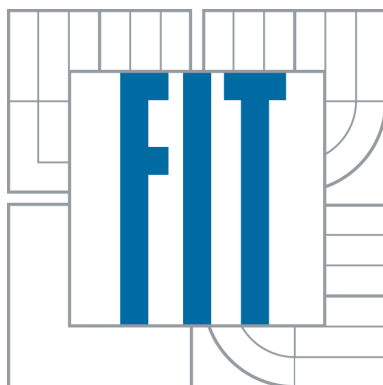


# FAKULTA INFORMAČNÍCH TECHNOLOGIÍ VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ



Dokumentace k projektu předmětu ISJ – Skriptovací jazyky  
Stahování dat z Twitteru a diskusního fóra

# Obsah

<b>1</b>	<b>Úvod</b>	<b>2</b>
<b>2</b>	<b>Twitter</b>	<b>2</b>
2.1	Načtení nových dat . . . . .	2
2.2	Zobrazení obsahu lokálně uložených dat . . . . .	2
2.3	Stažení URL adres . . . . .	2
2.3.1	Bez poskytnutí identifikátoru . . . . .	2
2.3.2	S identifikátorem . . . . .	3
2.4	Specifikovaný účet . . . . .	3
<b>3</b>	<b>Fórum</b>	<b>3</b>
3.1	Načtení dat . . . . .	3
3.2	Zobrazení dat . . . . .	3

# 1 Úvod

Dokumentace pojednává o skriptu napsaném v rámci předmětu Skriptovací jazyky.

Zadání bylo rozděleno na dvě části. Jednou z částí bylo napsat skript, který stáhne všechny dosavadní příspěvky diskuzního fóra a uloží je lokálně do souboru společně s relevantními metainformacemi. Skript by také měl být schopen rozpoznat nové příspěvky a zahájit jejich stahování. Druhá část zadání spočívala v úkolu napsat skript, jež bude sloužit ke sledování účtu na sociální síti Twitter. Příspěvky mají být dle zadání lokálně ukládány a mají být nalezeny případné odkazy na webové stránky a stažen jejich obsah.

Pro jednoduchost a fakt, že v obou částech řešení jsou využity totožné metody (práce s JSON soubory, zpracování parametrů, řešení chybových stavů), jsou obě části zadání implementovány v rámci jednoho souboru, s tím, že kontext pro práci s fórem a se sítí Twitter se přepíná pomocí parametru, jak je popsáno níže, nebo také v nápovědě skriptu, kterou lze vyvolat spuštěním s parametrem `-h` nebo `-help`.

## 2 Twitter

Rozhraní skriptu pro práci se sociální sítí Twitter je specifikováno pomocí parametru `twitter` uvedeným za přepínačem `-i`. Akce, která má být vykonána je specifikována parametrem `-a`.

### 2.1 Načtení nových dat

Je-li skript spuštěn s parametrem `load` za přepínačem `-a`, dojde ke stažení tzv. *tweetů* vytvořených sledovaným účtem.

Úvodním krokem je připojení se k API sítě Twitter a načtení zpráv uživatele. Pro umožnění připojení k API a úspěšné autentifikaci aplikace je použita knihovna třetí strany [oauth](#), jejíž použití pro autentifikaci je popsáno na webových stránkách Twitteru určených pro vývojáře. Tuto knihovnu je možné nainstalovat jako Ruby Gem příkazem `gem install oauth`

Po zaslání požadavku na [Twitter API](#) je skriptu poslána zpět odpověď ve formátu JSON, kterou následně skript parsuje a vyfiltruje z ní pouze požadované údaje (identifikátor tweetu, datum a čas vytvoření, text, URL adresy obsažené v textu). Tyto načtené zprávy jsou následně porovnány s případně již staženými, a jsou-li nalezeny novější, jsou přidány do souboru.

### 2.2 Zobrazení obsahu lokálně uložených dat

Při zadání parametru `show` za přepínačem `-a` skript zobrazí soubor s načtenými tweety. Zobrazeny jsou již výše zmíněné údaje:

- identifikátor tweetu
- datum a čas vytvoření
- text tweetu
- URL adresy obsažené v textu

O výpis souboru se stará externí knihovna [awesome\\_print](#), což je přehlednější modifikace standardní knihovny PrettyPrint. Knihovnu lze nainstalovat jako Gem příkazem `gem install awesome_print`.

### 2.3 Stažení URL adres

#### 2.3.1 Bez poskytnutí identifikátoru

Při spuštění skriptu s parametrem `download`, přičemž není dále specifikovaný identifikátor tweetu, se nejprve vykonají tytéž úkony, jako při spuštění s parametrem `load` (viz kapitola 2.1) a poté dojde ke stažení obsahu všech nalezených URL odkazů v nově načtených tweetech.

### 2.3.2 S identifikátorem

Pokud je s parametrem `download` zadáný i identifikátor pomocí přepínače `-s`, pokusí se skript vyhledat zprávu s tímto identifikátorem mezi tweety v lokálně uloženém souboru. Pokud případně nalezený tweet obsahuje alespoň jednu URL adresu, skript stáhne její obsah.

Stažené odkazy mají vždy jako součást prefixu identifikační číslo tweetu, z něhož byly načteny, díky čemuž je možné lokálně uložit obsah z více zdrojů při jediném načtení timeline uživatele.

## 2.4 Specifikovaný účet

Skript je rozšířen o možnost specifikování jiného, než defaultně zvoleného účtu, který je zapsán přímo ve zdrojovém kódu. Pokud je pomocí přepínač `-c` zadáný twitter účet, skript bude pracovat právě se specifikovaným účtem. Při načtení tweetu do lokálního souboru je součástí prefixu souboru také název sledovaného účtu, čímž je vyřešen problem s přepsáním načtených dat v lokálním souboru vyřešen.

## 3 Fórum

Stejně tak, jako je rozhraní pro práci se sítí Twitter specifikováno parametrem `twitter`, je rozhraní pro práci s diskuzním fórem specifikováno parametrem `forum` za přepínačem `-i`.

Nutným požadavkem při práci s fórem je zadání URL adresy požadovaného vlákna pomocí přepínače `-s` a operace, která se má provést přepínačem `-a`. Z adresy je pak regulárním výrazem vyfiltrováno identifikační číslo vlákna, které slouží k odlišení lokálně uloženého souboru, pro případ, že by uživatel měl zájem sledovat skriptem více vláken současně.

### 3.1 Načtení dat

Vybrané fórum [talk.manageiq.org](http://talk.manageiq.org) je provozováno aplikací [Discourse](https://discourse.org/), která nabízí v tuto chvíli ještě nepříliš propracované API, avšak dostačující ke získání potřebných dat z fóra. Ukázka práce s API byla demonstrována v [příspěvku](#) na diskuzním fóru aplikace.

Při spuštění skriptu s parametrem `load` dojde k načtení dat z API zadané URL adresy vlákna, z něhož jsou následně vyfiltrována požadovaná data příspěvku v daném vlákně:

- počet příspěvků ve vlákně
- identifikátor příspěvku
- autor příspěvku
- text příspěvku

Podle načteného počtu příspěvku ve vlákně se pak určí, zda byly od posledního načtení vlákna do lokálního souboru přidány nějaké nové příspěvky a zapíše se do již vytvořeného lokálního souboru, případně bude vytvořen nový soubor.

### 3.2 Zobrazení dat

Funkce parametru `show` je totožná jako při práci se sociální sítí Twitter. Lokálně uložená data jsou vypsány pomocí knihovny [awesome\\_print](#). Pokud lokální soubor pro dané vlákno ještě neexistuje, je na to uživatel upozorněn chybovým hlášením.