Building complete free and open source GIS infrastructure for hydrological computing and data publication using GIS.lab and Gisquick platforms

M. Landa[@*] P. Kavka[*] L. Strouhal[*] J. Cepicky[**]

[@]martin.landa@fsv.cvut.cz, [*]Faculty of Civil Engineering, CTU in Prague, [**]OpenGeoLabs s.r.o., Czech Republic

Introduction

Building a complete open source GIS infrastructure allowing operations from data preparation, analysis and computation to publishing results to end-user is a very complex task. This poster presents GIS.lab as an open source software solution which helps to build complete GIS infrastructure in an easy, but still fully customized manner.

GIS.lab as a Core Component

Figure 1: GIS.lab logo (source: GIS.lab Documentation)

GIS.lab (`http://web.gislab.io`) has been originally designed with a goal to enable simple, unbreakable deployment of a complete, centrally managed, horizontally scalable GIS infrastructure in the local network area (LAN), data center or cloud in a few steps. GIS.lab is able to turn diverse open source GIS software packages into a seamlessly integrated easy-to-use system. As a result GIS.lab significantly decreases deployment of such complex GIS infrastructure to absolute minimum, but still keeping the whole technology under a full control of the system operator.

Gisquick as a Publication Platform

Gisquick (`http://gisquick.org`) is a separate project not directly related to GIS.lab. It is a web application based on modern technologies as Django, Angular and OpenLayers 3 with fully responsive design optimized also for mobile devices. The main purpose of Gisquick is to provide the capability for easy publishing the QGIS projects on the web.

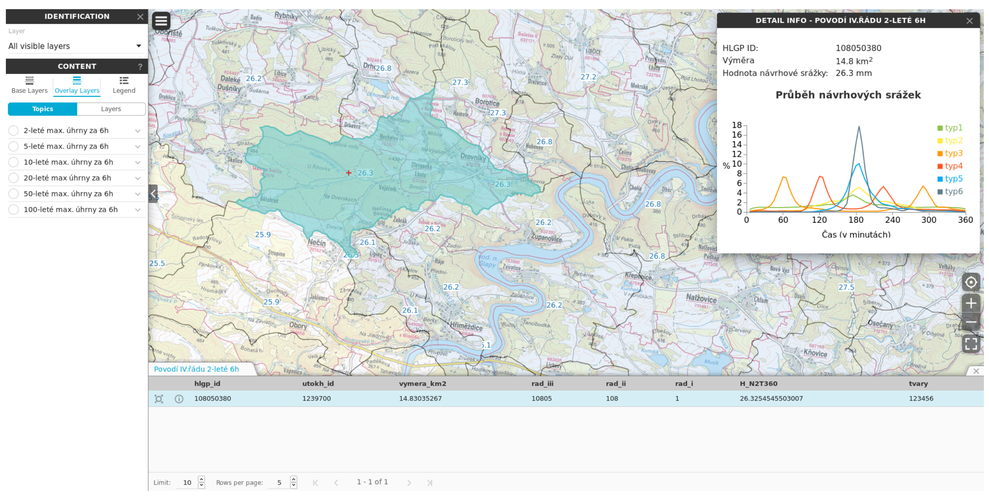

Figure 2: Gisquick web application interface (source: author)

Case study

Desired system should allow the user to collect, prepare, and preprocess data from heterogeneous data sources for hydrological computation using GIS software packages. The results of hydrological data processing can be easily published via web services from user desktop environment and ideally also as the interactive web mapping applications provided by the server component (master node) in the infrastructure.

Most of the requirements are already fulfilled by GIS.lab platform. Missing features can be easily integrated into GIS.lab eco-system by customizing the deployment of the both master/server and desktop client components.

The procedure consists of **three major steps**:

1. Deployment of a Master Node

Master node (playing a role of a server) of geospatial cluster can be automatically deployed thanks to GIS.lab technology.

2. Master Node Customization

Customization rules are defined by *Ansible Playbooks* similarly how GIS.lab provision works. Example below demonstrates how can be implemented a role for installing and configuring PyPWS4.

```
- name: Install PyWPS4
  pip:
    name: pywps
    version: 4.0.0

- name: Set up PyWPS4 configuration
  template:
    src: pywps.cfg.j2
    dest: /opt/pywps4/pywps.cfg
    mode: 0644
```

3. Gisquick integration

Gisquick project provides core *Docker images* for successful running of this publishing platform. It significantly simplifies Gisquick integration into GIS.lab infrastructure. Docker images can be automatically composed on GIS.lab master node by *docker-compose* command performed by customized provision roles.

Conclusions

Combining GIS.lab and Gisquick technologies leads to a complete, seamlessly integrated platform capable to prepare the input data, perform geospatial analysis as services and publish results easily on the web in the sense of interactive web mapping applications.

Acknowledgements This work has been supported by the research project QJ1520265 - "Variability of Short-term Precipitation and Runoff in Small Czech Drainage Basins and its Influence on Water Resources Management"