Control System Studio Training - Alarm System Setup

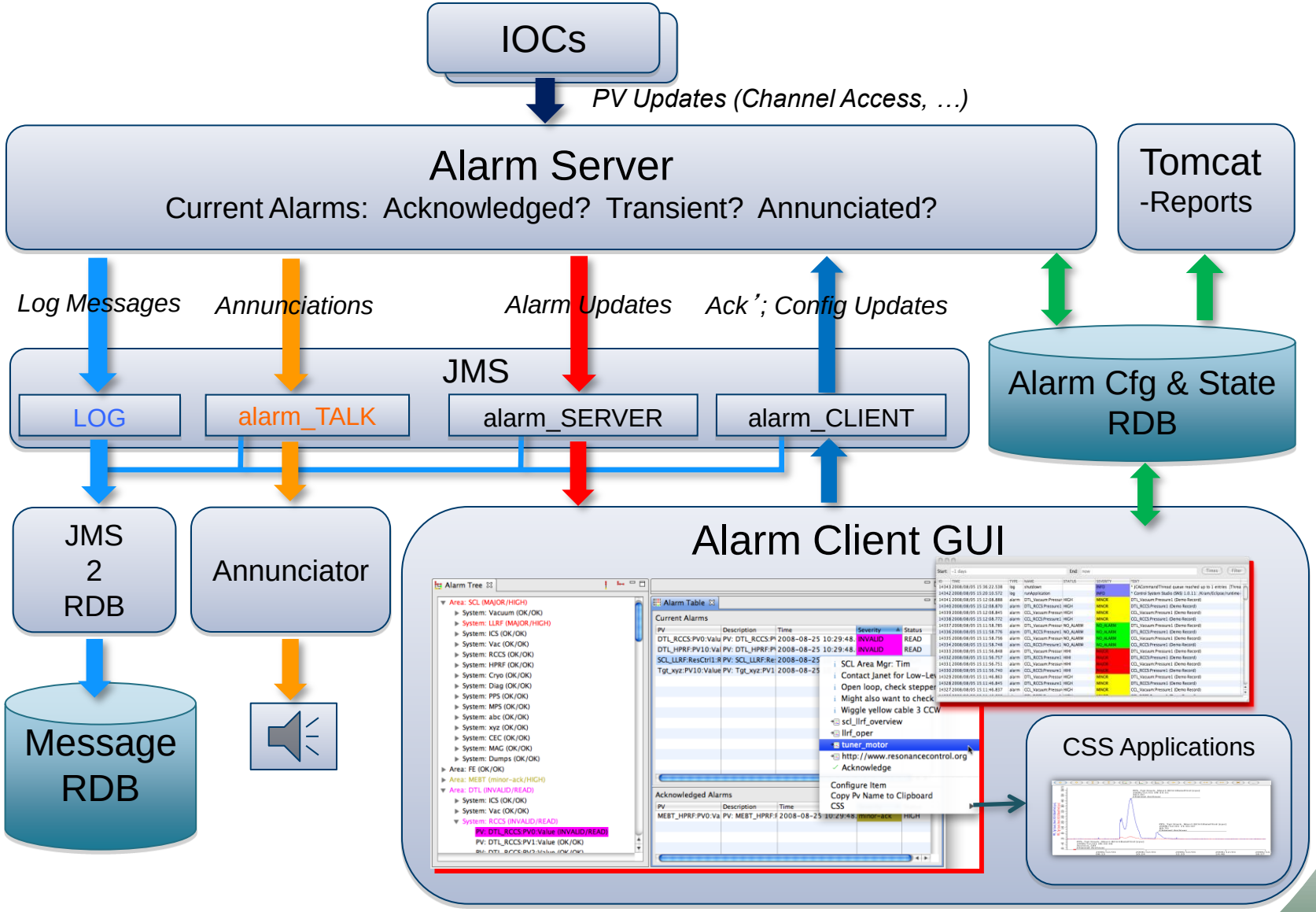
Kay Kasemir

ORNL/SNS

kasemirk@ornl.gov

2012, April at SLAC

Alarm System Overview



Initial Setup similar to archive system

1. Prepare RDB
2. Run JMS Server
3. Create and import initial configuration
4. Run Alarm Server
5. Create CSS product with
   - Authentication, Authorization
   - Alarm client GUI
6. View & edit in CSS

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 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

Java Message Server, JMS

- **Specifically: Apache ActiveMQ**
- **Start**

```
cd [activemq_install_dir]
bin/activemq start
```

  - **CheckL**

```
Netstat -an | fgrep 61616
```

- **URL for CSS clients**

```
failover:(tcp://localhost:61616)
```

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)

CSS Auth & Auth

- **Editing alarms requires**
  - **Authentication: Log in**
  - **Authorization: Being allowed to edit**
- **Can even be required to acknowledge alarms**
- **Could use LDAP, Kerberos, ….**
  - **See chapter in http://cs-studio.sourceforge.net/docbook/**

Dummy Authentication

- **Include plugins in CSS product:**
  `org.csstudio.platform.jaasAuthentication`
  `org.csstudio.platform.jaasAuthentication.ui`
- **Configure like this in plugin_customization.ini of CSS product:**

```
# Select 'Dummy' JAAS Authentication
org.csstudio.platform.jaasAuthentication/jaas_config_source=File
org.csstudio.platform.jaasAuthentication/jaas_config_file_entry=Dummy
```

Now any user and password will work

– **Except user name “fail”, which can be used for tests**

Dummy Authorization

- **Include plugin in CSS product:**
  `org.csstudio.sns.dummyAuthorization`
  - **and include only this *authorization* plugin!**

Now any user and password will work

– **Still needs to log on, though, but any user name and password will be accepted**

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

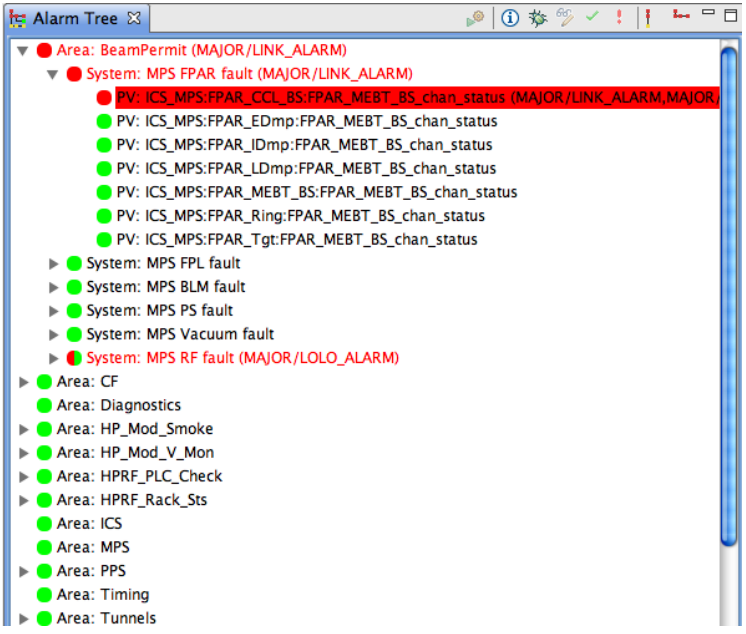
Nice

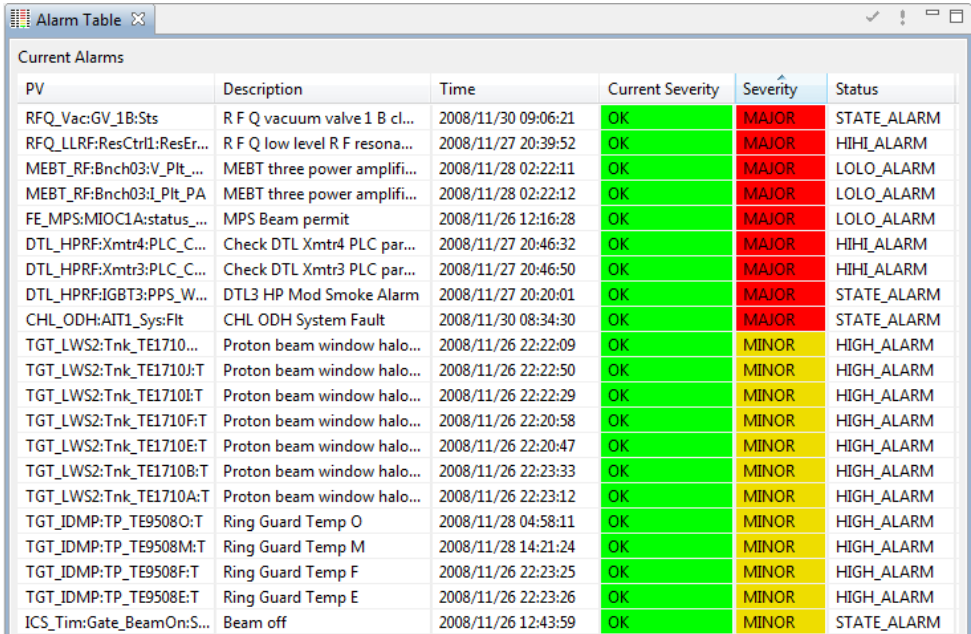
- plugin_customization.ini of CSS product:

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

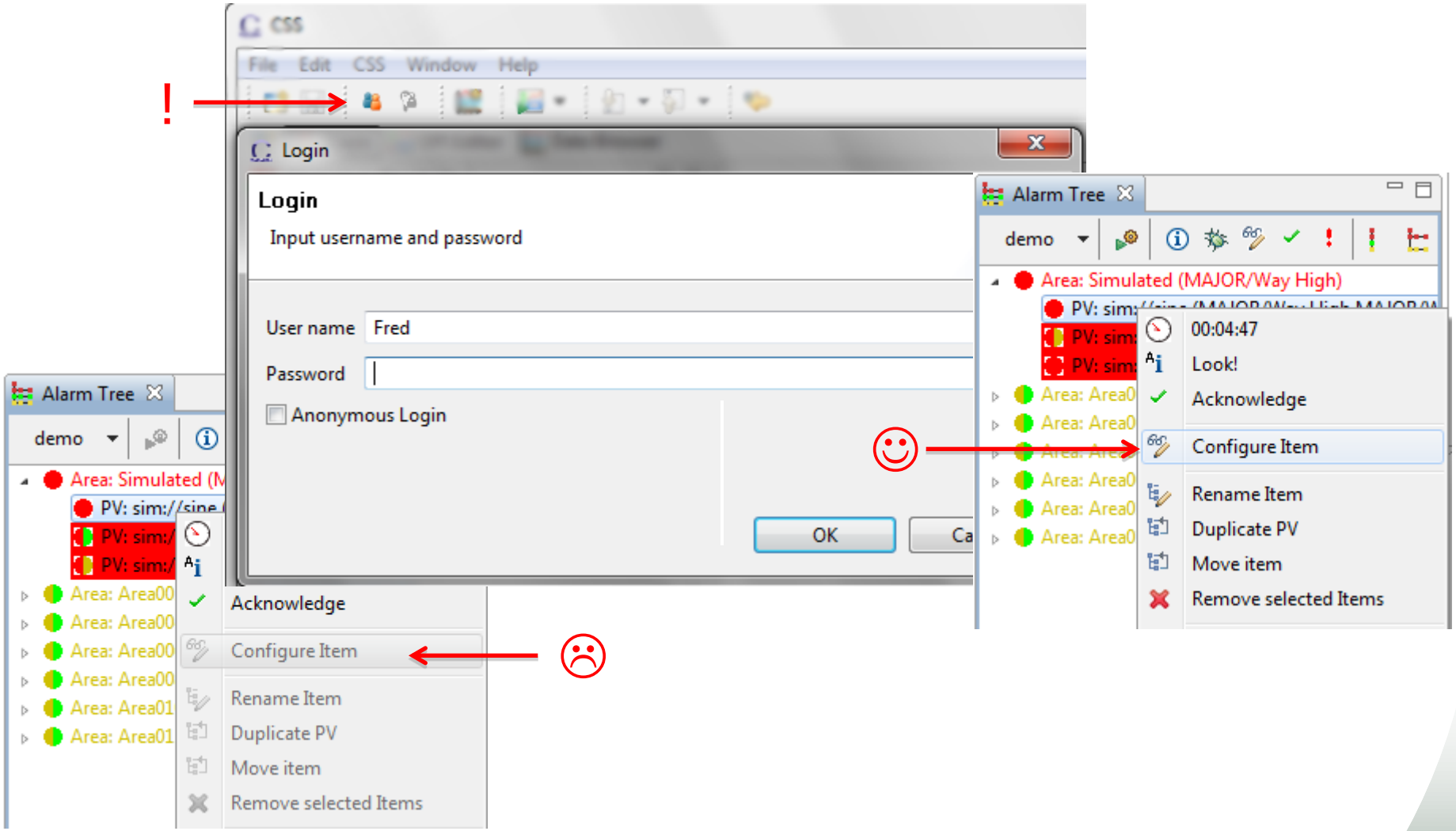
Use CSS Alarm Tree, Alarm Table, …

- **Open Menu CSS/Alarm/…**
  - **Alarm Tree, Table, Annunciator, Area Panel**
- **In alarm tree, use context menu to add to configuration**
  - **After log in**





Authorization Required



Only authorized users can change the configuration

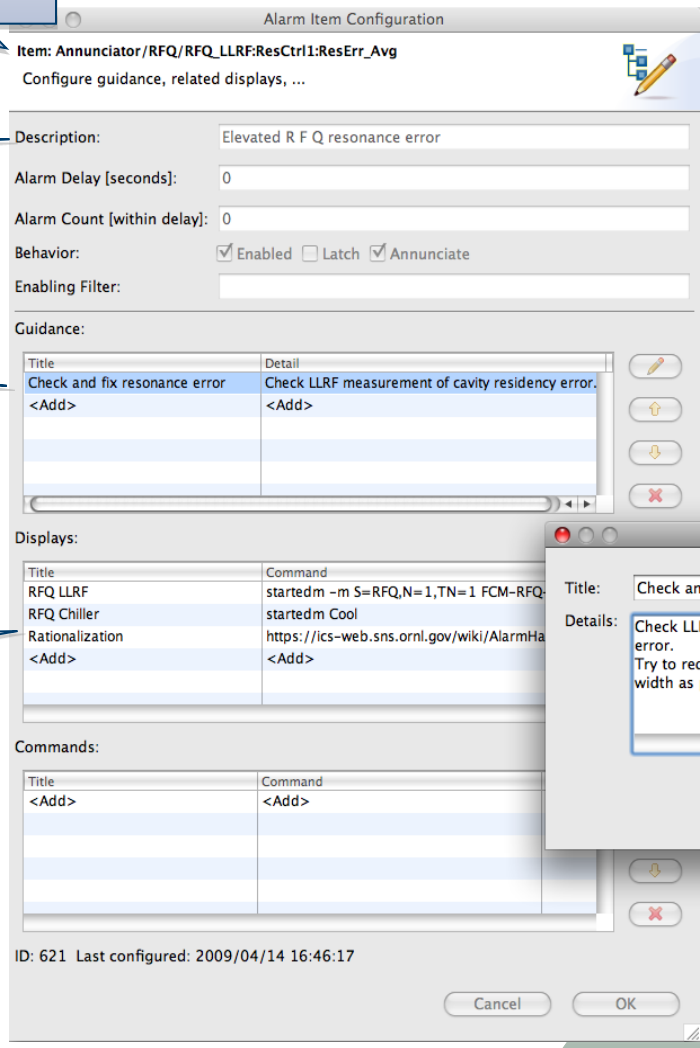
PV Configuration

Full Path to PV in Alarm Tree

Description: Also used for Annunciation

Guidance: Simple Title & Detail that should help operators handle the alarm

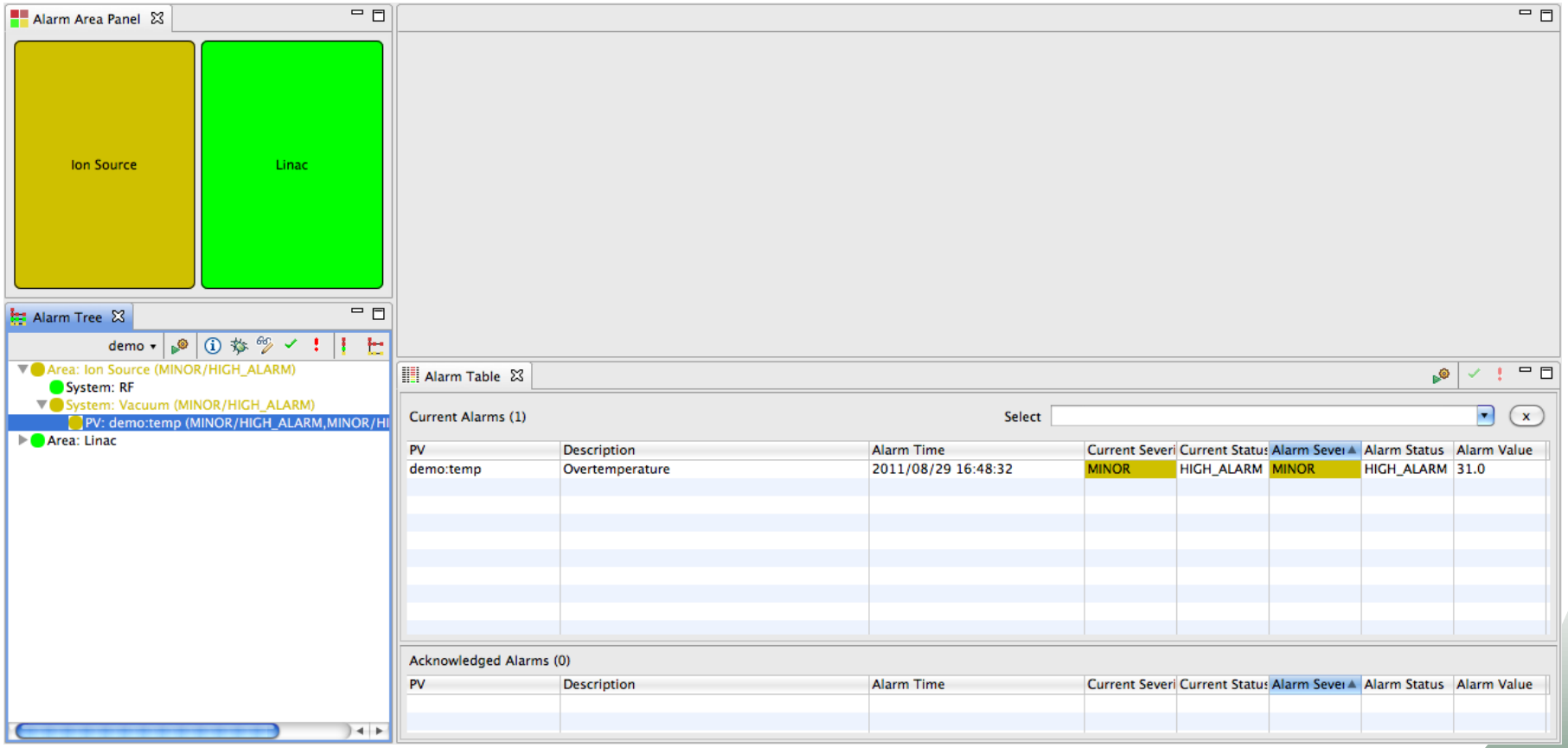
Display Link Options:
/CSS/path/to/display.opi
http://www.google.com
https://some.host.org
scriptname arg1 arg2



See online help for more details

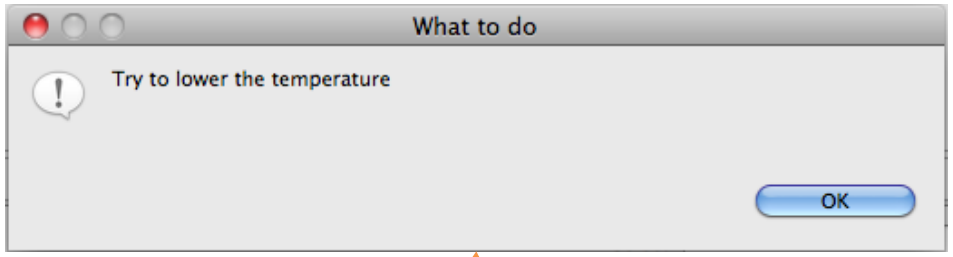
An Alarm Triggers…

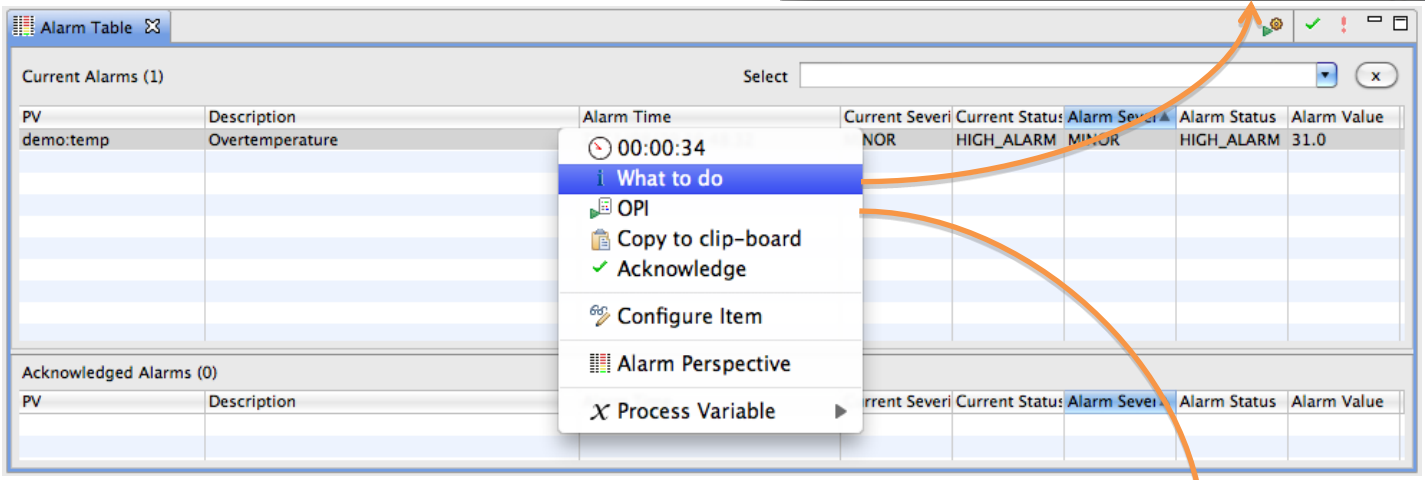
Some operators prefer just the Alarm Table, others also like to look at Area Panel or Tree View



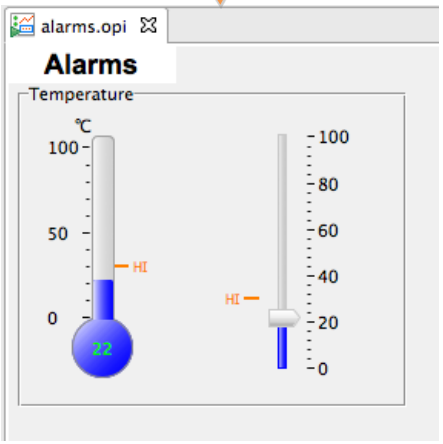
Context menu of Alarm

- Guidance

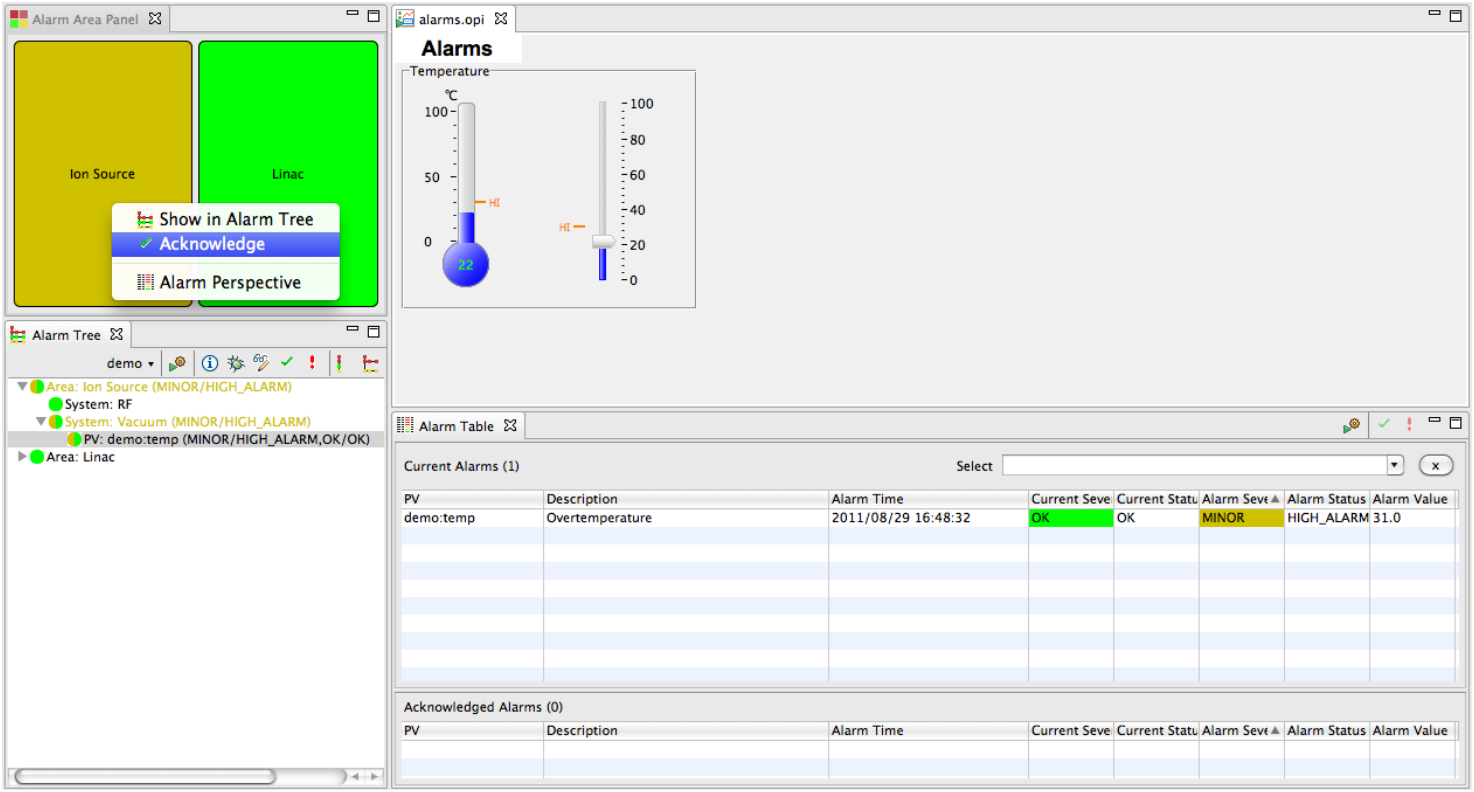




- Links to related OPIs

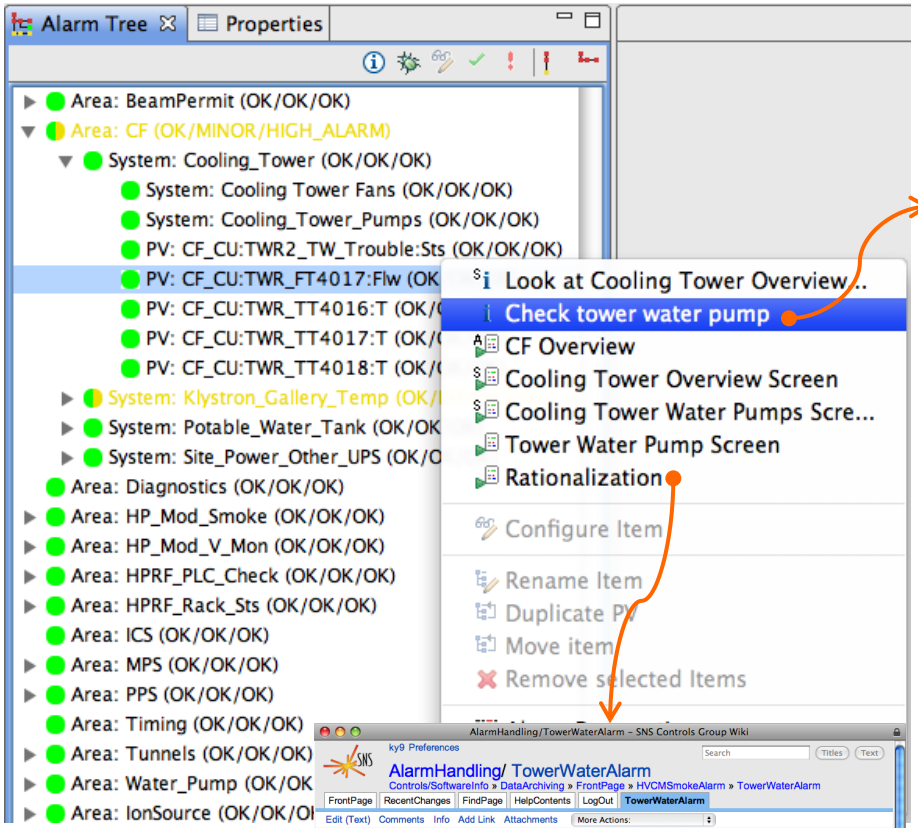


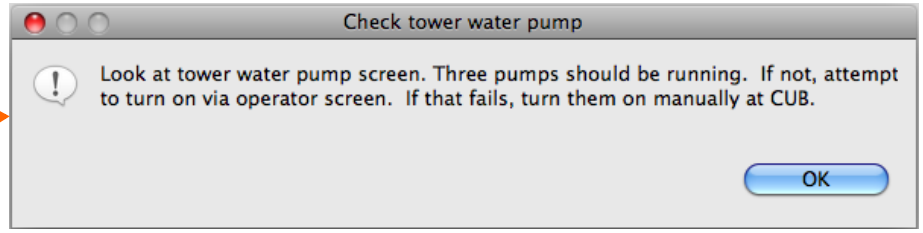
The Problem is fixed, Alarm clears



- **By default, the alarm system latches alarms**
  - **"Current" severity of PV is OK, but MINOR alarm is remembered until alarm is ✔ Acknowledged**

Guidance, Related Displays, Commands





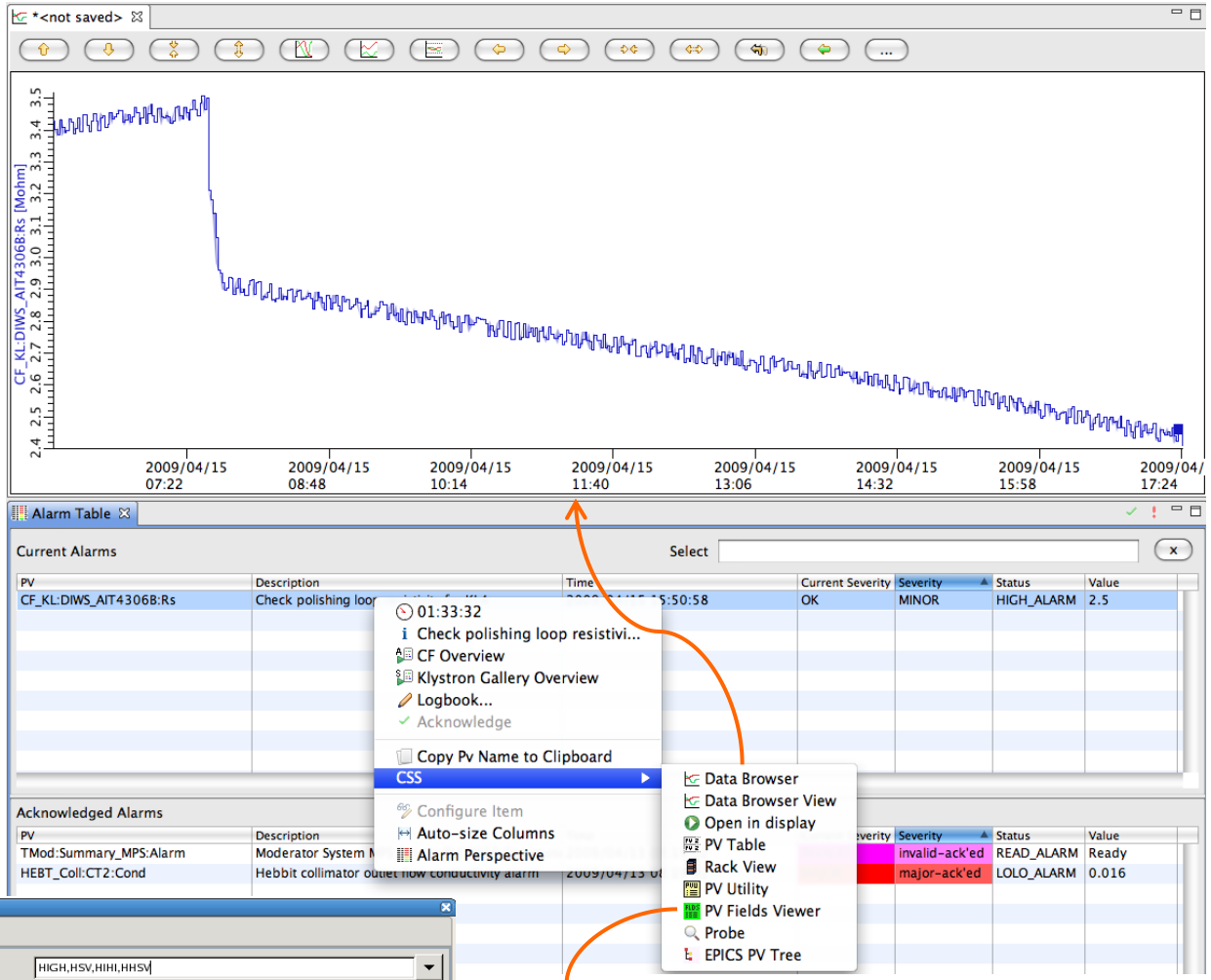
- ☑ **Basic Text**
- ☑ **Open EDM/OPI screen**
- ☑ **Open web page**
- ☑ **Run ext. command**

Hierarchical:
Including info of parent entries

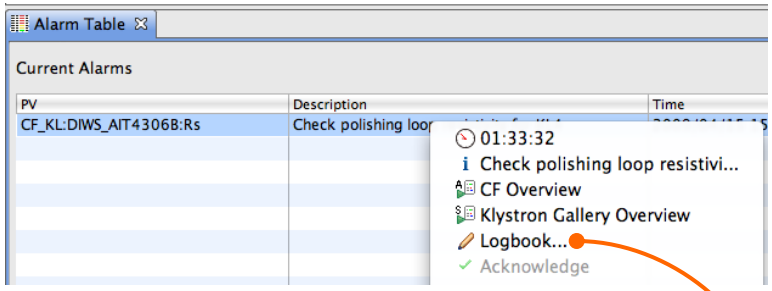
Merges Guidance etc. from all selected alarms

CSS Context Menus Connect the Tools

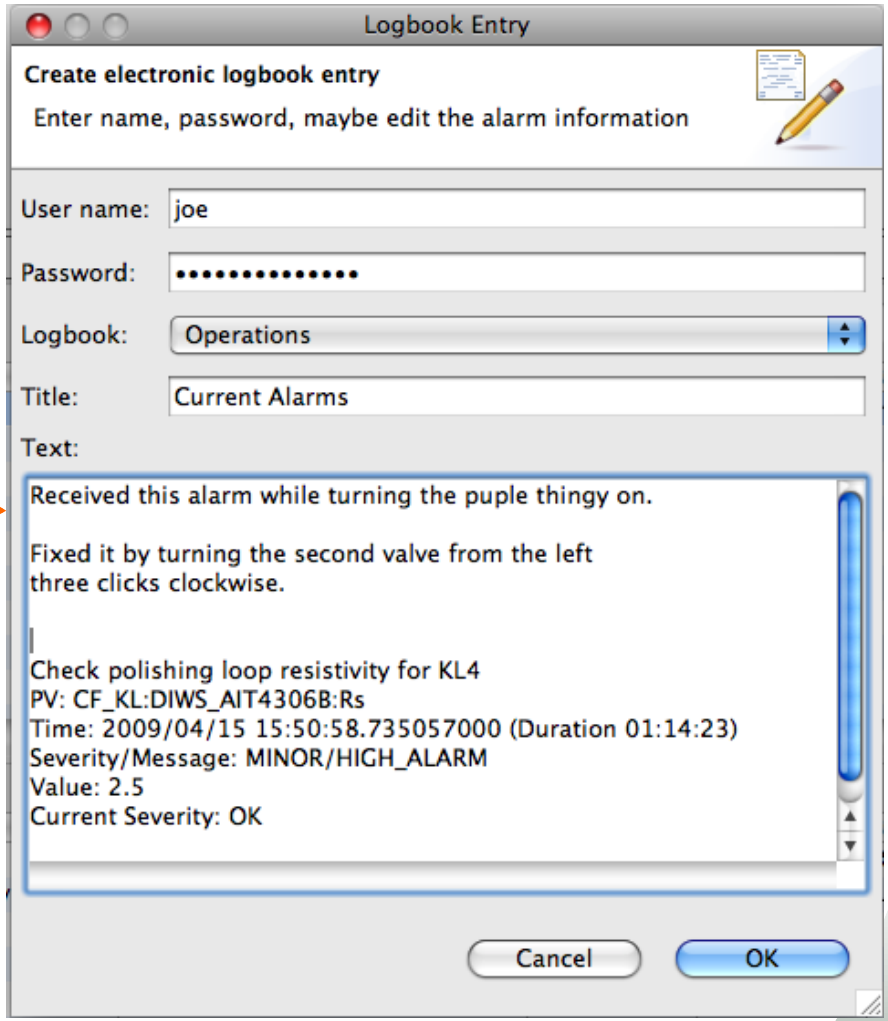
Send alarm PV to any other CSS PV tool



E-Log Entries



- **"Logbook" from context menu creates text w/ basic info about selected alarms. Edit, submit.**



- **Pluggable implementation, not limited to Oracle-based SNS ELog**