Towards the integrated ALICE Online-Offline monitoring subsystem

Adam Wegrzynek
for the ALICE Collaboration
ALICE O²

First Level Processors

- 3.3 TB/s
- 9000 fibers
- 19 detectors
- 270 nodes

Event Processing Nodes

- 500 GB/s
- 1500 nodes

Storage

- 100 GB/s
- 60 PB storage

Synchronous post-processing

Asynchronous post-processing
Comparison

1. Modular stack
   - Spark
   - InfluxDB

2. MonALISA
   - Monitoring Agents using a Large Integrated Services Architecture

3. ZABBIX

- Performance requirements
- Functional architecture
- Experience at CERN
1. **Modular stack**

1. **collectd**
   - System performance metrics
   - Hardware monitoring

2. **Flume**
   - Metric routing

3. **Apache Spark**
   - In memory data processing

4. **InfluxDB**
   - Time series database

5. **Grafana**
   - Visualization tool

6. **RIEMANN**
   - Alarming

Currently used at INFN Bari, CERN IT
2. MonALISA

▶ Distributed data collector infrastructure
▶ Discovery mechanism
▶ Aggregation, filtering, alerts
▶ Real-time data distribution
▶ In memory buffers
▶ SQL database
▶ Currently used by ALICE Offline

Courtesy of Costin Grigoraş
3. Zabbix

- Agent-server
- Push via Zabbix protocol
- Community support
- Currently used in ALICE HLT and DAQ for computing infrastructure monitoring
## Comparison table (1)

<table>
<thead>
<tr>
<th></th>
<th>Modular Stack</th>
<th>MonALISA</th>
<th>Zabbix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference OS (CC7)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Documentation</td>
<td>Good</td>
<td>Insufficient</td>
<td>Good</td>
</tr>
<tr>
<td>Support and maintenance</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Running in isolation</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>600 kHz rate</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Scalable &gt;&gt;600 kHz</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Handle 100k sources</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Storage size</td>
<td>~30 bytes</td>
<td>~75 bytes</td>
<td>90-500 bytes</td>
</tr>
</tbody>
</table>

- **Modular Stack**
- **MonALISA**
- **Zabbix**
<table>
<thead>
<tr>
<th></th>
<th>Modular Stack</th>
<th>MonALISA</th>
<th>Zabbix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Functional arch.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>System sensors</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Metric processing</td>
<td>Batch and stream</td>
<td>Stream</td>
<td>Batch</td>
</tr>
<tr>
<td>Historical dashboard</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Real-time dashboard</td>
<td>No (RFC)</td>
<td>Yes (obsolete)</td>
<td>No</td>
</tr>
<tr>
<td>Alarming</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Storage downsampling</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
1. **Modular stack**

![](Spark.png) ![](InfluxDB.png) ![](RIEMANN.png)

2. Remains for Grid job monitoring

3. ![](ZABBIX.png)

Adam Wegrzynek | CHEP18 | Towards the integrated ALICE Online-Offline monitoring subsystem
Modular stack metric flow

Computing node
- System sensors
- Application monitoring
- CollectD
- Processing device
- Monitoring lib
- Derived Metric
- Process Monitor

Monitoring backend
- Collection, processing
- Flume
- Processing device
- InfluxDB
- Spark
- Monitoring lib
- Process Monitor
- Derived Metric

Storage
- Riemann
- Visualizing
  - Real-time, historical

Alarming
Monitoring library

- Push metrics to a backend
- Monitors the process
- Derived metrics
- Tags

**AliceO2Group/Monitoring**

<table>
<thead>
<tr>
<th>metric type</th>
<th>value</th>
<th>timestamp</th>
<th>hostname</th>
<th>role</th>
<th>detector</th>
</tr>
</thead>
<tbody>
<tr>
<td>myMetric</td>
<td>0 10</td>
<td>1530099250985</td>
<td>test.cern</td>
<td>readout</td>
<td>TPC</td>
</tr>
</tbody>
</table>
Flume routing

- **HTTP Source**
- **UDP/JSON Source**
- **MonALISA Source**

- **CollectD JSON Handler**
- **Channel Selector**
- **Memory Channel**
- **IFQL UDP Sink**
- **WebSocket Sink**
- **HTTP Sink**
- **Avro Sink**

- Courtesy of **Gioacchino Vino**
Spark jobs

- Higher level metrics
- Written in Scala
- Operates on Flume events
- Configurable

Readout rate of a detector

Sum rate of each detector link

Flume

Spark

avro

UDP
InfluxDB timeseries storage

- Up to 700 kHz writes
  - 8 data streams
  - 2x SSD drives RAID0, 25 GbE

- Continuous Queries
  - Downsampling high resolution data
    (merge 12 points into 1 by applying average)

- Retention Policies
  - Drop high resolution data after 30 days
  - Keep low resolution data for 1 year
Grafana

**Network**
- 2018-07-01 16:04:30
- Incoming data: 529 Bps
- Outgoing data: 510 Bps
- Incoming packets: 2 pps
- Outgoing packets: 2 pps

**Disk Throughput**
- Min: 0 Bps
- Max: 183.5 Bps
- Current: 28.7 Bps

**CPU Load**
- Min: 0
- Max: 0.066
- Avg: 0.041

**Disk IOPS**
- Min: 2.45 iops
- Max: 7.70 iops
- Avg: 0.44 iops
- Current: 0.00 iops
Integration with O² Software

- Quality Control
- Data Processing Layer
  
  Evolution of the ALICE Software Framework for LHC Run 3
  
  Giulio Eulisse, Tuesday 10 July 14:15, Hall 3

- Readout
  
  Readout software for the ALICE integrated Online-Offline (O2) system
  
  Filippo Costa, Thursday 12 July 11:00, Hall 3.1
Conclusion

- 3 options compared
- Modular Stack selected for $O^2$ farm monitoring
- Defined interfaces between tools
- Deployed in the detector commissioning facilities
Future steps

- Alarming
  - Define thresholds and patterns
- Grafana real-time data source
  - Display critical metrics in real time
- Sensors to custom hardware
  - Monitor status of custom FPGA board