



TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií ■

Editor konfiguračních souborů Flow123d

Diplomová práce

Studijní program: N2612 – Elektrotechnika a informatika

Studijní obor: 1802T007 – Informační technologie

Autor práce: **Bc. Tomáš Křížek**

Vedoucí práce: doc. Ing. Jiřina Královcová, Ph.D.





TECHNICAL UNIVERSITY OF LIBEREC
Faculty of Mechatronics, Informatics
and Interdisciplinary Studies ■

Editor for Flow123d configuration files

Master thesis

Study programme: N2612 – Electrotechnology and informatics

Study branch: 1802T007 – Information technology

Author: **Bc. Tomáš Křížek**

Supervisor: doc. Ing. Jiřina Královcová, Ph.D.



Tento list nahrad'te
originálem zadání.

Prohlášení

Byl jsem seznámen s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu TUL.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Diplomovou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím mé diplomové práce a konzultantem.

Současně čestně prohlašuji, že tištěná verze práce se shoduje s elektronickou verzí, vloženou do IS STAG.

Datum:

Podpis:

Abstrakt

Český abstrakt

Abstract

English abstract

Poděkování

Obsah

Seznam zkratek	8
1 Rešerše	11
1.1 Datové struktury pro popis dat	11
1.2 Formát dat	11
1.3 Jazyky pro popis dat	11
2 Analýza	12
2.1 Konfigurační soubory	12
2.2 Input Structure Tree	12
2.3 Autokonverze	12
3 Návrh	13
4 Implementace	14

Seznam zkratk

EU Evropská unie

Úvod

Existuje celá řada softwarů, která ke své činnosti potřebuje správné nastavení, podle kterého daný program nakonfiguruje svoji činnost. Může se jednat o počáteční, jako je tomu například u serverových aplikací, e-mailových klientů. Tato konfigurace se dále mění pouze v případě, že se má změnit činnost aplikace.

Oproti tomu existují programy, od kterých se očekává, že budou spouštěny v různých konfiguracích. U těchto aplikací se typicky konfigurace předává při spuštění jako jeden ze vstupních parametrů. Takové programy pak přizpůsobují svoji činnost dle zadaného nastavení. Příkladem je například validátor XML, který určí, zda dokument odpovídá danému DTD či XML schématu.

Takovou aplikací je i simulátor Flow123d, který pracuje se zadanou výpočetní sítí, na které provádí výpočty dle definované úlohy. Úloha se definuje pomocí konfiguračního souboru, která pak spolu s výpočetní sítí a ostatními daty tvoří vstupní data pro aplikaci Flow123d.

Kromě toho, že je nutné každou úlohu konfigurovat zvlášť, vyvstává i otázka toho, zda je vůbec zvolená konfigurace validní. Chyba může nastat hned na několika úrovních. V nejběžnějším případě může jít o překlep. V konfiguraci však může být i logická chyba, kdy jsou sice splněny všechny syntaktické požadavky, ale kombinace některých hodnot je nepřípustná.

Náročnost odhalení těchto chyb závisí na složitosti možného nastavení. V případě Flow123d má samotná referenční příručka, která popisuje formát konfiguračních souborů, několik desítek stran. Pro uživatele tedy není triviální rozhodnout, zda je jím nadefinovaná úloha správně zadaná, nebo jestli např. zapomněl vyplnit nějaký povinný atribut. Na uživatele jsou tedy kladeny velké nároky – buď musí mít se softwarem rozsáhlé zkušenosti, nebo musí trávit čas hledáním a pročítáním dokumentace.

Celá situace je dále komplikována tím, že formát konfiguračních souborů se mění s tím, jak se vyvíjí funkce softwaru Flow123d. Některé změny bohužel nejsou ani zpětně kompatibilní. Tím pádem soubor, který byl validní pro starší verzi už ne-

musí být validní pro nově vydanou verzi. Uživatele tedy opět čeká studium rozsáhlé referenční dokumentace, aby zjistil, jaké změny má provést.

V případě, že uživatel spustí Flow123d, tak se o chybě dozví. Jelikož se ale jedná o výpočetně náročné úlohy, které se často pouští na vzdáleném výpočetním clusteru, je tento proces poměrně zdlouhavý. Pokud se úloha spouští vzdáleně, musí dojít k navázání komunikace a přidělení zdrojů, než může být vůbec úloha spuštěna. Odhalování takovýchto chyb je pak časově náročné a uživatelsky nepříjemné.

Tyto důvody byly hlavní motivací ke vzniku speciálního editoru, který značně zjednoduší práci s konfiguračními soubory pro Flow123d. Aplikace by měla poskytovat přívětivé rozhraní, které uživateli zjednoduší přístup k dokumentaci. Místo hledání příslušné sekce v několika stránkové referenční dokumentaci by měl mít uživatel k dispozici tu část dokumentace, která bezprostředně souvisí s tou částí konfiguračního souboru, kterou upravuje.

Editor by měl dále usnadnit a zrychlit proces odstranění chyb v konfiguračních souborech. Vzhledem k tomu, že aplikace Flow123d může být časově náročná na spuštění, obzvlášť pokud je spouštěna vzdáleně, by bylo ideální chybám předcházet a odhalit je ještě před tím, než je konfigurační soubor předložen aplikaci Flow123d. Editor by měl umožňovat ověření, zda v konfiguračním souboru nejsou chyby, které jsou buď syntaktické, nebo v rozporu s formátem, který Flow123d očekává.

Dále by editor měl umožnit vizualizaci datové struktury, která tvoří konfigurační soubor. Kromě toho se od editoru očekávají základní funkce pro práci s textovými soubory, jako je podpora operací se schránkou, možnost vrátit či opakovat změny, vyhledávání či nahrazení textu a další. Editor by měl podporovat platformy Windows a Linux.

1 Rešerše

1.1 Datové struktury pro popis dat

Obecný popis struktur - pole, záznam, hodnota.

Vnořování dat - stromové struktury.

1.2 Formát dat

Semantika - přidání významu datům.

Specifické pro aplikační doménu.

1.3 Jazyky pro popis dat

XML, DTD, XMLSchema JSON, JSON Schema YAML

2 Analýza

2.1 Konfigurační soubory

Přiblížení konfiguračních souborů a jejich formátu.

Starý formát CON a přechod na nový formát YAML.

Diskuze XML/YAML(JSON).

2.2 Input Structure Tree

Popis formátu datových souborů.

Definice konkrétních datových typů.

Specifikovat možné atributy.

2.3 Autokonverze

Popsat možné konverze, uvést do souvislosti s XML transformacemi?

3 Návrh

Diagramy pro zpracování YAML (inspirace SAX vs DOM).

4 Implementace

Závěr

Test zaver v zaveru v zaveru v zaveru v zaveru v zaveru v zaveru v zaveru v zav
v zaveru v zaveru