

CSS – Control System Studio

Workshop Report

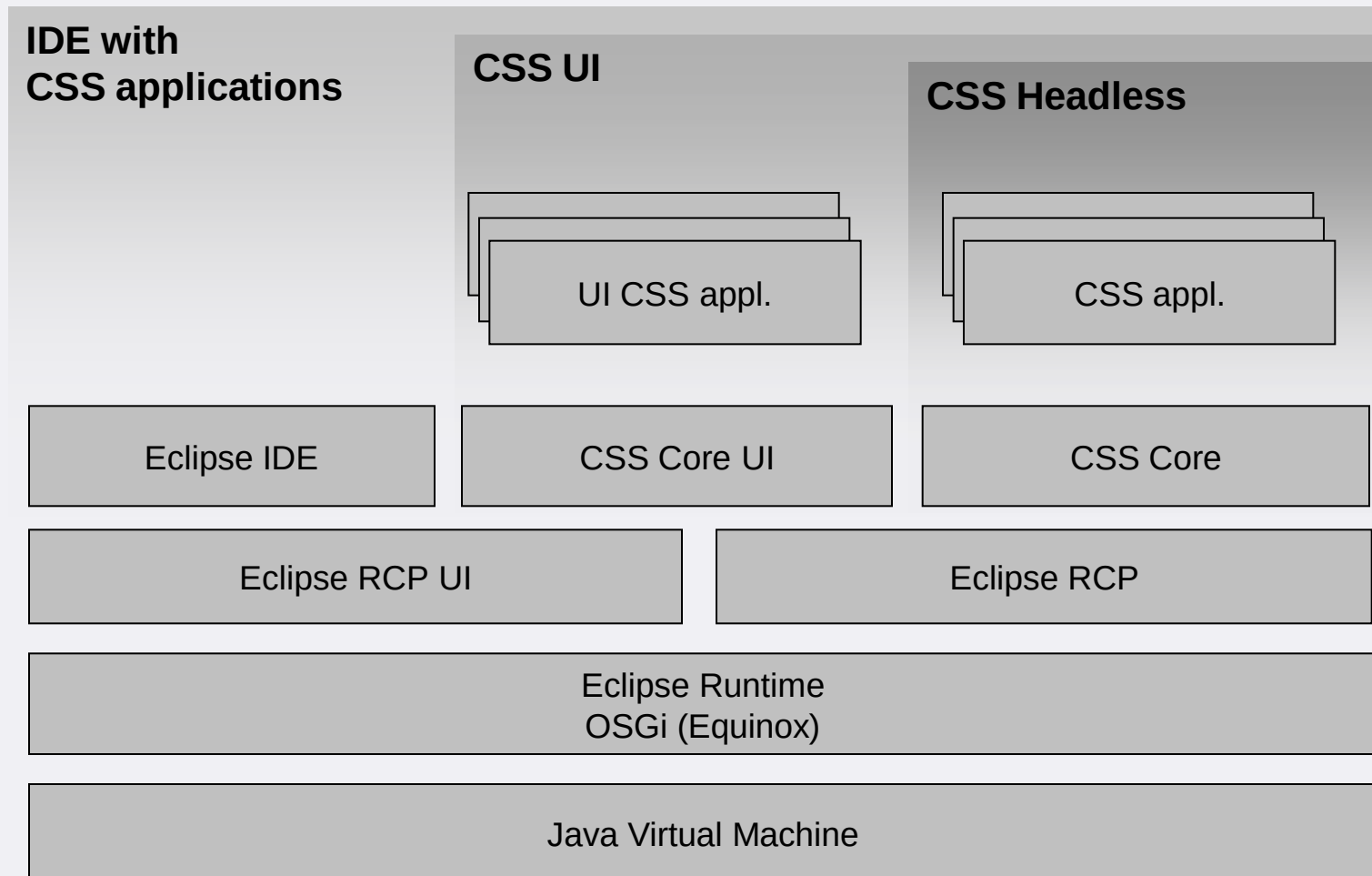
PCaPAC, 5-8 October 2010

Jan Hatje (DESY / MKS-2)

Overview

- Eclipse and OSGI technology
- CSS Core features
- Hudson Build Server
- Database Creation Tool
- Synoptic Display Studio
- Databrowser

CSS Design

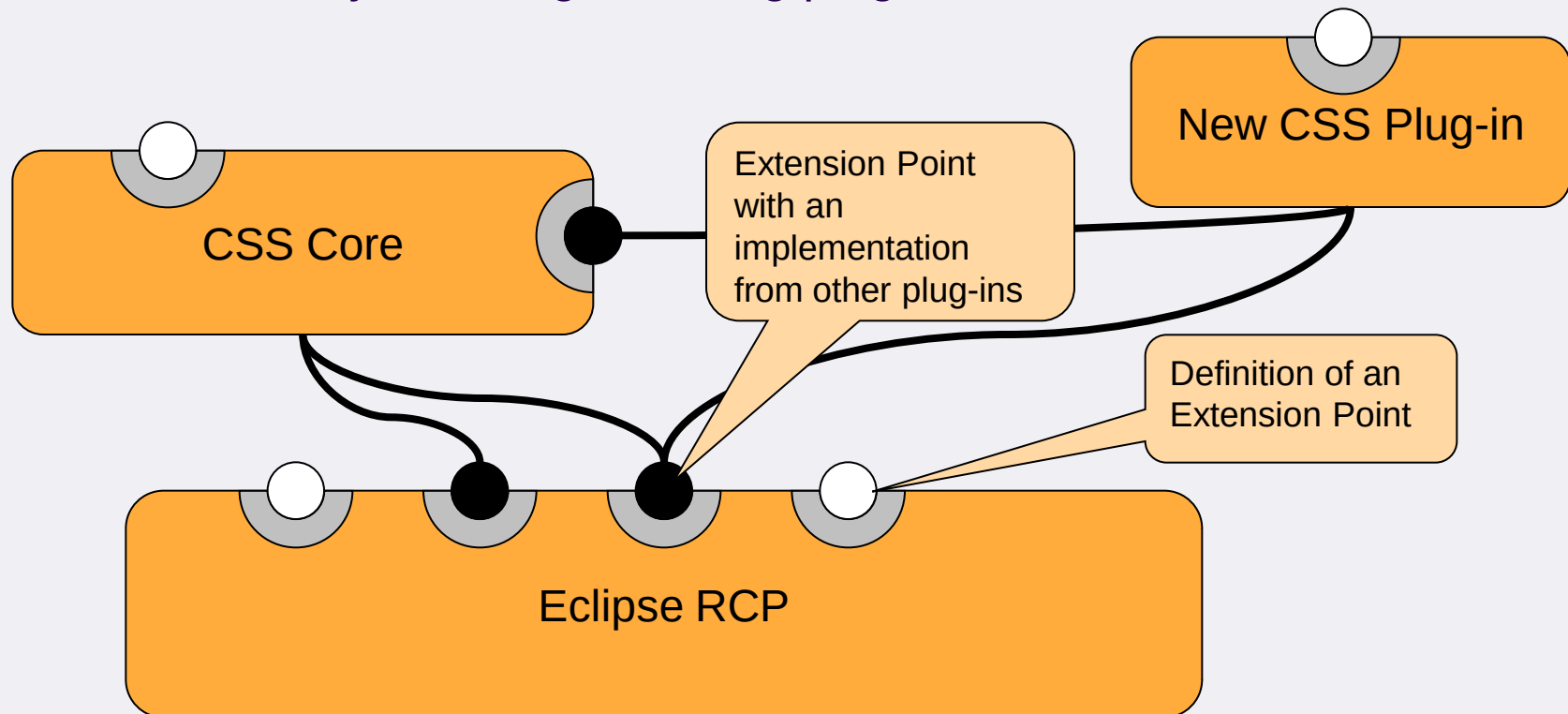


Eclipse

- Written in Java → operation system independent
- Based on Equinox an OSGi implementation
- OSGi is an open standard organization 'Open Services Gateway initiative'
- Plug-in technology
- Using SWT/JFace for UI
- Provides a Rich Client Platform (RCP) for general applications
- Add new functionality by extension points
- Many helpful features (Drag and Drop, Help System, Update site, Language support, ...)

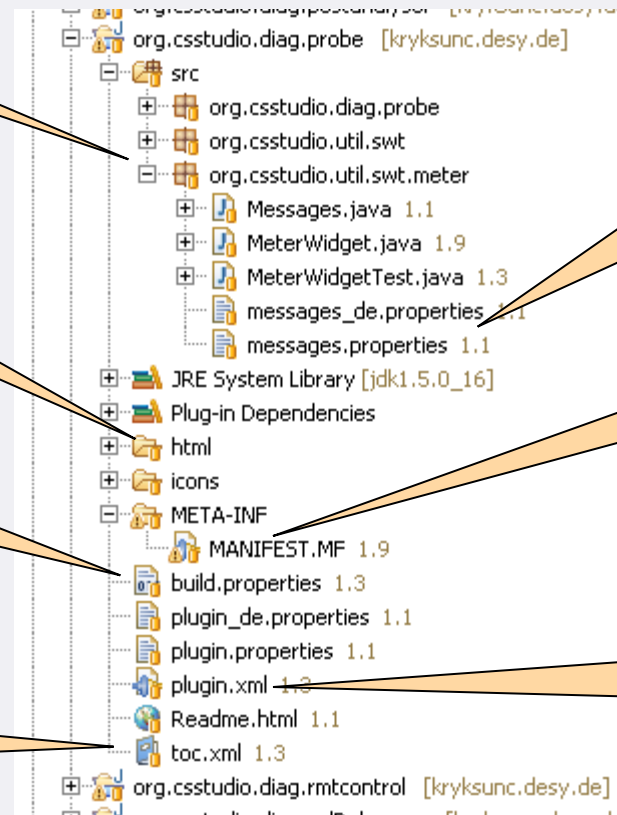
Eclipse Extension Points

- A plug-in can use extension points to add new functionality
- An extension point is defined by a XML-file and usually a Java interface
- Not necessary to change existing plug-ins



Eclipse Plug-in

An Eclipse plug-in is a set of definition- and java files



Java classes

HTML files for
help pages

Settings for build
process of the
plug-in

Content structure
of plug-in help files

Strings in UI are hold in
separated files for
different languages

File for OSGi bundle
configuration

File for Eclipse related
configuration (on top of
OSGi bundle)

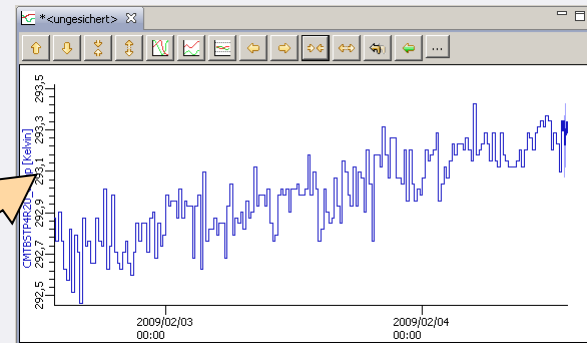
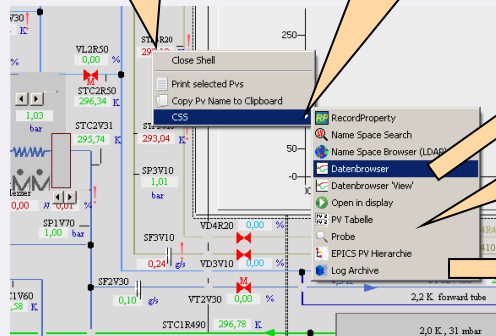
Object Contribution, Drag and Drop

- Eclipse provides Object Contribution and Drag and Drop for data types
- Plug-ins define data types they can handle
- Eclipse manages context menu and drop targets depending on existing plug-ins

Context menu on right mouse click on a process variable

Menu "CSS" shows all applications applicable for process variable

Call applications and send selected process variable



Console - JMS Log archive viewer

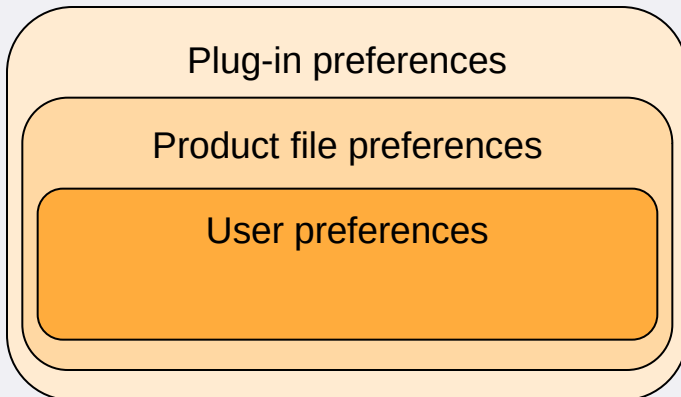
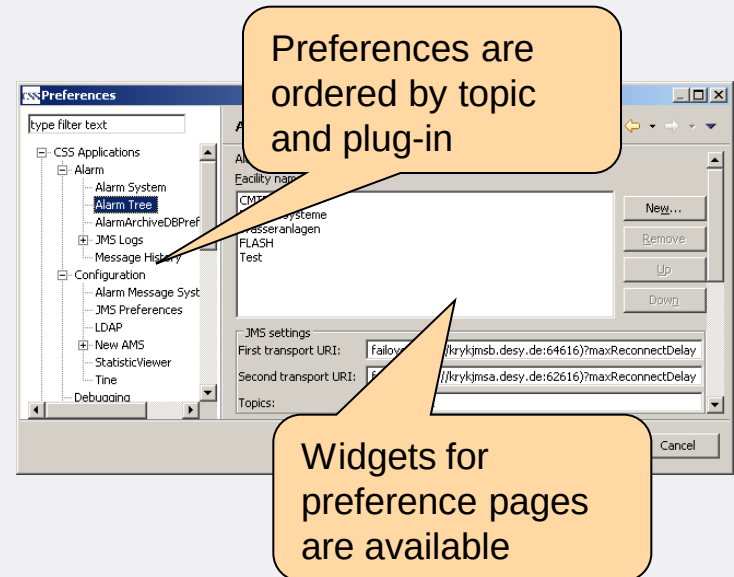
Zeitraum: 3 Tage Woche Benutzer: Experte von 2009-02-03 13:40:35.221 bis 2009-02-04 13:40:35.221 Anz

ACK	TYPE	EVENTTIME	NAME	VALUE	H
	event	2009-02-04 11:32:22.196	MTPI709_ai	4.94247	mtkg
	event	2009-02-04 11:32:31.185	MTPI709_ai	4.94186	mtkg
	event	2009-02-04 11:01:25.989	MTPI709_ai	4.94186	mtkg
	event	2009-02-04 06:08:51.605	MTPI709_ai	4.94064	mtkg
	event	2009-02-04 06:08:50.591	MTPI709_ai	4.94024	mtkg
	event	2009-02-04 03:20:45.281	MTPI709_ai	4.944	mtkg
	event	2009-02-04 03:20:44.249	MTPI709_ai	4.92823	mtkg
	event	2009-02-03 21:43:46.230	MTPI709_ai	4.95041	mtkg
	event	2009-02-03 21:43:45.612	MTPI709_ai	4.95068	mtkg
	event	2009-02-03 20:36:58.948	MTPI709_ai	4.95224	mtkg
	event	2009-02-03 20:36:57.900	MTPI709_ai	4.95134	mtkg
	event	2009-02-03 19:27:15.300	MTPI709_ai	4.95621	mtkg
	event	2009-02-03 19:27:14.258	MTPI709_ai	4.95621	mtkg
	event	2009-02-03 19:22:42.014	MTPI709_ai	4.95621	mtkg
	event	2009-02-03 19:22:40.962	MTPI709_ai	4.95417	mtkg

Preferences

Eclipse provides a Preference Store for all plug-in preferences

- 1) Define plug-in preferences
- 2) Create an initializer for default values (optional)
- 3) Create preference page (optional)



- Several levels for preference definition
- Easy to administer preference settings
- Different preferences for different CSS installations by *.ini files

Views, Editors and Perspectives

Workbench: Users can arrange views and editors

File types are connected to dedicated editors

Detached view outside of CSS workbench

Perspective: Predefined and individual arrangements of views and editors

Editor: connected to workspace resources

View: basic UI part

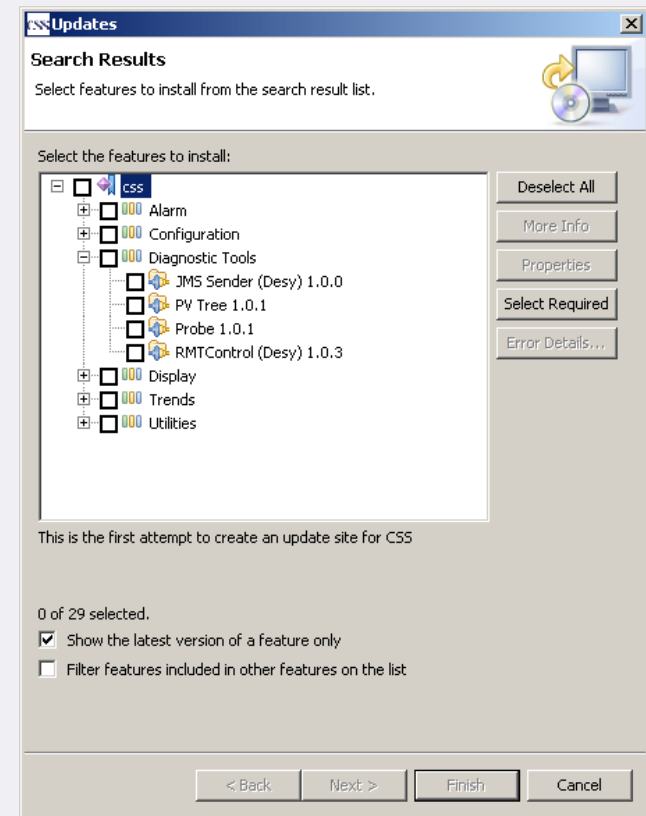
The screenshot displays the CSS Standard workbench. The main workspace shows a schematic diagram of a beamline with components like magnets (VH5W67, VH5W62, SF3V110, etc.) and a transfer section. A callout points to the 'Ausbauklappe 1,4bar Dichtung prüfen' (Check 1.4bar seal) component. Below the workspace is a 'JMS Log archive viewer' showing a table of events.

ACK	TYPE	EVENTTIME	NAME	VALUE	
event	2009-02-04 14:14:42.131	TMTHLIR26A_ao	19.92	tdn	
event	2009-02-04 14:14:42.131	TMTHLIR26A_p_cak	45	tdn	
event	2009-02-04 14:14:42.131	TMTHLIR26A_pid	50.0412	tdn	
event	2009-02-04 14:12:42.189	TSMSTAT3_mbbd	0000 0000 0000...	tdn	
event	2009-02-04 14:12:42.189	TSMSTAT3_mbbd	0000 0000 0000...	tdn	
event	2009-02-04 14:11:15.895	TMTHLIR26_calc	0	tdn	
event	2009-02-04 14:11:15.895	TMTHLIR26_ao	0	tdn	
event	2009-02-04 14:11:15.895	TMTHLIR26A_calc	19.92	tdn	
event	2009-02-04 14:11:15.895	TMTHLIR26A_ao	0	tdn	

On the right, the 'Widget Properties' panel shows settings for a 'Record' widget, including 'Name', 'Background Color', 'Position', and 'Namespace Browser'. A detached view titled 'Record Property' is also visible, showing a table of field names and values.

Update Site

- Install additional features for CSS
- Update existing plug-ins
- Eclipse checks for dependencies of plug-ins
- Easy to create an update site for your own applications
- Update site can be located on web- or file server
- Remote update → next presentation



CSS Core

- Data types for control systems (e. g. process variable, time stamp) to use Eclipse object contribution and drag and drop
- Logging Service on basis of Log4J
- Different destinations for log messages (console, file, JMS)
- Common libraries for JDBC, JMS, EPICS, ...
- A service layer for database and messaging is under development

CSS Core - Logging

The screenshot shows the 'Preferences' dialog box for the CSS Core. The 'Logging Settings' tab is active, showing options to use the CSS Console Appender, File Appender, and JMS Appender. The 'File Appender' is selected. The 'File Logging Settings' tab is also visible, showing the Log-Level set to DEBUG, the Pattern set to '%d %d %-5p [%t] %c: %m%n', the Log file set to 'logs/cssLog.log', and the Backup index set to 10.

Logging Settings

- ☒ Use CSS Console Appender
- ☒ Use CSS File Appender
- ☐ Use CSS JMS Appender

File Logging Settings

- Log-Level: ☒ DEBUG, ☐ INFO, ☐ WARN, ☐ ERROR, ☐ FATAL
- Pattern: %d %d %-5p [%t] %c: %m%n
- Log file: logs/cssLog.log
- Backup index: 10

Console

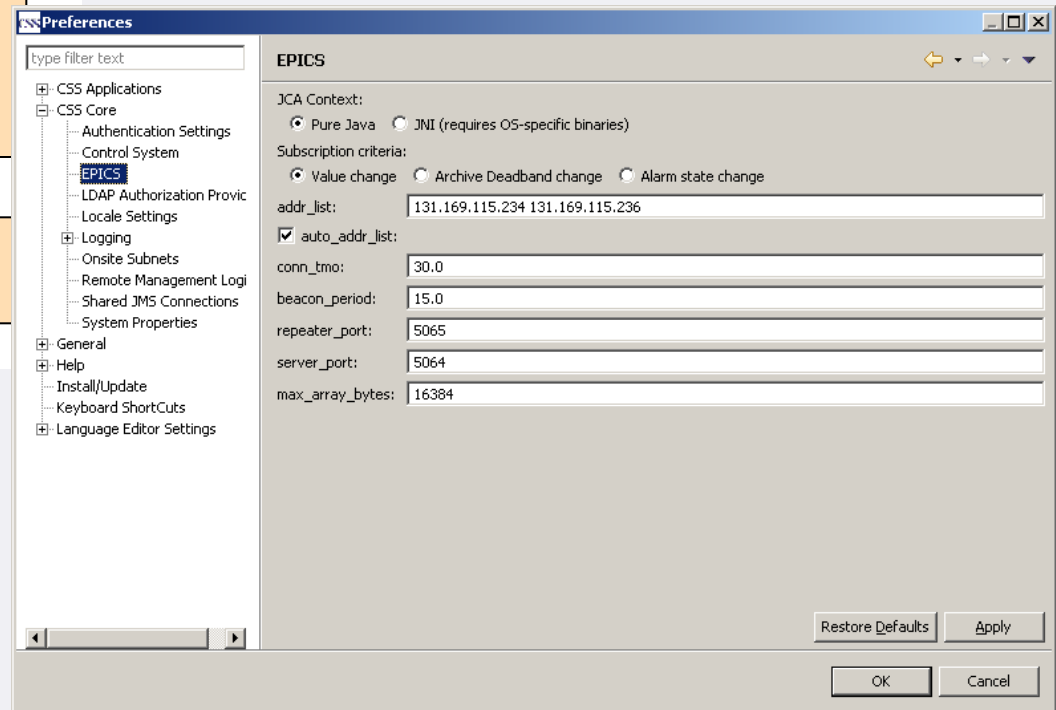
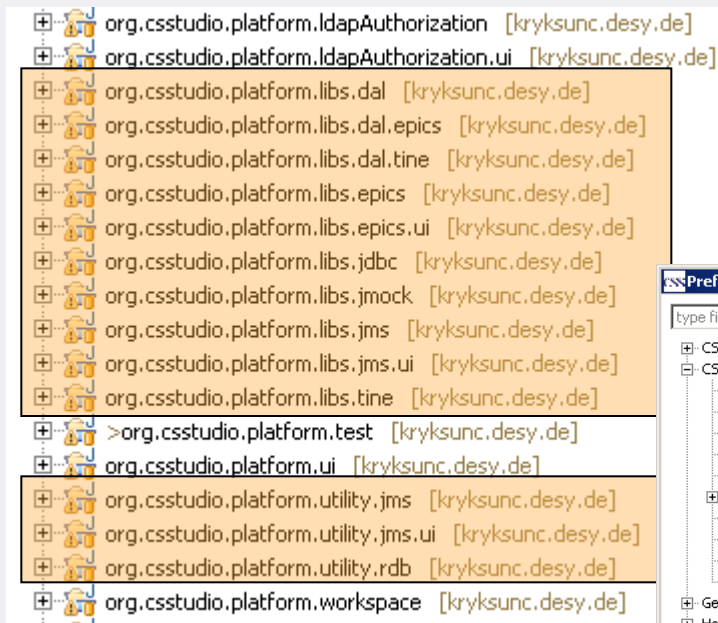
```

2009-10-28 08:12:11,392 2009-10-28 08:12:11,392 INFO [main] org.csstudio.alarm.table.jms.JmsAlarmMessageReceiver: Initialize JMS connection with topics: ALARM,ACK
2009-10-28 08:12:35,877 2009-10-28 08:12:35,877 DEBUG [ActiveMQ Session Task] org.csstudio.alarm.table.jms.JmsAlarmMessageReceiver: Received map message: EVENTTIME: 2009-10-28 08:12:35.872 NAME: TSM
2009-10-28 08:12:35,893 2009-10-28 08:12:35,893 DEBUG [main] org.csstudio.alarm.table.ui.messageable.MessageTableContentProvider$1: Add Message, Number of Msg in Model: 1; Number of Msg in Table: 1
2009-10-28 08:12:43,080 2009-10-28 08:12:43,080 DEBUG [ActiveMQ Session Task] org.csstudio.alarm.table.jms.JmsAlarmMessageReceiver: Received map message: EVENTTIME: 2009-10-28 08:12:43.075 NAME: TSM
2009-10-28 08:12:44,127 2009-10-28 08:12:44,127 DEBUG [main] org.csstudio.alarm.table.ui.messageable.MessageTableContentProvider$3: Remove Message, Number of Msg in Model: 2; Number of Msg in Table: 1
2009-10-28 08:12:44,127 2009-10-28 08:12:44,127 DEBUG [main] org.csstudio.alarm.table.ui.messageable.MessageTableContentProvider$1: Add Message, Number of Msg in Model: 2; Number of Msg in Table: 1
2009-10-28 08:12:44,127 2009-10-28 08:12:44,127 DEBUG [main] org.csstudio.alarm.table.ui.messageable.MessageTableContentProvider$1: Add Message, Number of Msg in Model: 2; Number of Msg in Table: 2
  
```

```

121         .parseInt(fontDataString[2]), Integer
122         .parseInt(fontDataString[1]));
123     } catch (Exception e) {
124         CentralLogger.getInstance().error(this, "Cannot create font, " + e.getMessage());
125     }
126     if (font[0] == null) {
127         font[0] = new FontDataString();
  
```

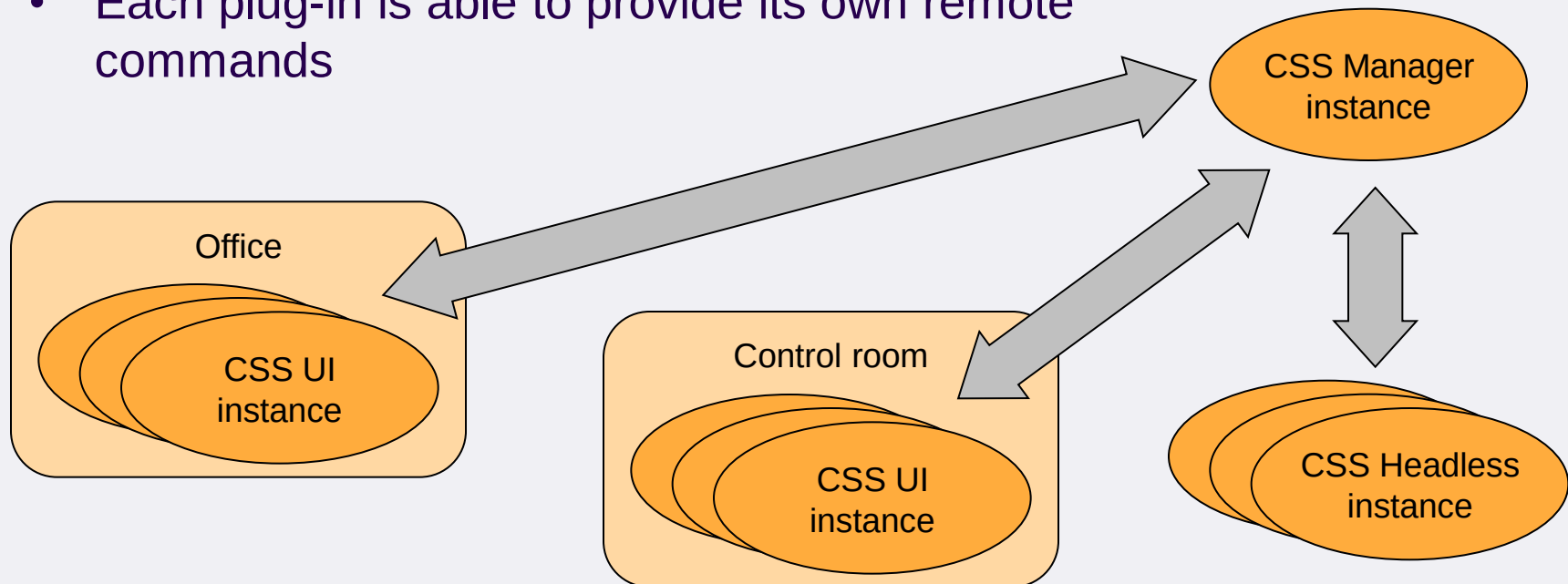
CSS Core – Common Libraries



Remote Management

- Management of CSS instances

- All remote features are located in separated plug-ins → CSS can easily be built with or without remote management
- CSS Core provides common remote commands (e.g. update plug-in, write preference, ...)
- Each plug-in is able to provide its own remote commands



Remote Management

The screenshot shows the 'Remote Eclipse RCP Management' application. It features a 'Contacts' tree on the left, a 'Selected Contacts' list in the center, and a 'Provisioning Editor' on the right. The 'Provisioning Editor' contains a table for user preferences.

Available commands of selected instance: Points to the list of commands for the selected instance 'testb@kryloxmpp.desy.de/ecf.+PysBTW'.

Available remote commands: Points to the 'Selected Contacts' list.

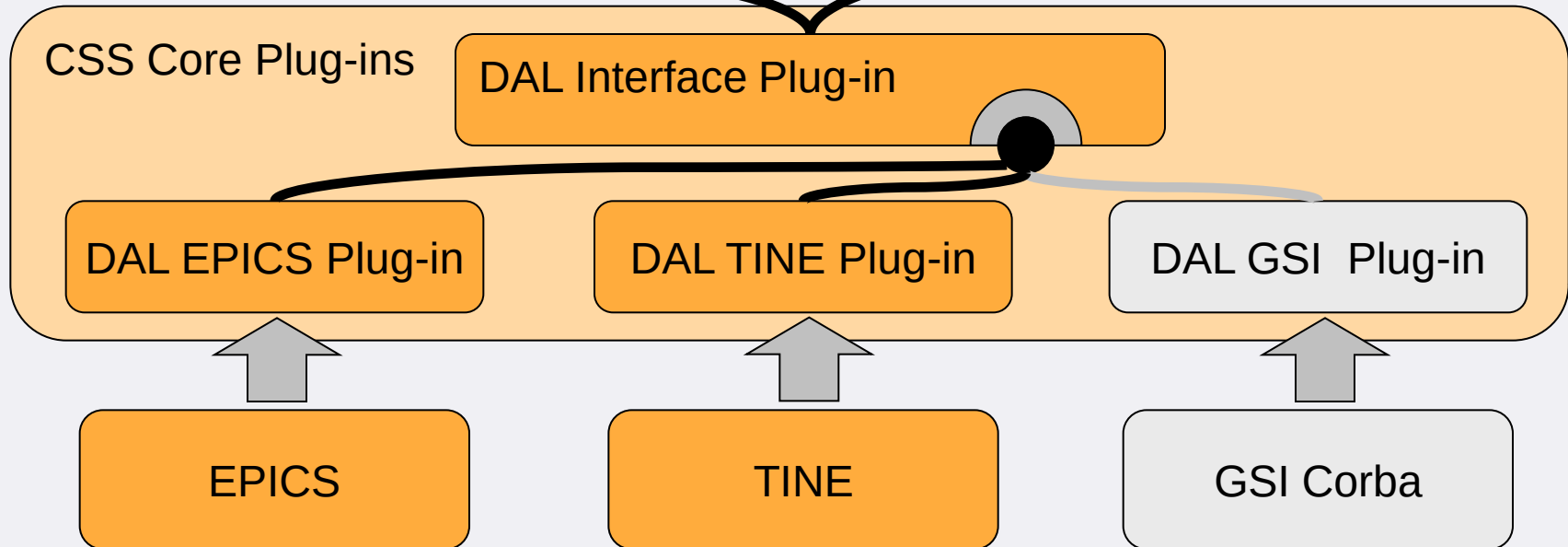
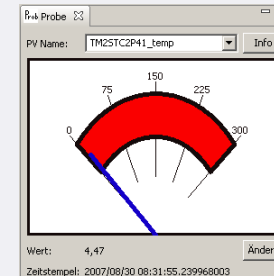
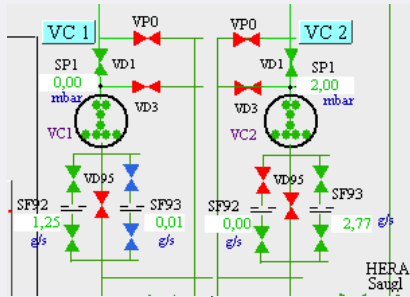
Editor to handle specific remote command: Points to the 'Provisioning Editor' table.

All (online and offline) instances: Points to the 'Contacts' tree.

Selected instances to be managed: Points to the 'Selected Contacts' list.

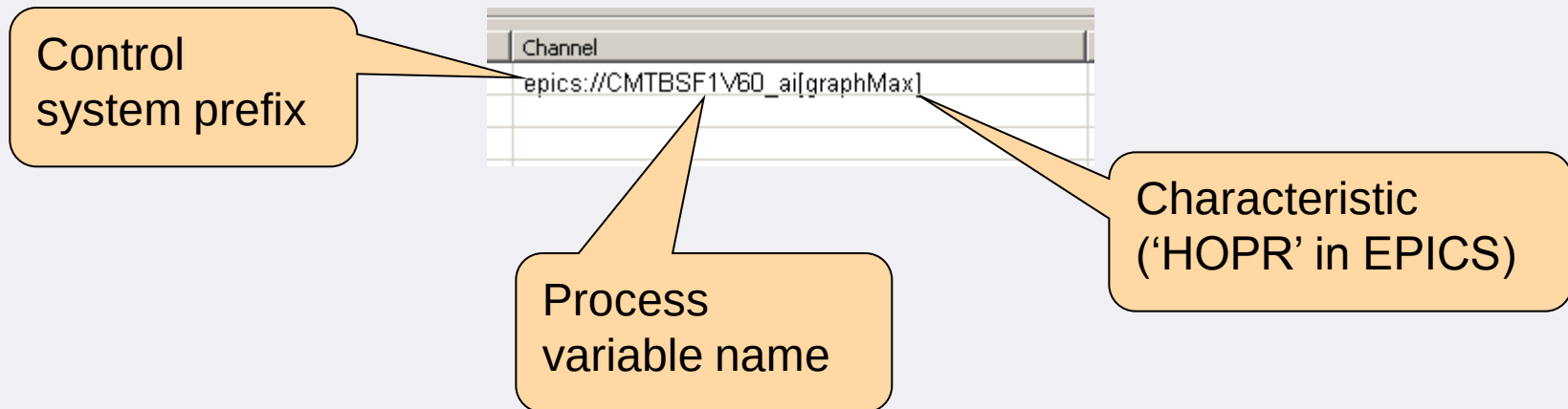
Key	Remote value	Transfer	Local value
org.remotercp.login/lastElement	kryloxmpp.desy.de		
org.remotercp.login/server/kryloxmpp.desy.de			
org.remotercp.login/username	testb		

Data Access Layer (DAL)



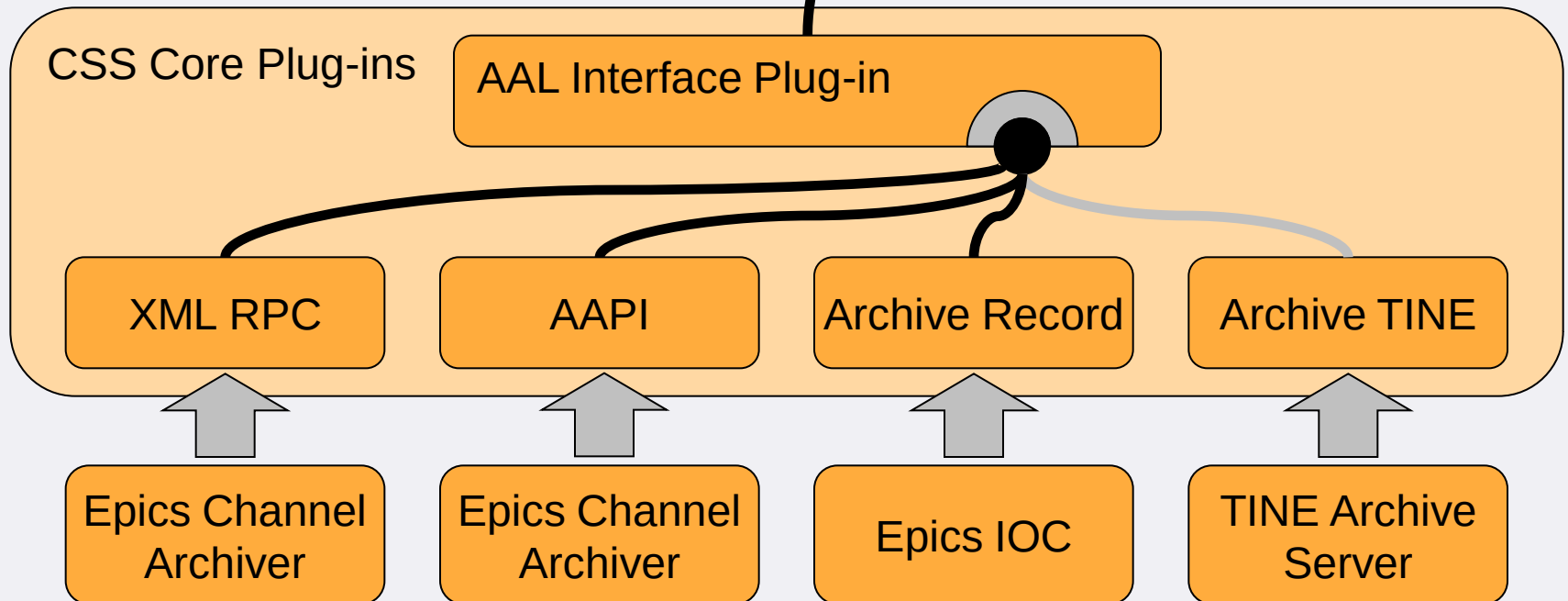
Data Access Layer (DAL)

- Extension Point for implementations of control systems
- Prefix for control system type
- Caching of characteristics (no separated connection)
- Connection state (connected, disconnected, failed)
- Cosylab will create a example DAL plug-in for serial communication



Archive Access Layer (AAL)

- Defines an extension point and interfaces for archive access



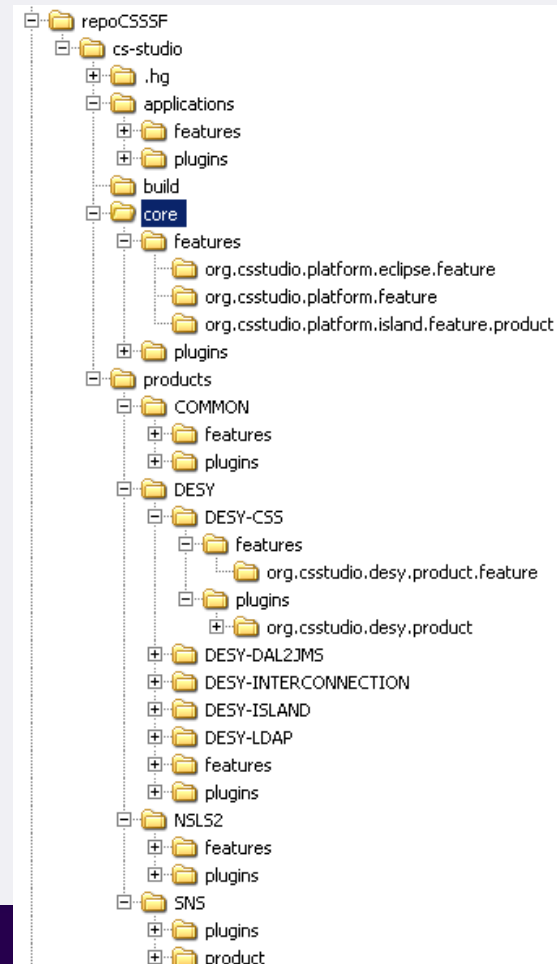
CSS Repository

Mercurial repository on SourceForge:

`ssh://YOUR_ACCOUNT_NAME@cs-studio.hg.sourceforge.net/hgroot/cs-studio/cs-studio`

Sorce Code structure:

- Core – CSS Platform features and plugins
- Products – Plugins and features for site specific products
- Applications – Plugins and features for CSS
- Build – Script files for CSS product build

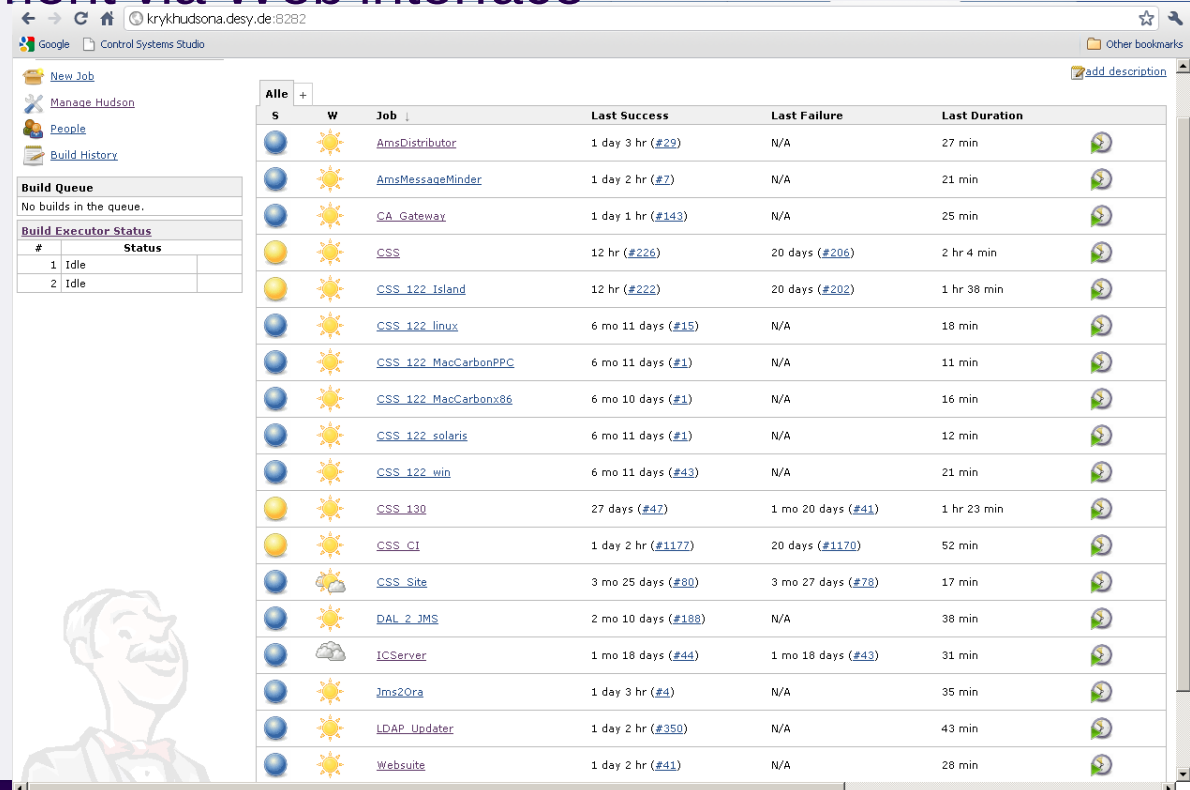


Headless Build

- Control Headless Build with Hudson
- Build is started periodically or by Mercurial commit
- Ant scripts for headless build are predefined in eclipse
- Check CSS plug-ins for compile errors
- Run JUnit tests (also in plug-in environment)
- Run integration tests
- Create a CSS product for deployment
- Notify developers of build result

Hudson

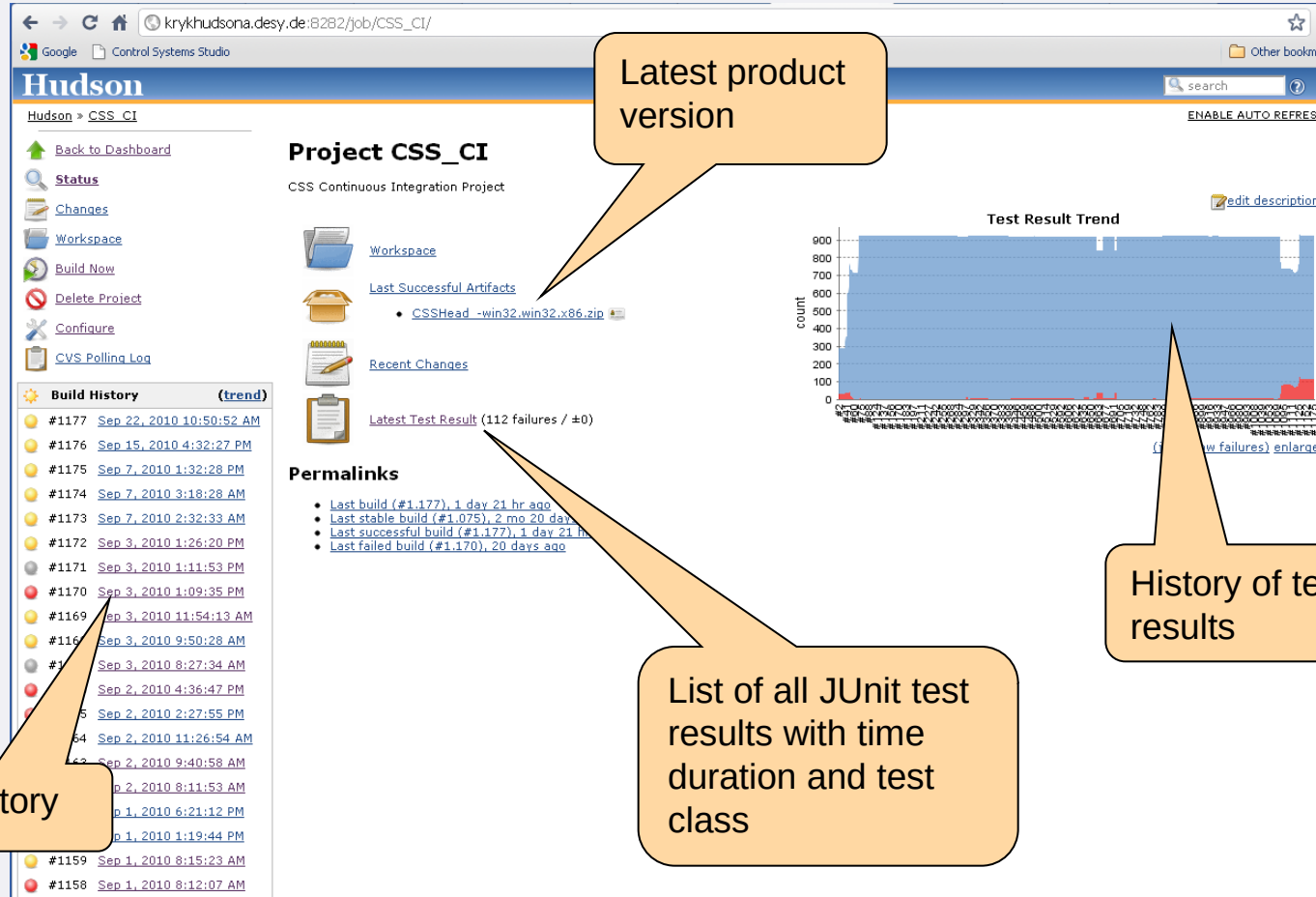
- Continuous integration tool implemented in Java
- Runs in a servlet container like Tomcat
- Access and management via Web interface
- Starts Ant script for build process
- Many plugins available e. g. for Mercurial
- E-Mail notification



The screenshot shows the Hudson web interface at the URL `krykudsona.desy.de:8282`. The interface includes a sidebar with navigation links: [New Job](#), [Manage Hudson](#), [People](#), and [Build History](#). Below these links, there are sections for **Build Queue** (showing 'No builds in the queue.') and **Build Executor Status** (showing 1 Idle and 2 Idle executors). The main content area displays a table of build jobs.

S	W	Job	Last Success	Last Failure	Last Duration
Blue ball	Sun	AmsDistributor	1 day 3 hr (#29)	N/A	27 min
Blue ball	Sun	AmsMessageMinder	1 day 2 hr (#7)	N/A	21 min
Blue ball	Sun	CA_Gateway	1 day 1 hr (#143)	N/A	25 min
Yellow ball	Sun	CSS	12 hr (#226)	20 days (#206)	2 hr 4 min
Yellow ball	Sun	CSS_122_Island	12 hr (#222)	20 days (#202)	1 hr 38 min
Blue ball	Sun	CSS_122_linux	6 mo 11 days (#15)	N/A	18 min
Blue ball	Sun	CSS_122_MacCarbonPPC	6 mo 11 days (#1)	N/A	11 min
Blue ball	Sun	CSS_122_MacCarbonx86	6 mo 10 days (#1)	N/A	16 min
Blue ball	Sun	CSS_122_solaris	6 mo 11 days (#1)	N/A	12 min
Blue ball	Sun	CSS_122_win	6 mo 11 days (#43)	N/A	21 min
Yellow ball	Sun	CSS_130	27 days (#47)	1 mo 20 days (#41)	1 hr 23 min
Yellow ball	Sun	CSS_CI	1 day 2 hr (#1177)	20 days (#1170)	52 min
Blue ball	Sun	CSS_Site	3 mo 25 days (#80)	3 mo 27 days (#78)	17 min
Blue ball	Sun	DAL_2_JMS	2 mo 10 days (#188)	N/A	38 min
Blue ball	Cloud	ICServer	1 mo 18 days (#44)	1 mo 18 days (#43)	31 min
Blue ball	Sun	Jms2Ora	1 day 3 hr (#4)	N/A	35 min
Blue ball	Sun	LDAP_Updater	1 day 2 hr (#350)	N/A	43 min
Blue ball	Sun	Websuite	1 day 2 hr (#41)	N/A	28 min

Hudson Project



Hudson Configuration

Hudson

Project name: `CSS_130`

Description: `CSS_130 300 / Mac / Linux`

☐ Discard Old Builds

☐ This build is parameterized

☐ Disable Build (No new builds will be executed until the project is re-enabled.)

☐ Execute concurrent builds if necessary (beta)

Advanced Project Options

Source Code Management

☐ None

☒ CVS

CVSROOT: `/usr/local/cvs/cvs@xray.mpg.de/group/m/this/cvs`

Module(s): `CSS_130`

Branch: `CSSProduct130`

☐ This is a tag, not a branch

☐ Subversion

☐ File System

Build Triggers

☐ Build after other projects are built

☐ Build periodically

☐ Poll SCM

Build

☒ Invoke Ant

Ant Version: `[Default]`

Targets:

Post-build Actions

☐ Publish Javadoc

☒ Archive the artifacts

Files to archive: `artifacts/*.zip`

☐ Aggregate downstream test results

☒ Publish JUnit test result report

Test report XMLs: `artifacts/*.xml`

☐ Build other projects

☐ Record fingerprints of files to track usage

☒ E-mail Notification

Recipients: `jan.hatje@desy.de`

White-space separated list of recipient addresses. E-mail will be sent when a build fails.



Hudson Configuration

```
<!--
  Set Targetplatform in Buckminster / Resolve & Materialize $buckcquery
-->
<target name="buckminster-import" depends="preparebuckminster">
  <java classname="org.eclipse.equinox.launcher.Main" fork="true" failonerror="true">
    <arg value="-data"/>
    <arg value="${buildDirectorySrc}"/>
    <arg value="setpreference"/>
    <arg value="org.eclipse.buckminster.pde.targetPlatformPath=${baseLocation}"/>
    <classpath>
      <pathelement location="${buckminsterLocation}/plugins/org.eclipse.equinox.launcher_${buckminsterEquinoxLauncherPluginVersion}.jar" />
    </classpath>
  </java>
  <java classname="org.eclipse.equinox.launcher.Main" fork="true" failonerror="true">
    <arg value="-data"/>
    <arg value="${buildDirectorySrc}"/>
    <arg value="import"/>
    <arg value="${buckminsterCquery}"/>
    <jvmarg value="-Xmx512m"/>
    <classpath>
      <pathelement location="${buckminsterLocation}/plugins/org.eclipse.equinox.launcher_${buckminsterEquinoxLauncherPluginVersion}.jar" />
    </classpath>
  </java>
</target>

<!--
  PDE Build win
-->
<target name="pde-build_win">
  <copy todir="${base}/bwin">
    <fileset dir="${buildDirectorySrc}"/>
  </copy>
  <java classname="org.eclipse.equinox.launcher.Main" fork="true" failonerror="true">
    <arg value="-application" />
    <arg value="org.eclipse.ant.core.antRunner" />
    <arg value="-buildfile" />
    <arg value="${base}/productBuild.xml" />
    <arg value="-DbaseLocation=${baseLocation}" />
    <arg value="-DbuildDirectory=${base}/bwin" />
    <arg value="-Dconfigs=win32, win32, x86" />
    <arg value="-Dtimestamp=${tstamp}" />
    <classpath>
      <pathelement location="${eclipseLocation}/plugins/org.eclipse.equinox.launcher_${equinoxLauncherPluginVersion}.jar" />
    </classpath>
  </java>
</target>
```

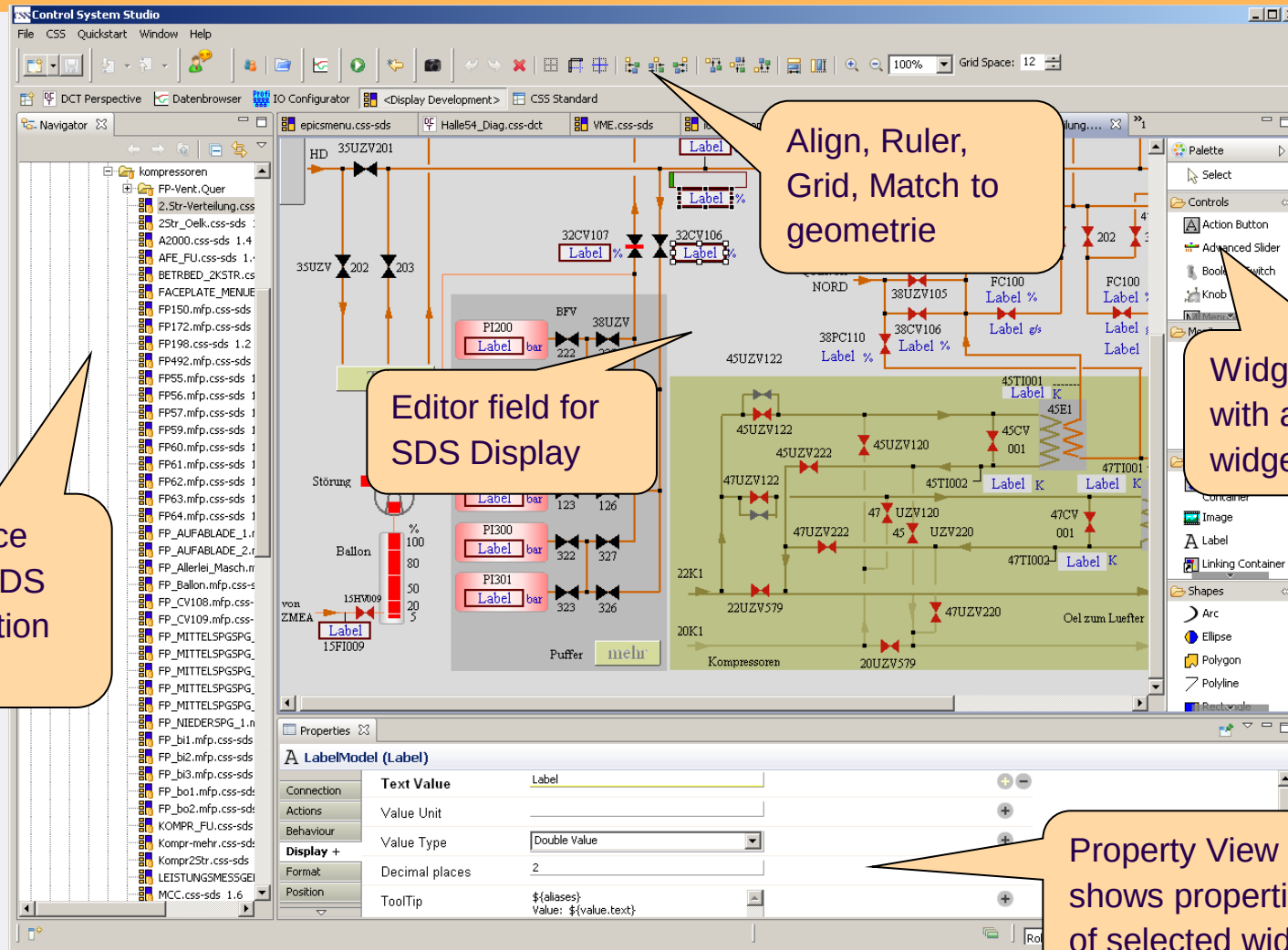
Ant build for CSS

1. Delete files from previous build
2. Configure Buckminster
3. Start Buckminster to resolve plug-in dependencies
4. Start Eclipse integrated Ant builds for each target platform
5. Copy additional files in the CSS directory and create the zip-archives
6. Run JUnit tests
7. Copy archive files and tests results in Hudson artifacts directories

Synoptic Display Studio

- Synoptic Display Studio (SDS) is a set of CSS plugins
- SDS is based on GEF
- Easy to add new widgets
- Connection to control system via DAL
- Behaviors for widgets
- CVS support for the management of displays
- ADL converter to reuse MEDM Displays

SDS Workbench

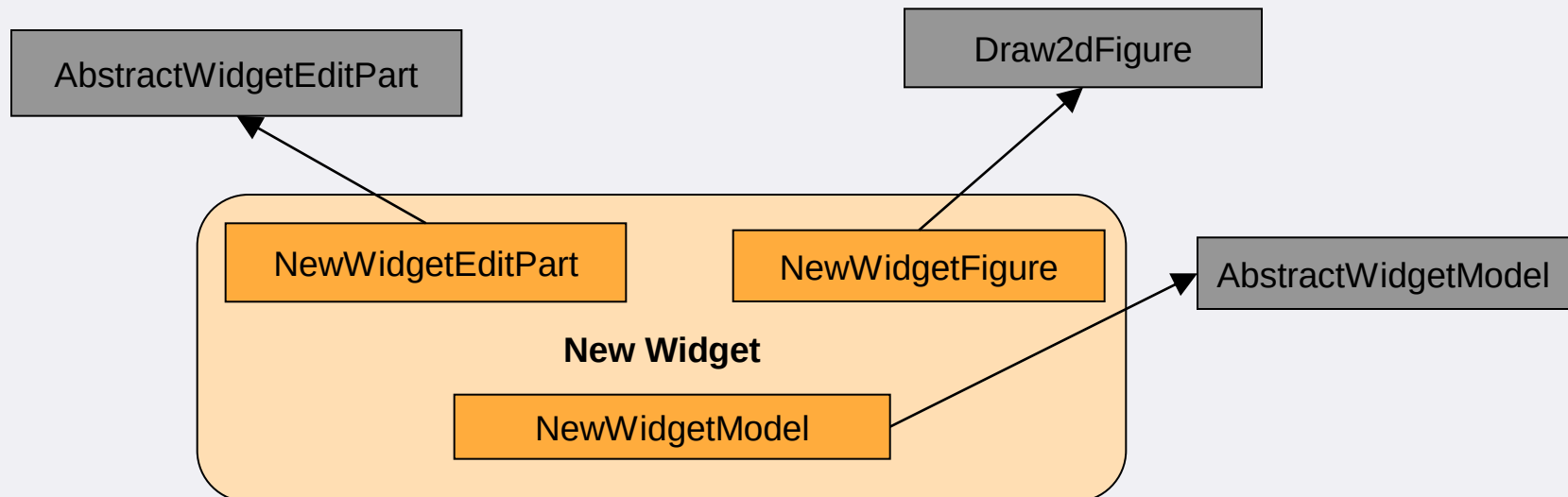


SDS Design

SDS is build on GEF / Draw2D framework and uses the Model – View – Controller Pattern.

New widgets has to implement just three classes:

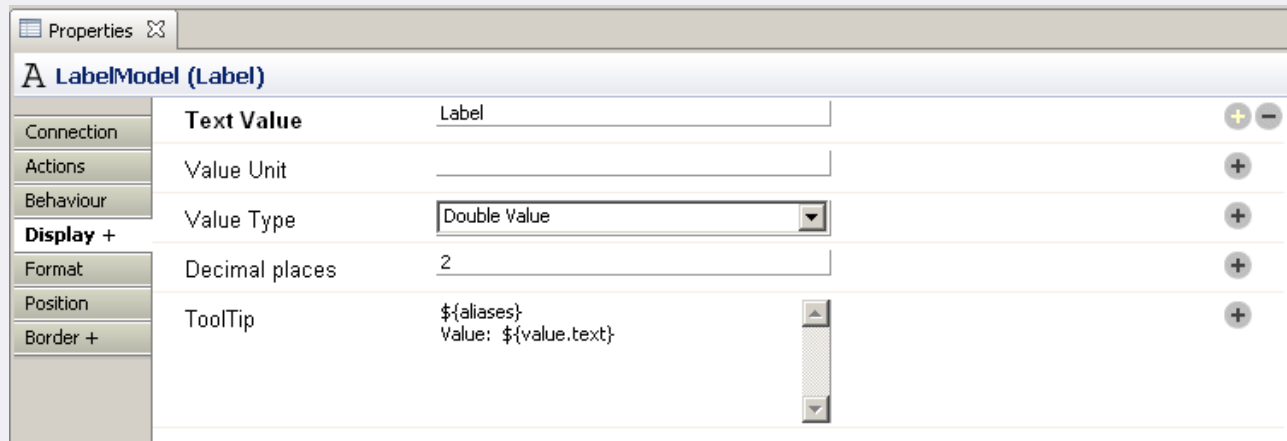
- **WidgetModel** (Model) – Holds all available properties for the widget.
- **WidgetFigure** (View) – Sets appearance of the widget.
- **WidgetEditPart** (Controller) – Defines behavior for user actions and events.



SDS Properties and Dynamisation

Dynamisation of a property means that the property depends on a value of a process variable

- Properties are ordered in categories
- Dynamisation is indicated by an additional icon
- Every widget property can be dynamic
- Widget behaviors simplifies the setup
- Aliases can be used for PV names



SDS Behaviors

Behaviors defines a site specific default set up for widgets

- A Behavior is related to one widget
- Widgets can have many behaviors
- Behaviors set default values for properties
- Properties can be hidden from the user
- For e.g. new alarm colors just the behavior has to be changed
- Behaviors are implemented as java classes

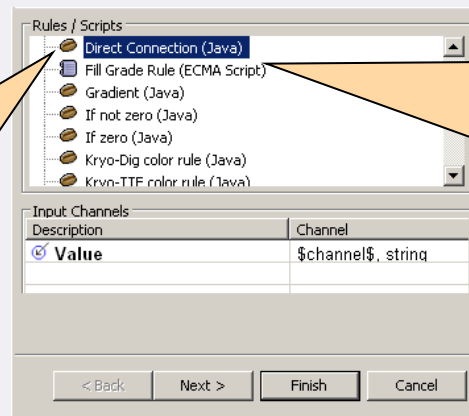
SDS Rules

Rules calculate or convert values from the control system

For instance a double value of a PV is converted by a rule to a color for the widget background

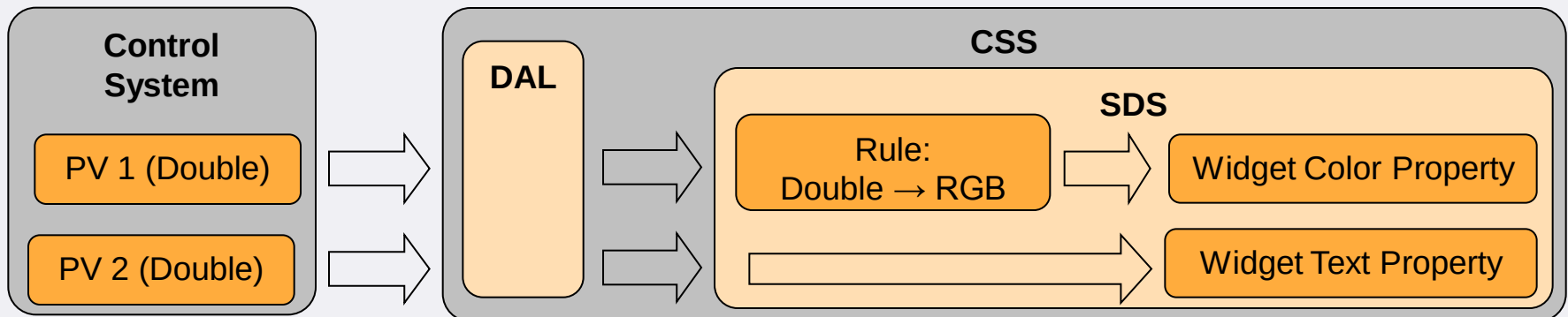
Rule implemented as Java class:

- High performance
- For new rules recompilation is necessary



Rule implemented as Java Script:

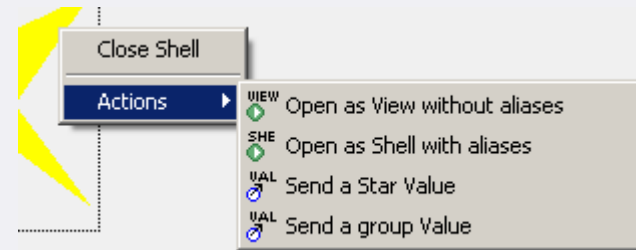
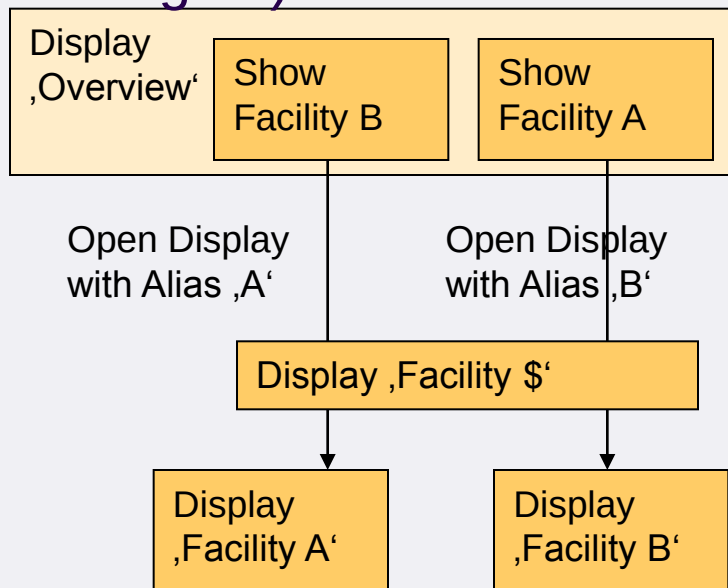
- Each rule is available in the workspace
- Rules can be added by the user
- Performance for high update rates?



SDS Action Data

Actions are related to widgets

- A widget can hold many actions
- Actions are inherited (Actions of display are available in all widgets)



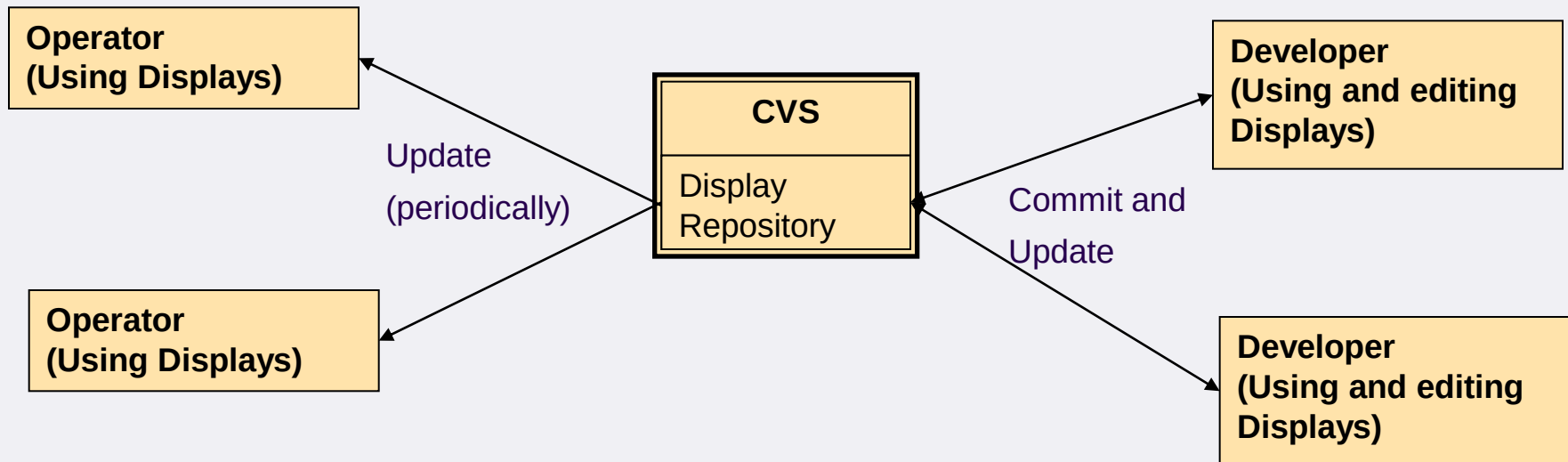
Action type 'Open as View/Shell'

- Opens a display from workspace
- Call new display with parameters

Action type 'Send Value'

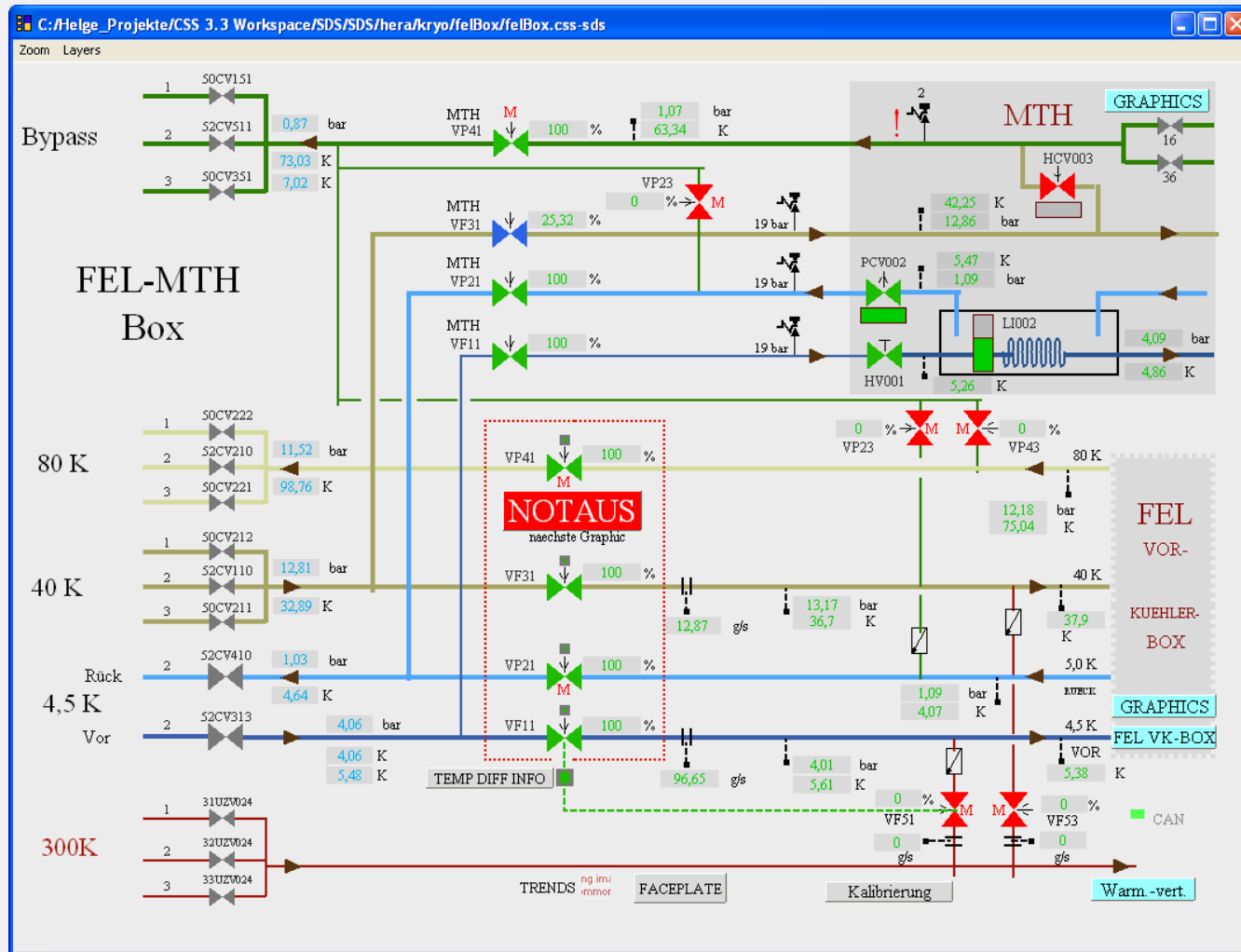
- Writes a value to a process variable

SDS CVS support



- CVS repository holds all displays files
- Operators get periodically an update that they are using always the current display versions
- No parallel versions of displays (Developers do not work on copies of the displays)
- History of all changes

SDS Example



Database Creation Tool (DCT)

- Hierarchical structure for EPICS databases
- Arrange EPICS records in logical components (e.g. vacuum pressure, oil temperature)
- Prototype - Instance relation to reuse components
- DCT perspective in CSS

Editor field for selected element in outline view

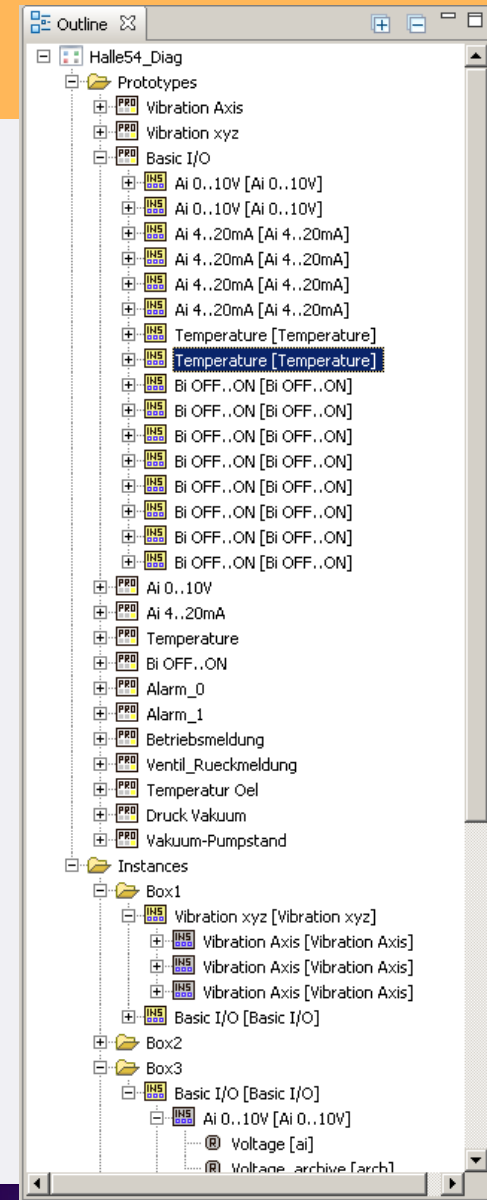
Outline View with EPICS database structure

List of all errors/problems in selected database

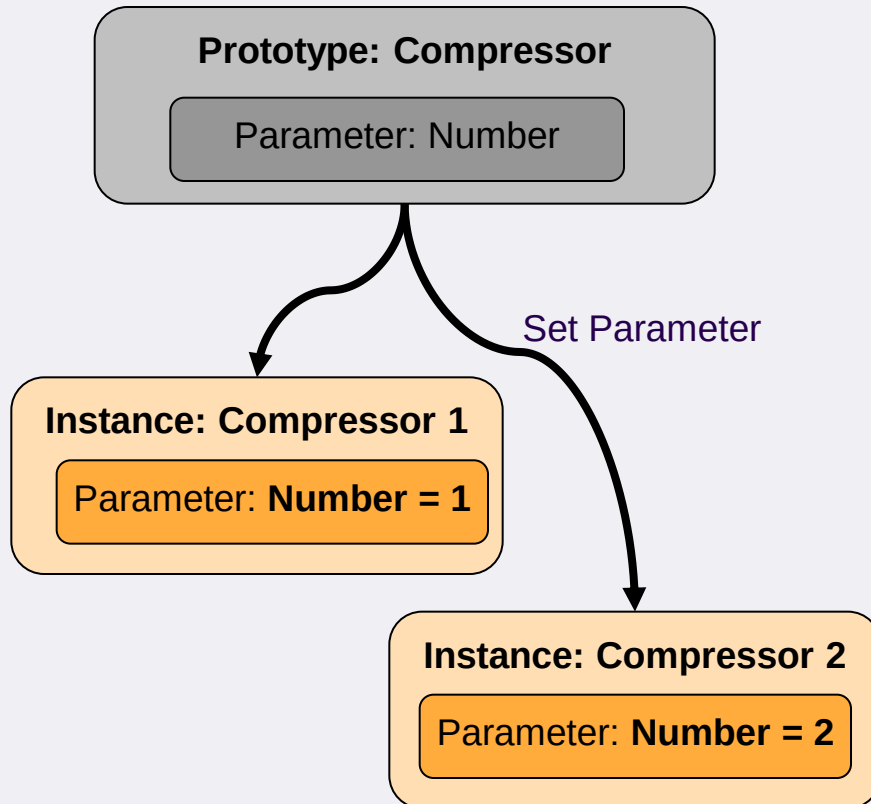
Instance of view to show all available instances

DCT Database structure

- Each prototype can contain records and instances of prototypes
- Levels of nested Prototype instances are unlimited
- Modifications in prototypes will change also the instances
- Drag and Drop and Copy / Paste in Database tree
- Specific values can be overwritten in instance level
- Organize database structure with free definable folders



DCT Prototype / Instance



Prototype

SimpleDemo > Prototypes > Sawtooth

Common Settings	
Identifier	1b892b58-8858-446b-8072-084hh8371
Name	Sawtooth

Properties	
Parameter	
dispMin	
dispMax	
lolo	
hihi	
lo	
hi	
minVal	
maxVal	
name	
increment	

Instance

SimpleDemo > Instances > SawAndIncr1 > Sawtooth
(Link to prototype or parent instance)

Common Settings	
Identifier	a53980dc-961c-47f3-b92e-21bfea51d2
Name	Sawtooth
Type	Sawtooth

Parameter Values	
dispMin	1000
dispMax	10000
lolo	2000
hihi	8000
lo	3000
hi	7000
minVal	1000
maxVal	10000
name	\$(name)
increment	Incrementer\$(name) Increment

DCT Records

- *.dbd file defines EPICS record fields and default values
- Disable specific records in prototypes and instances
- Online support for functions (eval, datalink, forwardlink)
- Autocompletion for defined Parameters

calc-Record
SimpleDemo > Prototypes > Sawtooth > Calculator

Common Settings

Identifier	ff29c15a-5f3f-4
Name	Calculator
Type	calc
Epics Name	Calc\$(name)
Disabled	false

Properties

Fields
Group: All ☐ Hide Defaults

DESC - Descriptor	Calc\$(name)
ASG - Access Security Group	<empty>
SCAN - Scan Mechanism	1 second
PINI - Process at iocInit	NO
PHAS - Scan Phase	1
EVNT - Event Number	<empty>
TSE - Time Stamp Event	<empty>
TSEL - Time Stamp Link	<empty>
DTYP - Device Type	<empty>
DISV - Disable Value [1]	1
SDIS - Scanning Disable	<empty>
ACKT - Alarm Ack Transient [Y...	YES
DISS - Disable Alarm Sevrty	NO_ALARM
PRI0 - Scheduling Priority	LOW
UDF - Undefined [1]	1
FLNK - Forward Process Link	<empty>
CALC - Calculation [0]	A<D?A+B:C
INPA - Input A	Calc\$(name).V
INPB - Input B	\$(increment)

calc-Record (CalcSawAndIncr0)
SimpleDemo > Instances > SawAndIncr0 > Sawtooth > Calculator

Common Settings

Identifier	36036613-0812-4b51-8382-86a8268
Name	Calculator
Type	calc
Epics Name	Calc\$(name)
Disabled	false

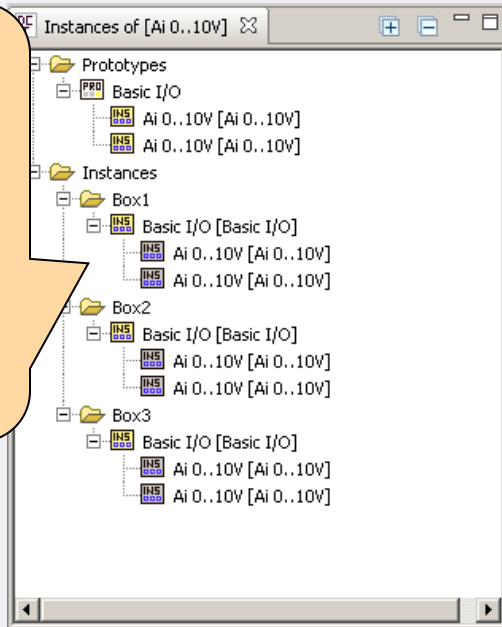
Properties

Fields
Group: All ☐ Hide Defaults

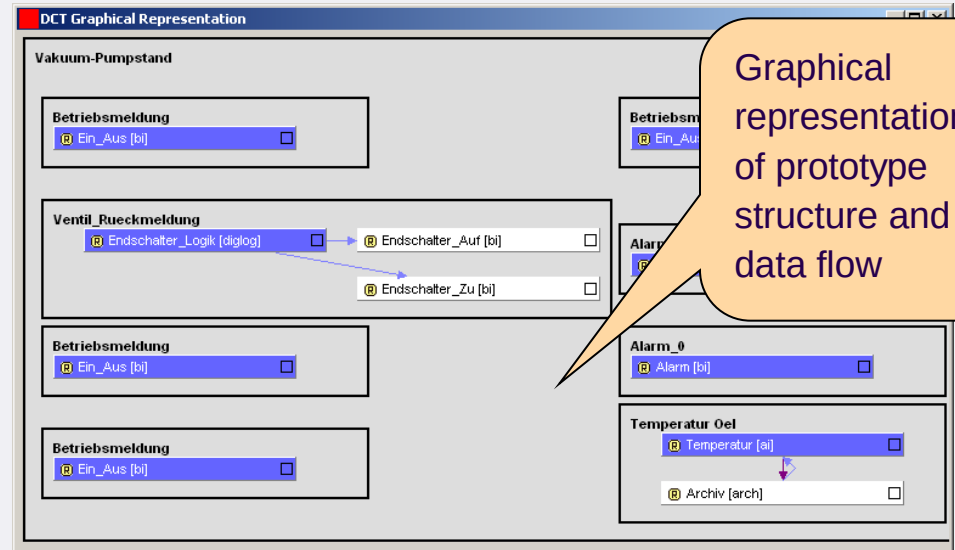
DESC - Descriptor	CalcSawAndIncr0
ASG - Access Security Group	<empty>
SCAN - Scan Mechanism	1 second
PINI - Process at iocInit	NO
PHAS - Scan Phase	1
EVNT - Event Number	<empty>
TSE - Time Stamp Event	<empty>
TSEL - Time Stamp Link	<empty>
DTYP - Device Type	<empty>
DISV - Disable Value [1]	1
SDIS - Scanning Disable	<empty>
ACKT - Alarm Ack Transient [Y...	YES
DISS - Disable Alarm Sevrty	NO_ALARM
PRI0 - Scheduling Priority	LOW
UDF - Undefined [1]	1
FLNK - Forward Process Link	<empty>
CALC - Calculation [0]	A<D?A+B:C
INPA - Input A	CalcSawAndIncr0.VAL NPP MS
INPB - Input B	IncrementerSawAndIncr0.VAL

DCT Utilities

Show all instances of a prototype. Selection updates outline view with database structure



Graphical representation of prototype structure and data flow



Overview of all errors / problems in database. Selection jump in outline view to related prototype

The screenshot shows a window titled 'Problems' with a table of 6 items. The table has the following columns: Description, Resource, Path, Location, Type, and an empty column.

Description	Resource	Path	Location	Type	
6 items					
■ Infos (6 items)					
■ Record name "H54DIAG:CD_bi" is not unique.	Halle54_Diag.css-dct	/DCT Demo Project	16df34...	Problem	
■ Record name "H54DIAG:CD_bi" is not unique.	Halle54_Diag.css-dct	/DCT Demo Project	58cc2d...	Problem	
■ Record name "H54DIAG:CD_bi" is not unique.	Halle54_Diag.css-dct	/DCT Demo Project	652fd2...	Problem	
■ Variable "iNrxx" cannot be resolved.	Halle54_Diag.css-dct	/DCT Demo Project	2fbc32...	Problem	
■ Variable "iNrxx" cannot be resolved.	Halle54_Diag.css-dct	/DCT Demo Project	58e00...	Problem	
■ Variable "iNrxx" cannot be resolved.	Halle54_Diag.css-dct	/DCT Demo Project	a850d...	Problem	

```
# # Halle54_Diag : Instances > Box1 > Vibration xyz > Vibration Axis > System Alarax
record(bi_ "H54DIAG1.CBX43VIBX_SYS_OK_bi") {
    field(DESC, "Vibr CBX43 System ok?")
    field(DTYPE, "PBDEP")
    field(INP, "@PROFI1.41/13 T=INT8.B=0'")
    field(OSV, "MAJOR")
    field(ZNAM, "OK")
    field(ONAM, "Fehler")
}

# # Halle54_Diag : Instances > Box1 > Vibration xyz > Vibration Axis > Kabelbruch
record(bi_ "H54DIAG1.CBX43VIBX_ERR_f_bi") {
    field(DESC, "Fehler Kabelb./Kurzschl.")
    field(SCAN, "1 second")
    field(DTYPE, "PBDEP")
    field(INP, "@PROFI1.41/13 T=INT8.B=3'")
    field(OSV, "MAJOR")
    field(ZNAM, "OK")
    field(ONAM, "Fehler")
}

# # Halle54_Diag : Instances > Box1 > Vibration xyz > Vibration Axis > Kommunikation_0
record(bi_ "H54DIAG1.CBX43VIBX_RCOMB_bi") {
    field(DESC, "Prozess/Register Komm.")
    field(SCAN, "1 second")
    field(DTYPE, "PBDEP")
    field(INP, "@PROFI1.41/13 T=INT8.B=7'")
    field(ZNAM, "Process")
    field(ONAM, "Register")
}

# # Halle54_Diag : Instances > Box1 > Vibration xyz > Vibration Axis > RMS_Wert
record(arch_ "H54DIAG1.CBX43VIBX_PMS_ai_h") {
    field(DESC, "Effektivwert Schwingung")
    field(SCAN, "1 second")
    field(DTYPE, "PBDEP")
    field(FILNK, "H54DIAG1.CBX43VIBX_PMS_ai_h")
    field(INP, "@PROFI1.41/14 T=INT16")
    field(EGU, "m/s")
    field(HOPR, "100")
    field(ASLO, "0.1")
    field(HYST, "0.5")
    field(ADEL, "0.5")
}

# # Halle54_Diag : Instances > Box1 > Vibration xyz > Vibration Axis > RMS_Wert Archiv
record(arch_ "H54DIAG1.CBX43VIBX_RMS_ai_h") {
    field(DESC, "Effektivwert Schwingung")
    field(INP, "Error 1")
    field(AVAR, "0.2")
    field(RVAR, "0.5")
    field(PCAB, "abs && rel")
    field(HOPR, "100")
    field(PREC, "1")
}
```

Databrowser

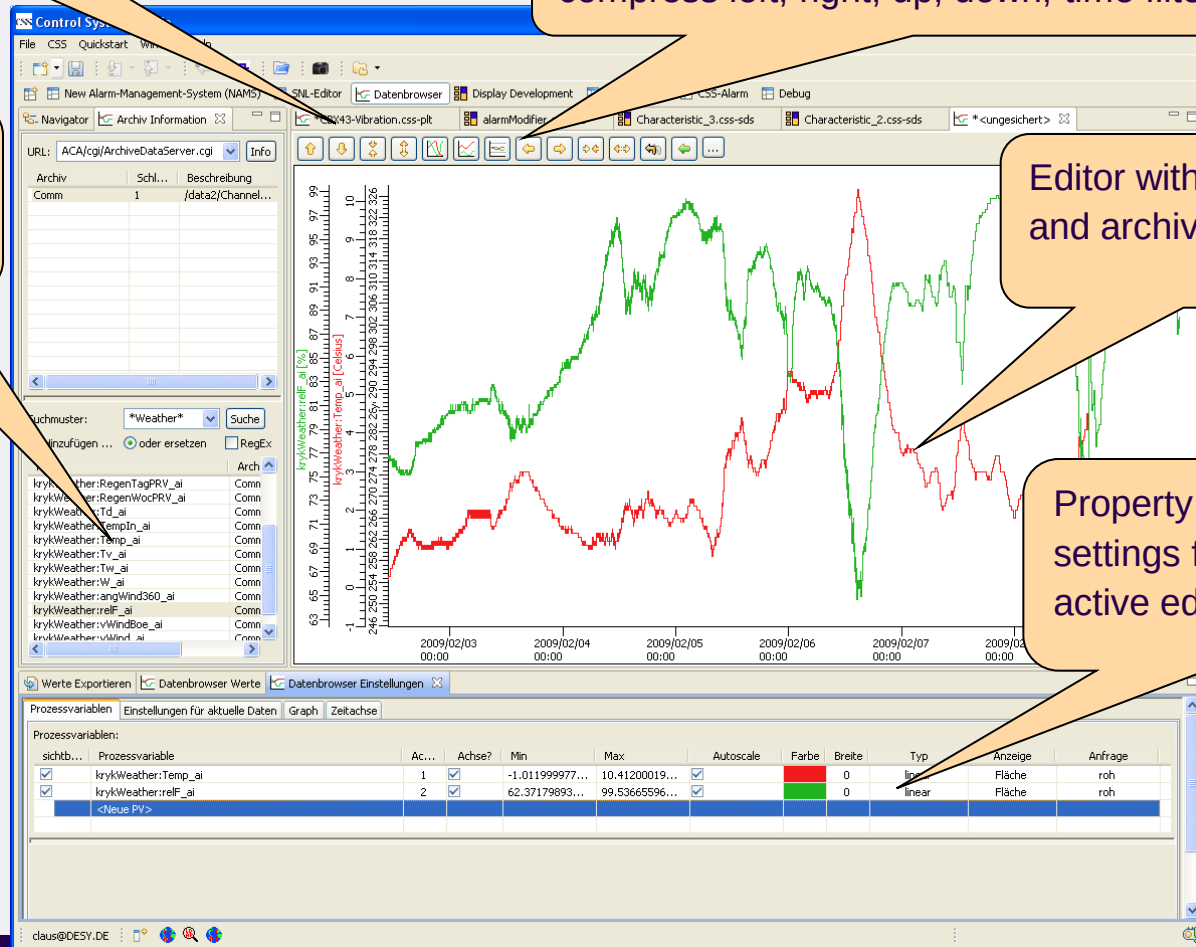
CSS Perspective
with DB Views

List of archived
PVs. Filter is
available

Navigation in chart: Shift left, right, expand,
compress left, right, up, down, time filter, undo

Editor with chart of live
and archived values

Property View with
settings for each PV of
active editor



Databrowser Property view

Set PV name and name on chart

Axis the PV should belong to

Settings for y-Axis: Range, Autoscale

Chart style (dot, line, bar) representation (linear, log) and type of archive access

sic...	Prozessvariable	Anzeigename	A...	Ac...	Min	M...	Autoscale	F...	Br...	Typ	Anzeige	Anfrage
☒	krykWeather:Temp_ai	krykWeather:Temp_ai	0	☒	8.046	23.754	☒	☒	0	linear	Fläche	optimiert
☒	krykWeather:PLuft_ai	krykWeather:PLuft_ai	1	☒	997.65...	1020.5...	☒	☒	0	linear	Fläche	optimiert
☒	krykWeather:RegenStdPRV_ai	krykWeather:RegenStdPRV_ai	2	☒	-0.066	6.6659...	☒	☒	0	linear	Fläche	optimiert
☒	krykWeather:vWind_ai	krykWeather:vWind_ai	3	☒	-0.0936...	10.204...	☒	☒	0	linear	Fläche	optimiert
	<Neue PV>											

Archiv für gewählte Prozessvariable:

Archiv	Schlüssel	URL
Comm	1	xnds://www-kryo.desy.de/applets/LACA/cgi/ArchiveDataServer.cgi

PV archive values in table representation

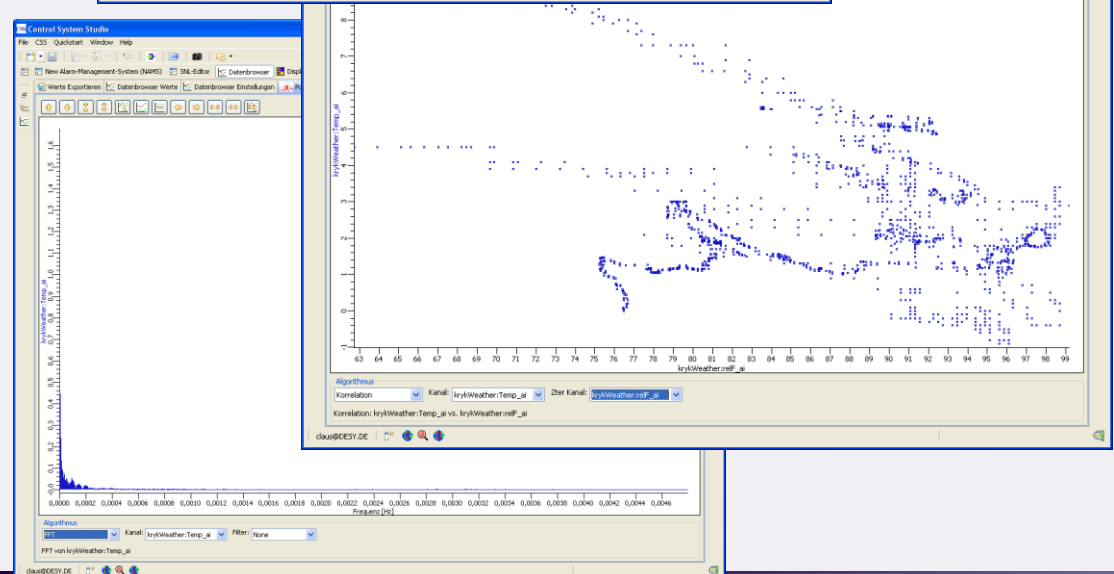
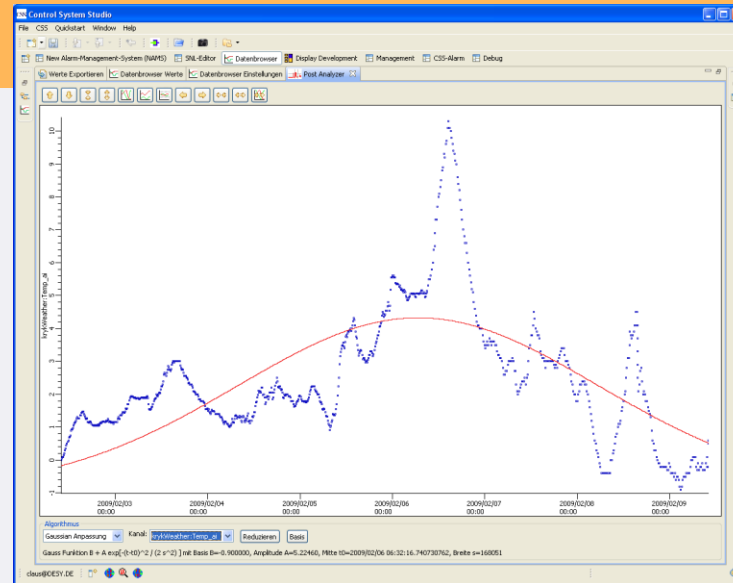
Datenbrowser Einstellungen Werte Exportieren Contacts Datenbrowser Werte Aktualisieren

PV: krykWeather:vWind_ai

Zeit	Wert	Güte, Status	Qualität	Datenquelle
2010/09/17 10:36:55.452502522	3,0	OK, OK	Interpoliert	Channel Archiver
2010/09/17 10:39:55.446151912	6,7	OK, OK	Interpoliert	Channel Archiver
2010/09/17 10:42:55.439601303	4,6	OK, OK	Interpoliert	Channel Archiver
2010/09/17 10:43:55.437775703	5,8	OK, OK	Interpoliert	Channel Archiver
2010/09/17 10:46:55.431437688	4,6	OK, OK	Interpoliert	Channel Archiver
2010/09/17 10:49:55.425099674	6,2	OK, OK	Interpoliert	Channel Archiver
2010/09/17 10:52:55.418761659	5,9	OK, OK	Interpoliert	Channel Archiver
2010/09/17 10:53:55.416728859	5,8	OK, OK	Interpoliert	Channel Archiver
2010/09/17 10:56:55.410358995	3,9	OK, OK	Interpoliert	Channel Archiver
2010/09/17 10:59:55.403989131	6,7	OK, OK	Interpoliert	Channel Archiver
2010/09/17 11:02:55.397619267	6,7	OK, OK	Interpoliert	Channel Archiver
2010/09/17 11:05:55.391437688	6,7	OK, OK	Interpoliert	Channel Archiver

Postanalyzer

- Call from DataBrowser with object contribution
- Gaussian Fit, Correlation, FFT available for data analysis
- Chart widget is based on DataBrowser



CSS Workshop

Thank you!