Presentation layer of CMS Online Monitoring System

Data Visualization Framework

Mantas Stankevičius (FNAL) on behalf of OMS development group
CMS OMS - Online Monitoring System

Upgrade and successor to the CMS Web-based Monitoring (WBM) system

- Aggregate and integrate different sources of information into a central place
- Allow users to view, compare and correlate information
- Display real-time and historical information
Roadmap

- **2015 Dec**: WBM Review
  Decided to upgrade

- **2016 Nov**: OMS kickoff workshop

- **2017 June**: OMS prototype #1

- **2017 Dec**: OMS prototype #2

- **2018 March**: OMS in production.
  Runs in parallel with WBM
CMS OMS structure

OMS is divided into two separate layers. Layers communicate via RESTful API.

**Aggregation layer**
Fetch data from database and expose it via RESTful JSON:API (LHC Fills, CMS Runs, Lumisections, Trigger rates, ...)

**Presentation layer**
Fetch data from multiple sources and display it via web interface (Datatables, interactive charts, images, ...)

[Diagram showing the CMS OMS structure with layers and communication through RESTful API.]
Presentation layer

**Metadata API** provides portal configuration:

- Workspaces, Folders, Pages
- Page layout (controller & portlets & relations)
- Portlet type, size, position
- User privileges

**Web application** is graphical user interface:

- User friendly
- Interactive
- Responsive
- Customizable
Metadata API

python + Flask + SQLAlchemy

✔ RESTful
✔ CRUD
✔ JSON:API specification
✔ Test coverage
✔ Scalable
Web Application

State of the art technologies:

- ES6
- ReactJS
- Redux
- Material-UI
- Highcharts

- ✔ Lightweight
- ✔ Responsive
- ✔ Interactive
- ✔ Scalable
Content organisation

Workspace – top-level container defines organizational unit, sub-detector, etc.

Folder – lower-level container contains Pages organized by context.

Page – lowest-level content container contains one or more Portlets.

Controller - (optional) component allows user to filter result set.

Portlet - a content presentation component.
Components: Controller

Controller is a **reusable** filtering component.

User selected filter is applied to all Portlets within the Page.
Components: Portlet

Portlet is a **reusable** content presentation component.

Displays **single** well-defined aspect of information: chart, datatable, text, etc.

- Description tooltip
- Refresh button
- Collapse mode
- Full Screen mode
### Components: Portlet

#### Metadata API

<table>
<thead>
<tr>
<th>Run</th>
<th>Duration</th>
<th>Start Time</th>
<th>End Time</th>
<th>Del. Lumi pk^-1</th>
<th>Rec. Lumi pk^-1</th>
<th>L1 Triggers</th>
<th>L1/HLT Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>318879</td>
<td>01:03:28</td>
<td>2018-06-28 19:44:24</td>
<td>2018-06-29 00:47:52</td>
<td>47.401</td>
<td>44.185</td>
<td>233865054</td>
<td>collisions2018</td>
</tr>
<tr>
<td>318874</td>
<td>02:04:55</td>
<td>2018-06-28 17:38:51</td>
<td>2018-06-29 00:43:46</td>
<td>100.164</td>
<td>94.792</td>
<td>505486155</td>
<td>collisions2018</td>
</tr>
<tr>
<td>318872</td>
<td>01:52:08</td>
<td>2018-06-28 15:44:01</td>
<td>2018-06-29 00:36:09</td>
<td>94.516</td>
<td>88.548</td>
<td>418474880</td>
<td>collisions2018</td>
</tr>
<tr>
<td>318871</td>
<td>00:00:58</td>
<td>2018-06-28 15:40:44</td>
<td>2018-06-28 20:41:42</td>
<td>0.000</td>
<td>0.000</td>
<td>418474880</td>
<td>collisions2018</td>
</tr>
</tbody>
</table>

#### Aggregation API

- **Duration**: 14617
- **start_time**: "2018-06-28T20:48:42Z"
- **l1_triggers_counter**: 876437745
- **l1_hlt_mode**: "collisions2018"
- **end_time**: "2018-06-29T00:52:19Z"
- **delivered_lumi**: 157.346391
- **recorded_lumi**: 152.885063

### JSON Representation

```json
{ "label": "Run", "name": "run_number", "props": { "href": "/cms/runs/report?cms_run=%", "params": ["run_number"], "type": "link" } }

{ "label": "Duration", "name": "duration", "props": { "type": "duration" } }

{ "label": "Del. Lumi", "name": "delivered_lumi", "show_units": true, "props": { "ndigits": 3, "type": "float" } }
```
“Master-detail”

Similar to Master-Detail, but it is Many-to-Many relationship.

Portlets can be assigned into groups. Multiple groups can be within a Page.

Portlets communicate with each other only within a group scope.

Action in one Portlet causes other Portlets update accordingly.
“Master-detail” example
Portal: Edit Mode

Still under heavy development.

Privileged users can **realtime:**

- Add/Remove/Resize/Rearrange existing Portlets within a Page.
- Create/Update/Delete/Reorder Folders and Pages

No need to redeploy!
Future plan

OMS in production. Runs in parallel with WBM

2018 March

2018

Long Shutdown 2 (2019 - 2020)

2021

OMS in production. Shutdown WBM

Content migration