

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE  
FAKULTA STAVEBNÍ



# DIPLOMOVÁ PRÁCE

Workflow Builder

Vypracoval: Zdeněk Růžička

Vedoucí práce: Ing. Martin Landa, Ph.D.

Rok: Praha, 2012

## Prohlášení

Prohlašuji, že jsem svou diplomovou práci na téma **Workflow Builder** vypracoval samostatně s pomocí svého vedoucího práce a za použití literatury a zdrojů uvedených v příloženém seznamu na konci práce.

V Praze dne \_\_\_\_\_

\_\_\_\_\_  
podpis

## Poděkování

Především děkuji vedoucímu mé diplomové práce Ing. Martinu Landovi, Ph.D. za odborné vedení, rychlé reakce na mé dotazy a ochotu hledat na ně odpovědi. Dále bych chtěl poděkovat Camilo Polimeris za napsání QGIS Processing Frameworku jehož je tato práce součástí. V neposlední řadě bych chtěl poděkovat rodině a kamarádům za důvěru a podporu během studií.

## Abstrakt

Diplomová práce si vytyčila za cíl vytvořit v prostředí Quantum GIS ( QGIS ) nástroj, který by umožňoval uživateli grafické propojování modulů z frameworku **QGIS Processing Framework**. V úvodních dvou kapitolách je představeno prostředí QGIS a popsána práce s QGIS Processing Frameworkem jak z pozice uživatele GISu, tak z pozice vývojáře nových modulů. Dále je zmíněn SAGA Plugin, který byl napsán v rámci onoho frameworku.

V dalších částech diplomové práce představím z rychlíku knihovnu Qt, resp. její verzi PyQt pro jazyk Python, ve kterém byl celý Workflow Builder napsán.

V poslední kapitole diplomové práce je představena samotná aplikace umožňující grafické propojování modulů z QGIS Processing Framework Manageru do grafu, jejich postupné spuštění a případné uložení jako nový modul.

### Klíčová slova

Quantum QGIS, workflow, open source, GIS, PyQt, SAGA, QGIS Processing Framework

## Abstract

### Key words

Quantum QGIS, workflow, open source, GIS, PyQt, SAGA, QGIS Processing Framework

# Obsah

<b>Úvod</b>	<b>1</b>
<b>1 Teorie</b>	<b>3</b>
1.1 Quantum GIS . . . . .	3
1.1.1 Správa pluginů . . . . .	4
1.1.2 Psaní vlastního pluginu . . . . .	5
1.1.3 Python plugin . . . . .	7
1.1.4 C++ plugin . . . . .	7
1.2 QGIS Processing Framework . . . . .	8
1.2.1 Psaní pluginu . . . . .	8
1.3 Python . . . . .	9
1.4 Qt . . . . .	10
1.4.1 Model/View Programming . . . . .	10
1.4.2 Graphics View Interface . . . . .	10
1.4.3 VisTrails, Orange . . . . .	10
1.5 .xml.dom . . . . .	11
<b>2 Workflow Builder</b>	<b>12</b>
2.1 Tvorba workflow . . . . .	12
2.2 Uložení workflow . . . . .	12
2.2.1 Popsání výstupního xml souboru . . . . .	12
2.3 Načtení workflow do PF Manageru . . . . .	12

<b>3</b>	<b>SEXTANTE</b>	<b>13</b>
3.1	Srovnání QGIS Processing Framework v SEXTANTE . . . . .	13
3.2	Srovnání Workflow Builder v SEXTANTE Modeler . . . . .	13
	<b>Rejstřík</b>	<b>I</b>
	<b>Literatura</b>	<b>III</b>

# Úvod

V dnešní době se můžeme setkat s geoinformačními ( GIS ) technologiemi na každém kroku. V různých oblastech krajinného inženýrství, při plánování výstavby silnic, v územním plánování, při řešení krizových situací či plánování záchranných akcí. Uživatel si může vybrat z nepřeberného množství již existujících GIS nástrojů, řešení. A je pěkné, že svobodná řešení, nejen v oblasti geoinformačních technologií, drží krok s těmi proprietárními. Uživatel tedy nemusí sahát hluboko do kapsy. Co se týče nástrojů pro prohlížení, zpracování a analýzu geodat, můžeme jmenovat například GRASS GIS, gvSIG, Quantum GIS či SAGA GIS. Tato práce si ale nekladla za cíl srovnat GIS nástroje, ale implementací nástroje do programu Quantum GIS, který by uživateli umožňoval vytvářet vlastní funkce spojováním již existujících funkcí.

Můžeme se také setkat s pojmy jako *model builder* či *chaining*, v té práci budu používat pojem *workflow builder*. Takzvané *workflow buildery* dávají uživateli možnost vytvářet si vlastní moduly za pomoci spojování výstupů a vstupů modulů již existujících. Uživatel tak nemusí spouštět každý modul zvlášť a starat se o výstupy, nová data, která se vytvoří jen dočasně a která uživatel v konečném výsledku nepotřebuje. Dále je pro uživatele je mnohem pohodlnější, pakliže může najít všechny funkce na jednom místě (tzv. *toolbox*), nežli při hledání procházet všechny možné pluginy.

V době psaní této diplomové práce existoval projekt QGIS Processing Framework studenta Camilo Polymeris z univerzity Universidad de Concepción. QGIS Processing Framework si kladl za cíl být frameworkem, který by sdružoval moduly z pluginů pro QGIS na jednom místě (tzv. *toolbox*). Odtud by byly jednotlivé moduly volány, pomocí *workflow builderu* spojovány, ukládány atp. V rámci tohoto projektu začala vznikat podpora pro použití modulů z jiného GIS nástroje - System for Automated

---

Geoscientific Analysis ( SAGA GIS ). V době psaní existovala podpora pro 170 modulů, ne všechny ale byly testovány a fungovaly správně. Ale i přesto se mohlo začít s prací na workflow builderu.

Aktuální verzi workflow builderu můžete najít zde:

<https://github.com/CzendaZdenda/qgis>



# Kapitola 1

## Teorie

V této kapitole představím jeden ze svobodných systémů pro práci s geografickými daty - Quantum GIS a možnost rozšiřování funkcionality pomocí zásuvných modulů, tzv. pluginů. V druhé části popíši projekt QGIS Processing Framework, co bylo jeho cílem a také možnosti jeho rozšiřování. Dále se stručně charakterizuji jazyka Python, ve kterém vznikal Workflow Builder. V další části se budu věnovat knihovně Qt, respektive její verze pro Python - PyQt, kde popíši nástroje, které jsem využil při psaní Workflow Builderu. Mezi tyto nástroje patří Model/View Programming a Graphics View Framework. V poslední části představím modul xml.dom jazyka Python pro práci s objekty ve formátu XML formátem.

### 1.1 Quantum GIS



Quantum GIS je nejen prohlížečka geografických dat dostupná pro MS Windows, GNU/Linux, Unix, BSD a Mac OS X. Quantum GIS podporuje díky knihovně OGR většinu vektorových formátů dat jako například ESRI Shapefile, GRASS, MapInfo či GML a díky knihovně GDAL mnoho rastrových formátů jako TIFF, ArcInfo, GRASS

raster, ERDAS a další. Přes Quantum GIS můžeme také přistupovat k datům uložených v geodatabázích PostGIS a SpatiaLite či k datům dostupných přes WMS a WFS služby. Quantum GIS je šířen pod licencí GNU Public Licence.<sup>1</sup>

Program je napsán v jazyce C++. Poslední stabilní verze nese označení 1.7.4. Quantum GIS je lehce rozšířitelný program pomocí pluginů, které mohou být psány na jazyce Python nebo C++. Quantum GIS má poměrně dobře zdokumentované API a nutno také podotknout, že komunita kolem Quantum GISu je aktivní a podpora skrze mailinglisty je na vysoké úrovni.

Systém začal vyvíjet v roce 2002 Gary Sherman. Mělo jít o nenáročnou prohlížečku geodat pro Linux s širokou podporou datových formátů. Dlouhou dobu byl Quantum GIS brán převážně jako grafická nadstavba pro jiný desktopový GIS - GRASS GIS. Přes GRASS Plugin QGIS zpřístupňuje řadu modulů GRASS GIS. V současnosti se na vývoji nejvíce podílí skupina vývojářů kolem organizace<sup>2</sup> Faunalia.

Funkcionalitu Quantum GIS rozšiřuje množství pluginů. Jako základní pluginy bych označil<sup>3</sup> **fTools**, který umožňuje pokročilé prostorové analýzy nad vektorovými daty,<sup>4</sup> **GdalTools** pro práci s rastrovými daty a již zmíněný<sup>5</sup> **GRASS Plugin** plugin, který zpřístupňuje funkce GRASSu uživatelům Quantum GIS.

### 1.1.1 Správa pluginů

Jak už bylo zmíněno Quantum GIS umožňuje uživatelům rozšiřovat funkce programu dle jejich potřeb v podobě zásuvných modulů. Díky dobře zdokumentovanému API může uživatel pohodlně psát pluginy v jazyce C++ nebo Python. Pluginy píše jak vývojáři Quantum GISu, tak i obyčejní uživatelé. Pluginy si můžete stáhnout z oficiálních či neoficiálních repositářů. Pro instalování pluginů napsaných v jazyce Python a správu repositářů slouží nástroj **QGIS Python Plugin Installer**, dostupný přes *Plugins → Fetch Python Plugins...*

---

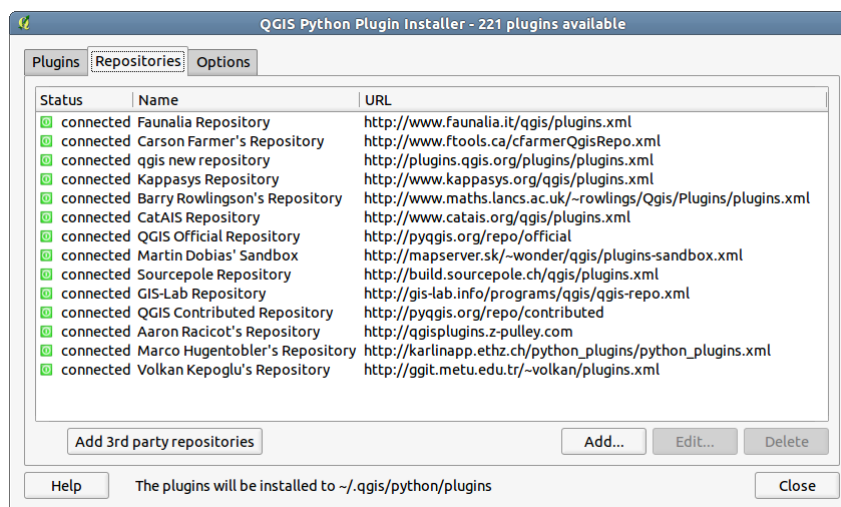
<sup>1</sup><http://qgis.org/about-qgis/features.html>

<sup>2</sup><http://www.faunalia.co.uk/quantumgis>

<sup>3</sup><http://www.ftools.ca/>

<sup>4</sup><http://www.faunalia.co.uk/gdaltools>

<sup>5</sup>[http://grass.osgeo.org/wiki/GRASS\\_and\\_QGIS](http://grass.osgeo.org/wiki/GRASS_and_QGIS)



Obrázek 1.1: QGIS Python Plugin Installer - správa repozitářů

Jak je vidět z [Obr. 1.1], takto nainstalované pluginy se stáhnou do adresáře:

- `$HOME/.qgis/python/plugins` - v případě OS GNU/Linux
- `C:\Documents and Settings\USER\.qgis/python/plugins` - v případě OS Windows bývá cesta podobná této

Pakliže uživatel napíše plugin v jazyce Python, doporučuji ho uložit do tohoto adresáře.

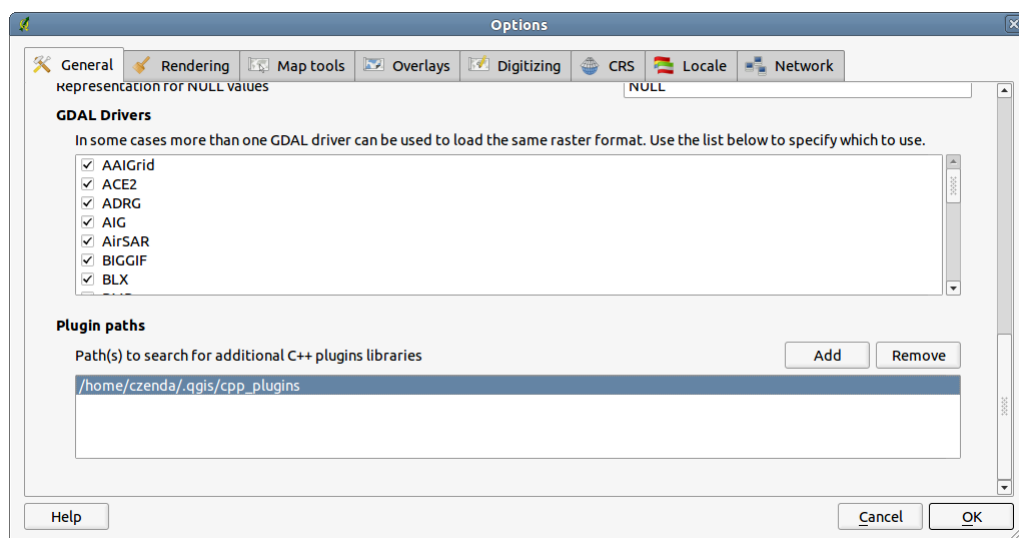
Je také sice možnost uložit plugin do adresáře `$QGIS_INSTALL_DIR\share\qgis/python/plugins`, ale při případné opětovné kompilaci by byly změny ztraceny.

Pluginy psané v C++ se po přeložení ukládají standardně v `$QGIS_INSTALL_DIR\lib\qgis/plugins` případně uživatel může přidat nová úložiště pomocí *Settings* → *Options* a v záložce *Generals* zadat cestu [Obr. 1.2].

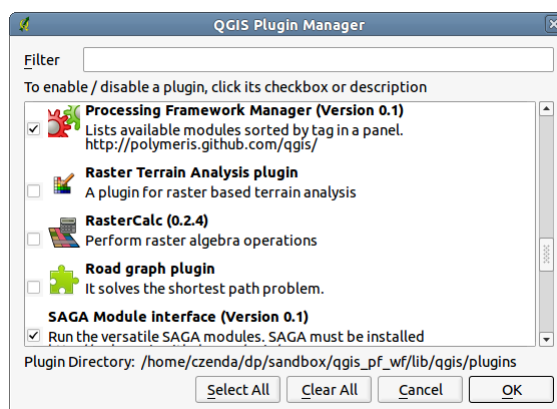
Všechny nainstalované pluginy, ať psané v jazyce C++ či Python, potom může uživatel spravovat přes **QGIS Plugin Manager** - *Plugins* → *Manage Plugins...* [Obrázek 1.3.

### 1.1.2 Psaní vlastního pluginu

Pluginy mohou být psány na jazyce C++ a Python. Již z charakteristiky daných jazyků vyplývá, že pro jednoduché, nenáročné či na začátku vývoje pluginy, se bude hodit spíše jazyk Python, který se nemusí kompilovat a píše se v něm rychleji než v



Obrázek 1.2: *Settings*→*Options*→*Generals* - přidání nové cesty k pluginům psaných na jazyce C++



Obrázek 1.3: *Plugins*→*Manage Plugins...* - správa pluginů

jazyce C++. Pro rozsáhlejší projekty je lepší sáhnout po jazyce C++. Obecně jsou programy psané v kompilovaných jazycích mnohem rychlejší než programy psané v jazycích interpretovaných.

### 1.1.3 Python plugin

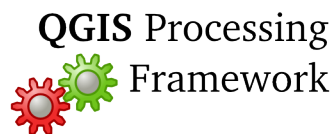
Při psaní pluginu v jazyce Python využíváme nástroje PyQGIS. Kromě dokumentace k Quantum GIS API také doporučuji [2].

```
1 def name():
2     return "My testing plugin"
3
4 def description():
5     return "Popis pluginu."
6
7 def version():
8     return "Version 0.1"
9
10 def qgisMinimumVersion():
11     return "1.0"
12
13 def authorName():
14     return "Developer"
15
16 def classFactory(iface):
17     # load TestPlugin class from file testplugin.py
18     from plugin import TestPlugin
19     return TestPlugin(iface)
```

Ukázka kódu 1.1: `--init--`.py - inicializační soubor

### 1.1.4 C++ plugin

## 1.2 QGIS Processing Framework



QGIS Processing Framework vznikl v rámci projektu <sup>6</sup>GSoC 2011 . Student Camilo Polymeris z univerzity Universidad de Concepción si kladl za cíl napsat framework, který ...

### 1.2.1 Psaní pluginu

---

<sup>6</sup>Google Summer of Code. Projekt společnosti Google na podporu studentu. více na <http://code.google.com/soc/>

## **1.3 Python**

## **1.4 Qt**

### **1.4.1 Model/View Programming**

### **1.4.2 Graphics View Interface**

### **1.4.3 VisTrails, Orange**



## **1.5 .xml.dom**

## Kapitola 2

# Workflow Builder

### 2.1 Tvorba workflow

### 2.2 Uložení workflow

#### 2.2.1 Popsání výstupního xml souboru

### 2.3 Načtení workflow do PF Manageru

## **Kapitola 3**

# **SEXTANTE**

### **3.1 Srovnání QGIS Processing Framework v SEXTANTE**

### **3.2 Srovnání Workflow Builder v SEXTANTE Modeler**

# Závěr

Dále musím zmínit existenci druhého frameworku, který se objevil v konci psaní této práce. SEXTANT pro Quantum GIS. Daný framework má v podstatě podobné cíle a v současné době se zdá být on tou pravou cestou pro qgis, ikdyž ne všechny moduly jsou momentálně plně funkční.

# Ukázky kódu

1.1	<code>__init__.py</code> - inicializační soubor . . . . .	7
-----	---	---

# Rejstřík

Faunalia, 4

fTools, 4

GDAL, 3

GDAL, GdalTools, 4

geodata, 1

GIS, 1

GRASS Plugin, 4

OGR, 3

PyQGIS, 7

QGIS Processing Framework, 1

QGIS,Quantum GIS, 3

SAGA GIS, 1

SEXTANT, 10

# Bibliography

- [1] *Official homepage of Quantum GIS*. URL: <http://www.qgis.org>.
- [2] *PyQGIS Developer Cookbook*. 2012. URL: <http://www.qgis.org/pyqgis-cookbook/index.html>.
- [3] *Quantum GIS API Documentation*. URL: <http://qgis.org/api/>.