

OPENBACH, OPEN METROLOGY TESTING FRAMEWORK

Global technical presentation

E. Dubois (CNES), D. Pradas (Viveris Technologies)

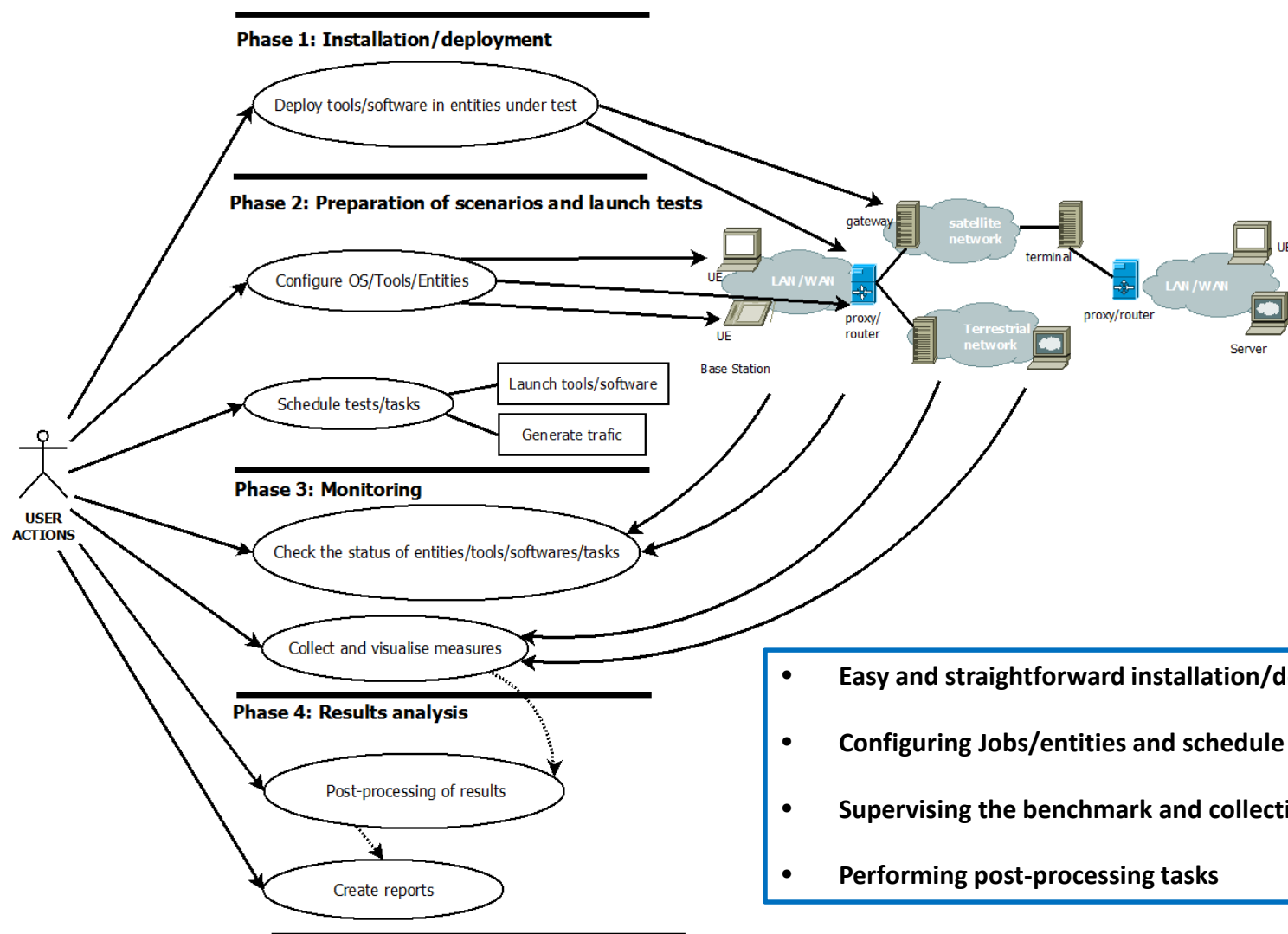


GENERAL SPECIFICATIONS



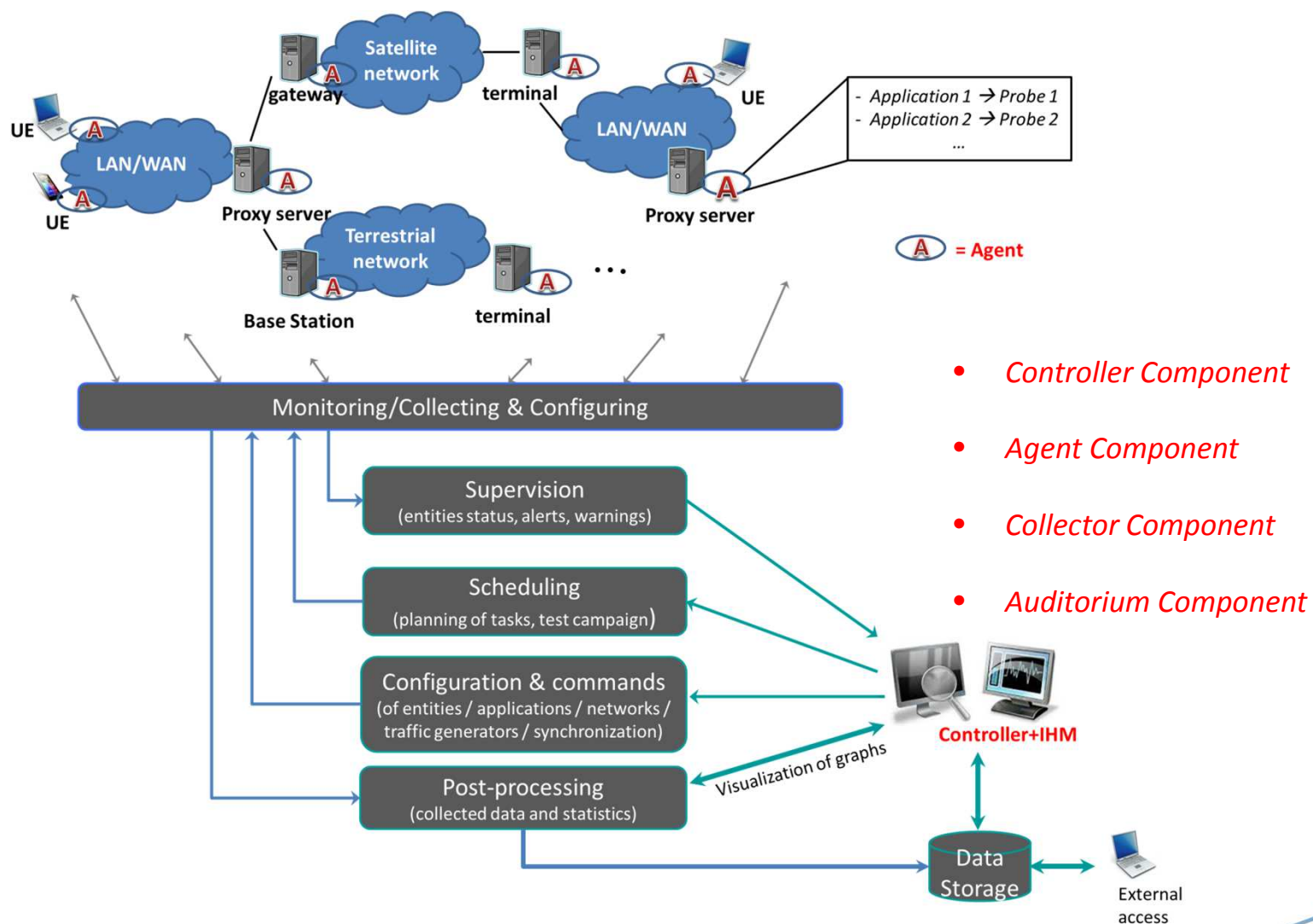
- The benchmark shall be based as much as possible on open-source components.
- The benchmark shall be able to be integrated in different types of equipments, servers, clients, hardware and software, with minimal adaptation effort.
- The benchmark shall be easily scalable.
- OpenBACH shall have a modular structure to facilitate the addition of new:
 - software tools
 - monitoring parameters
 - tasks
- All components/entities synchronized

METROLOGY GENERAL USE CASES



- Easy and straightforward installation/deployment
- Configuring Jobs/entities and schedule scenarios
- Supervising the benchmark and collecting measures
- Performing post-processing tasks

PRELIMINARY DESIGN



OPENBACH TERMINOLOGY



Job: A number of individual tasks (one or more) with a common purpose and to be executed in a sole Agent.

Job instance: An execution of a job configured with a set of parameters.

Openbach-function: Function defined and executed by the Controller allowing to start/stop job/scenario instances, install agents/Jobs, perform information/status requests, etc.

Scenario: Set of openbach-functions that allow to perform different tasks might be executed in different Agents to accomplish an action.

Scenario instance: An execution of a scenario with a set of parameters.

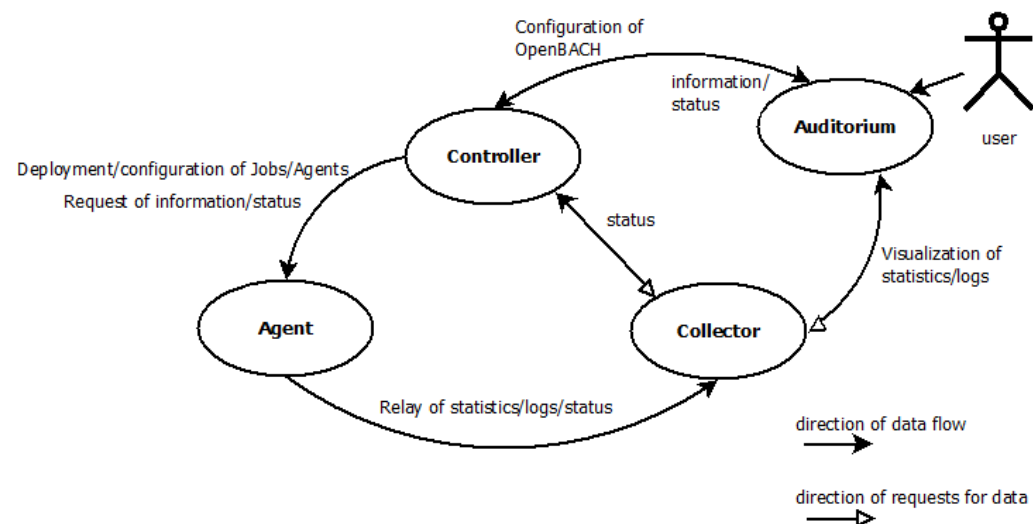
Project : Allows for scenario organization in OpenBACH

Entity : Agent representation in a project.

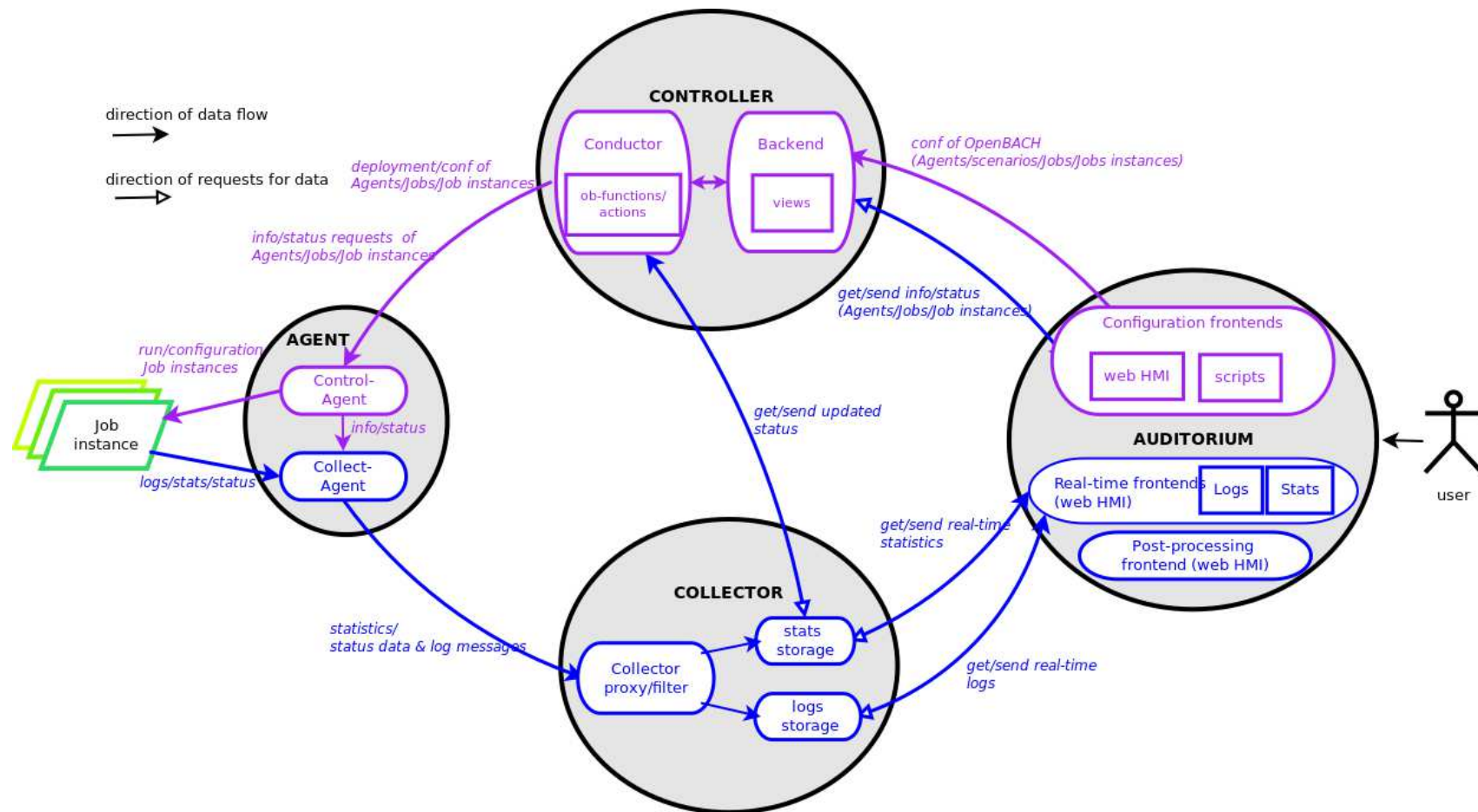
DESIGN

Based on 4 components:

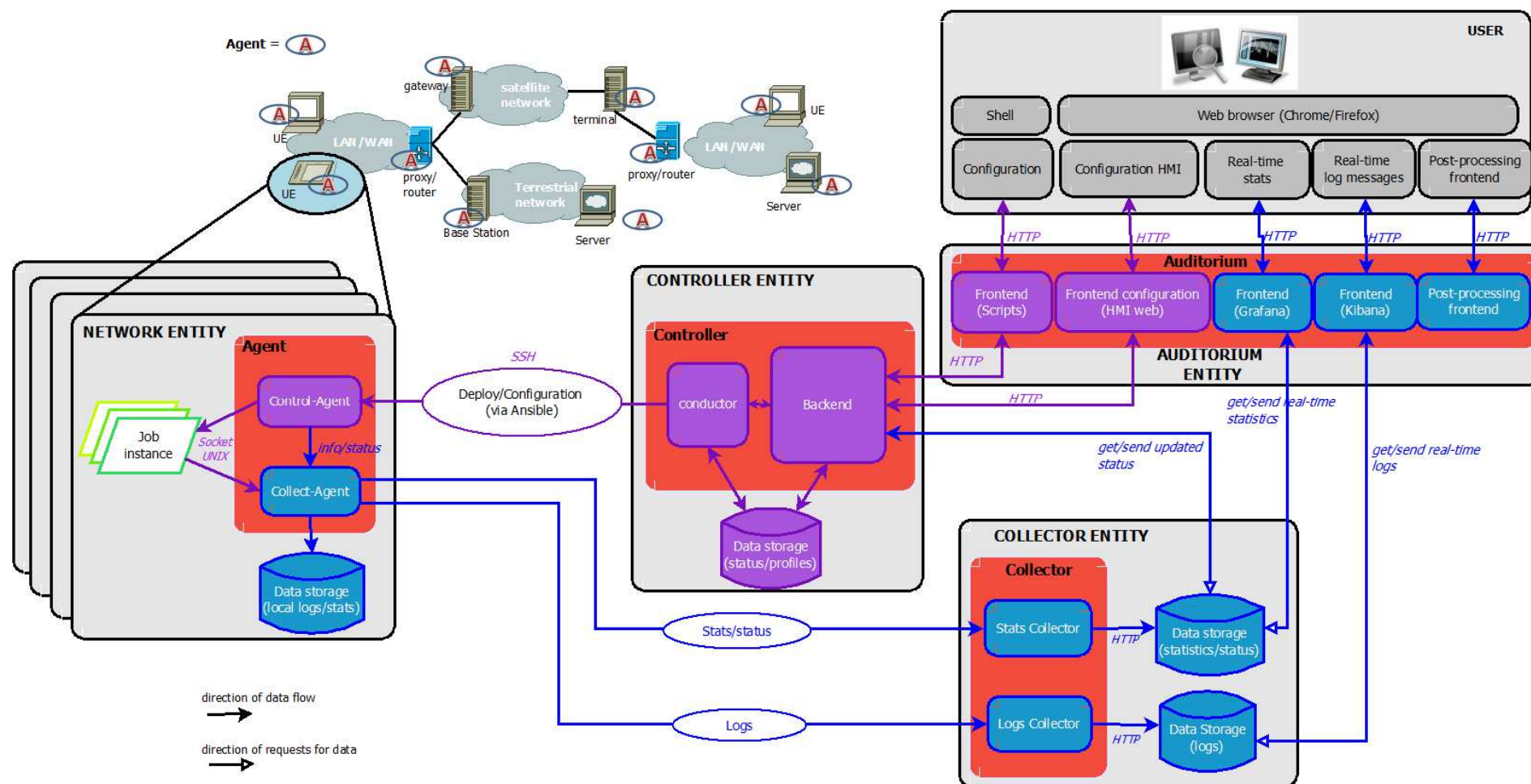
- **Controller**: Centralise and deploy configuration, including jobs/scenarios.
- **Collector**: Centralise the collection of data/status and log messages.
- **Agent**: deployed in the entities under test (WS, ST, proxy, server, etc.)
- **Auditorium**: Centralise the frontends/web interfaces of configuration and monitoring.



FUNCTIONAL ARCHITECTURE



DETAILED ARCHITECTURE



Available functions in the Controller:

- Add/install (delete/remove) Agents and Jobs to/from the benchmark
- List the available Agents and the available jobs per Agent.
- Create/modify/delete a scenario.
- Configure/launch/stop scenario instances.
- List the available scenario and scenario instances and their status.
- Send commands of schedule/start/stop of Jobs instances to the corresponding Agents .
- List the scheduled/started job instances and their status.

How to deploy new Agents, new jobs in the entities (and their dependencies)?


Ansible! Off-the-shelf framework for deployment of configuration, automation of common tasks, etc.

- Via SSH
- Idempotence concept

AUDITORIUM HMI




- Administration


 OpenBach Administration

EXTERNAL LINKS ▾ | HELP ▾ | ADMIN ▾

Agents

Name	Address	Collector	Uninstall
Controller	172.20.34.39	172.20.34.39	

Projects topology



```
graph TD; Controller[Controller] --- DITG[DITG project];
```

Add agent

Name

IP Address

Username

Password

Collector

ADD

- Project

 'MPTCP' Project

EXTERNAL LINKS ▾

HELP ▾

ADMIN ▾

PROJECT

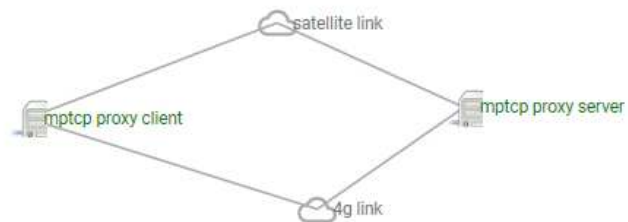
TOPOLOGY

SCENARIOS

INSTANCES


NO SCENARIO SELECTED

Topology



Entity

mptcp proxy server
The MPTCP proxy on the server side



- Description: **The MPTCP proxy on the server side**
- OS: **Linux**
- Networks:

satellite link 4g link ▾
- Agent:
 - Name: **p6_agent3**
 - IP: **172.20.34.36**
 - Username: **exploit**
 - Collector: **172.20.34.32**
 - Installed jobs: No jobs installed

INSTALL JOBS

UNINSTALL

Select... ▾

Select... ▾

- Scenario status

ping

ping

(id: 278) 2017-05-19 10:04:09 -> 2017-05-19 10:04:28 [Duration of 0:00:18]



fping *start_job_instance*



fping (172.20.34.39)
(id: 240)



hping *start_job_instance*



hping (172.20.34.39)
(id: 241)

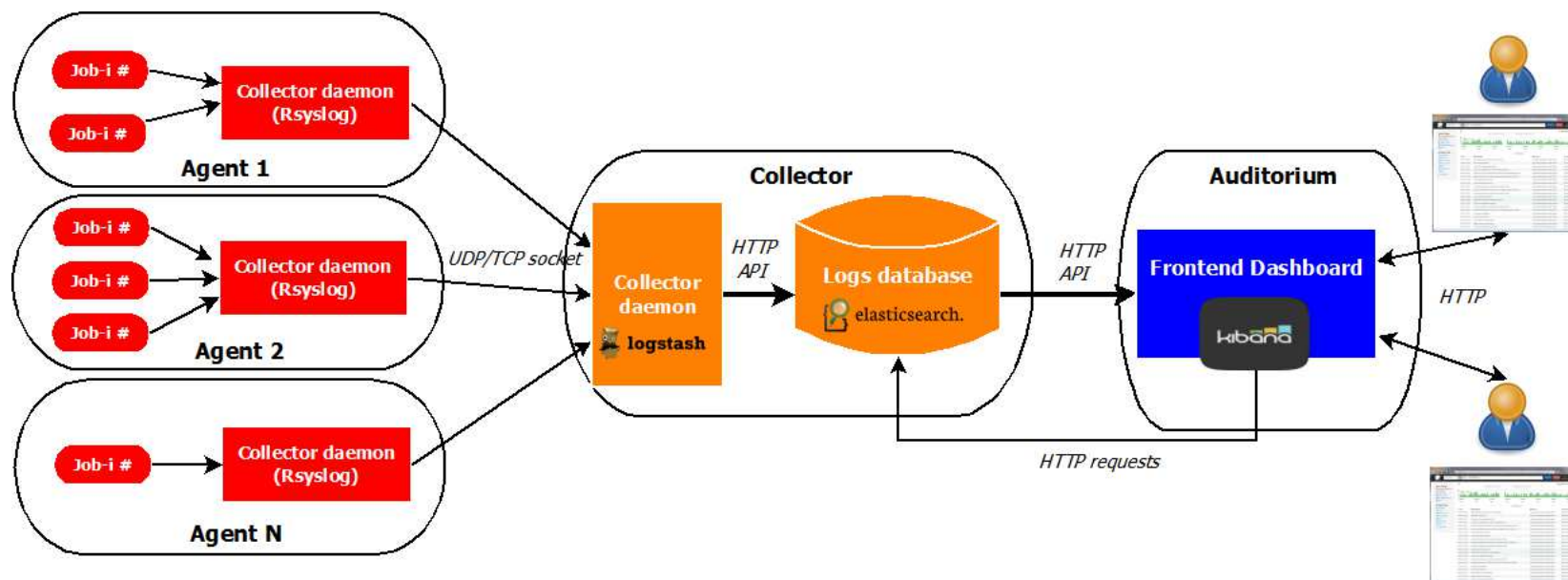


SHOW STATISTICS

OK

LOGS

- Collection and display of log messages

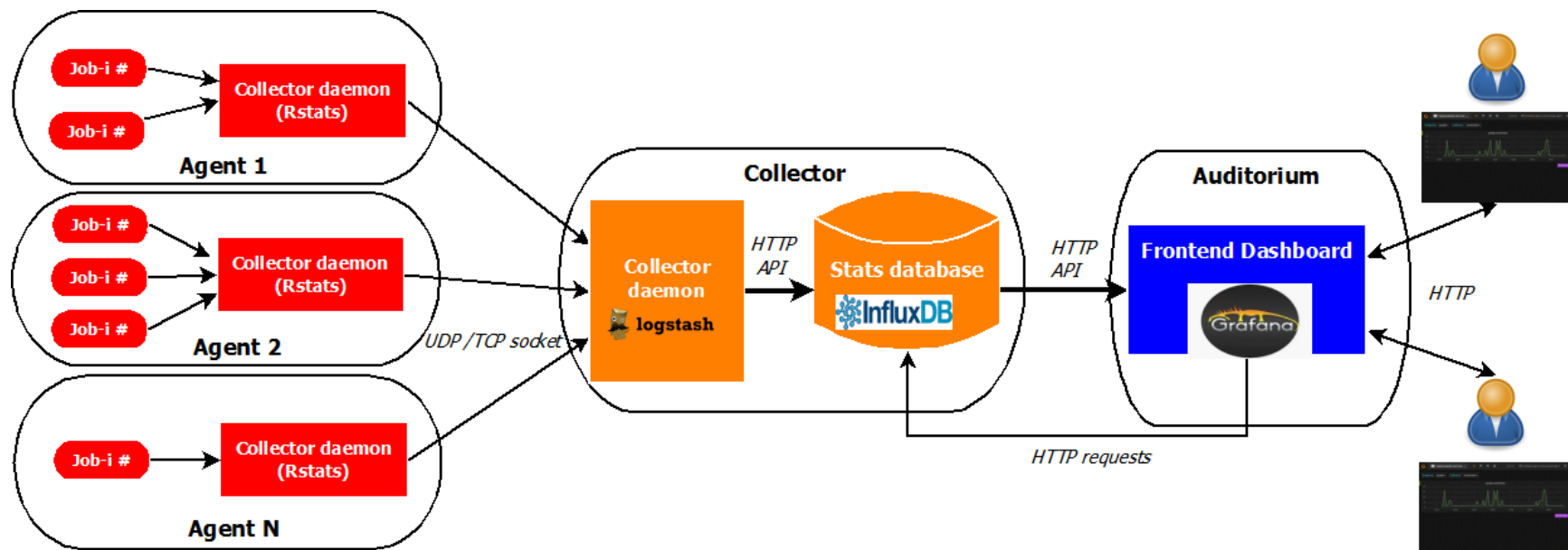


LOGS

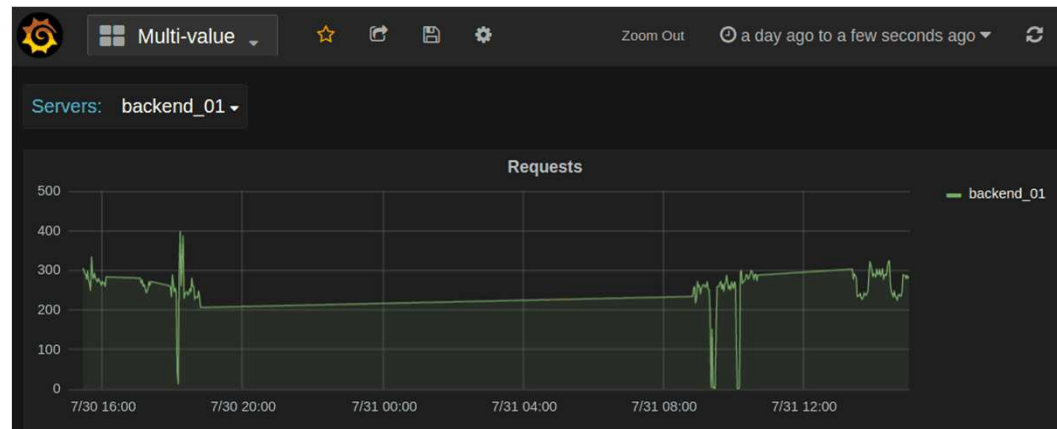
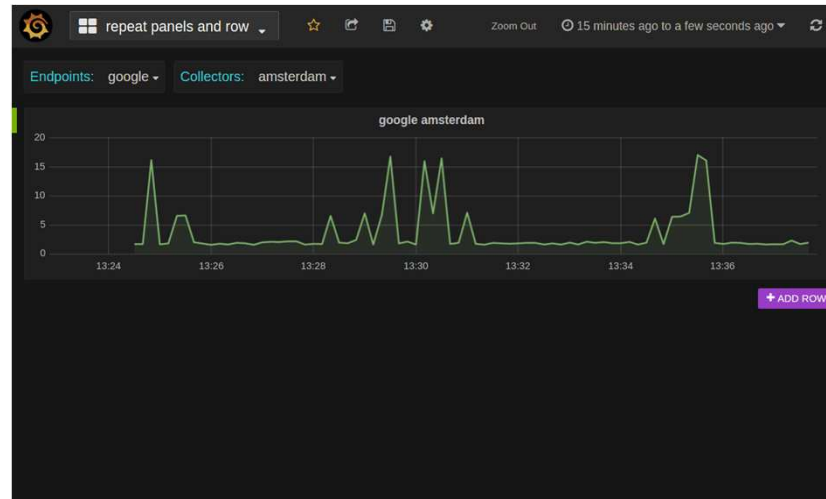


STATS

- Collection and display of statistics/measures



STATS



TECHNICAL PERSPECTIVES



- Main technical points in progress :
 - Technical Jobs
 - Network integration/configuration
 - Postprocessing
 - System configuration
 - Multi-user benchmark
 - System integration
 - HMI Evolutions