Detector controls meets JEE on the web

ICALEPCS 2015

Frank Glege
Part 1: Web based Remote access to controls systems

Part 2: JEE for controls
About CERN
The CMS detector

Compact Muon Solenoid

Length: ~20m
Diameter: ~15m
Weight: 12500 t
Magnet: 4T superconducting
CMS DCS facts

- Working since several years at high efficiency.
- SCADA: WinCC OA developed by ETM
- ~3 million parameters
- ~700,000 lines of code
- ~35,000 finite state machine nodes
- ~34 SCADA systems
- 29 redundant PC pairs (Windows)
- ~50 DB schemas (ORACLE)
- O(TB) of data in schemas

SCADA: supervisory control and data acquisition
The logical structure of the system is built with a hierarchy of finite state machine nodes. The image shows a circular view of this hierarchy with the top node in the center and sub nodes distributed equally on consecutive circles.
Remote control

- 4000 collaboration members should be able to access the DCS information
- All collaboration members should be able to read information
- Experts should be able to access information about their system in read/write mode
- Visualization should be live
- Authentication and authorization is a must
Visualization in WinCC OA

• In WinCC OA visualization is done through so-called panels.
• Panels are developed using a graphical editor allowing to position graphics elements (text, buttons, lines, etc.)
• Each element can be enhanced by scripts
• The scripts have access to process data
• Panels are visualized by a WinCC OA application using Qt
Example WinCC OA panel
Bringing it to the Web

• Thousands of panels have been created
• Redoing panels in HTML is not an option
• Panels should be visualized on the WEB without any modification

• Intercept drawing commands in Qt and forward them to an HTML5 canvas
• Forward HTML5 canvas events to Qt

20.10.2015  Frank Glege
Painting a panel

Layout manager

Graphics element

1. Reserve space
2. Draw rectangle
3. Color rectangle
4. Draw text

CMS_CSC_PLUS

Graphics element components

CMS_CSC_PLUS
Command redirection

Canvas

clip
draw rectangle
Stroke
draw line

...
Command redirection

Qt platform abstraction

Canvas

... clip draw rectangle Stroke draw line ...

HTML 5

... clip draw rectangle Stroke draw line ...

20.10.2015

Frank Glege
Web server as gateway

• Using a web server as interface between Qt and the HTML5 canvas allows for
  – Easy authentication and authorization
  – Multi user access
  – Fire walling using well tested applications

• Additionally required components:
  – Resource broker
  – Fan out

• Single user read/write
• Multi user read only
Why JEE?

• JEE (JAVA enterprise edition) provides authorization

• JEE provides EJBs (Enterprise Java Beans):
  – EJBs model the business logic of a JEE application
  – EJBs are not limited to a request
  – EJBs provide intercommunication
  – EJBs allow modularity and scalability
  – EJBs allow for distributed systems
Work flow

Authorization

Web server infrastructure

Custom EJB

Resource broker

Web socket proxy

Panel request

Client HTML

Start panel
Performance

• Without optimization not all panels provided a sufficiently low latency in rendering due to high data rate.

• For a reference panel the data rate from Qt to the browser could be optimized as follows:
  – Before optimization: 1.1MB/s required
  – After optimization: 20kB/s required
Optimizations

- Command sequence caching
- Command indexing
- Image caching
- Optimize TCP packet usage
- Optimize (reduce) event sending
- Reduce update frequency to human perceivable rate
Next step

• Allow to add pure HTML objects to visualization
  – Allows for navigation and simple animation
  – Enables restricted multi user read/write access
  – Will further improve performance
  – Allows for object based authorization
Adding pure HTML objects

- Buttons
- Text
- Color coded fields
Final goal

• Pure HTML visualization
  – Requires reworking or new creation of PVSS panels
  – Enables full multi user read/write
  – Will provide maximum performance
  – Requires object based authorization
  – Feasibility still to be proven (can all panel designs be reproduced with pure HTML?)
Summary of part 1

• We implemented a solution for web based remote control mirroring WinCC OA panels using Qt platform abstraction

• No modification of panels is needed to provide single user read/write or multi user read only in a first step with sufficient performance

• An evolution towards a fully HTML based remote visualization is possible and foreseen
Controls system architecture

System N
System 2
System 1

Programming Interface
(API, scripting, panels, ...)

Visualization

Processing core

Persistent storage

Front end interfaces

Controls System API

Device API

20.10.2015
Frank Glege
JEE APIs

Extract from the ORACLE JEE documentation
JEE architecture

Web server N
Web server 2
Web server 1

Servlet, JSP

EJB

Persistent storage

Web browsers

Device interfaces

JNI, JAVA, TCP, Etc.

Device API

20.10.2015

Frank Glege
JEE for controls?

- The concept fits
- The visualization is quite powerful and provides many ready to use components
- Access to different persistent storage systems is built in and uses a common API
- Different communication mechanisms between JEE components and towards the outside are provided
- Provides authentication and authorization
- Programmable in JAVA

20.10.2015

Frank Glege
JEE for controls?

- JEE is not a fully functional controls software
- JEE is a concept for which different implementations exist
- JEE provides many components and interfaces required to build a controls system
- JEE is well documented, widely used and often free of charges

JEE seems well suited as a basis for the development of a new controls system

20.10.2015
Frank Glege