

Task 5: Dynamic ZABBIX maps

A ZABBIX installation should be used to dynamically adapt and change ZABBIX maps with a world map as background to add and delete telescope positions.

Single tasks:

- Install Ubuntu Linux as Windows Wubi installation
- Install ZABBIX on Linux
- Understand ZABBIX
- Understand the ZABBIX API
- Write a program or script to create a new map and to add or delete a telescope position

The project has been built mainly based on the Zabbix as backend and Grafana to provide the front-end of the maps.

The concept is similar to NASA approach which is shown here (Page 14, http://www.jive.eu/jumpingjive/lib/exe/fetch.php?media=wiki:jumping_jive_d8.4.pdf)

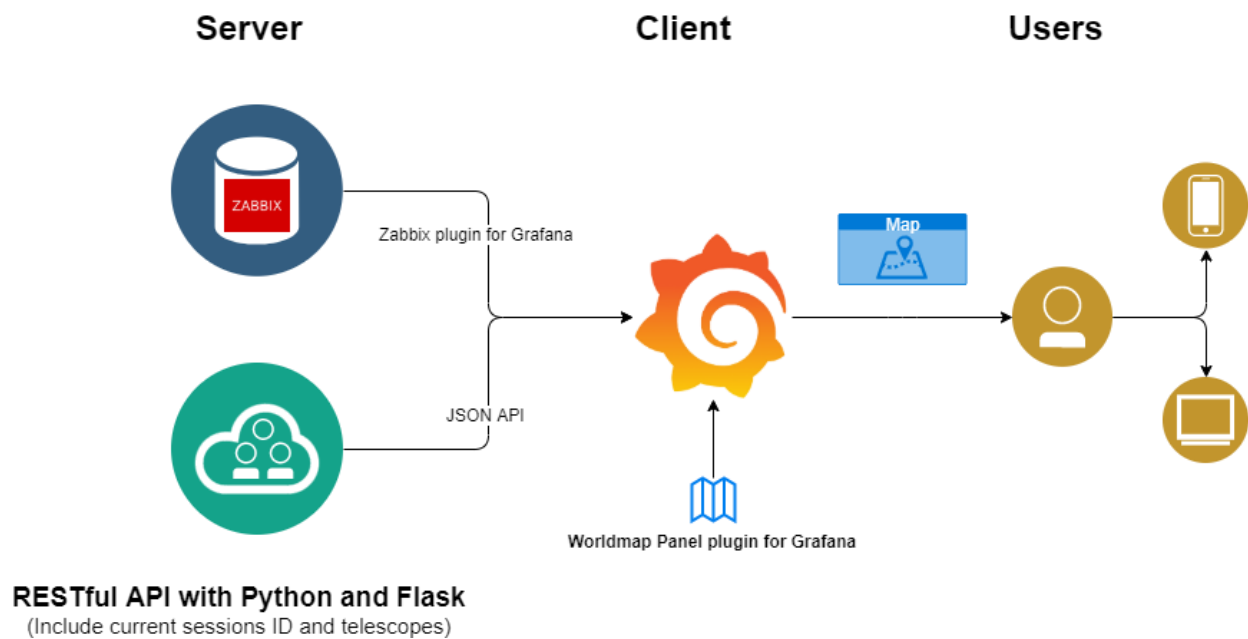


Figure 1: Workflow chart

Docker

To provide a better solution for deployment and easier for creation containers were used based on Docker to create different apps for each application. Then Docker-Compose is used for defining and running multi-container Docker applications, Grafana, RESTful API Server, Zabbix, Database.

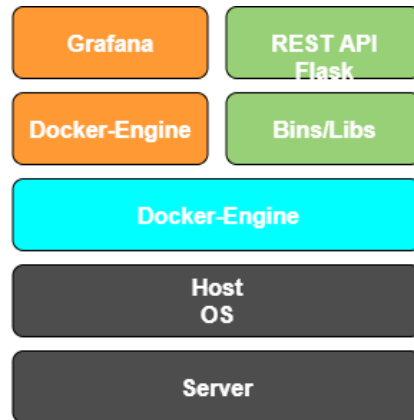


Figure 2: Docker Apps

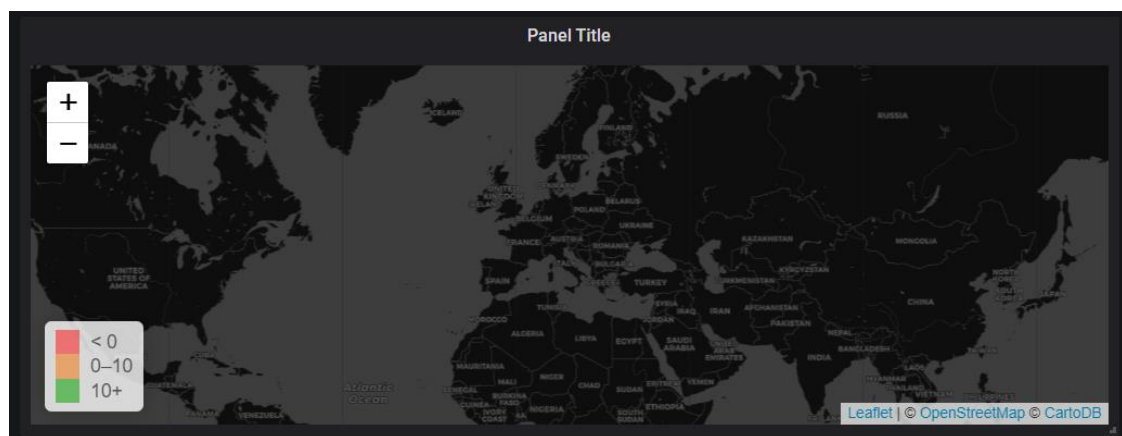
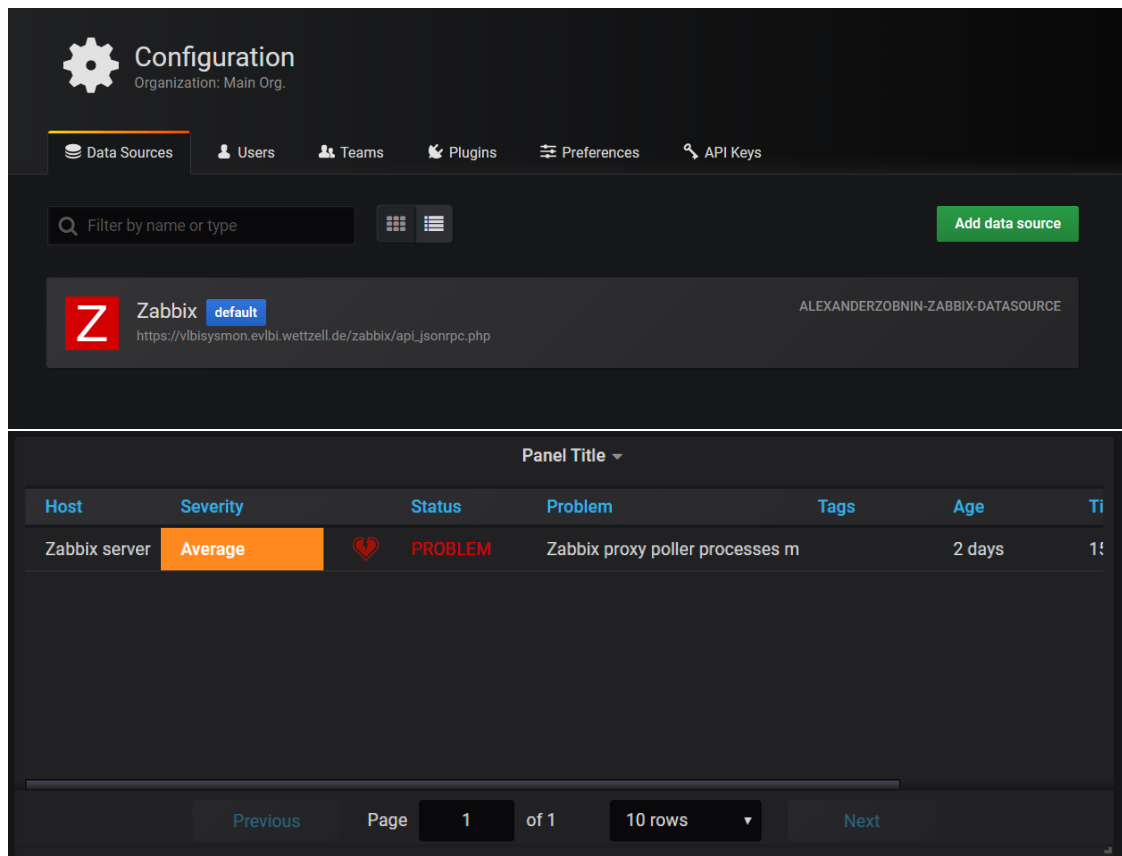
Tasks:

1. Create Docker for each app

Description	Progress	Timeline	Comments
Create Dockerfile for Zabbix, Grafana, and Flask on Linux, , mainly based on Zabbix	100%	07/2019-09/2019	This is mainly to create multi-container for each app

2. Install Plugin for maps and Zabbix

Description	Progress	Timeline	Comments
Install needed Plugins on <i>Grafana</i> to connect to <i>Zabbix</i> , <i>Worldmap Panel</i> for showing the map	100%	07/2019-08/2019	This is mainly acquire the data from the server.



3. Create a script to Scrap all sessions information and current

Description	Progress	Timeline	Comments
Create Python script for creating a dataframe from all the sessions and current session as well.	100%	08/2019-09/2019	This is mainly acquiring the sessions from the server side. https://ivscc.gsfc.nasa.gov/sessions/2019/

	Name	Code	Start	DOY	Dur	Stations	DB Code	Ops Center	Correlator	Status	Analysis	start_date	start_time	Split_Stations
1	IVS-R1875	R1875	2019-01-02 17:00	2.0	24:00	Ht Is Ke Kk Kv Ny On Wn Yg Ag Ma	XA	NASA	BONN	Released	NASA	2019-01-02	17:00:00	[Ht, Is, Ke, Kk, Kv, Ny, On, Wn, Yg, Ag, Ma]
2	IVS-R4875	R4875	2019-01-03 18:30	3.0	24:00	Ft Ht Is Kk Ma Ny Wn Wz Yg Ke	XE	USNO	WASH	Released	USNO	2019-01-03	18:30:00	[Ft, Ht, Is, Kk, Ma, Ny, Wn, Wz, Yg, Ke]
3	IVS-R1876	R1876	2019-01-07 17:00	7.0	24:00	Ht Ke Kk Kv Ma Wn Yg Ag	XA	NASA	BONN	Released	NASA	2019-01-07	17:00:00	[Ht, Ke, Kk, Kv, Ma, Wn, Yg, Ag]
4	VLBA133	RV133	2019-01-07 17:00	7.0	24:00	Hh Ny On Va Wz	XB	NASA	VLBA	Processing session	NASA	2019-01-07	17:00:00	[Hh, Ny, On, Va, Wz]
5	IVS-R8D-1	RD1901	2019-01-08 17:30	8.0	24:00	Ft Hh Kb Kk Ma Ny On Wz Ho	XA	NASA	HAYS	Waiting for data	NASA	2019-01-08	17:30:00	[Ft, Hh, Kb, Kk, Ma, Ny, On, Wz, Ho]

Current Session:

	Name	Code	Start	DOY	Dur	Stations	DB Code	Ops Center	Correlator	Status	Analysis	start_date	start_time	Split_Stations
142	AOV039	AOV039	2019-09-17 17:30	260.0	24:00	Ho Is K1 Kg Sh Sy Vm Ww Yg Ke	XA	GSI	GSI	NaN	NASA	2019-09-17	17:30:00	[Ho, Is, K1, Kg, Sh, Sy, Vm, Ww, Yg, Ke]

4. Create a database each telescope location, longitude, latitude

Description	Progress	Timeline	Comments
Create database for each of the unique 40 telescopes location on the map, including longitude and latitude	10%	09/2019-??	This task is currently blocked because I could not find website which provide information about the telescope location.

This NASA website provide all Network Stations Code and Name but without any location information <https://ivscg.gsfc.nasa.gov/sessions/stations/>

Where can the network station location be found?

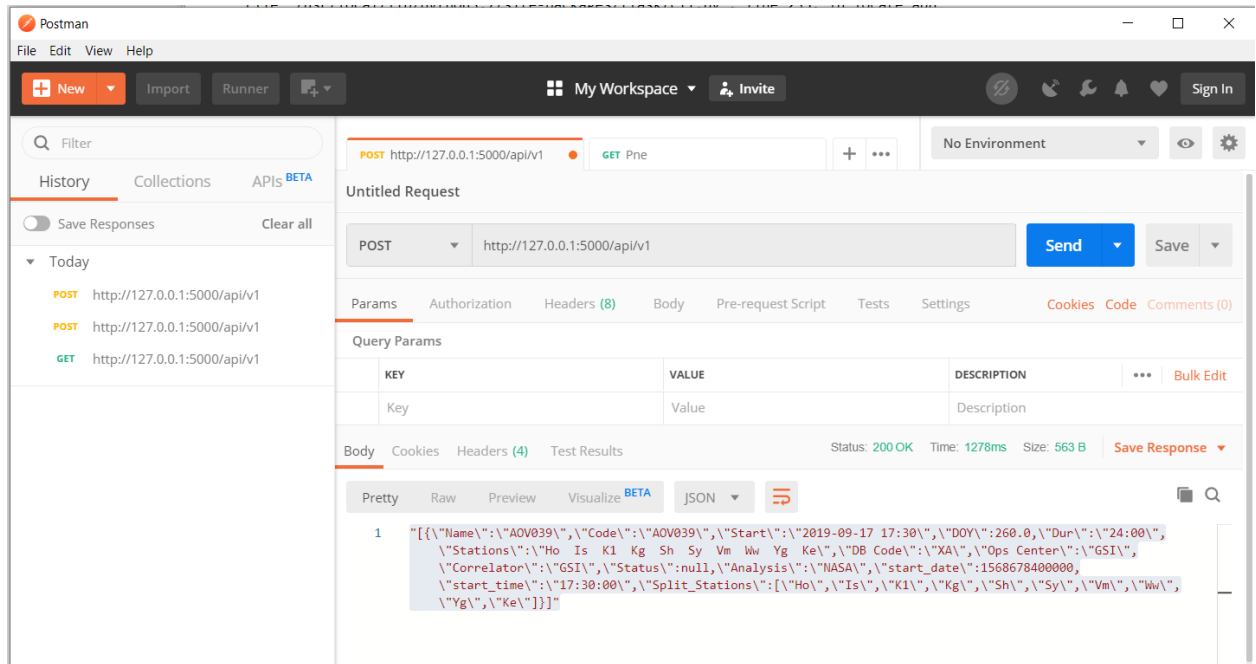
5. Script for parse current session as JSON

Description	Progress	Timeline	Comments
Python script for each session as JSON	100%	08/2019-09/2019	The JSON script is working correctly and providing the current and all sessions.

```
{
  "Name": "AOV039",
  "Code": "AOV039",
  "Start": "2019-09-17 17:30",
  "DOY": 260.0,
  "Dur": "24:00",
  "Stations": "Ho Is K1 Kg Sh Sy Vm Ww Yg Ke",
  "DB Code": "XA",
  "Ops Center": "GSI",
  "Correlator": "GSI",
  "Status": null,
  "Analysis": "NASA",
  "start_date": "2019-09-17",
  "start_time": "17:30:00",
  "Split_Stations": ["Ho", "Is", "K1", "Kg", "Sh", "Sy", "Vm", "Ww", "Yg", "Ke"]
}
```

6. Flask server to create a REST API

Description	Progress	Timeline	Comments
Create docker and launch basic Flask server	60%	08/2019-09/2019	Flask server as an API for POST and GET sessions. Still, it is a basic server.



7. JSON server for all network stations location and code

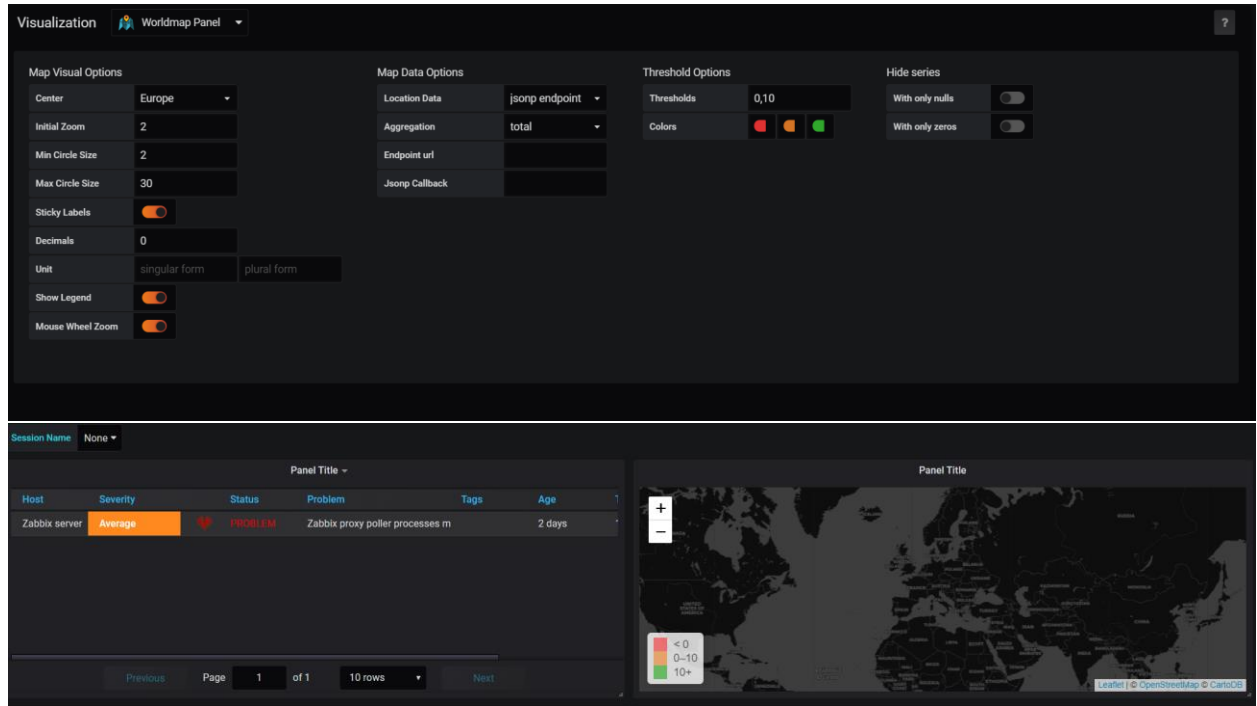
Description	Progress	Timeline	Comments
Create JSON server for station code and location	10%	09/2019-??	This will be used as jsonp endpoint for worldmap location -I need to learn more and JSON and POST GET for the API

8. JSON server for current sessions information

Description	Progress	Timeline	Comments
JSON server for providing current session ID and participating stations	10%	09/2019-??	I need to learn and test more and JSON and POST GET for the API

9. Modify Worldmap datasource from the Flask JSON Server and Zabbix

Description	Progress	Timeline	Comments
Checking the main concept of Worldmap	20%	08/2019-??	Worldmap requires the location of each satellite to be created in jsonp server, which include the location of each station. Input here from task 7



Current Running Servers:

Postgres Server

<http://localhost:8080>

Zabbix

<http://localhost:8090>

Grafana

<http://localhost:3000>

Flask

<http://localhost:5000>