Control System Studio Training - Alarm System Setup

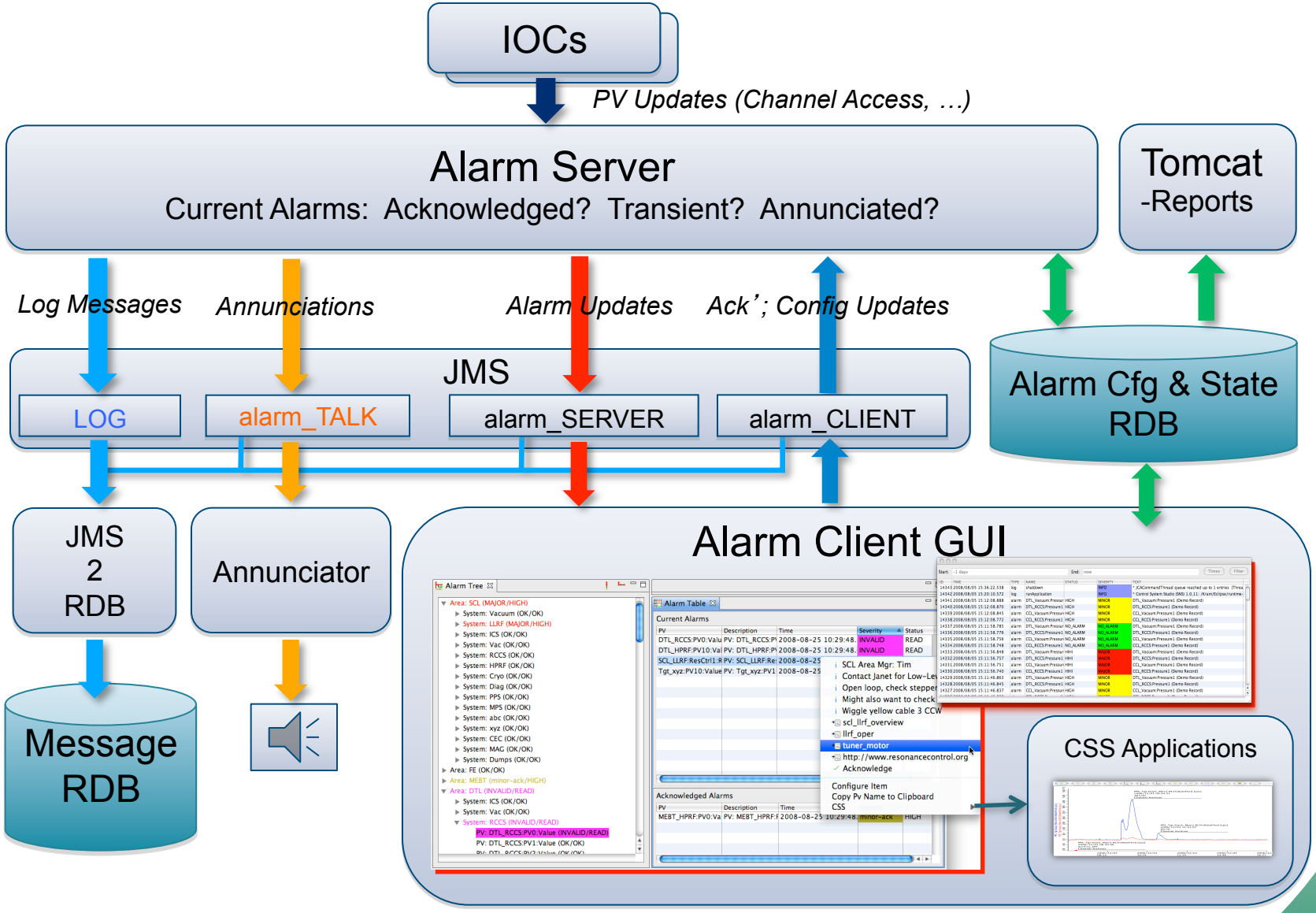
Kay Kasemir

ORNL/SNS

kasemirk@ornl.gov

Jan. 2013

Alarm System Overview



Initial Setup similar to archive system

1. Prepare RDB
2. Start JMS Server
3. Create and import initial configuration
4. Run Alarm Server
5. View in CSS Alarm Tree, Alarm Table

Prepare RDB

Plugin `org.csstudio.alarm.beast`,
folder `dbd/`:

Copy/paste the commands for the following from `MYSQL_USER.sql` and `ALARM_MYSQL.sql` into a mysql shell:

1. Create “alarm” user with password “$alarm”
2. Allow “report” user to read alarm tables
3. Create “alarm” data base
4. Create tables, insert some demo data

Create initial configuration

- **Minimum XML File**

```
<config name=”Demo">
</config>
```

- **More elaborate Example**

```
<config name=”Demo">
    <component name="Simulated">
        <pv name="sim://ramp">
            <description>Ramp</description>
            <latching>true</latching>
            <annunciating>true</annunciating>
        </pv>
    </component>
    <component name="Heater Demo">
        <pv name="demo1:heat_V">
            <description>Heater at maximum</description>
            <latching>false</latching>
            <annunciating>true</annunciating>
        </pv>
    </component>
</config>
```

Either one can then be edited from CSS GUI

settings.ini for alarm tools

Add to settings.ini:

```
# Alarm RDB (Config Tool, Alarm Server)
org.csstudio.alarm.beast/rdb_url=jdbc:mysql://localhost/alarm
org.csstudio.alarm.beast/rdb_user=alarm
org.csstudio.alarm.beast/rdb_password=$alarm
org.csstudio.alarm.beast/rdb_schema=


# JMS Connection
org.csstudio.alarm.beast/jms_url=failover:(tcp://localhost:61616)


# Specify alarm configuration (root element)
org.csstudio.alarm.beast/root_component=Demo

# Annunciator
org.csstudio.alarm.beast.annunciator/jms_url=failover:(tcp://localhost:61616)
org.csstudio.alarm.beast.annunciator/jms_topic=demo_TALK

# Channel Access (Alarm Server and Archive Engine)
org.csstudio.platform.libs.epics/addr_list=127.0.0.1
```

Import XML Configuration

Alarm Config Tool:

```
AlarmConfigTool -pluginCustomization /path/to/settings.ini \
    -root Demo   -file /path/demo.xml   -import
```

- 'root' :
Database can contain multiple alarm configuration trees, identified by name of root element

- Consistency check:
Name of `<config name="Demo">` and command-line argument `-root Demo` must match

Run Alarm Server

Start:

```
AlarmServer -pluginCustomization /path/to/settings.ini \
            -root Demo
```

Monitor:

- Primarily just use CSS Alarm GUI
- Console output
- Send ‘debug’ message from CSS Alarm Tree
- org.csstudio.debugging.jmsmonitor

Stop:

Kill the process (Ctrl-C)

Add Alarm GUI to CSS

- Add alarm GUI plugins to CSS product
  - org.csstudio.alarm.beast
  - org.csstudio.alarm.beast.annunciator
  - org.csstudio.alarm.beast.ui
  - org.csstudio.alarm.beast.ui.alarmtable
  - org.csstudio.alarm.beast.ui.alarmtree
  - org.csstudio.alarm.beast.ui.areapanel
  - org.csstudio.utility.speech

To hear alarms

Most important

Nice

<u>The above also combined in alarm.feature</u>

- plugin_customization.ini of CSS product:

  Same org.csstudio.alarm.beast/* settings as used by Alarm Server (settings.ini)