

BI-SKJ Programátorská dokumentace

Dominik Soukup

11. 4. 2015

Obsah

1	Úvod	3
2	Popis skriptu	3
2.1	Testování potřebných programů	3
2.2	Načtení vstupních parametrů	3
2.2.1	Parsování přepínačů na příkazové řádce	3
2.2.2	Parsování konfiguračního souboru	3
2.3	Kontrola souborů a adresáře	4
2.3.1	Vytvoření adresáře	4
2.3.2	Kontrola datových souborů	4
2.4	Sjednocení dat	4
2.4.1	Datové soubory	5
2.5	Time, Speed, Fps	5
2.6	Nastavení gnuplotu	6
2.6.1	Příkaz plot	6
2.6.2	Vlastnosti gnuplotu	6
2.7	Tvorba obrázků	6
2.8	Tvorba animace	7
2.9	Ukončení	7
2.10	Návratové kódy	7

1 Úvod

Hlavním cílem tohoto dokumentu je detailněji vysvětlit fungování skriptu po programátorské stránce. Dokument vysvětluje jednotlivé části skriptu, tudíž umožňuje pochopení jeho běhu.

2 Popis skriptu

2.1 Testování potřebných programů

Na začátku běhu skriptu je potřeba ověřit potřebné programy. Proto se ještě před zpracováním uživatelského vstupu provádí kontrola programů pro tvorbu animace (*gnuplot*, *wget*, *ffmpeg*). Pokud některý program chybí, tak se skript okamžitě ukončí.

2.2 Načtení vstupních parametrů

Ještě před samotným parsováním přepínačů se provádí kontrola opakování přepínačů. Opakovat se mohou pouze přepínače *--gnuplotparams*, *--effectparams* a *--criticalvalue*.

Pozn. přepínač *--criticalvalue* je na příkazové řádce přijímán a kontrolován, avšak jeho funkce na výslednou animaci implementovaná není.

2.2.1 Parsování přepínačů na příkazové řádce

Vstupní přepínače se parsují pomocí modulu *argparse*. Tento modul se stará o veškerou kontrolu vstupních přepínačů. Jediné hodnoty přepínačů, které se nekontrolují jsou *--xmin*, *--xmax* a *--timeformat*. Pro kontrolu hodnot přepínačů *--xmin* a *--xmax* je potřeba zajistit, aby se provedla až po zadání hodnoty *--timeformat*. Jelikož přepínače mohou být zadávány v libovolném pořadí je kontrola *--xmin* a *--xmax* prováděna později. Pro ověření přepínače *--timeformat* je potřeba znát časové hodnoty ze vstupního souboru. Z tohoto důvodu je kontrola přepínače prováděna až při načítání vstupních dat.

2.2.2 Parsování konfiguračního souboru

Pokud je zadán přepínač *-f* a tedy konfigurační soubor, tak se provede jeho zpracování. Z konfiguračního souboru se nejdříve odfiltrují nadbytečné části, což jsou komentáře a prázdné řádky. Po tomto zpracování zbydou pouze řádky, kde je klíč a hodnota. Tyto řádky se uloží do slovníku se vstupními parametry. Kontrola parametrů se zde zatím ještě neprovádí, protože načtené parametry mohou být přepsány přepínači z příkazové řádky.

Následně se do slovníku se vstupními parametry přidají načtené hodnoty přepínačů z příkazové řádky. Tím se zajistí, že příkazová řádka bude mít větší platnost než konfigurační soubor.

Nyní, když jsou všechny vstupní parametry zpracované, se nastaví implicitní hodnoty pro ty přepínače, které nebyly zadány vůbec. Také se mohou zpracovat vynechané přepínače `--xmin` a `--xmax` a následně se provede kontrola parametrů načtených z konfiguračního souboru.

2.3 Kontrola souborů a adresáře

Jestliže jsou všechny vstupní informace správně zpracované, tak se může přejít ke kontrole datových souborů a vytvoření výstupního adresáře.

2.3.1 Vytvoření adresáře

Jméno adresáře zjistíme podle názvu spuštěného skriptu nebo podle hodnoty přepínače `--name`. Pokud název obsahuje adresářovou cestu je provedeno rozdělení názvu na část cesty a koncový název adresáře. Následně se provádí kontrola existence jména adresáře v zadané cestě. Pokud již nějaký soubor se stejným jménem existuje, je jméno adresáře rozšířeno o číslo s hodnotou o jedna větší než je dosud pro existující soubor maximální. Po vytvoření jména adresáře se skript pokusí daný adresář vytvořit. Jestliže během vytváření došlo k jakékoliv chybě, tak se skript ukončí.

2.3.2 Kontrola datových souborů

Jako první se kontroluje, zda jsou soubory z internetu nebo ne. Pokud jméno začíná na `http`, tak se skript pokusí soubor stáhnout z internetu. Stažený soubor je uložen mezi dočasné soubory. Jestliže při stahování alespoň jednoho vstupního souboru dojde k chybě, tak je program ukončen. Ovšem s použitím přepínače `--ignoreerrors` se dá tato chyba potlačit, a jestliže existuje ještě nějaký další vstupní soubor, tak se pokračuje v jeho zpracování. Pokud žádný další soubor neexistuje je skript ukončen.

Jestliže již nejsou žádné další webové soubory nejsou k dispozici, tak se provádí kontrola na prázdnot dat. Pokud je zjištěno, že nějaký soubor je prázdný, tak je skript ukončen. Ovšem s přepínačem `--ignoreerrors` se tato chyba dá potlačit, a jestliže existuje ještě nějaký další vstupní soubor, tak se pokračuje v jeho zpracovávání.

2.4 Sjedení dat

Po ověření vstupních souborů je možné začít pracovat s daty. Postupně se prochází soubor po souboru a každý řádek je rozdělen na klíč (časová hodnota) a hodnotu (konkrétní hodnota). Jednotlivé hodnoty jsou pečlivě kontrolovány a provádí se potřebné výpočty. Podle zadaného rozsahu, dle přepínačů `--xmin` a `--xmax`, jsou nevyhovující řádky ignorovány (např. při zmenšení osy X), dále jsou zjišťovány maximální a minimální hodnoty pro osy X a Y, provádí se detekce překryvu dat a počítají se meze pro barevné vykreslování.

Nakonec se vytvoří slovník, kde klíčem jsou platné časové formáty a hodnotou je seznam, který prozatím obsahuje tolik hodnot X kolik je platných vstupních souborů. Slovník se snaží napodobit vhodnou strukturu pro zpracování *gnuplotem*, kde první sloupec odpovídá ose X a následně každý další sloupec odpovídá hodnotám na ose Y.

Jakmile jsou všechny soubory zpracovány, tak se může připravený slovník vyplnit platnými hodnotami. Opět se projdou všechny soubory a na pozice ve slovníku, určené klíčem a číslem souboru, jsou vkládány hodnoty. Tím se nám zajistí korektní seřazení dat podle času a snadno se vykreslují překrývající i navazující grafy.

2.4.1 Datové soubory

Po naplnění slovníku validními daty se slovník seřadí v sestupném i vzešupném pořadí. Různé pořadí je potřeba pro efekt směru vykreslování. Seřazené slovníky se už jen vypíší do dočasných souborů. Při vypisování do souboru je ještě potřeba pro každý sloupec s hodnotami osy Y přidat tři sloupce vypočítaných hodnot pro barevné vykreslování. Důvod přidání vypočítaných hodnot je následující: jestliže je detekován překryv dat mimo stejnou hodnotu osy X, tak při tomto formátu uložených dat (viz 2.4) a při přičítání hodnot pro barevné vykreslování až ve fázi *plotování* grafu *gnuplot* překrývající data nevykresluje jako spojitou křivku. Tudíž vybarvování křivek není správné. Proto pokud se data překrývají mimo stejnou hodnotu osy X, musíme příkazu *plot* předávat již dopředu vypočtené souřadnice a následně se křivky vykreslí spojitě. Při implementaci jsem se rozhodl pro dva styly příkazu *plot* (viz 2.6.1), protože pokud se data nepřekrývají nebo se překrývají na stejných hodnotách osy X a v souboru se objeví nečíselné hodnoty, tak mým úmyslem bylo, aby se na místě neplatných dat nevykreslovalo nic a pokračovalo se opět až na číselných hodnotách.

2.5 Time, Speed, Fps

Hodnoty přepínačů *--time*, *--speed* a *--fps* slouží k ovlivňování rychlosti animace. Uživatel může nastavit libovolnou kombinaci těchto hodnot a skript ověřuje jejich správnost a případně dopočítává nezadané hodnoty. Pro tyto hodnoty platí následující rovnost:

$$\frac{\text{počet_řádek}}{\text{speed}} = \text{fps} \cdot \text{time} \quad (1)$$

Pokud je kombinace hodnot vyhodnocena jako podezřelá, tzn. čas animace je buď větší než 300 sekund nebo menší než 5 sekund, je uživateli vypsáno upozornění s podezřelými hodnotami. Jestliže je daná kombinace nesmyslná je skript ukončen. Ovšem pomocí přepínače *--ignoreerrors* lze tuto chybu

potlačit a pokračovat v generování animace. Při potlačení se místo chybných hodnot dosadí defaultní hodnoty.

2.6 Nastavení gnuplotu

Nastavení *gnuplotu* je rozděleno do dvou částí:

- 1. příprava příkazu *plot*
- 2. nastavení vlastností *gnuplotu*

Výsledkem je příkaz, který lze přímo ze skriptu spustit.

2.6.1 Příkaz plot

Jak již bylo řečeno v 2.4.1 jsou vytvořeny dva *ploty*. První pro případ, když se data nepřekrývají nebo se překrývají na stejných hodnotách osy X. A druhý pro případ, kdy se data překrývají mimo stejné hodnoty na ose X. Kromě doby provádění výpočtu pro barevné vykreslování jsou oba *ploty* stejné. Příkaz *plot* kreslí vždy pro každý soubor čtyři křivky a plocha mezi nimi je vybarvena barvou, která je automaticky zvolena gnuplotem. Do nastavení plotu je ještě nastavena sytost barvy, aby při výsledné animaci byly snadno rozpoznatelné meze mezi soubory.

2.6.2 Vlastnosti gnuplotu

V této části je pouze nastavené globální nastavení pro vykreslování data. Pokud byl zadán přepínač *--gnuplotparams* je jeho hodnota přidána mezi tyto vlastnosti.

Následně se nastavený *plot* a nastavené vlastnosti *gnuplotu* spojí do jedné proměnné, která reprezentuje příkaz pro tvorbu obrázku

2.7 Tvorba obrázků

Tato část skriptu patří mezi výpočetně nejnáročnější. Nejdříve se určí podle hodnoty přepínače *--effectparams*, které soubory se použijí (soubory byly vytvořeny v 2.4). Možnosti jsou tři:

1. grafy se budou vykreslovat zleva doprava - zvolí se sestupně řazený soubor
2. grafy se budou vykreslovat současně zleva i zprava - zvolí se oba typy souborů a každá část grafu bude vykreslována jiným
3. grafy se budou vykreslovat zprava doleva - zvolí se vzestupně řazený soubor

Pro vytvoření jednoho obrázku animace se načte z vybraného souboru počet řádek podle hodnoty přepínače *--speed* a tyto řádky se uloží do pomocného dočasného souboru. Na pomocný soubor je zavolán příkaz *gnuplot* vytvořený v části 2.6 a výsledný obrázek je uložen do dočasného adresáře. Pro vytvoření dalšího obrázku se načte $(speed \times 2)$ řádek. Během vytváření snímků se na standartní výstup zobrazuje procentuální část zpracování souboru.

2.8 Tvorba animace

O tvorbu výsledné animace se stará program *ffmpeg*. Skript pouze doplní potřebné přepínače jako jsou: hodnota *--fps*, vstupní soubory a umístění animace. Následně se zavolá *ffmpeg* a animace je vytvořena.

2.9 Ukončení

Po úspěšném dokončení skriptu nebo při nějaké chybě v době zpracování je volána uzavírací funkce, která se stará o úklid vytvořených souborů a vrácení příslušného návratového kódu. Význam návratových kódů je popsán v 2.10.

2.10 Návratové kódy

Význam chybových kódů:

1. podezřelá hodnota přepínačů *--time*, *--speed*, *--fps*
2. ukončení po přijetí signálu, chyba při zpracování přepínačů *argparse-rem*
3. chyba souboru (práva, stažení, vytvoření)
4. neplatný rozsah nějaké osy
5. nesprávný *timeformat*
6. neplatný argument
7. chyba konfiguračního souboru
8. chybí potřebný software
9. ostatní