O^2 Control , Configuration, Monitoring and Logging

Teo Mrnjavac

Introduction

Update from last workshop in June

https://indico.cern.ch/event/687364/contributions/3022314/

Overview

Control

- Orchestration of O^2 processes
- Integration with detectors, trigger, LHC
- Resource management
- Scheduling
- Automation

Configuration

- Configuration of CRU firmware
- Configuration of O^2 software
- Configuration of O^2 services
- Configuration of FLPs and EPNs

Monitoring

- Collection, processing, storage and visualization of monitoring data
- Infrastructure, hardware, system, process and application

Logging

- Collection, annotation, transport, storage and visualization of application logs
- Analytics of system and service logs (NEW)

Status

June

Control	Configuration	Monitoring	Logging
• In progress	• v1 available	• v1 available	• v1 available

Now

Control	Configuration	Monitoring	Logging
• v1 available	• v1 available	• v1 available	• v1 available (application logs)

Control

More details in next presentation

Configuration

Overview

- Configuration library
  - Support for file formats
  - Support for Consul

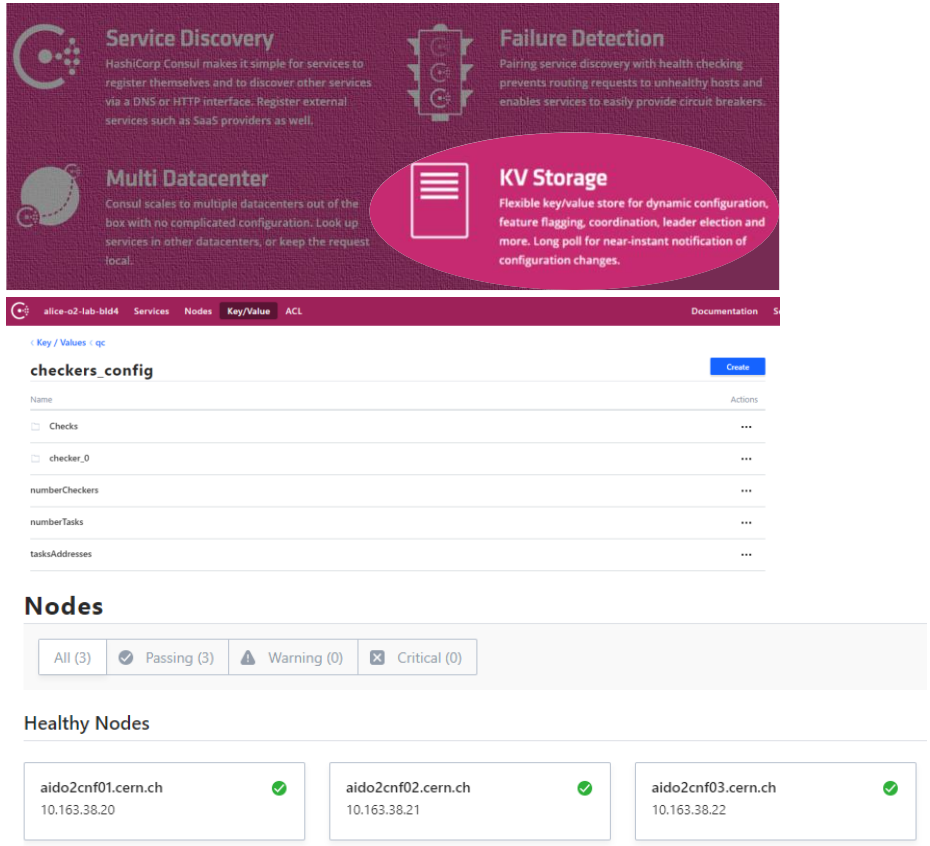
```
auto conf = ConfigurationFactory::getConfiguration("ini:///temp/config.ini");
int value = conf->get<int>("my_dir.my_key");
```

```
auto conf = ConfigurationFactory::getConfiguration("consul://localhost:8500");

// Get a tree
boost::property_tree::ptree subTree = conf->getRecursive("my_dir");
subTree.get<int>("my_key");

// Get flat key-value map
std::unordered_map<std::string, std::string> map = conf->getRecursiveMap("my_dir");
map["my_key"];
```

- Consul as configuration store



Configuration

Changes since last Workshop

- Configuration library main changes
  - Use of boost::ptree instead of custom tree as data structure
  - Support for arrays
  - Dot (.) as default separator (e.g. `its.param1`)
  - JSON ⇔ Consul conversion tool
  - Better documentation
- Integration expanding
  - Readout
  - Quality Control

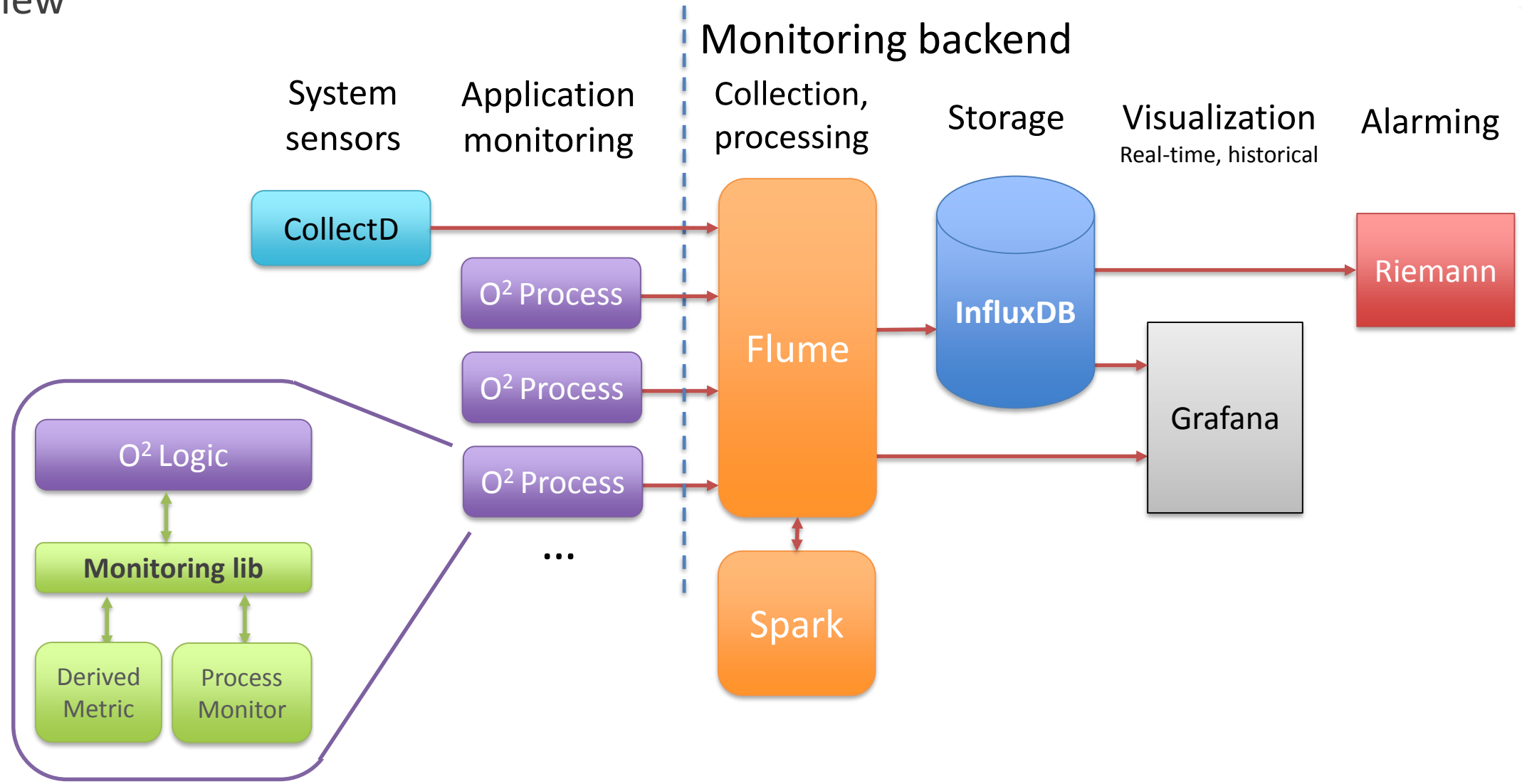
Configuration

Next steps

- DPL integration
- Gain experience using Consul as backend
- Custom GUI

Monitoring

Overview



Monitoring

Changes since last Workshop

- Monitoring library main changes
  - Improved process monitoring
  - Auto push metrics
  - Production/debug verbosity modes
- Sensor portfolio growing
  - IPMI, SMART (hard drives), network, CRU
- Integration expanding
  - DPL
- Dashboard portfolio growing
  - CR0, network, internal monitoring

Monitoring

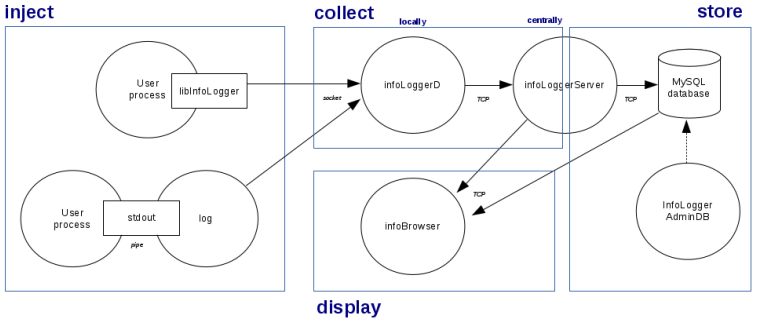
Next steps

- More dashboards
- More sensors
- Spark jobs for aggregation/transformation/enrichment
- Alarms

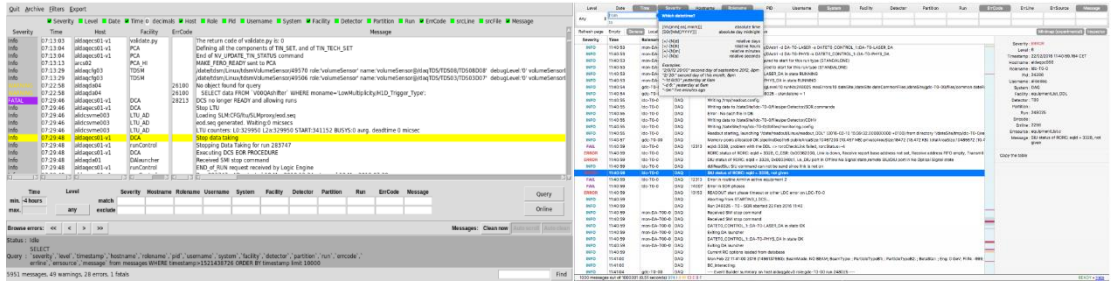
Logging

Overview

- Application logging
  - Evolution from existing DAQ infoLogger



  - Desktop and web GUI



- System and service logging
  - Lead by our colleagues from KMUTT in Thailand
  - More details on presentation by Tiranee Achalakul

Logging

Changes since last Workshop

- Application logging library main changes
  - Better support for Mac platforms

Conclusion

- Infrastructure services becoming available
  - Control, Configuration, Monitoring, Application Logging

- Will continue to improve in the coming months

References

- Control: https://github.com/AliceO2Group/Control
- Configuration: https://github.com/AliceO2Group/Configuration
- Monitoring: https://github.com/AliceO2Group/Monitoring
- Application Logging: https://github.com/AliceO2Group/InfoLogger
- Contacts
  - New requirements: alice-o2-wp8@cern.ch
  - Support: https://alice-talk.web.cern.ch