

Register a callback function that will be called after certain actions are taken on this database. The function must take three arguments: (*EventType* (stránka 351), library_id, event_type_specific_data)

add_notes_resource(*path_or_stream_or_data*, *name: str*, *mtime: float = None*) → int

Add the specified resource so it can be referenced by notes and return its content hash

all_annotation_types()

Return a tuple of all annotation types in the database.

all_annotation_users()

Return a tuple of all (user_type, user name) that have annotations.

all_annotations(*restrict_to_user=None*, *limit=None*, *annotation_type=None*, *ignore_removed=False*, *restrict_to_book_ids=None*)

Return a tuple of all annotations matching the specified criteria. *ignore_removed* controls whether removed (deleted) annotations are also returned. Removed annotations are just a skeleton used for merging of annotations.

all_annotations_for_book(*book_id*)

Return a tuple containing all annotations for the specified book_id as a dict with keys: *format*, *user_type*, *user*, *annotation*. Here, annotation is the annotation data.

all_book_ids(*type=<class 'frozenset'>*)

Frozen set of all known book ids.

all_field_for(*field*, *book_ids*, *default_value=None*)

Same as field_for, except that it operates on multiple books at once

all_field_ids(*name*)

Frozen set of ids for all values in the field `name`.

all_field_names(*field*)

Frozen set of all fields names (should only be used for many-one and many-many fields)

annotation_count_for_book(*book_id*)

Return the number of annotations for the specified book available in the database.

annotations_map_for_book(*book_id*, *fmt*, *user_type='local'*, *user='viewer'*)

Return a map of annotation type -> annotation data for the specified book_id, format, user and user_type.

author_data(*author_ids=None*)

Return author data as a dictionary with keys: name, sort, link

If no authors with the specified ids are found an empty dictionary is returned. If author_ids is None, data for all authors is returned.

author_sort_from_authors(*authors*, *key_func=<function make_change_case_func.<locals>.change_case>*)

Given a list of authors, return the author_sort string for the authors, preferring the author sort associated with the author over the computed string.

books_for_field(*name*, *item_id*)

Return all the books associated with the item identified by `item_id`, where the item belongs to the field `name`.

Returned value is a set of book ids, or the empty set if the item or the field does not exist.

books_in_virtual_library(*vl*, *search_restriction=None*, *virtual_fields=None*)

Return the set of books in the specified virtual library

compress_covers(*book_ids*, *jpeg_quality=100*, *progress_callback=None*)

Compress the cover images for the specified books. A compression quality of 100 will perform lossless compression, otherwise lossy compression.

The progress callback will be called with the book_id and the old and new sizes for each book that has been processed. If an error occurs, the new size will be a string with the error details.

copy_cover_to(*book_id*, *dest*, *use_hardlink=False*, *report_file_size=None*)

Copy the cover to the file like object `dest`. Returns False if no cover exists or dest is the same file as the current cover. dest can also be a path in which case the cover is copied to it if and only if the path is different from the current path (taking case sensitivity into account).

copy_format_to(*book_id*, *fmt*, *dest*, *use_hardlink=False*, *report_file_size=None*)

Copy the format `fmt` to the file like object `dest`. If the specified format does not exist, raises `NoSuchFormat` error. dest can also be a path (to a file), in which case the format is copied to it, iff the path is different from the current path (taking case sensitivity into account).

cover(*book_id*, *as_file=False*, *as_image=False*, *as_path=False*, *as_pixmap=False*)

Return the cover image or None. By default, returns the cover as a bytestring.

WARNING: Using as_path will copy the cover to a temp file and return the path to the temp file. You should delete the temp file when you are done with it.

Parametry

- **as_file** – If True return the image as an open file object (a SpooledTemporaryFile)
- **as_image** – If True return the image as a QImage object
- **as_pixmap** – If True return the image as a QPixmap object
- **as_path** – If True return the image as a path pointing to a temporary file

data_for_find_identical_books()

Return data that can be used to implement *find_identical_books()* (stránka 355) in a worker process without access to the db. See db.utils for an implementation.

data_for_has_book()

Return data suitable for use in *has_book()* (stránka 357). This can be used for an implementation of *has_book()* (stránka 357) in a worker process without access to the db.

delete_annotations(*annot_ids*)

Delete annotations with the specified ids.

delete_custom_book_data(*name*, *book_ids=()*)

Delete data for name. By default deletes all data, if you only want to delete data for some book ids, pass in a list of book ids.

delete_trash_entry(*book_id*, *category*)

Delete an entry from the trash. Here category is ‚b' for books and ‚f' for formats.

embed_metadata(*book_ids*, *only_fmts=None*, *report_error=None*, *report_progress=None*)

Update metadata in all formats of the specified book_ids to current metadata in the database.

expire_old_trash()

Expire entries from the trash that are too old

export_note(*field*, *item_id*) → str

Export the note as a single HTML document with embedded images as data: URLs

fast_field_for(*field_obj*, *book_id*, *default_value=None*)

Same as field_for, except that it avoids the extra lookup to get the field object

field_for(*name*, *book_id*, *default_value=None*)

Return the value of the field `name` for the book identified by `book_id`. If no such book exists or it has no defined value for the field `name` or no such field exists, then `default_value` is returned.

`default_value` is not used for title, title_sort, authors, author_sort and series_index. This is because these always have values in the db. `default_value` is used for all custom columns.

The returned value for is_multiple fields are always tuples, even when no values are found (in other words, default_value is ignored). The exception is identifiers for which the returned value is always a dictionary. The returned tuples are always in link order, that is, the order in which they were created.

field_ids_for(*name*, *book_id*)

Return the ids (as a tuple) for the values that the field `name` has on the book identified by `book_id`. If there are no values, or no such book, or no such field, an empty tuple is returned.

field_supports_notes(*field=None*) → bool

Return True iff the specified field supports notes. If field is None return frozenset of all fields that support notes.

find_identical_books(*mi*, *search_restriction=''*, *book_ids=None*)

Finds books that have a superset of the authors in mi and the same title (title is fuzzy matched). See also *data_for_find_identical_books()* (stránka 354).

format(*book_id*, *fmt*, *as_file=False*, *as_path=False*, *preserve_filename=False*)

Return the e-book format as a bytestring or *None* if the format doesn't exist, or we don't have permission to write to the e-book file.

Parametry

- **as_file** – If True the e-book format is returned as a file object. Note that the file object is a SpooledTemporaryFile, so if what you want to do is copy the format to another file, use *copy_format_to()* (stránka 354) instead for performance.
- **as_path** – Copies the format file to a temp file and returns the path to the temp file
- **preserve_filename** – If True and returning a path the filename is the same as that used in the library. Note that using this means that repeated calls yield the same temp file (which is re-created each time)

format_abspath(*book_id*, *fmt*)

Return absolute path to the e-book file of format *format*. You should almost never use this, as it breaks the threadsafe promise of this API. Instead use, *copy_format_to()* (stránka 354).

Currently used only in calibredb list, the viewer, edit book, compare_format to original format, open with, bulk metadata edit and the catalogs (via get_data_as_dict()).

Apart from the viewer, open with and edit book, I don't believe any of the others do any file write I/O with the results of this call.

format_hash(*book_id*, *fmt*)

Return the hash of the specified format for the specified book. The kind of hash is backend dependent, but is usually SHA-256.

format_metadata(*book_id*, *fmt*, *allow_cache=True*, *update_db=False*)

Return the path, size and mtime for the specified format for the specified book. You should not use path unless you absolutely have to, since accessing it directly breaks the threadsafe guarantees of this API. Instead use the *copy_format_to()* (stránka 354) method.

Parametry

- **allow_cache** – If `True` cached values are used, otherwise a slow filesystem access is done. The cache values could be out of date if access was performed to the filesystem outside of this API.
- **update_db** – If `True` The max_size field of the database is updated for this book.

formats(*book_id*, *verify_formats=True*)

Return tuple of all formats for the specified book. If verify_formats is True, verifies that the files exist on disk.

get_all_items_that_have_notes(*field_name=None*) → set[int] | dict[str, set[int]]

Return all item_ids for items that have notes in the specified field or all fields if field_name is None

get_all_link_maps_for_book(*book_id*)

Returns all links for all fields referenced by book identified by book_id. If book_id doesn't exist then the method returns {}.

Example: Assume author A has link X, author B has link Y, tag S has link F, and tag T has link G. If book 1 has author A and tag T, this method returns {‚authors‘:{‚A‘:‘X‘}, ‚tags‘:{‚T‘, ‚G‘}}. If book 2's author is neither A nor B and has no tags, this method returns {}.

Parametry

book_id – the book id in question.

Vrací

{field: {field_value, link_value}, … for all fields with a field_value having a non-empty link value for that book

get_categories(*sort='name'*, *book_ids=None*, *already_fixed=None*, *first_letter_sort=False*, *uncollapsed_categories=None*)

Used internally to implement the Tag Browser

get_custom_book_data(*name*, *book_ids=()*, *default=None*)

Get data for name. By default returns data for all book_ids, pass in a list of book ids if you only want some data. Returns a map of book_id to values. If a particular value could not be decoded, uses default for it.

get_id_map(*field*)

Return a mapping of id numbers to values for the specified field. The field must be a many-one or many-many field, otherwise a ValueError is raised.

get_ids_for_custom_book_data(*name*)

Return the set of book ids for which name has data.

get_item_id(*field*, *item_name*, *case_sensitive=False*)

Return the item id for item_name or None if not found. This function is very slow if doing lookups for multiple names use either get_item_ids() or get_item_name_map(). Similarly, case sensitive lookups are faster than case insensitive ones.

get_item_ids(*field*, *item_names*, *case_sensitive=False*)

Return a dict mapping item_name to the item id or None

get_item_name(*field*, *item_id*)

Return the item name for the item specified by item_id in the specified field. See also *get_id_map()* (stránka 356).

get_item_name_map(*field*, *normalize_func=None*)

Return mapping of item values to ids

get_link_map(*for_field*)

Return a dictionary of links for the supplied field.

Parametry

for_field – the lookup name of the field for which the link map is desired

Vrací

{field_value:link_value, …} for non-empty links

get_metadata(*book_id*, *get_cover=False*, *get_user_categories=True*, *cover_as_data=False*)

Return metadata for the book identified by book_id as a *calibre.ebooks.metadata.book.base.Metadata* (stránka 210) object. Note that the list of formats is not verified. If get_cover is True, the cover is returned, either a path to temp file as mi.cover or if cover_as_data is True then as mi.cover_data.

get_next_series_num_for(*series*, *field='series'*, *current_indices=False*)

Return the next series index for the specified series, taking into account the various preferences that control next series number generation.

Parametry

- **field** – The series-like field (defaults to the builtin series column)
- **current_indices** – If True, returns a mapping of book_id to current series_index value instead.

get_notes_resource(*resource_hash*) → dict | None

Return a dict containing the resource data and name or None if no resource with the specified hash is found

get_proxy_metadata(*book_id*)

Like *get_metadata()* (stránka 357) except that it returns a ProxyMetadata object that only reads values from the database on demand. This is much faster than get_metadata when only a small number of fields need to be accessed from the returned metadata object.

get_usage_count_by_id(*field*)

Return a mapping of id to usage count for all values of the specified field, which must be a many-one or many-many field.

has_book(*mi*)

Return True iff the database contains an entry with the same title as the passed in Metadata object. The comparison is case-insensitive. See also *data_for_has_book()* (stránka 354).

has_format(*book_id*, *fmt*)

Return True iff the format exists on disk

has_id(*book_id*)

Return True iff the specified book_id exists in the db

import_note(*field*, *item_id*, *path_to_html_file*, *path_is_data=False*)

Import a previously exported note or an arbitrary HTML file as the note for the specified item

init()

Initialize this cache with data from the backend.

items_with_notes_in_book(*book_id: int*) → dict[str, dict[int, str]]

Return a dict of field to items that have associated notes for that field for the specified book

link_for(*field*, *item_id*)

Return the link, if any, for the specified item or None if no link is found

list_extra_files(*book_id*, *use_cache=False*, *pattern=''*) → tuple[ExtraFile, ...]

Get information about extra files in the book's directory.

Parametry

- **book_id** – the database book id for the book
- **pattern** – the pattern of filenames to search for. Empty pattern matches all extra files. Patterns must use / as separator. Use the DATA_FILE_PATTERN constant to match files inside the data directory.

Vrací

A tuple of all extra files matching the specified pattern. Each element of the tuple is ExtraFile(relpath, file_path, stat_result). Where relpath is the relative path of the file to the book directory using / as a separator. stat_result is the result of calling os.stat() on the file.

merge_annotations_for_book(*book_id*, *fmt*, *annots_list*, *user_type='local'*, *user='viewer'*)

Merge the specified annotations into the existing annotations for book_id, fm, user_type, and user.

merge_extra_files(*dest_id*, *src_ids*, *replace=False*)

Merge the extra files from src_ids into dest_id. Conflicting files are auto-renamed unless replace=True in which case they are replaced.

move_book_from_trash(*book_id*)

Undelete a book from the trash directory

move_format_from_trash(*book_id*, *fmt*)

Undelete a format from the trash directory

multisort(*fields*, *ids_to_sort=None*, *virtual_fields=None*)

Return a list of sorted book ids. If ids_to_sort is None, all book ids are returned.

fields must be a list of 2-tuples of the form (field_name, ascending=True or False). The most significant field is the first 2-tuple.

notes_data_for(*field*, *item_id*) → str

Return all notes data as a dict or None if note does not exist

notes_for(*field*, *item_id*) → str

Return the notes document or an empty string if not found

notes_resources_used_by(*field*, *item_id*)

Return the set of resource hashes of all resources used by the note for the specified item

pref(*name*, *default=None*, *namespace=None*)

Return the value for the specified preference or the value specified as `default` if the preference is not set.

read_backup(*book_id*)

Return the OPF metadata backup for the book as a bytestring or None if no such backup exists.

remove_books(*book_ids*, *permanent=False*)

Remove the books specified by the book_ids from the database and delete their format files. If `permanent` is False, then the format files are placed in the per-library trash directory.

remove_extra_files(*book_id: int*, *relpaths: Iterable[str]*, *permanent=False*) → dict[str, Exception | None]

Delete the specified extra files, either to Recycle Bin or permanently.

remove_formats(*formats_map*, *db_only=False*)

Remove the specified formats from the specified books.

Parametry

- **formats_map** – A mapping of book_id to a list of formats to be removed from the book.
- **db_only** – If True, only remove the record for the format from the db, do not delete the actual format file from the filesystem.

Vrací

A map of book id to set of formats actually deleted from the filesystem for that book

remove_items(*field*, *item_ids*, *restrict_to_book_ids=None*)

Delete all items in the specified field with the specified ids. Returns the set of affected book ids. `restrict_to_book_ids` is an optional set of books ids. If specified the items will only be removed from those books.

rename_extra_files(*book_id*, *map_of_relpath_to_new_relpath*, *replace=False*)

Rename extra data files

rename_items(*field*, *item_id_to_new_name_map*, *change_index=True*, *restrict_to_book_ids=None*)

Rename items from a many-one or many-many field such as tags or series.

Parametry

- **change_index** – When renaming in a series-like field also change the series_index values.
- **restrict_to_book_ids** – An optional set of book ids for which the rename is to be performed, defaults to all books.

restore_book(*book_id*, *mi*, *last_modified*, *path*, *formats*, *annotations=()*)

Restore the book entry in the database for a book that already exists on the filesystem

restore_original_format(*book_id*, *original_fmt*)

Restore the specified format from the previously saved ORIGINAL_FORMAT, if any. Return True on success. The ORIGINAL_FORMAT is deleted after a successful restore.

property safe_read_lock

A safe read lock is a lock that does nothing if the thread already has a write lock, otherwise it acquires a read lock. This is necessary to prevent DowngradeLockErrors, which can happen when updating the search cache in the presence of composite columns. Updating the search cache holds an exclusive lock, but searching a composite column involves reading field values via ProxyMetadata which tries to get a shared lock. There may be other scenarios that trigger this as well.

This property returns a new lock object on every access. This lock object is not recursive (for performance) and must only be used in a with statement as `with cache.safe_read_lock:` otherwise bad things will happen.

save_original_format(*book_id*, *fmt*)

Save a copy of the specified format as ORIGINAL_FORMAT, overwriting any existing ORIGINAL_FORMAT.

search(*query*, *restriction=''*, *virtual_fields=None*, *book_ids=None*)

Search the database for the specified query, returning a set of matched book ids.

Parametry

- **restriction** – A restriction that is ANDed to the specified query. Note that restrictions are cached, therefore the search for a AND b will be slower than a with restriction b.

- **virtual_fields** – Used internally (virtual fields such as on_device to search over).
- **book_ids** – If not None, a set of book ids for which books will be searched instead of searching all books.

search_annotations(*fts_engine_query*, *use_stemming=True*, *highlight_start=None*, *highlight_end=None*, *snippet_size=None*, *annotation_type=None*, *restrict_to_book_ids=None*, *restrict_to_user=None*, *ignore_removed=False*)

Return of a tuple of annotations matching the specified Full-text query.

search_notes(*fts_engine_query=''*, *use_stemming=True*, *highlight_start=None*, *highlight_end=None*, *snippet_size=None*, *restrict_to_fields=()*, *return_text=True*, *result_type=<class 'tuple'>*, *process_each_result=None*, *limit=None*)

Search the text of notes using an FTS index. If the query is empty return all notes.

set_annotations_for_book(*book_id*, *fmt*, *annots_list*, *user_type='local'*, *user='viewer'*)

Set all annotations for the specified book_id, fmt, user_type and user.

set_conversion_options(*options*, *fmt='PIPE'*)

options must be a map of the form {book_id:conversion_options}

set_cover(*book_id_data_map*)

Set the cover for this book. The data can be either a QImage, QPixmap, file object or bytestring. It can also be None, in which case any existing cover is removed.

set_field(*name*, *book_id_to_val_map*, *allow_case_change=True*, *do_path_update=True*)

Set the values of the field specified by `name`. Returns the set of all book ids that were affected by the change.

Parametry

- **book_id_to_val_map** – Mapping of book_ids to values that should be applied.
- **allow_case_change** – If True, the case of many-one or many-many fields will be changed. For example, if a book has the tag `tag1` and you set the tag for another book to `Tag1` then the both books will have the tag `Tag1` if allow_case_change is True, otherwise they will both have the tag `tag1`.
- **do_path_update** – Used internally, you should never change it.

set_link_map(*field*, *value_to_link_map*, *only_set_if_no_existing_link=False*)

Sets links for item values in field. Note: this method doesn't change values not in the value_to_link_map

Parametry

- **field** – the lookup name
- **value_to_link_map** – dict(field_value:link, …). Note that these are values, not field ids.

Vrací

books changed by setting the link

set_metadata(*book_id*, *mi*, *ignore_errors=False*, *force_changes=False*, *set_title=True*, *set_authors=True*, *allow_case_change=False*)

Set metadata for the book *id* from the *Metadata* object *mi*

Setting force_changes=True will force set_metadata to update fields even if mi contains empty values. In this case, ‚None' is distinguished from ‚empty'. If mi.XXX is None, the XXX is not replaced, otherwise it is. The tags, identifiers, and cover attributes are special cases. Tags and identifiers cannot be set to None so they will always be replaced if force_changes is true. You must ensure that mi contains the values you want the book to have. Covers are always changed if a new cover is provided, but are never deleted. Also note that force_changes has no effect on setting title or authors.

set_notes_for(*field*, *item_id*, *doc: str*, *searchable_text: str = ''*, *resource_hashes=()*, *remove_unused_resources=False*) → int

Set the notes document. If the searchable text is different from the document, specify it as searchable_text. If the document references resources their hashes must be present in resource_hashes. Set remove_unused_resources to True to cleanup unused resources, note that updating a note automatically cleans up resources pertaining to that note anyway.

set_pref(*name*, *val*, *namespace=None*)

Set the specified preference to the specified value. See also *pref()* (stránka 358).

split_if_is_multiple_composite(*f*, *val*)

If f is a composite column lookup key and the column is is_multiple then split v into unique non-empty values. The comparison is case sensitive. Order is not preserved. Return a list() for compatibility with proxy metadata field getters, for example tags.

tags_older_than(*tag*, *delta=None*, *must_have_tag=None*, *must_have_authors=None*)

Return the ids of all books having the tag `tag` that are older than the specified time. tag comparison is case insensitive.

Parametry

- **delta** – A timedelta object or None. If None, then all ids with the tag are returned.
- **must_have_tag** – If not None the list of matches will be restricted to books that have this tag
- **must_have_authors** – A list of authors. If not None the list of matches will be restricted to books that have these authors (case insensitive).

unretire_note_for(*field*, *item_id*) → int

Unretire a previously retired note for the specified item. Notes are retired when an item is removed from the database

update_annotations(*annot_id_map*)

Update annotations.

user_categories_for_books(*book_ids*, *proxy_metadata_map=None*)

Return the user categories for the specified books. proxy_metadata_map is optional and is useful for a performance boost, in contexts where a ProxyMetadata object for the books already exists. It should be a mapping of book_ids to their corresponding ProxyMetadata objects.

14.9.2 Dokumentace API pro nástroje pro úpravu e-knih

Nástroje pro úpravu e-knih se skládají z objektu *calibre.ebooks.oeb.polish.container.Container* (stránka 362), který představuje knihu jako kolekci souborů HTML a zdrojů a různých nástrojů, které lze použít k provádění operací na kontejneru. Všechny nástroje jsou ve formě funkcí na úrovni modulů v různých modulech `calibre.ebooks.oeb.polish.*`. Objekt kontejneru pro knihu v cestě získáte takto:

```
from calibre.ebooks.oeb.polish.container import get_container
container = get_container('Path to book file', tweak_mode=True)
```

Pokud píšete modul pro Editor e-knih, získáte aktuální kontejner pro upravovanou knihu takto:

```
from calibre.gui2.tweak_book import current_container
container = current_container()
if container is None:
    report_error # No book has been opened yet
```

Objekt kontejneru

class calibre.ebooks.oeb.polish.container.**Container**(*rootpath=None*, *opfpath=None*, *log=<calibre.utils.logging.Log object>*, *clone_data=None*)

A container represents an open e-book as a folder full of files and an OPF file. There are two important concepts:

- The root folder. This is the base of the e-book. All the e-books files are inside this folder or in its sub-folders.
- Names: These are paths to the books' files relative to the root folder. They always contain POSIX separators and are unquoted. They can be thought of as canonical identifiers for files in the book. Most methods on the container object work with names. Names are always in the NFC Unicode normal form.
- Clones: the container object supports efficient on-disk cloning, which is used to implement checkpoints in the e-book editor. In order to make this work, you should never access files on the filesystem directly. Instead, use *raw_data()* (stránka 364) or *open()* (stránka 364) to read/write to component files in the book.

When converting between hrefs and names use the methods provided by this class, they assume all hrefs are quoted.

abspath_to_name(*fullpath*, *root=None*)

Convert an absolute path to a canonical name relative to `root`

Parametry

root – The base folder. By default the root for this container object is used.

add_file(*name*, *data=b''*, *media_type=None*, *spine_index=None*, *modify_name_if_needed=False*, *process_manifest_item=None*, *suggested_id=''*)

Add a file to this container. Entries for the file are automatically created in the OPF manifest and spine (if the file is a text document)

add_name_to_manifest(*name*, *process_manifest_item=None*, *suggested_id=''*)

Add an entry to the manifest for a file with the specified name. Returns the manifest id.

add_properties(*name*, **properties*)

Add the specified properties to the manifest item identified by name.

apply_unique_properties(*name*, **properties*)

Ensure that the specified properties are set on only the manifest item identified by name. You can pass None as the name to remove the property from all items.

book_type = 'oeb'

Typ knihy (epub pro soubory EPUB a azw3 pro soubory AZW3)

commit(*outpath=None*, *keep_parsed=False*)

Commit all dirtied parsed objects to the filesystem and write out the e-book file at outpath.

Parametry

- **output** – The path to write the saved e-book file to. If None, the path of the original book file is used.
- **keep_parsed** – If True the parsed representations of committed items are kept in the cache.

commit_item(*name*, *keep_parsed=False*)

Commit a parsed object to disk (it is serialized and written to the underlying file). If `keep_parsed` is True the parsed representation is retained in the cache. See also: *parsed()* (stránka 364)

dirty(*name*)

Mark the parsed object corresponding to name as dirty. See also: *parsed()* (stránka 364).

exists(*name*)

True iff a file/folder corresponding to the canonical name exists. Note that this function suffers from the limitations of the underlying OS filesystem, in particular case (in)sensitivity. So on a case insensitive filesystem this will return True even if the case of name is different from the case of the underlying filesystem file. See also *has_name()* (stránka 363)

filesize(*name*)

Return the size in bytes of the file represented by the specified canonical name. Automatically handles dirtied parsed objects. See also: *parsed()* (stránka 364)

generate_item(*name*, *id_prefix=None*, *media_type=None*, *unique_href=True*)

Add an item to the manifest with href derived from the given name. Ensures uniqueness of href and id automatically. Returns generated item.

get_file_path_for_processing(*name*, *allow_modification=True*)

Similar to open() except that it returns a file path, instead of an open file object.

property guide_type_map

Mapping of guide type to canonical name

has_name(*name*)

Return True iff a file with the same canonical name as that specified exists. Unlike *exists()* (stránka 362) this method is always case-sensitive.

href_to_name(*href*, *base=None*)

Convert an href (relative to base) to a name. base must be a name or None, in which case self.root is used.

insert_into_xml(*parent*, *item*, *index=None*)

Insert item into parent (or append if index is None), fixing indentation. Only works with self closing items.

is_dir = False

Pokud tento kontejner představuje rozbalenou knihu (adresář)

iterlinks(*name*, *get_line_numbers=True*)

Iterate over all links in name. If get_line_numbers is True the yields results of the form (link, line_number, offset). Where line_number is the line_number at which the link occurs and offset is the number of characters from the start of the line. Note that offset could actually encompass several lines if not zero.

make_name_unique(*name*)

Ensure that *name* does not already exist in this book. If it does, return a modified version that does not exist.

manifest_has_name(*name*)

Return True if the manifest has an entry corresponding to name

property manifest_id_map

Mapping of manifest id to canonical names

manifest_items_of_type(*predicate*)

The names of all manifest items whose media-type matches predicate. *predicate* can be a set, a list, a string or a function taking a single argument, which will be called with the media-type.

manifest_items_with_property(*property_name*)

All manifest items that have the specified property

property manifest_type_map

Mapping of manifest media-type to list of canonical names of that media-type

property mi

The metadata of this book as a Metadata object. Note that this object is constructed on the fly every time this property is requested, so use it sparingly.

name_to_abspath(*name*)

Convert a canonical name to an absolute OS dependent path

name_to_href(*name*, *base=None*)

Convert a name to a href relative to base, which must be a name or None in which case self.root is used as the base

property names_that_must_not_be_changed

Set of names that must never be renamed. Depends on the e-book file format.

property names_that_must_not_be_removed

Set of names that must never be deleted from the container. Depends on the e-book file format.

property names_that_need_not_be_manifested

Set of names that are allowed to be missing from the manifest. Depends on the e-book file format.

open(*name*, *mode='rb'*)

Open the file pointed to by name for direct read/write. Note that this will commit the file if it is dirtied and remove it from the parse cache. You must finish with this file before accessing the parsed version of it again, or bad things will happen.

property opf

The parsed OPF file

opf_get_or_create(*name*)

Convenience method to either return the first XML element with the specified name or create it under the opf:package element and then return it, if it does not already exist.

property opf_version

The version set on the OPF's <package> element

property opf_version_parsed

The version set on the OPF's <package> element as a tuple of integers

opf_xpath(*expr*)

Convenience method to evaluate an XPath expression on the OPF file, has the opf: and dc: namespace prefixes pre-defined.

parsed(*name*)

Return a parsed representation of the file specified by name. For HTML and XML files an lxml tree is returned. For CSS files a css_parser stylesheet is returned. Note that parsed objects are cached for performance. If you make any changes to the parsed object, you must call *dirty()* (stránka 362) so that the container knows to update the cache. See also *replace()* (stránka 365).

raw_data(*name*, *decode=True*, *normalize_to_nfc=True*)

Return the raw data corresponding to the file specified by name

Parametry

- **decode** – If True and the file has a text based MIME type, decode it and return a unicode object instead of raw bytes.
- **normalize_to_nfc** – If True the returned unicode object is normalized to the NFC normal form as is required for the EPUB and AZW3 file formats.

relpath(*path*, *base=None*)

Convert an absolute path (with os separators) to a path relative to base (defaults to self.root). The relative path is *not* a name. Use *abspath_to_name()* (stránka 362) for that.

remove_from_spine(*spine_items*, *remove_if_no_longer_in_spine=True*)

Remove the specified items (by canonical name) from the spine. If `remove_if_no_longer_in_spine` is True, the items are also deleted from the book, not just from the spine.

remove_from_xml(*item*)

Removes item from parent, fixing indentation (works only with self closing items)

remove_item(*name*, *remove_from_guide=True*)

Remove the item identified by name from this container. This removes all references to the item in the OPF manifest, guide and spine as well as from any internal caches.

rename(*current_name*, *new_name*)

Renames a file from current_name to new_name. It automatically rebases all links inside the file if the folder the file is in changes. Note however, that links are not updated in the other files that could reference this file. This is for performance, such updates should be done once, in bulk.

replace(*name*, *obj*)

Replace the parsed object corresponding to name with obj, which must be a similar object, i.e. an lxml tree for HTML/XML or a css_parser stylesheet for a CSS file.

replace_links(*name*, *replace_func*)

Replace all links in name using replace_func, which must be a callable that accepts a URL and returns the replaced URL. It must also have a ‚replaced‘ attribute that is set to True if any actual replacement is done. Convenient ways of creating such callables are using the `LinkReplacer` and `LinkRebaser` classes.

serialize_item(*name*)

Convert a parsed object (identified by canonical name) into a bytestring. See *parsed()* (stránka 364).

set_spine(*spine_items*)

Set the spine to be spine_items where spine_items is an iterable of the form (name, linear). Will raise an error if one of the names is not present in the manifest.

property spine_items

An iterator yielding the path for every item in the books‘ spine. See also: *spine_iter* (stránka 365) and *spine_items* (stránka 365).

property spine_iter

An iterator that yields item, name is_linear for every item in the books‘ spine. item is the lxml element, name is the canonical file name and is_linear is True if the item is linear. See also: *spine_names* (stránka 365) and *spine_items* (stránka 365).

property spine_names

An iterator yielding name and is_linear for every item in the books‘ spine. See also: *spine_iter* (stránka 365) and *spine_items* (stránka 365).

Správa souborů komponent v kontejneru

calibre.ebooks.oeb.polish.replace.**replace_links**(*container*, *link_map*, *frag_map=<function <lambda>>*, *replace_in_opf=False*)

Replace links to files in the container. Will iterate over all files in the container and change the specified links in them.

Parametry

- **link_map** – A mapping of old canonical name to new canonical name. For example: `{'images/old.png': 'images/new.png'}`
- **frag_map** – A callable that takes two arguments `(name, anchor)` and returns a new anchor. This is useful if you need to change the anchors in HTML files. By default, it does nothing.
- **replace_in_opf** – If False, links are not replaced in the OPF file.

calibre.ebooks.oeb.polish.replace.**rename_files**(*container*, *file_map*)

Rename files in the container, automatically updating all links to them.

Parametry

file_map – A mapping of old canonical name to new canonical name, for example: `{'text/chapter1.html': 'chapter1.html'}`.

calibre.ebooks.oeb.polish.replace.**get_recommended_folders**(*container*, *names*)

Return the folders that are recommended for the given filenames. The recommendation is based on where the majority of files of the same type are located in the container. If no files of a particular type are present, the recommended folder is assumed to be the folder containing the OPF file.

Pěkný tisk a automatická analýza opravy chyb

calibre.ebooks.oeb.polish.pretty.**fix_html**(*container*, *raw*)

Fix any parsing errors in the HTML represented as a string in raw. Fixing is done using the HTML5 parsing algorithm.

calibre.ebooks.oeb.polish.pretty.**fix_all_html**(*container*)

Fix any parsing errors in all HTML files in the container. Fixing is done using the HTML5 parsing algorithm.

calibre.ebooks.oeb.polish.pretty.**pretty_html**(*container*, *name*, *raw*)

Pretty print the HTML represented as a string in raw

calibre.ebooks.oeb.polish.pretty.**pretty_css**(*container*, *name*, *raw*)

Pretty print the CSS represented as a string in raw

calibre.ebooks.oeb.polish.pretty.**pretty_xml**(*container*, *name*, *raw*)

Pretty print the XML represented as a string in raw. If `name` is the name of the OPF, extra OPF-specific prettying is performed.

calibre.ebooks.oeb.polish.pretty.**pretty_all**(*container*)

Pretty print all HTML/CSS/XML files in the container

Správa přebalů knih

calibre.ebooks.oeb.polish.jacket.**remove_jacket**(*container*)

Remove an existing jacket, if any. Returns False if no existing jacket was found.

calibre.ebooks.oeb.polish.jacket.**add_or_replace_jacket**(*container*)

Either create a new jacket from the book's metadata or replace an existing jacket. Returns True if an existing jacket was replaced.

Rozdělování a slučování souborů

calibre.ebooks.oeb.polish.split.**split**(*container*, *name*, *loc_or_xpath*, *before=True*, *totals=None*)

Split the file specified by name at the position specified by loc_or_xpath. Splitting automatically migrates all links and references to the affected files.

Parametry

- **loc_or_xpath** – Should be an XPath expression such as //h:div[@id="split_here"]. Can also be a *loc* which is used internally to implement splitting in the preview panel.
- **before** – If True the split occurs before the identified element otherwise after it.
- **totals** – Used internally

calibre.ebooks.oeb.polish.split.**multisplit**(*container*, *name*, *xpath*, *before=True*)

Split the specified file at multiple locations (all tags that match the specified XPath expression). See also: *split()* (stránka 366). Splitting automatically migrates all links and references to the affected files.

Parametry

before – If True the splits occur before the identified element otherwise after it.

calibre.ebooks.oeb.polish.split.**merge**(*container*, *category*, *names*, *master*)

Merge the specified files into a single file, automatically migrating all links and references to the affected files. The file must all either be HTML or CSS files.

Parametry

- **category** – Must be either `'text'` for HTML files or `'styles'` for CSS files
- **names** – The list of files to be merged
- **master** – Which of the merged files is the *master* file, that is, the file that will remain after merging.

Správa obálek

calibre.ebooks.oeb.polish.cover.**set_cover**(*container*, *cover_path*, *report=None*, *options=None*)

Set the cover of the book to the image pointed to by cover_path.

Parametry

- **cover_path** – Either the absolute path to an image file or the canonical name of an image in the book. When using an image in the book, you must also set options, see below.
- **report** – An optional callable that takes a single argument. It will be called with information about the tasks being processed.
- **options** – None or a dictionary that controls how the cover is set. The dictionary can have entries: **keep_aspect**: True or False (Preserve aspect ratio of covers in EPUB) **no_svg**: True or False (Use an SVG cover wrapper in the EPUB titlepage) **existing**: True or False (`cover_path` refers to an existing image in the book)

calibre.ebooks.oeb.polish.cover.**mark_as_cover**(*container*, *name*)

Mark the specified image as the cover image.

calibre.ebooks.oeb.polish.cover.**mark_as_titlepage**(*container*, *name*, *move_to_start=True*)

Mark the specified HTML file as the titlepage of the EPUB.

Parametry

move_to_start – If True the HTML file is moved to the start of the spine

Práce s CSS

calibre.ebooks.oeb.polish.fonts.**change_font**(*container*, *old_name*, *new_name=None*)

Change a font family from old_name to new_name. Changes all occurrences of the font family in stylesheets, style tags and style attributes. If the old_name refers to an embedded font, it is removed. You can set new_name to None to remove the font family instead of changing it.

calibre.ebooks.oeb.polish.css.**remove_unused_css**(*container*, *report=None*, *remove_unused_classes=False*, *merge_rules=False*, *merge_rules_with_identical_properties=False*, *remove_unreferenced_sheets=False*)

Remove all unused CSS rules from the book. An unused CSS rule is one that does not match any actual content.

Parametry

- **report** – An optional callable that takes a single argument. It is called with information about the operations being performed.
- **remove_unused_classes** – If True, class attributes in the HTML that do not match any CSS rules are also removed.
- **merge_rules** – If True, rules with identical selectors are merged.
- **merge_rules_with_identical_properties** – If True, rules with identical properties are merged.
- **remove_unreferenced_sheets** – If True, stylesheets that are not referenced by any content are removed

calibre.ebooks.oeb.polish.css.**filter_css**(*container*, *properties*, *names=()*)

Remove the specified CSS properties from all CSS rules in the book.

Parametry

- **properties** – Set of properties to remove. For example: `{'font-family', 'color'}`.
- **names** – The files from which to remove the properties. Defaults to all HTML and CSS files in the book.

Práce s obsahem

calibre.ebooks.oeb.polish.toc.**from_xpaths**(*container*, *xpaths*, *prefer_title=False*)

Generate a Table of Contents from a list of XPath expressions. Each expression in the list corresponds to a level of the generate ToC. For example: `['//h:h1', '//h:h2', '//h:h3']` will generate a three level Table of Contents from the `<h1>`, `<h2>` and `<h3>` tags.

calibre.ebooks.oeb.polish.toc.**from_links**(*container*)

Generate a Table of Contents from links in the book.

calibre.ebooks.oeb.polish.toc.**from_files**(*container*)

Generate a Table of Contents from files in the book.

calibre.ebooks.oeb.polish.toc.**create_inline_toc**(*container*, *title=None*)

Create an inline (HTML) Table of Contents from an existing NCX Table of Contents.

Parametry

title – The title for this table of contents.

Nástroj na úpravu knih

class calibre.gui2.tweak_book.plugin.**Tool**

Bases: `object`

The base class for individual tools in an Edit Book plugin. Useful members include:

- `self.plugin`: A reference to the *calibre.customize.Plugin* (stránka 256) object to which this tool belongs.

- self. *boss* (stránka 369)
- self. *gui* (stránka 369)

Methods that must be overridden in sub classes:

- *create_action()* (stránka 369)
- *register_shortcut()* (stránka 369)

name = None

Set this to a unique name it will be used as a key

allowed_in_toolbar = True

If True the user can choose to place this tool in the plugins toolbar

allowed_in_menu = True

If True the user can choose to place this tool in the plugins menu

toolbar_button_popup_mode = 'delayed'

The popup mode for the menu (if any) of the toolbar button. Possible values are ‚delayed', ‚instant', ‚button'

property boss

The *calibre.gui2.tweak_book.boss.Boss* (stránka 370) object. Used to control the user interface.

property gui

The main window of the user interface

property current_container

Return the current *calibre.ebooks.oeb.polish.container.Container* (stránka 362) object that represents the book being edited.

register_shortcut(*qaction*, *unique_name*, *default_keys=()*, *short_text=None*, *description=None*, ***extra_data*)

Register a keyboard shortcut that will trigger the specified `qaction`. This keyboard shortcut will become automatically customizable by the user in the Keyboard shortcuts section of the editor preferences.

Parametry

- **qaction** – A QAction object, it will be triggered when the configured key combination is pressed by the user.
- **unique_name** – A unique name for this shortcut/action. It will be used internally, it must not be shared by any other actions in this plugin.
- **default_keys** – A list of the default keyboard shortcuts. If not specified no default shortcuts will be set. If the shortcuts specified here conflict with either builtin shortcuts or shortcuts from user configuration/other plugins, they will be ignored. In that case, users will have to configure the shortcuts manually via Preferences. For example: `default_keys=('Ctrl+J', 'F9')`.
- **short_text** – An optional short description of this action. If not specified the text from the QAction will be used.
- **description** – An optional longer description of this action, it will be used in the preferences entry for this shortcut.

create_action(*for_toolbar=True*)

Create a QAction that will be added to either the plugins toolbar or the plugins menu depending on `for_toolbar`. For example:

```
def create_action(self, for_toolbar=True):
    ac = QAction(get_icons('myicon.png'), 'Do something')
    if for_toolbar:
        # We want the toolbar button to have a popup menu
        menu = QMenu()
        ac.setMenu(menu)
        menu.addAction('Do something else')
        subaction = menu.addAction('And another')

        # Register a keyboard shortcut for this toolbar action be
        # careful to do this for only one of the toolbar action or
        # the menu action, not both.
        self.register_shortcut(ac, 'some-unique-name', default_keys=('Ctrl+K',
↪))
    return ac
```

Viz také

Method *register_shortcut()* (stránka 369).

Ovládání uživatelského rozhraní editoru

Uživatelské rozhraní editoru e-knih je ovládáno jediným globálním *Hlavním* objektem. To má mnoho užitečných metod, které lze použít v kódu modulu k provádění běžných úkolů.

class calibre.gui2.tweak_book.boss.**Boss**(*parent*, *notify=None*)

add_savepoint(*msg*)

Create a restore checkpoint with the name specified as `msg`

apply_container_update_to_gui(*mark_as_modified=True*)

Update all the components of the user interface to reflect the latest data in the current book container.

Parametry

mark_as_modified – If True, the book will be marked as modified, so the user will be prompted to save it when quitting.

close_editor(*name*)

Close the editor that is editing the file specified by `name`

commit_all_editors_to_container()

Commit any changes that the user has made to files open in editors to the container. You should call this method before performing any actions on the current container

property currently_editing

Return the name of the file being edited currently or None if no file is being edited

edit_file(*name*, *syntax=None*, *use_template=None*)

Open the file specified by name in an editor

Parametry

- **syntax** – The media type of the file, for example, `'text/html'`. If not specified it is guessed from the file extension.
- **use_template** – A template to initialize the opened editor with

open_book(*path=None*, *edit_file=None*, *clear_notify_data=True*, *open_folder=False*, *search_text=None*)

Open the e-book at `path` for editing. Will show an error if the e-book is not in a supported format or the current book has unsaved changes.

Parametry

edit_file – The name of a file inside the newly opened book to start editing. Can also be a list of names.

rewind_savepoint()

Undo the previous creation of a restore checkpoint, useful if you create a checkpoint, then abort the operation with no changes

save_book()

Save the book. Saving is performed in the background

set_modified()

Mark the book as having been modified

show_current_diff(*allow_revert=True*, *to_container=None*)

Show the changes to the book from its last checkpointed state

Parametry

- **allow_revert** – If True the diff dialog will have a button to allow the user to revert all changes
- **to_container** – A container object to compare the current container to. If None, the previously checkpointed container is used

show_editor(*name*)

Show the editor that is editing the file specified by `name`

sync_preview_to_editor()

Sync the position of the preview panel to the current cursor position in the current editor

KAPITOLA 15

Správa digitálních práv (DRM)

Správa digitálních práv (DRM) je všeobecný pojem pro technologie řízení přístupu, které mohou výrobci hardwaru, vydavatelé, držitelé autorských práv a jednotlivci použít k pokusu o vynucení omezení na používání digitálního obsahu a zařízení. Někdy je také pohrdavě popsán jako Správa digitálních omezení. Pojem se používá k popisu jakékoliv technologie, která brání (právního nebo jiného) použití digitálního obsahu, který poskytovatel obsahu nepožadoval ani nepředvídal. Pojem se obecně nevztahuje na jiné formy ochrany proti kopírování, které lze obejít bez úpravy souboru nebo zařízení, jako jsou sériová čísla nebo soubory klíčů. Může také odkazovat na omezení spojená s konkrétními případy digitálních děl nebo zařízení. Technologie DRM se snaží kontrolovat použití digitálních médií tím, že koncovým uživatelům brání v přístupu, kopírování nebo převodu do jiných formátů. Viz Wikipedia[140].

15.1 Co z DRM vyplývá pro mě osobně?

Když si koupíte e-knihu s DRM, ve skutečnosti ji nevlastníte, ale zakoupili jste povolení k jejímu používání způsobem, který vám diktuje prodejce. DRM omezuje to, co můžete dělat s e-knihami, které jste „koupili“. Lidé, kteří nakupují knihy s DRM, často nevědí o rozsahu těchto omezení. Tato omezení vám brání v přeformátování e-knihy podle vašich představ, včetně provedení stylistických změn, jako je úprava velikosti písma, i když existuje software, který vám umožňuje dělat takové věci pro knihy bez DRM. Lidé jsou často překvapeni, že e-knihu, kterou zakoupili v určitém formátu, nelze převést na jiný formát, pokud má e-kniha DRM. Takže pokud máte Amazon Kindle a koupíte si knihu od Barnes a Nobles, měli byste vědět, že pokud tato e-kniha má DRM, nebudete ji moci číst na svém Kindle. Všimněte si, že mluvím o knize, kterou kupujete, ne kradete nebo pirátíte, ale KUPUJETE.

15.2 Co přináší DRM autorům?

Vydavatelé e-knih chráněných DRM tvrdí, že DRM je v zájmu autorů a chrání jejich uměleckou integritu a zabraňuje pirátství. DRM však NEZABRAŇUJE pirátství. Lidé, kteří chtějí pirátit obsah nebo používají pirátský obsah, to stále dělají a jsou v tom úspěšní. Tři hlavní schémata DRM pro e-knihy dnes provozují Amazon, Adobe a Barnes and Noble a všechna tři schémata DRM byla prolomena. Jediné, co DRM dělá, je to, že přináší nepohodlí oprávněným uživatelům. Lze tvrdit, že to vlastně poškozuje autory, protože lidé, kteří by knihu zakoupili, se rozhodnou najít pirátskou verzi, protože nejsou ochotni se smířit s DRM. Ti, kteří by pirátili i bez DRM, tak činí také v jeho přítomnosti. Znovu opakuji,

[140] https://en.wikipedia.org/wiki/Digital_rights_management

klíčovým bodem je, že DRM *nezabraňuje pirátství *. DRM tedy není jen zbytečné a škodlivé pro kupující e-knihy, ale také plýtvání penězi.

15.3 DRM a svoboda

Přestože digitální obsah může být použit pro snadné zpřístupnění informací a také tvůrčích děl všem a posílení lidskosti, není to v zájmu některých vydavatelů, kteří chtějí lidi odvrátit od této možnosti svobody, jen aby si udrželi svůj význam ve světě, který se vyvíjí tak rychle, že se jim nedaří držet krok.

15.4 Proč Calibre nepodporuje DRM?

Calibre je software s otevřeným zdrojovým kódem, zatímco DRM je ze své podstaty uzavřené. Pokud by mělo Calibre podporovat otevírání nebo zobrazování souborů DRM, mohlo by být snadno upraveno pro použití jako nástroje pro odebírání DRM, což je podle dnešních zákonů nezákonné. Software s otevřeným zdrojovým kódem a DRM jsou střetem principů. Zatímco DRM je jen o ovládání uživatele, software s otevřeným zdrojovým kódem je o posílení postavení uživatele. Oba jednoduše nemohou spolu existovat.

15.5 Jak Calibre nahlíží na poskytovatele obsahu?

Pevně věříme, že autoři a ostatní poskytovatelé obsahu by měli být za své úsilí odměněni, ale DRM není cestou, jak toho dosáhnout. Tuto databázi e-knih bez DRM vyvíjíme z různých zdrojů, abychom vám pomohli najít alternativy bez DRM a pomohli nezávislým autorům a vydavatelům e-knih bez DRM propagovat jejich obsah. Doufáme, že to shledáte užitečným, a žádáme vás, abyste nepirátili obsah, který je vám zde k dispozici.

15.6 Jak mohu pomoci v boji proti DRM?

Jako někdo, kdo čte a kupuje e-knihy, můžete pomoci v boji proti DRM. Nekupujte e-knihy s DRM. Někteří vydavatelé vydávají e-knihy bez DRM. Snažte se zjistit, zda nabízejí e-knihu, kterou hledáte. Pokud máte rádi knihy od některých nezávislých autorů, kteří prodávají e-knihy bez DRM, a můžete si to dovolit, věnujte jim finanční dar. Jsou to dobře vynaložené peníze, protože jejich e-knihy bývají levnější (mohou existovat výjimky), než ty, které byste si koupili od vydavatelů knih chráněných DRM, a pravděpodobně budou v budoucnu fungovat na všech zařízeních, která vlastníte, čímž vám ušetří náklady na nové zakoupení e-knihy. Neodstrašujte vydavatele a autory e-knih bez DRM pirátěním jejich obsahu. Poskytovatelé obsahu si za své úsilí zaslouží odměnu. Netrestejte je za to, že se snaží zlepšit váš zážitek ze čtení zpřístupněním e-knih bez DRM. Z dlouhodobého hlediska je to pro vás škodlivé. Pokud jste si zakoupili knihy od prodejců, kteří nabízejí knihy jak s DRM, tak i bez DRM, aniž byste věděli, zda mají DRM nebo ne, bylo by vhodné zanechat komentář nebo recenzi na webových stránkách, abyste informovali budoucí kupující o stavu DRM. Mnoho prodejců si nemyslí, že je důležité jasně označit svým kupujícím, zda e-kniha obsahuje DRM nebo ne. *Zde <https://www.defectivebydesign.org/guide/ebooks>* najdete průvodce životem bez DRM.

KAPITOLA 16

Slovník pojmů

RSS

RSS *(Really Simple Syndication)* je formát webového informačního kanálu, který se používá k publikování často aktualizovaného obsahu, jako jsou novinové články, příspěvky blogu atd. Je to formát, který je obzvlášť vhodný pro čtení počítači, a je proto upřednostňovaným způsob získávání obsahu z webu do e-knihy. Existuje mnoho dalších formátů informačních kanálů používaných na internetu a calibre rozumí většině z nich. Zejména má dobrou podporu pro formát *ATOM*, který se obvykle používá pro blogy.

předpis

Předpis je sada instrukcí, která učí calibre, jak převést na e-knihu online zdroje zpráv, jako jsou časopisy nebo blogy. Předpis je v podstatě kód Pythonu[141]. Jako takový je schopný převést na e-knihy libovolně složitý zdroj zpráv. Na nejjednodušší úrovni je to pouze sada proměnných, jako jsou URL, které dávají calibre dostatek informací pro přejití na internet a stažení zpráv.

HTML

HTML *(značkovací jazyk pro hypertext = HyperText Markup Language)*, podmnožina SGML (univerzální značkovací metajazyk = Standard Generalized Markup Language) pro elektronické publikování, je konkrétní standard použití pro web.

CSS

CSS *(šablony stylů = Cascading Style Sheets)* je jazyk použitý pro popsání, jak by měl být vykreslen dokument *HTML* (vizuální stylování).

API

API *(rozhraní pro programování aplikací = Application Programming Interface)* je rozhraní zdrojového kódu, které knihovna poskytuje pro podporu požadavků na služby, které na něj mají být provedeny počítačovými programy.

LRF

LRF Formát e-knihy, který je čten čtečkami e-knih SONY.

URL

URL *(jednotná adresa zdroje = Uniform Resource Locator)* například: `http://example.com`

regexp

Regulární výrazy poskytují stručný a pružný prostředek pro identifikaci řetězců zkoumaného textu, jako jsou

[141] https://www.python.org

určité znaky, slova nebo vzory znaků. Úvod do regulárních výrazů naleznete na *the tutorial* (stránka 214).

Rejstřík modulů Pythonu

C

calibre.customize, 256
calibre.customize.conversion, 265
calibre.db.cache, 351
calibre.devices.interface, 268
calibre.ebooks.metadata.book.base, 210
calibre.ebooks.metadata.sources.base, 262
calibre.ebooks.oeb.polish.container, 361
calibre.ebooks.oeb.polish.cover, 367
calibre.ebooks.oeb.polish.css, 367
calibre.ebooks.oeb.polish.jacket, 366
calibre.ebooks.oeb.polish.pretty, 366
calibre.ebooks.oeb.polish.replace, 365
calibre.ebooks.oeb.polish.split, 366
calibre.ebooks.oeb.polish.toc, 368
calibre.gui2.tweak_book.boss, 370
calibre.utils.formatter_functions, 184
calibre.web.feeds.news, 41

Symboly

-1
    calibredb-add command line option, 315
-H
    ebook-polish command line option, 336
-I
    calibredb-add command line option, 315
    fetch-ebook-metadata command line option, 338
-S
    calibredb-add command line option, 315
-T
    calibredb-add command line option, 315
-U
    ebook-polish command line option, 337
--access-log
    calibre-server command line option, 307
--add
    calibredb-add command line option, 315
--add-alt-text-to-img
    ebook-convert command line option, 331
--add-plugin
    calibre-customize command line option, 304
--add-simple-plugin
    calibre-debug command line option, 305
--add-soft-hyphens
    ebook-polish command line option, 336
--ajax-timeout
    calibre-server command line option, 307
--all
    calibredb-backup_metadata command line option, 323
    calibredb-export command line option, 317
--allowed-plugin
    fetch-ebook-metadata command line option, 338
--append
    calibredb-set_custom command line option, 321
--as-extra-data-file
    calibredb-add_format command line option, 316
--as-opf
    calibredb-show_metadata command line option, 316
--ascending
    calibredb-list command line option, 313
--asciiize
    ebook-convert command line option, 327
--attachment
    calibre-smtp command line option, 311
--auth-mode
    calibre-server command line option, 307
--author-sort
    ebook-convert command line option, 332
    ebook-meta command line option, 334
--authors
    calibredb-add command line option, 314
    ebook-convert command line option, 332
    ebook-meta command line option, 334
    fetch-ebook-metadata command line option, 338
--auto-reload
    calibre-server command line option, 307
--automerge
    calibredb-add command line option, 314
--ban-after
    calibre-server command line option, 307
--ban-for
    calibre-server command line option, 307
--base-dir
    web2disk command line option, 340
--base-font-size
    ebook-convert command line option, 327
--book-list-mode
    calibre-server command line option, 307
--book-producer
    ebook-convert command line option, 332

ebook-meta command line option, 334
--build-plugin
calibre-customize command line option, 304
--cafile
calibre-smtp command line option, 311
--catalog-title
calibredb-catalog command line option, 318
--categories
calibredb-list_categories command line option, 322
--category
ebook-meta command line option, 334
--change-justification
ebook-convert command line option, 327
--chapter
ebook-convert command line option, 331
--chapter-mark
ebook-convert command line option, 331
--command
calibre-debug command line option, 305
--comments
ebook-convert command line option, 332
ebook-meta command line option, 334
--compress-images
ebook-polish command line option, 336
--compress-min-size
calibre-server command line option, 307
--continue
ebook-viewer command line option, 337
--cover
calibredb-add command line option, 315
ebook-convert command line option, 332
ebook-meta command line option, 334
ebook-polish command line option, 336
fetch-ebook-metadata command line option, 338
--cross-reference-authors
calibredb-catalog command line option, 318
--csv
calibredb-check_library command line option, 322
calibredb-list_categories command line option, 322
--custom-list-template
calibre-server command line option, 308
--customize-plugin
calibre-customize command line option, 304
--daemonize
calibre-server command line option, 308
--date
ebook-meta command line option, 334
--debug-device-driver
calibre-debug command line option, 305
--debug-pipeline
calibredb-catalog command line option, 319
ebook-convert command line option, 333
--default-programs
calibre-debug command line option, 305
--delay
web2disk command line option, 340
--detach
calibre command line option, 304
ebook-edit command line option, 334
ebook-viewer command line option, 337
--details
calibredb-custom_columns command line option, 321
--dialect
calibredb-list_categories command line option, 322
--diff
calibre-debug command line option, 305
--disable-allow-socket-preallocation
calibre-server command line option, 308
--disable-auth
calibre-server command line option, 308
--disable-dehyphenate
ebook-convert command line option, 330
--disable-delete-blank-paragraphs
ebook-convert command line option, 330
--disable-fallback-to-detected-interface
calibre-server command line option, 308
--disable-fix-indents
ebook-convert command line option, 330
--disable-font-rescaling
ebook-convert command line option, 327
--disable-format-scene-breaks
ebook-convert command line option, 330
--disable-hyphenation
lrfviewer command line option, 339
--disable-italicize-common-cases
ebook-convert command line option, 330
--disable-local-write
calibre-server command line option, 308
--disable-log-not-found
calibre-server command line option, 308
--disable-markup-chapter-headings
ebook-convert command line option, 330
--disable-plugin
calibre-customize command line option, 304
--disable-remove-fake-margins
ebook-convert command line option, 331

--disable-renumber-headings
    ebook-convert command line option, 330
--disable-unwrap-lines
    ebook-convert command line option, 330
--disable-use-bonjour
    calibre-server command line option, 308
--disable-use-sendfile
    calibre-server command line option, 308
--disallow-rendered-cover
    ebook-meta command line option, 335
--display
    calibredb-add_custom_column command line option, 320
--displayed-fields
    calibre-server command line option, 308
--do-not-match-on-related-words
    calibredb-fts_search command line option, 324
--dont-asciiize
    calibredb-export command line option, 317
--dont-download-stylesheets
    web2disk command line option, 340
--dont-output-resources
    lrf2lrs command line option, 339
--dont-replace
    calibredb-add_format command line option, 316
--dont-save-cover
    calibredb-export command line option, 317
--dont-save-extra-files
    calibredb-export command line option, 317
--dont-split-on-page-breaks
    ebook-convert command line option, 326
--dont-update-metadata
    calibredb-export command line option, 317
--dont-verify-server-certificate
    calibre-smtp command line option, 311
--dont-write-opf
    calibredb-export command line option, 317
--download-external-resources
    ebook-polish command line option, 336
--duplicate-links-in-toc
    ebook-convert command line option, 332
--duplicates
    calibredb-add command line option, 315
--edit-book
    calibre-debug command line option, 305
--embed-all-fonts
    ebook-convert command line option, 327
--embed-font-family
    ebook-convert command line option, 327
--embed-fonts
    ebook-polish command line option, 336
--empty
    calibredb-add command line option, 315
--enable-allow-socket-preallocation
    calibre-server command line option, 308
--enable-auth
    calibre-server command line option, 308
--enable-fallback-to-detected-interface
    calibre-server command line option, 308
--enable-heuristics
    ebook-convert command line option, 330
--enable-local-write
    calibre-server command line option, 308
--enable-log-not-found
    calibre-server command line option, 308
--enable-plugin
    calibre-customize command line option, 304
--enable-use-bonjour
    calibre-server command line option, 308
--enable-use-sendfile
    calibre-server command line option, 308
--encoding
    web2disk command line option, 340
--encryption-method
    calibre-smtp command line option, 311
--epub-flatten
    ebook-convert command line option, 326
--epub-inline-toc
    ebook-convert command line option, 326
--epub-max-image-size
    ebook-convert command line option, 326
--epub-toc-at-end
    ebook-convert command line option, 326
--epub-version
    ebook-convert command line option, 326
--exclude-genre
    calibredb-catalog command line option, 319
--exclusion-rules
    calibredb-catalog command line option, 319
--exec-file
    calibre-debug command line option, 305
--expand-css
    ebook-convert command line option, 328
--explode-book
    calibre-debug command line option, 305
--export-all-calibre-data
    calibre-debug command line option, 306
--extra-css
    ebook-convert command line option, 328
--extract-to
    ebook-convert command line option, 326
--field

calibredb-set_metadata command line option, 317
--fields
calibredb-list command line option, 313
--filter-css
ebook-convert command line option, 328
--filter-regexp
web2disk command line option, 340
--flow-size
ebook-convert command line option, 326
--font-size-mapping
ebook-convert command line option, 328
--for-machine
calibredb-list command line option, 314
--force
calibredb-remove_custom_column command line option, 321
--force-reload
ebook-viewer command line option, 337
--fork
calibre-smtp command line option, 310
--formats
calibredb-export command line option, 317
--from-opf
ebook-convert command line option, 333
ebook-meta command line option, 335
--full-screen
ebook-viewer command line option, 337
--fullscreen
ebook-viewer command line option, 337
--generate-authors
calibredb-catalog command line option, 319
--generate-descriptions
calibredb-catalog command line option, 319
--generate-genres
calibredb-catalog command line option, 319
--generate-recently-added
calibredb-catalog command line option, 319
--generate-series
calibredb-catalog command line option, 319
--generate-titles
calibredb-catalog command line option, 319
--genre-source-field
calibredb-catalog command line option, 319
--get-cover
ebook-meta command line option, 335
--gui
calibre-debug command line option, 306
--gui-debug
calibre-debug command line option, 306
--header-note-source-field
calibredb-catalog command line option, 319
--help
calibre command line option, 304
calibre-customize command line option, 304
calibre-debug command line option, 306
calibre-server command line option, 308
calibre-smtp command line option, 310
command line option, 313
ebook-convert command line option, 325
ebook-edit command line option, 334
ebook-meta command line option, 335
ebook-polish command line option, 336
ebook-viewer command line option, 337
fetch-ebook-metadata command line option, 338
lrf2lrs command line option, 339
lrfviewer command line option, 339
lrs2lrf command line option, 340
web2disk command line option, 340
--html-unwrap-factor
ebook-convert command line option, 330
--identifier
calibredb-add command line option, 315
ebook-meta command line option, 335
fetch-ebook-metadata command line option, 338
--ids
calibredb-catalog command line option, 318
--ignore
calibredb-add command line option, 315
--ignore_extensions
calibredb-check_library command line option, 322
--ignore_names
calibredb-check_library command line option, 322
--ignore-plugins
calibre command line option, 304
--ignored-fields
calibre-server command line option, 308
--implode-book
calibre-debug command line option, 306
--import-calibre-data
calibre-debug command line option, 306
--include-snippets
calibredb-fts_search command line option, 324

--index
    ebook-meta command line option, 335
--indexing-speed
    calibredb-fts_index command line option, 324
--indexing-threshold
    calibredb-fts_search command line option, 324
--input-encoding
    ebook-convert command line option, 326
--input-profile
    ebook-convert command line option, 325
--insert-blank-line
    ebook-convert command line option, 328
--insert-blank-line-size
    ebook-convert command line option, 328
--insert-metadata
    ebook-convert command line option, 331
--inspect-mobi
    calibre-debug command line option, 306
--is-multiple
    calibredb-add_custom_column command line option, 321
--isbn
    calibredb-add command line option, 315
    ebook-convert command line option, 332
    ebook-meta command line option, 335
    fetch-ebook-metadata command line option, 338
--item_count
    calibredb-list_categories command line option, 322
--jacket
    ebook-polish command line option, 336
--keep-ligatures
    ebook-convert command line option, 328
--kepubify
    calibre-debug command line option, 306
--language
    ebook-convert command line option, 333
    ebook-meta command line option, 335
--languages
    calibredb-add command line option, 315
--level1-toc
    ebook-convert command line option, 332
--level2-toc
    ebook-convert command line option, 332
--level3-toc
    ebook-convert command line option, 332
--library-path
    command line option, 313
--limit
    calibredb-list command line option, 314
    calibredb-search command line option, 323
--line-height
    ebook-convert command line option, 328
--line-width
    calibredb-list command line option, 314
--linearize-tables
    ebook-convert command line option, 328
--list-fields
    calibredb-set_metadata command line option, 317
--list-plugins
    calibre-customize command line option, 305
--list-recipes
    ebook-convert command line option, 325
--listen-on
    calibre-server command line option, 309
--localhost
    calibre-smtp command line option, 311
--log
    calibre-server command line option, 309
--lrf-bookid
    ebook-meta command line option, 335
--lrs
    lrs2lrf command line option, 340
--manage-users
    calibre-server command line option, 309
--margin-bottom
    ebook-convert command line option, 328
--margin-left
    ebook-convert command line option, 328
--margin-right
    ebook-convert command line option, 329
--margin-top
    ebook-convert command line option, 329
--match-end-marker
    calibredb-fts_search command line option, 324
--match-regexp
    web2disk command line option, 340
--match-start-marker
    calibredb-fts_search command line option, 324
--max-files
    web2disk command line option, 341
--max-header-line-size
    calibre-server command line option, 309
--max-job-time
    calibre-server command line option, 309
--max-jobs
    calibre-server command line option, 309
--max-log-size
    calibre-server command line option, 309
--max-opds-items
    calibre-server command line option, 309

--max-opds-ungrouped-items
    calibre-server command line option, 309
--max-recursions
    web2disk command line option, 341
--max-request-body-size
    calibre-server command line option, 309
--max-toc-links
    ebook-convert command line option, 332
--merge-comments-rule
    calibredb-catalog command line option, 319
--minimum-line-height
    ebook-convert command line option, 329
--new-instance
    ebook-viewer command line option, 337
--no-chapters-in-toc
    ebook-convert command line option, 332
--no-default-epub-cover
    ebook-convert command line option, 327
--no-svg-cover
    ebook-convert command line option, 327
--no-update-check
    calibre command line option, 304
--num-per-page
    calibre-server command line option, 309
--one-book-per-directory
    calibredb-add command line option, 315
--only-formats
    calibredb-embed_metadata command line option, 323
--open-at
    ebook-viewer command line option, 337
--opf
    ebook-polish command line option, 336
    fetch-ebook-metadata command line option, 338
--outbox
    calibre-smtp command line option, 311
--output
    lrf2lrs command line option, 339
    lrs2lrf command line option, 340
--output-format
    calibredb-fts_search command line option, 324
--output-profile
    calibredb-catalog command line option, 319
    ebook-convert command line option, 326
--page-breaks-before
    ebook-convert command line option, 331
--password
    calibre-smtp command line option, 311
    command line option, 313
--paths
    calibre-debug command line option, 306
--permanent
    calibredb-remove command line option, 316
--pidfile
    calibre-server command line option, 309
--port
    calibre-server command line option, 309
    calibre-smtp command line option, 311
--prefer-metadata-cover
    ebook-convert command line option, 331
--prefix
    calibredb-list command line option, 314
--prefix-rules
    calibredb-catalog command line option, 319
--preserve-cover-aspect-ratio
    ebook-convert command line option, 327
--preset
    calibredb-catalog command line option, 320
--pretty-print
    ebook-convert command line option, 327
--profile
    lrfviewer command line option, 339
--progress
    calibredb-export command line option, 318
--pubdate
    ebook-convert command line option, 333
--publisher
    ebook-convert command line option, 333
    ebook-meta command line option, 335
--raise-window
    ebook-viewer command line option, 338
--rating
    ebook-convert command line option, 333
    ebook-meta command line option, 335
--read-metadata-from-opf
    ebook-convert command line option, 333
--really-do-it
    calibredb-restore_database command line option, 321
--recurse
    calibredb-add command line option, 315
--relay
    calibre-smtp command line option, 311
--remove-first-image
    ebook-convert command line option, 331
--remove-jacket
    ebook-polish command line option, 336
--remove-paragraph-spacing
    ebook-convert command line option, 329
--remove-paragraph-spacing-indent-size
    ebook-convert command line option, 329
--remove-plugin

calibre-customize command line option, 305
--remove-soft-hyphens
ebook-polish command line option, 336
--remove-unused-css
ebook-polish command line option, 336
--replace-scene-breaks
ebook-convert command line option, 330
--replace-whitespace
calibredb-export command line option, 318
--report
calibredb-check_library command line option, 322
--restrict-to
calibredb-fts_search command line option, 324
--run-plugin
calibre-debug command line option, 306
--run-test
calibre-debug command line option, 306
--run-without-debug
calibre-debug command line option, 306
--search
calibredb-catalog command line option, 318
calibredb-list command line option, 314
--search-replace
ebook-convert command line option, 330
--search-the-net-urls
calibre-server command line option, 309
--select-text
ebook-edit command line option, 334
--separator
calibredb-list command line option, 314
--series
calibredb-add command line option, 315
ebook-convert command line option, 333
ebook-meta command line option, 335
--series-index
calibredb-add command line option, 315
ebook-convert command line option, 333
--shutdown-running-calibre
calibre command line option, 304
calibre-debug command line option, 306
--shutdown-timeout
calibre-server command line option, 310
--single-dir
calibredb-export command line option, 318
--smarten-punctuation
ebook-convert command line option, 329
ebook-polish command line option, 336
--sort-by
calibredb-list command line option, 314
--sr1-replace
ebook-convert command line option, 330
--sr1-search
ebook-convert command line option, 330
--sr2-replace
ebook-convert command line option, 330
--sr2-search
ebook-convert command line option, 331
--sr3-replace
ebook-convert command line option, 331
--sr3-search
ebook-convert command line option, 331
--ssl-certfile
calibre-server command line option, 310
--ssl-keyfile
calibre-server command line option, 310
--start-in-tray
calibre command line option, 304
--start-reading-at
ebook-convert command line option, 331
--subject
calibre-smtp command line option, 311
--subset-embedded-fonts
ebook-convert command line option, 329
--subset-font
calibre-debug command line option, 306
--subset-fonts
ebook-polish command line option, 337
--tags
calibredb-add command line option, 315
ebook-convert command line option, 333
ebook-meta command line option, 335
--template
calibredb-export command line option, 318
calibredb-list command line option, 314
--template_file
calibredb-list command line option, 314
--template_heading
calibredb-list command line option, 314
--test-build
calibre-debug command line option, 307
--thumb-width
calibredb-catalog command line option, 320
--timefmt
calibredb-export command line option, 318
--timeout
calibre-server command line option, 310
calibre-smtp command line option, 311
command line option, 313
fetch-ebook-metadata command line option, 338
web2disk command line option, 341
--timestamp
ebook-convert command line option, 333

--title
    calibredb-add command line option, 315
    ebook-convert command line option, 333
    ebook-meta command line option, 335
    fetch-ebook-metadata command line option, 338
--title-sort
    ebook-convert command line option, 333
    ebook-meta command line option, 335
--to-dir
    calibredb-export command line option, 318
--to-lowercase
    calibredb-export command line option, 318
--to-opf
    ebook-meta command line option, 335
--toc-filter
    ebook-convert command line option, 332
--toc-threshold
    ebook-convert command line option, 332
--toc-title
    ebook-convert command line option, 327
--transform-css-rules
    ebook-convert command line option, 329
--transform-html-rules
    ebook-convert command line option, 329
--trusted-ips
    calibre-server command line option, 310
--un-kepubify
    calibre-debug command line option, 307
--unsmarten-punctuation
    ebook-convert command line option, 329
--upgrade-book
    ebook-polish command line option, 337
--url-prefix
    calibre-server command line option, 310
--use-auto-toc
    ebook-convert command line option, 332
--use-existing-cover
    calibredb-catalog command line option, 320
--userdb
    calibre-server command line option, 310
--username
    calibre-smtp command line option, 311
    command line option, 313
--vacuum-fts-db
    calibredb-check_library command line option, 322
--verbose
    calibre command line option, 304
    calibre-smtp command line option, 311
    calibredb-catalog command line option, 318
    ebook-convert command line option, 333
    ebook-polish command line option, 337
    fetch-ebook-metadata command line option, 338
    lrf2lrs command line option, 339
    lrfviewer command line option, 339
    lrs2lrf command line option, 340
    web2disk command line option, 341
--version
    calibre command line option, 304
    calibre-customize command line option, 305
    calibre-debug command line option, 307
    calibre-server command line option, 310
    calibre-smtp command line option, 311
    command line option, 313
    ebook-convert command line option, 326
    ebook-edit command line option, 334
    ebook-meta command line option, 335
    ebook-polish command line option, 337
    ebook-viewer command line option, 338
    fetch-ebook-metadata command line option, 338
    lrf2lrs command line option, 339
    lrfviewer command line option, 339
    lrs2lrf command line option, 340
    web2disk command line option, 341
--viewer
    calibre-debug command line option, 307
--visual-debug
    lrfviewer command line option, 339
--wait-for-completion
    calibredb-fts_index command line option, 324
--white-background
    lrfviewer command line option, 339
--width
    calibredb-list_categories command line option, 322
--with-library
    calibre command line option, 304
    command line option, 313
--worker-count
    calibre-server command line option, 310
-a
    calibre-customize command line option, 304
    calibre-smtp command line option, 311
    calibredb-add command line option, 314
    calibredb-set_custom command line option, 321
    ebook-meta command line option, 334
    fetch-ebook-metadata command line option, 338
-b

calibre-customize command line option, 304
-c
calibre-debug command line option, 305
calibredb-add command line option, 315
calibredb-check_library command line option, 322
calibredb-list_categories command line option, 322
ebook-meta command line option, 334
ebook-polish command line option, 336
fetch-ebook-metadata command line option, 338
-d
calibre-debug command line option, 305
calibredb-add command line option, 315
calibredb-custom_columns command line option, 321
ebook-convert command line option, 333
ebook-meta command line option, 334
ebook-polish command line option, 336
fetch-ebook-metadata command line option, 338
web2disk command line option, 340
-e
calibre-debug command line option, 305
calibre-smtp command line option, 311
calibredb-add command line option, 315
calibredb-check_library command line option, 322
ebook-polish command line option, 336
-f
calibre-debug command line option, 306
calibre-smtp command line option, 310
calibredb-embed_metadata command line option, 323
calibredb-list command line option, 313
calibredb-remove_custom_column command line option, 321
calibredb-set_metadata command line option, 317
ebook-polish command line option, 337
ebook-viewer command line option, 337
-g
calibre-debug command line option, 306
-h
calibre command line option, 304
calibre-customize command line option, 304
calibre-debug command line option, 306
calibre-server command line option, 308
calibre-smtp command line option, 310
command line option, 313
ebook-convert command line option, 325
ebook-edit command line option, 334
ebook-meta command line option, 335
ebook-polish command line option, 336
ebook-viewer command line option, 337
fetch-ebook-metadata command line option, 338
lrf2lrs command line option, 339
lrfviewer command line option, 339
lrs2lrf command line option, 340
web2disk command line option, 340
-i
calibre-debug command line option, 306
calibredb-add command line option, 315
calibredb-catalog command line option, 318
calibredb-list_categories command line option, 322
ebook-meta command line option, 335
ebook-polish command line option, 336
fetch-ebook-metadata command line option, 338
-j
ebook-polish command line option, 336
-k
ebook-meta command line option, 334
-l
calibre-customize command line option, 305
calibre-smtp command line option, 311
calibredb-add command line option, 315
calibredb-search command line option, 323
calibredb-set_metadata command line option, 317
ebook-meta command line option, 335
-m
calibre-debug command line option, 306
calibredb-add command line option, 314
ebook-convert command line option, 333
-n
calibredb-check_library command line option, 322
web2disk command line option, 341
-o
calibre-smtp command line option, 311
ebook-polish command line option, 336
fetch-ebook-metadata command line option, 338
lrf2lrs command line option, 339
lrs2lrf command line option, 340
-p
calibre-smtp command line option, 311
ebook-meta command line option, 335
ebook-polish command line option, 336

fetch-ebook-metadata command line option, 338
-r
calibre-customize command line option, 305
calibre-debug command line option, 306
calibre-smtp command line option, 311
calibredb-add command line option, 315
calibredb-check_library command line option, 322
calibredb-list_categories command line option, 322
calibredb-restore_database command line option, 321
ebook-meta command line option, 335
web2disk command line option, 341
-s
calibre command line option, 304
calibre-debug command line option, 306
calibre-smtp command line option, 311
calibredb-add command line option, 315
calibredb-catalog command line option, 318
calibredb-list command line option, 314
ebook-meta command line option, 335
-t
calibre-debug command line option, 306
calibre-smtp command line option, 311
calibredb-add command line option, 315
calibredb-list command line option, 314
ebook-meta command line option, 335
fetch-ebook-metadata command line option, 338
web2disk command line option, 341
-u
calibre-smtp command line option, 311
ebook-polish command line option, 336
-v
calibre command line option, 304
calibre-smtp command line option, 311
calibredb-catalog command line option, 318
ebook-convert command line option, 333
fetch-ebook-metadata command line option, 338
-w
calibre-debug command line option, 307
calibredb-list command line option, 314
calibredb-list_categories command line option, 322
-x
calibre-debug command line option, 305

A

abort_article() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 42
abort_recipe_processing() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 42
abspath_to_name() (*metoda calibre.ebooks.oeb.polish.container.Container*), 362
accept_drag_move_event() (*metoda calibre.gui2.actions.InterfaceAction*), 282
accept_enter_event() (*metoda calibre.gui2.actions.InterfaceAction*), 282
accepts_drops (*atribut calibre.gui2.actions.InterfaceAction*), 282
action_add_menu (*atribut calibre.gui2.actions.InterfaceAction*), 282
action_menu_clone_qaction (*atribut calibre.gui2.actions.InterfaceAction*), 282
action_shortcut_name (*atribut calibre.gui2.actions.InterfaceAction*), 282
action_spec (*atribut calibre.gui2.actions.InterfaceAction*), 282
action_type (*atribut calibre.gui2.actions.InterfaceAction*), 282
add_annotation_to_library() (*metoda calibre.devices.usbms.device.Device*), 279
add_book() (*metoda calibre.devices.interface.BookList*), 275
add_books() (*metoda calibre.db.cache.Cache*), 352
add_books_to_metadata() (*metoda calibre.devices.usbms.driver.USBMS*), 280
add_books_to_metadata() (*třídní metoda calibre.devices.interface.DevicePlugin*), 272
add_custom_book_data() (*metoda calibre.db.cache.Cache*), 352
add_extra_files() (*metoda calibre.db.cache.Cache*), 352
add_file() (*metoda calibre.ebooks.oeb.polish.container.Container*), 362
add_format() (*metoda calibre.db.cache.Cache*), 352
add_listener() (*metoda calibre.db.cache.Cache*), 352
add_name_to_manifest() (*metoda calibre.ebooks.oeb.polish.container.Container*), 362
add_notes_resource() (*metoda calibre.db.cache.Cache*), 352
add_or_replace_jacket() (*v modulu calibre.ebooks.oeb.polish.jacket*), 366
add_properties() (*metoda calibre.ebooks.oeb.polish.container.Container*), 362
add_savepoint() (*metoda calibre.gui2.tweak_book.boss.Boss*), 370

add_toc_thumbnail() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 42
adeify_images() (*třídní metoda calibre.web.feeds.news.BasicNewsRecipe*), 41
all_annotation_types() (*metoda calibre.db.cache.Cache*), 353
all_annotation_users() (*metoda calibre.db.cache.Cache*), 353
all_annotations() (*metoda calibre.db.cache.Cache*), 353
all_annotations_for_book() (*metoda calibre.db.cache.Cache*), 353
all_book_ids() (*metoda calibre.db.cache.Cache*), 353
all_field_for() (*metoda calibre.db.cache.Cache*), 353
all_field_ids() (*metoda calibre.db.cache.Cache*), 353
all_field_keys() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 211
all_field_names() (*metoda calibre.db.cache.Cache*), 353
all_non_none_fields() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 211
allowed_in_menu (*atribut calibre.gui2.tweak_book.plugin.Tool*), 369
allowed_in_toolbar (*atribut calibre.gui2.tweak_book.plugin.Tool*), 369
annotation_count_for_book() (*metoda calibre.db.cache.Cache*), 353
annotations_map_for_book() (*metoda calibre.db.cache.Cache*), 353
API, **375**
apply_container_update_to_gui() (*metoda calibre.gui2.tweak_book.boss.Boss*), 370
apply_unique_properties() (*metoda calibre.ebooks.oeb.polish.container.Container*), 362
articles_are_obfuscated (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 46
ASK_TO_ALLOW_CONNECT (*atribut calibre.devices.interface.DevicePlugin*), 269
author (*atribut calibre.customize.InterfaceActionBase*), 284
author (*atribut calibre.customize.MetadataReaderPlugin*), 260
author (*atribut calibre.customize.MetadataWriterPlugin*), 261
author (*atribut calibre.customize.Plugin*), 256
author (*atribut calibre.customize.PreferencesPlugin*), 285
author (*atribut calibre.devices.usbms.driver.USBMS*), 279
author (*atribut calibre.ebooks.metadata.sources.base.Source*), 262
author_data() (*metoda calibre.db.cache.Cache*), 353
author_sort_from_authors() (*metoda calibre.db.cache.Cache*), 353
auto_cleanup (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 46
auto_cleanup_keep (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 46
auto_repeat (*atribut calibre.gui2.actions.InterfaceAction*), 282
auto_trim_covers (*atribut calibre.ebooks.metadata.sources.base.Source*), 262

B

BasicNewsRecipe (*třída v calibre.web.feeds.news*), 41
BCD (*atribut calibre.devices.interface.DevicePlugin*), 268
BCD (*atribut calibre.devices.usbms.device.Device*), 277
book_class (*atribut calibre.devices.usbms.driver.USBMS*), 279
book_created (*atribut calibre.db.cache.Cache.EventType*), 351
book_edited (*atribut calibre.db.cache.Cache.EventType*), 351
book_type (*atribut calibre.ebooks.oeb.polish.container.Container*), 362
BookList (*třída v calibre.devices.interface*), 275
booklist_class (*atribut calibre.devices.usbms.driver.USBMS*), 279
books() (*metoda calibre.devices.interface.DevicePlugin*), 272
books() (*metoda calibre.devices.usbms.driver.USBMS*), 280
books_for_field() (*metoda calibre.db.cache.Cache*), 353
books_in_virtual_library() (*metoda calibre.db.cache.Cache*), 353
books_removed (*atribut calibre.db.cache.Cache.EventType*), 351
boss (*calibre.gui2.tweak_book.plugin.Tool property*), 369
Boss (*třída v calibre.gui2.tweak_book.boss*), 370
browser_type (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 46
BuiltinAdd (*třída v calibre.utils.formatter_functions*), 184
BuiltinAnd (*třída v calibre.utils.formatter_functions*), 185
BuiltinAnnotationCount (*třída v calibre.utils.formatter_functions*), 193
BuiltinApproximateFormats (*třída v calibre.utils.formatter_functions*), 193
BuiltinArguments (*třída v calibre.utils.formatter_functions*), 203
BuiltinAssign (*třída v calibre.utils.formatter_functions*), 203
BuiltinAuthorLinks (*třída v calibre.utils.formatter_functions*), 193

BuiltinAuthorSorts (*třída v calibre.utils.formatter_functions*), 194
BuiltinBookCount (*třída v calibre.utils.formatter_functions*), 186
BuiltinBooksize (*třída v calibre.utils.formatter_functions*), 194
BuiltinBookValues (*třída v calibre.utils.formatter_functions*), 187
BuiltinCapitalize (*třída v calibre.utils.formatter_functions*), 186
BuiltinCeiling (*třída v calibre.utils.formatter_functions*), 184
BuiltinCharacter (*třída v calibre.utils.formatter_functions*), 205
BuiltinCheckYesNo (*třída v calibre.utils.formatter_functions*), 205
BuiltinCmp (*třída v calibre.utils.formatter_functions*), 204
BuiltinConnectedDeviceName (*třída v calibre.utils.formatter_functions*), 194
BuiltinConnectedDeviceUUID (*třída v calibre.utils.formatter_functions*), 194
BuiltinContains (*třída v calibre.utils.formatter_functions*), 205
BuiltinCount (*třída v calibre.utils.formatter_functions*), 199
BuiltinCurrentLibraryName (*třída v calibre.utils.formatter_functions*), 194
BuiltinCurrentLibraryPath (*třída v calibre.utils.formatter_functions*), 194
BuiltinCurrentVirtualLibraryName (*třída v calibre.utils.formatter_functions*), 195
BuiltinDateArithmetic (*třída v calibre.utils.formatter_functions*), 189
BuiltinDaysBetween (*třída v calibre.utils.formatter_functions*), 190
BuiltinDivide (*třída v calibre.utils.formatter_functions*), 185
BuiltinEncodeForURL (*třída v calibre.utils.formatter_functions*), 208
BuiltinEval (*třída v calibre.utils.formatter_functions*), 204
BuiltinExtraFileModtime (*třída v calibre.utils.formatter_functions*), 187
BuiltinExtraFileNames (*třída v calibre.utils.formatter_functions*), 187
BuiltinExtraFileSize (*třída v calibre.utils.formatter_functions*), 188
BuiltinField (*třída v calibre.utils.formatter_functions*), 195
BuiltinFieldExists (*třída v calibre.utils.formatter_functions*), 206
BuiltinFieldListCount (*třída v calibre.utils.formatter_functions*), 199
BuiltinFinishFormatting (*třída v calibre.utils.formatter_functions*), 190
BuiltinFirstMatchingCmp (*třída v calibre.utils.formatter_functions*), 204
BuiltinFirstNonEmpty (*třída v calibre.utils.formatter_functions*), 197
BuiltinFloor (*třída v calibre.utils.formatter_functions*), 185
BuiltinFormatDate (*třída v calibre.utils.formatter_functions*), 190
BuiltinFormatDateField (*třída v calibre.utils.formatter_functions*), 191
BuiltinFormatDuration (*třída v calibre.utils.formatter_functions*), 192
BuiltinFormatNumber (*třída v calibre.utils.formatter_functions*), 193
BuiltinFormatsModtimes (*třída v calibre.utils.formatter_functions*), 195
BuiltinFormatsPaths (*třída v calibre.utils.formatter_functions*), 196
BuiltinFormatsPathSegments (*třída v calibre.utils.formatter_functions*), 195
BuiltinFormatsSizes (*třída v calibre.utils.formatter_functions*), 196
BuiltinFractionalPart (*třída v calibre.utils.formatter_functions*), 185
BuiltinGetLink (*třída v calibre.utils.formatter_functions*), 188
BuiltinGetNote (*třída v calibre.utils.formatter_functions*), 188
BuiltinGlobals (*třída v calibre.utils.formatter_functions*), 203
BuiltinHasCover (*třída v calibre.utils.formatter_functions*), 196
BuiltinHasExtraFiles (*třída v calibre.utils.formatter_functions*), 189
BuiltinHasNote (*třída v calibre.utils.formatter_functions*), 189
BuiltinHumanReadable (*třída v calibre.utils.formatter_functions*), 193
BuiltinIdentifierInList (*třída v calibre.utils.formatter_functions*), 198
BuiltinIfempty (*třída v calibre.utils.formatter_functions*), 206
BuiltinInList (*třída v calibre.utils.formatter_functions*), 198
BuiltinIsDarkMode (*třída v calibre.utils.formatter_functions*), 203
BuiltinIsMarked (*třída v calibre.utils.formatter_functions*), 196
BuiltinLanguageCodes (*třída v calibre.utils.formatter_functions*), 196
BuiltinLanguageStrings (*třída v calibre.utils.formatter_functions*), 196

BuiltinListCountMatching (*třída v calibre.utils.formatter_functions*), 199
BuiltinListDifference (*třída v calibre.utils.formatter_functions*), 200
BuiltinListEquals (*třída v calibre.utils.formatter_functions*), 200
BuiltinListIntersection (*třída v calibre.utils.formatter_functions*), 200
BuiltinListitem (*třída v calibre.utils.formatter_functions*), 198
BuiltinListJoin (*třída v calibre.utils.formatter_functions*), 200
BuiltinListRe (*třída v calibre.utils.formatter_functions*), 201
BuiltinListReGroup (*třída v calibre.utils.formatter_functions*), 201
BuiltinListRemoveDuplicates (*třída v calibre.utils.formatter_functions*), 201
BuiltinListSort (*třída v calibre.utils.formatter_functions*), 201
BuiltinListSplit (*třída v calibre.utils.formatter_functions*), 201
BuiltinListUnion (*třída v calibre.utils.formatter_functions*), 201
BuiltinLookup (*třída v calibre.utils.formatter_functions*), 198
BuiltinLowercase (*třída v calibre.utils.formatter_functions*), 186
BuiltinMakeUrl (*třída v calibre.utils.formatter_functions*), 208
BuiltinMakeUrlExtended (*třída v calibre.utils.formatter_functions*), 208
BuiltinMod (*třída v calibre.utils.formatter_functions*), 185
BuiltinMultiply (*třída v calibre.utils.formatter_functions*), 185
BuiltinNot (*třída v calibre.utils.formatter_functions*), 186
BuiltinOndevice (*třída v calibre.utils.formatter_functions*), 197
BuiltinOr (*třída v calibre.utils.formatter_functions*), 186
BuiltinPrint (*třída v calibre.utils.formatter_functions*), 204
BuiltinQueryString (*třída v calibre.utils.formatter_functions*), 209
BuiltinRange (*třída v calibre.utils.formatter_functions*), 202
BuiltinRatingToStars (*třída v calibre.utils.formatter_functions*), 193
BuiltinRawField (*třída v calibre.utils.formatter_functions*), 197
BuiltinRawList (*třída v calibre.utils.formatter_functions*), 197
BuiltinRe (*třída v calibre.utils.formatter_functions*), 206
BuiltinReGroup (*třída v calibre.utils.formatter_functions*), 206
BuiltinRound (*třída v calibre.utils.formatter_functions*), 185
BuiltinSelect (*třída v calibre.utils.formatter_functions*), 199
BuiltinSeriesSort (*třída v calibre.utils.formatter_functions*), 197
BuiltinSetGlobals (*třída v calibre.utils.formatter_functions*), 204
BuiltinShorten (*třída v calibre.utils.formatter_functions*), 206
BuiltinStrcat (*třída v calibre.utils.formatter_functions*), 206
BuiltinStrcatMax (*třída v calibre.utils.formatter_functions*), 207
BuiltinStrcmp (*třída v calibre.utils.formatter_functions*), 205
BuiltinStrcmpcase (*třída v calibre.utils.formatter_functions*), 205
BuiltinStrInList (*třída v calibre.utils.formatter_functions*), 199
BuiltinStrlen (*třída v calibre.utils.formatter_functions*), 207
BuiltinSubitems (*třída v calibre.utils.formatter_functions*), 202
BuiltinSublist (*třída v calibre.utils.formatter_functions*), 203
BuiltinSubstr (*třída v calibre.utils.formatter_functions*), 207
BuiltinSubtract (*třída v calibre.utils.formatter_functions*), 185
BuiltinSwapAroundArticles (*třída v calibre.utils.formatter_functions*), 207
BuiltinSwapAroundComma (*třída v calibre.utils.formatter_functions*), 207
BuiltinSwitch (*třída v calibre.utils.formatter_functions*), 198
BuiltinSwitchIf (*třída v calibre.utils.formatter_functions*), 198
BuiltinTemplate (*třída v calibre.utils.formatter_functions*), 204
BuiltinTest (*třída v calibre.utils.formatter_functions*), 207
BuiltinTitlecase (*třída v calibre.utils.formatter_functions*), 186
BuiltinToday (*třída v calibre.utils.formatter_functions*), 190
BuiltinToHex (*třída v calibre.utils.formatter_functions*), 210
BuiltinTransliterate (*třída v calibre.utils.formatter_functions*), 207
BuiltinUppercase (*třída v calibre.utils.formatter_functions*), 186

BuiltinUrlsFromIdentifiers (*třída v calibre.utils.formatter_functions*), 210
BuiltinUserCategories (*třída v calibre.utils.formatter_functions*), 197
BuiltinVirtualLibraries (*třída v calibre.utils.formatter_functions*), 197

C

Cache (*třída v calibre.db.cache*), 351
Cache.EventType (*třída v calibre.db.cache*), 351
cached_cover_url_is_reliable (*atribut calibre.ebooks.metadata.sources.base.Source*), 262
calibre command line option
    --detach, 304
    --help, 304
    --ignore-plugins, 304
    --no-update-check, 304
    --shutdown-running-calibre, 304
    --start-in-tray, 304
    --verbose, 304
    --version, 304
    --with-library, 304
    -h, 304
    -s, 304
    -v, 304
calibre.customize
    module, 256
calibre.customize.conversion
    module, 265
calibre.db.cache
    module, 351
calibre.devices.interface
    module, 268
calibre.ebooks.metadata.book.base
    module, 210
calibre.ebooks.metadata.sources.base
    module, 262
calibre.ebooks.oeb.polish.container
    module, 361
calibre.ebooks.oeb.polish.cover
    module, 367
calibre.ebooks.oeb.polish.css
    module, 367
calibre.ebooks.oeb.polish.jacket
    module, 366
calibre.ebooks.oeb.polish.pretty
    module, 366
calibre.ebooks.oeb.polish.replace
    module, 365
calibre.ebooks.oeb.polish.split
    module, 366
calibre.ebooks.oeb.polish.toc
    module, 368
calibre.gui2.tweak_book.boss
    module, 370
calibre.utils.formatter_functions
    module, 184
calibre.web.feeds.news
    module, 41
calibre-customize command line option
    --add-plugin, 304
    --build-plugin, 304
    --customize-plugin, 304
    --disable-plugin, 304
    --enable-plugin, 304
    --help, 304
    --list-plugins, 305
    --remove-plugin, 305
    --version, 305
    -a, 304
    -b, 304
    -h, 304
    -l, 305
    -r, 305
calibre-debug command line option
    --add-simple-plugin, 305
    --command, 305
    --debug-device-driver, 305
    --default-programs, 305
    --diff, 305
    --edit-book, 305
    --exec-file, 305
    --explode-book, 305
    --export-all-calibre-data, 306
    --gui, 306
    --gui-debug, 306
    --help, 306
    --implode-book, 306
    --import-calibre-data, 306
    --inspect-mobi, 306
    --kepubify, 306
    --paths, 306
    --run-plugin, 306
    --run-test, 306
    --run-without-debug, 306
    --shutdown-running-calibre, 306
    --subset-font, 306
    --test-build, 307
    --un-kepubify, 307
    --version, 307
    --viewer, 307
    -c, 305
    -d, 305
    -e, 305
    -f, 306
    -g, 306
    -h, 306
    -i, 306

-m, 306
-r, 306
-s, 306
-t, 306
-w, 307
-x, 305
calibre-server command line option
--access-log, 307
--ajax-timeout, 307
--auth-mode, 307
--auto-reload, 307
--ban-after, 307
--ban-for, 307
--book-list-mode, 307
--compress-min-size, 307
--custom-list-template, 308
--daemonize, 308
--disable-allow-socket-preallocation, 308
--disable-auth, 308
--disable-fallback-to-detected-interface, 308
--disable-local-write, 308
--disable-log-not-found, 308
--disable-use-bonjour, 308
--disable-use-sendfile, 308
--displayed-fields, 308
--enable-allow-socket-preallocation, 308
--enable-auth, 308
--enable-fallback-to-detected-interface, 308
--enable-local-write, 308
--enable-log-not-found, 308
--enable-use-bonjour, 308
--enable-use-sendfile, 308
--help, 308
--ignored-fields, 308
--listen-on, 309
--log, 309
--manage-users, 309
--max-header-line-size, 309
--max-job-time, 309
--max-jobs, 309
--max-log-size, 309
--max-opds-items, 309
--max-opds-ungrouped-items, 309
--max-request-body-size, 309
--num-per-page, 309
--pidfile, 309
--port, 309
--search-the-net-urls, 309
--shutdown-timeout, 310
--ssl-certfile, 310
--ssl-keyfile, 310
--timeout, 310
--trusted-ips, 310
--url-prefix, 310
--userdb, 310
--version, 310
--worker-count, 310
-h, 308
calibre-smtp command line option
--attachment, 311
--cafile, 311
--dont-verify-server-certificate, 311
--encryption-method, 311
--fork, 310
--help, 310
--localhost, 311
--outbox, 311
--password, 311
--port, 311
--relay, 311
--subject, 311
--timeout, 311
--username, 311
--verbose, 311
--version, 311
-a, 311
-e, 311
-f, 310
-h, 310
-l, 311
-o, 311
-p, 311
-r, 311
-s, 311
-t, 311
-u, 311
-v, 311
calibredb-add command line option
-1, 315
-I, 315
-S, 315
-T, 315
--add, 315
--authors, 314
--automerge, 314
--cover, 315
--duplicates, 315
--empty, 315
--identifier, 315
--ignore, 315
--isbn, 315
--languages, 315
--one-book-per-directory, 315
--recurse, 315
--series, 315

--series-index, 315
--tags, 315
--title, 315
-a, 314
-c, 315
-d, 315
-e, 315
-i, 315
-l, 315
-m, 314
-r, 315
-s, 315
-t, 315
calibredb-add_custom_column command line option
--display, 320
--is-multiple, 321
calibredb-add_format command line option
--as-extra-data-file, 316
--dont-replace, 316
calibredb-backup_metadata command line option
--all, 323
calibredb-catalog command line option
--catalog-title, 318
--cross-reference-authors, 318
--debug-pipeline, 319
--exclude-genre, 319
--exclusion-rules, 319
--generate-authors, 319
--generate-descriptions, 319
--generate-genres, 319
--generate-recently-added, 319
--generate-series, 319
--generate-titles, 319
--genre-source-field, 319
--header-note-source-field, 319
--ids, 318
--merge-comments-rule, 319
--output-profile, 319
--prefix-rules, 319
--preset, 320
--search, 318
--thumb-width, 320
--use-existing-cover, 320
--verbose, 318
-i, 318
-s, 318
-v, 318
calibredb-check_library command line option
--csv, 322
--ignore_extensions, 322
--ignore_names, 322
--report, 322
--vacuum-fts-db, 322
-c, 322
-e, 322
-n, 322
-r, 322
calibredb-custom_columns command line option
--details, 321
-d, 321
calibredb-embed_metadata command line option
--only-formats, 323
-f, 323
calibredb-export command line option
--all, 317
--dont-asciiize, 317
--dont-save-cover, 317
--dont-save-extra-files, 317
--dont-update-metadata, 317
--dont-write-opf, 317
--formats, 317
--progress, 318
--replace-whitespace, 318
--single-dir, 318
--template, 318
--timefmt, 318
--to-dir, 318
--to-lowercase, 318
calibredb-fts_index command line option
--indexing-speed, 324
--wait-for-completion, 324
calibredb-fts_search command line option
--do-not-match-on-related-words, 324
--include-snippets, 324
--indexing-threshold, 324
--match-end-marker, 324
--match-start-marker, 324
--output-format, 324
--restrict-to, 324
calibredb-list command line option
--ascending, 313
--fields, 313
--for-machine, 314
--limit, 314
--line-width, 314
--prefix, 314
--search, 314
--separator, 314
--sort-by, 314
--template, 314
--template_file, 314
--template_heading, 314
-f, 313
-s, 314

-t, 314
-w, 314
calibredb-list_categories command line option
--categories, 322
--csv, 322
--dialect, 322
--item_count, 322
--width, 322
-c, 322
-i, 322
-r, 322
-w, 322
calibredb-remove command line option
--permanent, 316
calibredb-remove_custom_column command line option
--force, 321
-f, 321
calibredb-restore_database command line option
--really-do-it, 321
-r, 321
calibredb-search command line option
--limit, 323
-l, 323
calibredb-set_custom command line option
--append, 321
-a, 321
calibredb-set_metadata command line option
--field, 317
--list-fields, 317
-f, 317
-l, 317
calibredb-show_metadata command line option
--as-opf, 316
can_be_disabled (*atribut calibre.customize.conversion.InputFormatPlugin*), 265
can_be_disabled (*atribut calibre.customize.conversion.OutputFormatPlugin*), 267
can_be_disabled (*atribut calibre.customize.InterfaceActionBase*), 284
can_be_disabled (*atribut calibre.customize.Plugin*), 257
can_be_disabled (*atribut calibre.customize.PreferencesPlugin*), 285
CAN_DO_DEVICE_DB_PLUGBOARD (*atribut calibre.devices.interface.DevicePlugin*), 268
can_get_multiple_covers (*atribut calibre.ebooks.metadata.sources.base.Source*), 262
can_handle() (*metoda calibre.devices.interface.DevicePlugin*), 270
can_handle_windows() (*metoda calibre.devices.interface.DevicePlugin*), 270
can_handle_windows() (*metoda calibre.devices.usbms.device.Device*), 278
CAN_SET_METADATA (*atribut calibre.devices.interface.DevicePlugin*), 268
CAN_SET_METADATA (*atribut calibre.devices.usbms.driver.USBMS*), 279
canonicalize_internal_url() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 42
capabilities (*atribut calibre.ebooks.metadata.sources.base.Source*), 262
card_prefix() (*metoda calibre.devices.interface.DevicePlugin*), 271
card_prefix() (*metoda calibre.devices.usbms.device.Device*), 277
CatalogPlugin (*třída v calibre.customize*), 261
category (*atribut calibre.customize.PreferencesPlugin*), 285
category_order (*atribut calibre.customize.PreferencesPlugin*), 285
center_navbar (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 46
change_font() (*v modulu calibre.ebooks.oeb.polish.fonts*), 367
changed_signal (*atribut calibre.gui2.preferences.ConfigWidgetBase*), 286
changed_signal (*atribut calibre.gui2.preferences.ConfigWidgetInterface*), 285
clean_downloaded_metadata() (*metoda calibre.ebooks.metadata.sources.base.Source*), 263
cleanup() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 42
CLI (*třída v calibre.devices.usbms.cli*), 279
cli_main() (*metoda calibre.customize.Plugin*), 258
cli_options (*atribut calibre.customize.CatalogPlugin*), 261
clone_browser() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 42
close_editor() (*metoda calibre.gui2.tweak_book.boss.Boss*), 370
command line option
--help, 313
--library-path, 313
--password, 313
--timeout, 313
--username, 313
--version, 313
--with-library, 313
-h, 313
commit() (*metoda calibre.ebooks.oeb.polish.container.Container*), 362

commit() (*metoda calibre.gui2.preferences.ConfigWidgetBase*), 287
commit() (*metoda calibre.gui2.preferences.ConfigWidgetInterface*), 286
commit_all_editors_to_container() (*metoda calibre.gui2.tweak_book.boss.Boss*), 370
commit_item() (*metoda calibre.ebooks.oeb.polish.container.Container*), 362
common_options (*atribut calibre.customize.conversion.InputFormatPlugin*), 266
common_options (*atribut calibre.customize.conversion.OutputFormatPlugin*), 267
compress_covers() (*metoda calibre.db.cache.Cache*), 353
compress_news_images (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 46
compress_news_images_auto_size (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 46
compress_news_images_max_size (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 46
config_help_message (*atribut calibre.ebooks.metadata.sources.base.Source*), 262
config_widget (*atribut calibre.customize.PreferencesPlugin*), 285
config_widget() (*metoda calibre.customize.Plugin*), 257
config_widget() (*metoda calibre.ebooks.metadata.sources.base.Source*), 263
config_widget() (*třídní metoda calibre.devices.interface.DevicePlugin*), 273
ConfigWidgetBase (*třída v calibre.gui2.preferences*), 286
ConfigWidgetInterface (*třída v calibre.gui2.preferences*), 285
Container (*třída v calibre.ebooks.oeb.polish.container*), 362
contains(), **160**
conversion_options (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 47
convert() (*metoda calibre.customize.conversion.InputFormatPlugin*), 266
convert() (*metoda calibre.customize.conversion.OutputFormatPlugin*), 267
copy_cover_to() (*metoda calibre.db.cache.Cache*), 354
copy_format_to() (*metoda calibre.db.cache.Cache*), 354
core_usage (*atribut calibre.customize.conversion.InputFormatPlugin*), 265
cover() (*metoda calibre.db.cache.Cache*), 354
cover_margins (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 47
create_action() (*metoda calibre.gui2.tweak_book.plugin.Tool*), 369
create_inline_toc() (*v modulu calibre.ebooks.oeb.polish.toc*), 368
create_menu_action() (*metoda calibre.gui2.actions.InterfaceAction*), 283
create_widget() (*metoda calibre.customize.PreferencesPlugin*), 285
CSS, **375**
current_container (*calibre.gui2.tweak_book.plugin.Tool property*), 369
currently_editing (*calibre.gui2.tweak_book.boss.Boss property*), 370
custom_field_keys() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 211
customization_help() (*metoda calibre.customize.Plugin*), 258
customization_help() (*metoda calibre.ebooks.metadata.sources.base.Source*), 262

D

data_for_find_identical_books() (*metoda calibre.db.cache.Cache*), 354
data_for_has_book() (*metoda calibre.db.cache.Cache*), 354
debug_managed_device_detection() (*metoda calibre.devices.interface.DevicePlugin*), 270
deepcopy() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 210
default_cover() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 42
delay (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 47
delete_annotations() (*metoda calibre.db.cache.Cache*), 354
delete_books() (*metoda calibre.devices.interface.DevicePlugin*), 272
delete_books() (*metoda calibre.devices.usbms.driver.USBMS*), 281
delete_custom_book_data() (*metoda calibre.db.cache.Cache*), 354
delete_trash_entry() (*metoda calibre.db.cache.Cache*), 354
description (*atribut calibre.customize.Plugin*), 256
description (*atribut calibre.customize.PreferencesPlugin*), 285

description (*atribut calibre.devices.usbms.driver.USBMS*), 279
description (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 47
description (*calibre.customize.conversion.OutputFormatPlugin property*), 267
detect_managed_devices() (*metoda calibre.devices.interface.DevicePlugin*), 269
Device (*třída v calibre.devices.usbms.device*), 276
DevicePlugin (*třída v calibre.devices.interface*), 268
dirty() (*metoda calibre.ebooks.oeb.polish.container.Container*), 362
do_user_config() (*metoda calibre.customize.Plugin*), 257
dont_add_to (*atribut calibre.gui2.actions.InterfaceAction*), 282
dont_remove_from (*atribut calibre.gui2.actions.InterfaceAction*), 282
download() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 42
download_cover() (*metoda calibre.ebooks.metadata.sources.base.Source*), 264
drop_event() (*metoda calibre.gui2.actions.InterfaceAction*), 283

E

ebook-convert command line option
--add-alt-text-to-img, 331
--asciiize, 327
--author-sort, 332
--authors, 332
--base-font-size, 327
--book-producer, 332
--change-justification, 327
--chapter, 331
--chapter-mark, 331
--comments, 332
--cover, 332
--debug-pipeline, 333
--disable-dehyphenate, 330
--disable-delete-blank-paragraphs, 330
--disable-fix-indents, 330
--disable-font-rescaling, 327
--disable-format-scene-breaks, 330
--disable-italicize-common-cases, 330
--disable-markup-chapter-headings, 330
--disable-remove-fake-margins, 331
--disable-renumber-headings, 330
--disable-unwrap-lines, 330
--dont-split-on-page-breaks, 326
--duplicate-links-in-toc, 332
--embed-all-fonts, 327
--embed-font-family, 327
--enable-heuristics, 330
--epub-flatten, 326
--epub-inline-toc, 326
--epub-max-image-size, 326
--epub-toc-at-end, 326
--epub-version, 326
--expand-css, 328
--extra-css, 328
--extract-to, 326
--filter-css, 328
--flow-size, 326
--font-size-mapping, 328
--from-opf, 333
--help, 325
--html-unwrap-factor, 330
--input-encoding, 326
--input-profile, 325
--insert-blank-line, 328
--insert-blank-line-size, 328
--insert-metadata, 331
--isbn, 332
--keep-ligatures, 328
--language, 333
--level1-toc, 332
--level2-toc, 332
--level3-toc, 332
--line-height, 328
--linearize-tables, 328
--list-recipes, 325
--margin-bottom, 328
--margin-left, 328
--margin-right, 329
--margin-top, 329
--max-toc-links, 332
--minimum-line-height, 329
--no-chapters-in-toc, 332
--no-default-epub-cover, 327
--no-svg-cover, 327
--output-profile, 326
--page-breaks-before, 331
--prefer-metadata-cover, 331
--preserve-cover-aspect-ratio, 327
--pretty-print, 327
--pubdate, 333
--publisher, 333
--rating, 333
--read-metadata-from-opf, 333
--remove-first-image, 331
--remove-paragraph-spacing, 329
--remove-paragraph-spacing-indent-size, 329
--replace-scene-breaks, 330
--search-replace, 330
--series, 333

--series-index, 333
--smarten-punctuation, 329
--sr1-replace, 330
--sr1-search, 330
--sr2-replace, 330
--sr2-search, 331
--sr3-replace, 331
--sr3-search, 331
--start-reading-at, 331
--subset-embedded-fonts, 329
--tags, 333
--timestamp, 333
--title, 333
--title-sort, 333
--toc-filter, 332
--toc-threshold, 332
--toc-title, 327
--transform-css-rules, 329
--transform-html-rules, 329
--unsmarten-punctuation, 329
--use-auto-toc, 332
--verbose, 333
--version, 326
-d, 333
-h, 325
-m, 333
-v, 333
ebook-edit command line option
--detach, 334
--help, 334
--select-text, 334
--version, 334
-h, 334
ebook-meta command line option
--author-sort, 334
--authors, 334
--book-producer, 334
--category, 334
--comments, 334
--cover, 334
--date, 334
--disallow-rendered-cover, 335
--from-opf, 335
--get-cover, 335
--help, 335
--identifier, 335
--index, 335
--isbn, 335
--language, 335
--lrf-bookid, 335
--publisher, 335
--rating, 335
--series, 335
--tags, 335
--title, 335
--title-sort, 335
--to-opf, 335
--version, 335
-a, 334
-c, 334
-d, 334
-h, 335
-i, 335
-k, 334
-l, 335
-p, 335
-r, 335
-s, 335
-t, 335
ebook-polish command line option
-H, 336
-U, 337
--add-soft-hyphens, 336
--compress-images, 336
--cover, 336
--download-external-resources, 336
--embed-fonts, 336
--help, 336
--jacket, 336
--opf, 336
--remove-jacket, 336
--remove-soft-hyphens, 336
--remove-unused-css, 336
--smarten-punctuation, 336
--subset-fonts, 337
--upgrade-book, 337
--verbose, 337
--version, 337
-c, 336
-d, 336
-e, 336
-f, 337
-h, 336
-i, 336
-j, 336
-o, 336
-p, 336
-u, 336
ebook-viewer command line option
--continue, 337
--detach, 337
--force-reload, 337
--full-screen, 337
--fullscreen, 337
--help, 337
--new-instance, 337
--open-at, 337
--raise-window, 338

--version, 338
-f, 337
-h, 337
edit_file() (*metoda calibre.gui2.tweak_book.boss.Boss*), 370
eject() (*metoda calibre.devices.interface.DevicePlugin*), 271
eject() (*metoda calibre.devices.usbms.device.Device*), 279
embed_metadata() (*metoda calibre.db.cache.Cache*), 354
encoding (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 47
exists() (*metoda calibre.ebooks.oeb.polish.container.Container*), 362
expire_old_trash() (*metoda calibre.db.cache.Cache*), 354
export_note() (*metoda calibre.db.cache.Cache*), 354
extra_css (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 47
extract_readable_article() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 42

F

fast_field_for() (*metoda calibre.db.cache.Cache*), 354
feeds (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 47
fetch-ebook-metadata command line option
-I, 338
--allowed-plugin, 338
--authors, 338
--cover, 338
--help, 338
--identifier, 338
--isbn, 338
--opf, 338
--timeout, 338
--title, 338
--verbose, 338
--version, 338
-a, 338
-c, 338
-d, 338
-h, 338
-i, 338
-o, 338
-p, 338
-t, 338
-v, 338
field_for() (*metoda calibre.db.cache.Cache*), 355
field_ids_for() (*metoda calibre.db.cache.Cache*), 355
field_supports_notes() (*metoda calibre.db.cache.Cache*), 355
file_type (*atribut calibre.customize.conversion.OutputFormatPlugin*), 267
file_types (*atribut calibre.customize.CatalogPlugin*), 261
file_types (*atribut calibre.customize.conversion.InputFormatPlugin*), 265
file_types (*atribut calibre.customize.FileTypePlugin*), 258
file_types (*atribut calibre.customize.MetadataReaderPlugin*), 260
file_types (*atribut calibre.customize.MetadataWriterPlugin*), 260
filename_callback() (*metoda calibre.devices.usbms.device.Device*), 279
filesize() (*metoda calibre.ebooks.oeb.polish.container.Container*), 363
FileTypePlugin (*třída v calibre.customize*), 258
filter_css() (*v modulu calibre.ebooks.oeb.polish.css*), 368
filter_regexps (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 47
find_identical_books() (*metoda calibre.db.cache.Cache*), 355
fix_all_html() (*v modulu calibre.ebooks.oeb.polish.pretty*), 366
fix_html() (*v modulu calibre.ebooks.oeb.polish.pretty*), 366
for_viewer (*atribut calibre.customize.conversion.InputFormatPlugin*), 265
format() (*metoda calibre.db.cache.Cache*), 355
format_abspath() (*metoda calibre.db.cache.Cache*), 355
format_added (*atribut calibre.db.cache.Cache.EventType*), 351
format_field() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 211
format_hash() (*metoda calibre.db.cache.Cache*), 355
format_metadata() (*metoda calibre.db.cache.Cache*), 355
FORMATS (*atribut calibre.devices.interface.DevicePlugin*), 268
FORMATS (*atribut calibre.devices.usbms.driver.USBMS*), 279
formats() (*metoda calibre.db.cache.Cache*), 356
formats_removed (*atribut calibre.db.cache.Cache.EventType*), 351
free_space() (*metoda calibre*

re.devices.interface.DevicePlugin), 272
free_space() (*metoda calibre.devices.usbms.device.Device*), 278
from_files() (*v modulu calibre.ebooks.oeb.polish.toc*), 368
from_links() (*v modulu calibre.ebooks.oeb.polish.toc*), 368
from_xpaths() (*v modulu calibre.ebooks.oeb.polish.toc*), 368

G

generate_item() (*metoda calibre.ebooks.oeb.polish.container.Container*), 363
genesis() (*metoda calibre.gui2.actions.InterfaceAction*), 284
genesis() (*metoda calibre.gui2.preferences.ConfigWidgetInterface*), 286
get_all_items_that_have_notes() (*metoda calibre.db.cache.Cache*), 356
get_all_link_maps_for_book() (*metoda calibre.db.cache.Cache*), 356
get_all_standard_metadata() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 211
get_all_user_metadata() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 211
get_annotations() (*metoda calibre.devices.usbms.device.Device*), 279
get_article_url() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 42
get_author_tokens() (*metoda calibre.ebooks.metadata.sources.base.Source*), 263
get_book_url() (*metoda calibre.ebooks.metadata.sources.base.Source*), 263
get_book_url_name() (*metoda calibre.ebooks.metadata.sources.base.Source*), 263
get_book_urls() (*metoda calibre.ebooks.metadata.sources.base.Source*), 263
get_browser() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 42
get_cached_cover_url() (*metoda calibre.ebooks.metadata.sources.base.Source*), 263
get_categories() (*metoda calibre.db.cache.Cache*), 356
get_collections() (*metoda calibre.devices.interface.BookList*), 276
get_cover_url() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 43
get_custom_book_data() (*metoda calibre.db.cache.Cache*), 356
get_device_information() (*metoda calibre.devices.interface.DevicePlugin*), 271
get_device_information() (*metoda calibre.devices.usbms.driver.USBMS*), 280
get_device_uid() (*metoda calibre.devices.interface.DevicePlugin*), 274
get_driveinfo() (*metoda calibre.devices.interface.DevicePlugin*), 271
get_extra_css() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 43
get_feeds() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 43
get_file() (*metoda calibre.devices.interface.DevicePlugin*), 273
get_file_path_for_processing() (*metoda calibre.ebooks.oeb.polish.container.Container*), 363
get_id_map() (*metoda calibre.db.cache.Cache*), 356
get_identifiers() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 210
get_ids_for_custom_book_data() (*metoda calibre.db.cache.Cache*), 356
get_images() (*metoda calibre.customize.conversion.InputFormatPlugin*), 266
get_item_id() (*metoda calibre.db.cache.Cache*), 356
get_item_ids() (*metoda calibre.db.cache.Cache*), 356
get_item_name() (*metoda calibre.db.cache.Cache*), 356
get_item_name_map() (*metoda calibre.db.cache.Cache*), 356
get_link_map() (*metoda calibre.db.cache.Cache*), 356
get_masthead_title() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 43
get_masthead_url() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 43
get_metadata() (*metoda calibre.customize.MetadataReaderPlugin*), 260
get_metadata() (*metoda calibre.db.cache.Cache*), 357
get_next_series_num_for() (*metoda calibre.db.cache.Cache*), 357
get_notes_resource() (*metoda calibre.db.cache.Cache*), 357
get_obfuscated_article() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 43
get_open_popup_message() (*třídní metoda calibre.devices.interface.DevicePlugin*), 269
get_option() (*metoda calibre.devices.interface.DevicePlugin*), 274
get_proxy_metadata() (*metoda calibre.db.cache.Cache*), 357
get_recommended_folders() (*v modulu calibre.ebooks.oeb.polish.replace*), 366
get_standard_metadata() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 211
get_title_tokens() (*metoda calib-*

re.ebooks.metadata.sources.base.Source), 263
get_url_specific_delay() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 44
get_usage_count_by_id() (*metoda calibre.db.cache.Cache*), 357
get_user_blacklisted_devices() (*metoda calibre.devices.interface.DevicePlugin*), 274
get_user_metadata() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 211
gui (*calibre.gui2.tweak_book.plugin.Tool property*), 369
gui_category (*atribut calibre.customize.PreferencesPlugin*), 285
gui_configuration_widget() (*metoda calibre.customize.conversion.InputFormatPlugin*), 266
gui_configuration_widget() (*metoda calibre.customize.conversion.OutputFormatPlugin*), 267
gui_layout_complete() (*metoda calibre.gui2.actions.InterfaceAction*), 284
gui_name (*atribut calibre.customize.PreferencesPlugin*), 285
guide_type_map (*calibre.ebooks.oeb.polish.container.Container property*), 363

H

handle_gzip (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 48
has_book() (*metoda calibre.db.cache.Cache*), 357
has_format() (*metoda calibre.db.cache.Cache*), 357
has_html_comments (*atribut calibre.ebooks.metadata.sources.base.Source*), 262
has_id() (*metoda calibre.db.cache.Cache*), 357
has_name() (*metoda calibre.ebooks.oeb.polish.container.Container*), 363
href_to_name() (*metoda calibre.ebooks.oeb.polish.container.Container*), 363
HTML, **375**

I

icon (*atribut calibre.customize.PreferencesPlugin*), 285
icon (*atribut calibre.devices.interface.DevicePlugin*), 268
id_from_url() (*metoda calibre.ebooks.metadata.sources.base.Source*), 264
identify() (*metoda calibre.ebooks.metadata.sources.base.Source*), 264
identify_results_keygen() (*metoda calibre.ebooks.metadata.sources.base.Source*), 264
ignore_connected_device() (*metoda calibre.devices.interface.DevicePlugin*), 274
ignore_duplicate_articles (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 48
ignore_ssl_errors (*atribut calibre.ebooks.metadata.sources.base.Source*), 262
image_url_processor() (*třídní metoda calibre.web.feeds.news.BasicNewsRecipe*), 41
import_note() (*metoda calibre.db.cache.Cache*), 357
index_to_soup() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 44
indexing_progress_changed (*atribut calibre.db.cache.Cache.EventType*), 352
init() (*metoda calibre.db.cache.Cache*), 357
initial_tab_changed() (*metoda calibre.gui2.preferences.ConfigWidgetInterface*), 286
initialization_complete() (*metoda calibre.gui2.actions.InterfaceAction*), 284
initialize() (*metoda calibre.customize.CatalogPlugin*), 261
initialize() (*metoda calibre.customize.Plugin*), 257
initialize() (*metoda calibre.gui2.preferences.ConfigWidgetBase*), 287
initialize() (*metoda calibre.gui2.preferences.ConfigWidgetInterface*), 286
InputFormatPlugin (*třída v calibre.customize.conversion*), 265
insert_into_xml() (*metoda calibre.ebooks.oeb.polish.container.Container*), 363
installation_type (*atribut calibre.customize.Plugin*), 257
InterfaceAction (*třída v calibre.gui2.actions*), 281
InterfaceActionBase (*třída v calibre.customize*), 284
InternalMetadataCompareKeyGen (*třída v calibre.ebooks.metadata.sources.base*), 265
is_configured() (*metoda calibre.ebooks.metadata.sources.base.Source*), 262
is_dir (*atribut calibre.ebooks.oeb.polish.container.Container*), 363
is_dynamically_controllable() (*metoda calibre.devices.interface.DevicePlugin*), 274
is_image_collection (*atribut calibre.customize.conversion.InputFormatPlugin*), 265
is_link_wanted() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 44
is_null() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 210
is_running() (*metoda calibre.devices.interface.DevicePlugin*), 275
is_usb_connected() (*metoda calibre.devices.interface.DevicePlugin*), 269

items_removed (*atribut calibre.db.cache.Cache.EventType*), 352
items_renamed (*atribut calibre.db.cache.Cache.EventType*), 352
items_with_notes_in_book() (*metoda calibre.db.cache.Cache*), 357
iterlinks() (*metoda calibre.ebooks.oeb.polish.container.Container*), 363

K

keep_only_tags (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 48

L

language (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 48
library_about_to_change() (*metoda calibre.gui2.actions.InterfaceAction*), 284
library_changed() (*metoda calibre.gui2.actions.InterfaceAction*), 284
link_for() (*metoda calibre.db.cache.Cache*), 357
links_changed (*atribut calibre.db.cache.Cache.EventType*), 352
list_extra_files() (*metoda calibre.db.cache.Cache*), 357
load_actual_plugin() (*metoda calibre.customize.InterfaceActionBase*), 284
load_resources() (*metoda calibre.customize.Plugin*), 257
load_resources() (*metoda calibre.gui2.actions.InterfaceAction*), 283
location_selected() (*metoda calibre.gui2.actions.InterfaceAction*), 284
LRF, **375**
lrf2lrs command line option
    --dont-output-resources, 339
    --help, 339
    --output, 339
    --verbose, 339
    --version, 339
    -h, 339
    -o, 339
lrfviewer command line option
    --disable-hyphenation, 339
    --help, 339
    --profile, 339
    --verbose, 339
    --version, 339
    --visual-debug, 339
    --white-background, 339
    -h, 339
lrs2lrf command line option
    --help, 340
    --lrs, 340
    --output, 340
    --verbose, 340
    --version, 340
    -h, 340
    -o, 340

M

make_name_unique() (*metoda calibre.ebooks.oeb.polish.container.Container*), 363
MANAGES_DEVICE_PRESENCE (*atribut calibre.devices.interface.DevicePlugin*), 269
manifest_has_name() (*metoda calibre.ebooks.oeb.polish.container.Container*), 363
manifest_id_map (*calibre.ebooks.oeb.polish.container.Container property*), 363
manifest_items_of_type() (*metoda calibre.ebooks.oeb.polish.container.Container*), 363
manifest_items_with_property() (*metoda calibre.ebooks.oeb.polish.container.Container*), 363
manifest_type_map (*calibre.ebooks.oeb.polish.container.Container property*), 363
mark_as_cover() (*v modulu calibre.ebooks.oeb.polish.cover*), 367
mark_as_titlepage() (*v modulu calibre.ebooks.oeb.polish.cover*), 367
masthead_url (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 48
match_regexps (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 48
max_articles_per_feed (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 48
MAX_PATH_LEN (*atribut calibre.devices.usbms.device.Device*), 277
merge() (*v modulu calibre.ebooks.oeb.polish.split*), 367
merge_annotations_for_book() (*metoda calibre.db.cache.Cache*), 358
merge_extra_files() (*metoda calibre.db.cache.Cache*), 358
Metadata (*třída v calibre.ebooks.metadata.book.base*), 210
metadata_changed (*atribut calibre.db.cache.Cache.EventType*), 352
metadata_for_field() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 211
MetadataReaderPlugin (*třída v calibre.customize*), 260
MetadataWriterPlugin (*třída v calibre.customize*), 260

mi (*calibre.ebooks.oeb.polish.container.Container property*), 363
minimum_calibre_version (*atribut calibre.customize.Plugin*), 257
model_metadata() (*třídní metoda calibre.devices.interface.DevicePlugin*), 269
model_metadata() (*třídní metoda calibre.devices.usbms.device.Device*), 277
module
    calibre.customize, 256
    calibre.customize.conversion, 265
    calibre.db.cache, 351
    calibre.devices.interface, 268
    calibre.ebooks.metadata.book.base, 210
    calibre.ebooks.metadata.sources.base, 262
    calibre.ebooks.oeb.polish.container, 361
    calibre.ebooks.oeb.polish.cover, 367
    calibre.ebooks.oeb.polish.css, 367
    calibre.ebooks.oeb.polish.jacket, 366
    calibre.ebooks.oeb.polish.pretty, 366
    calibre.ebooks.oeb.polish.replace, 365
    calibre.ebooks.oeb.polish.split, 366
    calibre.ebooks.oeb.polish.toc, 368
    calibre.gui2.tweak_book.boss, 370
    calibre.utils.formatter_functions, 184
    calibre.web.feeds.news, 41
move_book_from_trash() (*metoda calibre.db.cache.Cache*), 358
move_format_from_trash() (*metoda calibre.db.cache.Cache*), 358
multisort() (*metoda calibre.db.cache.Cache*), 358
multisplit() (*v modulu calibre.ebooks.oeb.polish.split*), 367

N

name (*atribut calibre.customize.Plugin*), 256
name (*atribut calibre.gui2.actions.InterfaceAction*), 282
name (*atribut calibre.gui2.tweak_book.plugin.Tool*), 369
name(), **160**
name_order (*atribut calibre.customize.PreferencesPlugin*), 285
name_to_abspath() (*metoda calibre.ebooks.oeb.polish.container.Container*), 364
name_to_href() (*metoda calibre.ebooks.oeb.polish.container.Container*), 364
names_that_must_not_be_changed (*calibre.ebooks.oeb.polish.container.Container property*), 364
names_that_must_not_be_removed (*calibre.ebooks.oeb.polish.container.Container property*), 364
names_that_need_not_be_manifested (*calibre.ebooks.oeb.polish.container.Container property*), 364
needs_subscription (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 48
NEWS_IN_FOLDER (*atribut calibre.devices.usbms.device.Device*), 277
no_stylesheets (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 48
normalize_path() (*třídní metoda calibre.devices.usbms.driver.USBMS*), 281
notes_changed (*atribut calibre.db.cache.Cache.EventType*), 352
notes_data_for() (*metoda calibre.db.cache.Cache*), 358
notes_for() (*metoda calibre.db.cache.Cache*), 358
notes_resources_used_by() (*metoda calibre.db.cache.Cache*), 358
NUKE_COMMENTS (*atribut calibre.devices.interface.DevicePlugin*), 269

O

oldest_article (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 49
on_import (*atribut calibre.customize.FileTypePlugin*), 258
on_postconvert (*atribut calibre.customize.FileTypePlugin*), 258
on_postdelete (*atribut calibre.customize.FileTypePlugin*), 258
on_postimport (*atribut calibre.customize.FileTypePlugin*), 258
on_postprocess (*atribut calibre.customize.FileTypePlugin*), 258
on_preprocess (*atribut calibre.customize.FileTypePlugin*), 258
open() (*metoda calibre.devices.interface.DevicePlugin*), 270
open() (*metoda calibre.devices.usbms.device.Device*), 278
open() (*metoda calibre.ebooks.oeb.polish.container.Container*), 364
open_book() (*metoda calibre.gui2.tweak_book.boss.Boss*), 370
OPEN_FEEDBACK_MESSAGE (*atribut calibre.devices.interface.DevicePlugin*), 269
opf (*calibre.ebooks.oeb.polish.container.Container property*), 364
opf_get_or_create() (*metoda calibre.ebooks.oeb.polish.container.Container*), 364
opf_version (*calibre.ebooks.oeb.polish.container.Container property*), 364
opf_version_parsed (*calibre.ebooks.oeb.polish.container.Container proper-*

ty), 364
opf_xpath() (*metoda calibre.ebooks.oeb.polish.container.Container*), 364
options (*atribut calibre.customize.conversion.InputFormatPlugin*), 266
options (*atribut calibre.customize.conversion.OutputFormatPlugin*), 267
options (*atribut calibre.ebooks.metadata.sources.base.Source*), 262
OSX_MAIN_MEM_VOL_PAT (*atribut calibre.devices.usbms.device.Device*), 277
output_encoding (*atribut calibre.customize.conversion.InputFormatPlugin*), 265
OutputFormatPlugin (*třída v calibre.customize.conversion*), 266

P

parse_feeds() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 44
parse_index() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 44
parsed() (*metoda calibre.ebooks.oeb.polish.container.Container*), 364
path_sep (*atribut calibre.devices.interface.DevicePlugin*), 268
Plugin (*třída v calibre.customize*), 256
populate_article_metadata() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 44
popup_type (*atribut calibre.gui2.actions.InterfaceAction*), 282
post_yank_cleanup() (*metoda calibre.devices.interface.DevicePlugin*), 271
post_yank_cleanup() (*metoda calibre.devices.usbms.device.Device*), 279
postadd() (*metoda calibre.customize.FileTypePlugin*), 259
postconvert() (*metoda calibre.customize.FileTypePlugin*), 259
postdelete() (*metoda calibre.customize.FileTypePlugin*), 259
postimport() (*metoda calibre.customize.FileTypePlugin*), 259
postprocess_book() (*metoda calibre.customize.conversion.InputFormatPlugin*), 266
postprocess_book() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 45
postprocess_html() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 45
pref() (*metoda calibre.db.cache.Cache*), 358
prefer_results_with_isbn (*atribut calibre.ebooks.metadata.sources.base.Source*), 262
PreferencesPlugin (*třída v calibre.customize*), 285
prepare_addable_books() (*metoda calibre.devices.interface.DevicePlugin*), 273
preprocess_html() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 45
preprocess_image() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 45
preprocess_raw_html() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 45
preprocess_regexps (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 49
pretty_all() (*v modulu calibre.ebooks.oeb.polish.pretty*), 366
pretty_css() (*v modulu calibre.ebooks.oeb.polish.pretty*), 366
pretty_html() (*v modulu calibre.ebooks.oeb.polish.pretty*), 366
pretty_xml() (*v modulu calibre.ebooks.oeb.polish.pretty*), 366
print_version() (*třídní metoda calibre.web.feeds.news.BasicNewsRecipe*), 41
priority (*atribut calibre.customize.Plugin*), 257
priority (*atribut calibre.gui2.actions.InterfaceAction*), 282
PRODUCT_ID (*atribut calibre.devices.interface.DevicePlugin*), 268
PRODUCT_ID (*atribut calibre.devices.usbms.device.Device*), 277
publication_date() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 45
publication_type (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 49
předpis, **375**

R

raw_data() (*metoda calibre.ebooks.oeb.polish.container.Container*), 364
re:test(), **160**
read_backup() (*metoda calibre.db.cache.Cache*), 358
recipe_disabled (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 49
recipe_specific_options (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 49
recommendations (*atribut calibre.customize.conversion.InputFormatPlugin*), 266
recommendations (*atribut calibre.customize.conversion.OutputFormatPlugin*), 267

recursions (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 49
refresh_gui() (*metoda calibre.gui2.preferences.ConfigWidgetInterface*), 286
regexp, **375**
register() (*metoda calibre.gui2.preferences.ConfigWidgetBase*), 286
register_shortcut() (*metoda calibre.gui2.tweak_book.plugin.Tool*), 369
relpath() (*metoda calibre.ebooks.oeb.polish.container.Container*), 364
remove_attributes (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 49
remove_book() (*metoda calibre.devices.interface.BookList*), 276
remove_books() (*metoda calibre.db.cache.Cache*), 358
remove_books_from_metadata() (*metoda calibre.devices.usbms.driver.USBMS*), 281
remove_books_from_metadata() (*třídní metoda calibre.devices.interface.DevicePlugin*), 273
remove_empty_feeds (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 49
remove_extra_files() (*metoda calibre.db.cache.Cache*), 358
remove_formats() (*metoda calibre.db.cache.Cache*), 358
remove_from_spine() (*metoda calibre.ebooks.oeb.polish.container.Container*), 365
remove_from_xml() (*metoda calibre.ebooks.oeb.polish.container.Container*), 365
remove_item() (*metoda calibre.ebooks.oeb.polish.container.Container*), 365
remove_items() (*metoda calibre.db.cache.Cache*), 359
remove_jacket() (*v modulu calibre.ebooks.oeb.polish.jacket*), 366
remove_javascript (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 49
remove_stale_user_metadata() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 211
remove_tags (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 50
remove_tags_after (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 50
remove_tags_before (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 50
remove_unused_css() (*v modulu calibre.ebooks.oeb.polish.css*), 367
rename() (*metoda calibre.ebooks.oeb.polish.container.Container*), 365
rename_extra_files() (*metoda calibre.db.cache.Cache*), 359
rename_files() (*v modulu calibre.ebooks.oeb.polish.replace*), 366
rename_items() (*metoda calibre.db.cache.Cache*), 359
replace() (*metoda calibre.ebooks.oeb.polish.container.Container*), 365
replace_links() (*metoda calibre.ebooks.oeb.polish.container.Container*), 365
replace_links() (*v modulu calibre.ebooks.oeb.polish.replace*), 365
requires_version (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 50
reset() (*metoda calibre.devices.interface.DevicePlugin*), 270
reset() (*metoda calibre.devices.usbms.device.Device*), 277
resolve_internal_links (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 50
restart_critical (*atribut calibre.gui2.preferences.ConfigWidgetBase*), 286
restart_critical (*atribut calibre.gui2.preferences.ConfigWidgetInterface*), 286
restore_book() (*metoda calibre.db.cache.Cache*), 359
restore_defaults() (*metoda calibre.gui2.preferences.ConfigWidgetBase*), 287
restore_defaults() (*metoda calibre.gui2.preferences.ConfigWidgetInterface*), 286
restore_defaults_desc (*atribut calibre.gui2.preferences.ConfigWidgetInterface*), 286
restore_original_format() (*metoda calibre.db.cache.Cache*), 359
reverse_article_order (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 50
rewind_savepoint() (*metoda calibre.gui2.tweak_book.boss.Boss*), 371
RSS, **375**
run() (*metoda calibre.customize.CatalogPlugin*), 261
run() (*metoda calibre.customize.FileTypePlugin*), 259

S

safe_read_lock (*calibre.db.cache.Cache property*), 359
sanitize_callback() (*metoda calibre.devices.usbms.device.Device*), 279
sanitize_path_components() (*metoda calibre.devices.usbms.device.Device*), 279
save_book() (*metoda calibre.gui2.tweak_book.boss.Boss*), 371

save_original_format() (*metoda calibre.db.cache.Cache*), 359
save_settings() (*metoda calibre.customize.Plugin*), 257
save_settings() (*metoda calibre.ebooks.metadata.sources.base.Source*), 263
save_settings() (*třídní metoda calibre.devices.interface.DevicePlugin*), 273
scale_news_images (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 50
scale_news_images_to_device (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 50
search() (*metoda calibre.db.cache.Cache*), 359
search_annotations() (*metoda calibre.db.cache.Cache*), 360
search_notes() (*metoda calibre.db.cache.Cache*), 360
serialize_item() (*metoda calibre.ebooks.oeb.polish.container.Container*), 365
set_all_user_metadata() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 211
set_annotations_for_book() (*metoda calibre.db.cache.Cache*), 360
set_conversion_options() (*metoda calibre.db.cache.Cache*), 360
set_cover() (*metoda calibre.db.cache.Cache*), 360
set_cover() (*v modulu calibre.ebooks.oeb.polish.cover*), 367
set_driveinfo_name() (*metoda calibre.devices.interface.DevicePlugin*), 273
set_driveinfo_name() (*metoda calibre.devices.usbms.driver.USBMS*), 280
set_field() (*metoda calibre.db.cache.Cache*), 360
set_identifier() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 211
set_identifiers() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 211
set_library_info() (*metoda calibre.devices.interface.DevicePlugin*), 274
set_link_map() (*metoda calibre.db.cache.Cache*), 360
set_metadata() (*metoda calibre.customize.MetadataWriterPlugin*), 261
set_metadata() (*metoda calibre.db.cache.Cache*), 360
set_modified() (*metoda calibre.gui2.tweak_book.boss.Boss*), 371
set_notes_for() (*metoda calibre.db.cache.Cache*), 360
set_option() (*metoda calibre.devices.interface.DevicePlugin*), 275
set_plugboards() (*metoda calibre.devices.interface.DevicePlugin*), 273
set_pref() (*metoda calibre.db.cache.Cache*), 361
set_progress_reporter() (*metoda calibre.devices.interface.DevicePlugin*), 271
set_progress_reporter() (*metoda calibre.devices.usbms.device.Device*), 277
set_spine() (*metoda calibre.ebooks.oeb.polish.container.Container*), 365
set_user_blacklisted_devices() (*metoda calibre.devices.interface.DevicePlugin*), 274
set_user_metadata() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 211
settings() (*třídní metoda calibre.devices.interface.DevicePlugin*), 273
show_current_diff() (*metoda calibre.gui2.tweak_book.boss.Boss*), 371
show_editor() (*metoda calibre.gui2.tweak_book.boss.Boss*), 371
shutdown() (*metoda calibre.devices.interface.DevicePlugin*), 274
shutting_down() (*metoda calibre.gui2.actions.InterfaceAction*), 284
simultaneous_downloads (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 50
skip_ad_pages() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 45
SLOW_DRIVEINFO (*atribut calibre.devices.interface.DevicePlugin*), 269
smart_update() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 211
sort_index_by() (*metoda calibre.web.feeds.news.BasicNewsRecipe*), 46
Source (*třída v calibre.ebooks.metadata.sources.base*), 262
specialize() (*metoda calibre.customize.conversion.InputFormatPlugin*), 266
specialize_css_for_output() (*metoda calibre.customize.conversion.OutputFormatPlugin*), 267
specialize_global_preferences() (*metoda calibre.devices.interface.DevicePlugin*), 274
specialize_options() (*metoda calibre.customize.conversion.OutputFormatPlugin*), 267
spine_items (*calibre.ebooks.oeb.polish.container.Container property*), 365
spine_iter (*calibre.ebooks.oeb.polish.container.Container property*), 365
spine_names (*calibre.ebooks.oeb.polish.container.Container property*), 365
split() (*v modulu calibre.ebooks.oeb.polish.split*), 366
split_if_is_multiple_composite() (*metoda ca-*

libre.db.cache.Cache), 361
split_jobs() (*metoda calibre.ebooks.metadata.sources.base.Source*), 263
standard_field_keys() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 211
STANDARD_METADATA_FIELDS (*v modulu calibre.ebooks.metadata.book.base*), 212
start_plugin() (*metoda calibre.devices.interface.DevicePlugin*), 274
startup() (*metoda calibre.devices.interface.DevicePlugin*), 274
stop_plugin() (*metoda calibre.devices.interface.DevicePlugin*), 274
summary_length (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 51
supported_platforms (*atribut calibre.customize.conversion.InputFormatPlugin*), 265
supported_platforms (*atribut calibre.customize.conversion.OutputFormatPlugin*), 267
supported_platforms (*atribut calibre.customize.InterfaceActionBase*), 284
supported_platforms (*atribut calibre.customize.MetadataReaderPlugin*), 260
supported_platforms (*atribut calibre.customize.MetadataWriterPlugin*), 260
supported_platforms (*atribut calibre.customize.Plugin*), 256
supported_platforms (*atribut calibre.customize.PreferencesPlugin*), 285
supported_platforms (*atribut calibre.devices.usbms.driver.USBMS*), 279
supported_platforms (*atribut calibre.ebooks.metadata.sources.base.Source*), 262
supports_collections() (*metoda calibre.devices.interface.BookList*), 275
supports_gzip_transfer_encoding (*atribut calibre.ebooks.metadata.sources.base.Source*), 262
supports_restoring_to_defaults (*atribut calibre.gui2.preferences.ConfigWidgetBase*), 286
supports_restoring_to_defaults (*atribut calibre.gui2.preferences.ConfigWidgetInterface*), 286
sync_booklists() (*metoda calibre.devices.interface.DevicePlugin*), 273
sync_booklists() (*metoda calibre.devices.usbms.driver.USBMS*), 281
sync_preview_to_editor() (*metoda calibre.gui2.tweak_book.boss.Boss*), 371
synchronize_with_db() (*metoda calibre.devices.interface.DevicePlugin*), 275

T

tag_browser_context_action() (*metoda calibre.gui2.actions.InterfaceAction*), 284
tag_to_string() (*třídní metoda calibre.web.feeds.news.BasicNewsRecipe*), 41
tags_older_than() (*metoda calibre.db.cache.Cache*), 361
template_css (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 51
template_to_attribute() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 211
temporary_file() (*metoda calibre.customize.Plugin*), 258
test_fields() (*metoda calibre.ebooks.metadata.sources.base.Source*), 263
THUMBNAIL_COMPRESSION_QUALITY (*atribut calibre.devices.interface.DevicePlugin*), 268
THUMBNAIL_HEIGHT (*atribut calibre.devices.interface.DevicePlugin*), 268
timefmt (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 51
timeout (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 51
title (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 51
to_html() (*metoda calibre.ebooks.metadata.book.base.Metadata*), 212
Tool (*třída v calibre.gui2.tweak_book.plugin*), 368
toolbar_button_popup_mode (*atribut calibre.gui2.tweak_book.plugin.Tool*), 369
total_space() (*metoda calibre.devices.interface.DevicePlugin*), 271
total_space() (*metoda calibre.devices.usbms.device.Device*), 278
touched_fields (*atribut calibre.ebooks.metadata.sources.base.Source*), 262
type (*atribut calibre.customize.CatalogPlugin*), 261
type (*atribut calibre.customize.conversion.InputFormatPlugin*), 265
type (*atribut calibre.customize.conversion.OutputFormatPlugin*), 266
type (*atribut calibre.customize.FileTypePlugin*), 259
type (*atribut calibre.customize.InterfaceActionBase*), 284
type (*atribut calibre.customize.MetadataReaderPlugin*), 260
type (*atribut calibre.customize.MetadataWriterPlugin*), 261
type (*atribut calibre.customize.Plugin*), 257
type (*atribut calibre.customize.PreferencesPlugin*), 285
type (*atribut calibre.devices.interface.DevicePlugin*), 268
type (*atribut calibre.ebooks.metadata.sources.base.Source*), 262

U

unretire_note_for() (*metoda calibre.db.cache.Cache*), 361
update_annotations() (*metoda calibre.db.cache.Cache*), 361
upload_books() (*metoda calibre.devices.interface.DevicePlugin*), 272
upload_books() (*metoda calibre.devices.usbms.driver.USBMS*), 280
upload_cover() (*metoda calibre.devices.usbms.driver.USBMS*), 280
URL, **375**
USBMS (*třída v calibre.devices.usbms.driver*), 279
use_embedded_content (*atribut calibre.web.feeds.news.BasicNewsRecipe*), 51
user_categories_for_books() (*metoda calibre.db.cache.Cache*), 361
user_feedback_after_callback (*atribut calibre.devices.interface.DevicePlugin*), 269
UserAnnotation (*atribut calibre.devices.interface.DevicePlugin*), 269

V

VENDOR_ID (*atribut calibre.devices.interface.DevicePlugin*), 268
VENDOR_ID (*atribut calibre.devices.usbms.device.Device*), 276
version (*atribut calibre.customize.MetadataReaderPlugin*), 260
version (*atribut calibre.customize.MetadataWriterPlugin*), 260
version (*atribut calibre.customize.Plugin*), 256
VIRTUAL_BOOK_EXTENSION_MESSAGE (*atribut calibre.devices.interface.DevicePlugin*), 269
VIRTUAL_BOOK_EXTENSIONS (*atribut calibre.devices.interface.DevicePlugin*), 269

W

WANTS_UPDATED_THUMBNAILS (*atribut calibre.devices.interface.DevicePlugin*), 268
web2disk command line option
    --base-dir, 340
    --delay, 340
    --dont-download-stylesheets, 340
    --encoding, 340
    --filter-regexp, 340
    --help, 340
    --match-regexp, 340
    --max-files, 341
    --max-recursions, 341
    --timeout, 341
    --verbose, 341
    --version, 341
    -d, 340
    -h, 340
    -n, 341
    -r, 341
    -t, 341
WINDOWS_CARD_A_MEM (*atribut calibre.devices.usbms.device.Device*), 277
WINDOWS_CARD_B_MEM (*atribut calibre.devices.usbms.device.Device*), 277
WINDOWS_MAIN_MEM (*atribut calibre.devices.usbms.device.Device*), 277
windows_sort_drives() (*metoda calibre.devices.usbms.device.Device*), 278