Control System Studio (CSS)
The next Generation of OPI

PCaPAC

Jefferson Lab 27. October 2006

Matthias Clausen, DESY

Agenda

- **Motivation**
- **CSS Design Criteria**
- **Applications**
- **CSS Development/ Collaboration**
- **Schedule**
- **Outlook**

Motivation – Current State

Have a look at current operator interfaces:

- **Many applications of different kind**
  - Implemented by different programmers
  - Different look and feel
  - Implemented in different programming languages
  - GUI with text in specific languages (difficulties for Japanese Operators)
  - Running only on specific operating systems
  - Running only in the context of a specific control system type
- **Which kind of data/ information transfer is possible?**
  - Drag and drop basic information as 'string'
  - Cut and paste 'strings'
  - Drag and drop 'objects' within the same program environment (Windows)

Matthias Clausen, DESY
CSS – Control System Studio

Motivation - Future

For the next projects at DESY

- *Upgrade of Cryogenic Plant (2007) - tentative*
- PERTA-III Utility Controls (2008)
- XFEL Test Facilities (2009)

Desired features:

- Common look and feel
- Common functionality
- Running on several operating systems
- Drag and drop complex (control system – specific) objects
- Find related applications/ information
- Running in the context of several control systems

Matthias Clausen, DESY
CSS – Control System Studio

CSS Design I

Environment:

- Java – to run on many platforms
- Eclipse – to use the RCP approach and to join a growing Eclipse developers community

Matthias Clausen, DESY
CSS – Control System Studio

Eclipse

Eclipse is an Open Source community

It was started in 2001 by IBM

- IBM donated a lot of research
- Controlled the early development, but later relinquished control
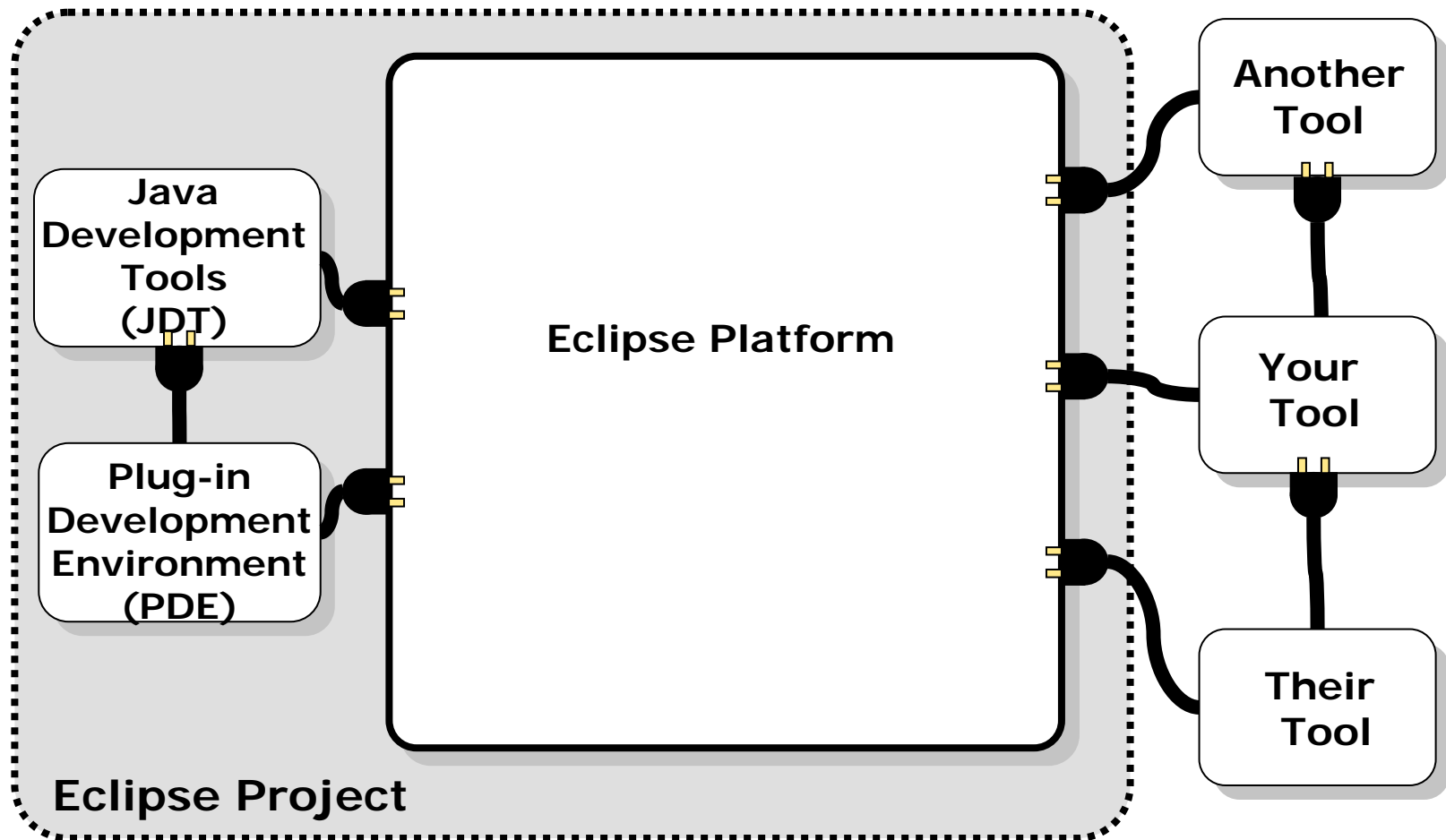
Out of the box it looks like a Java IDE

It is really a Plug-in manager

- That happens to come with Java Development plug-ins.
- You can take these out and put your own (and/or others) in

© Kenneth Evans, Jr.

Matthias Clausen, DESY
CSS – Control System Studio

Eclipse Consortium Strategic Members

* Strategic Consumer

Very Extensible and Very Flexible

Matthias Clausen, DESY
CSS – Control System Studio

Eclipse as a Java IDE

9
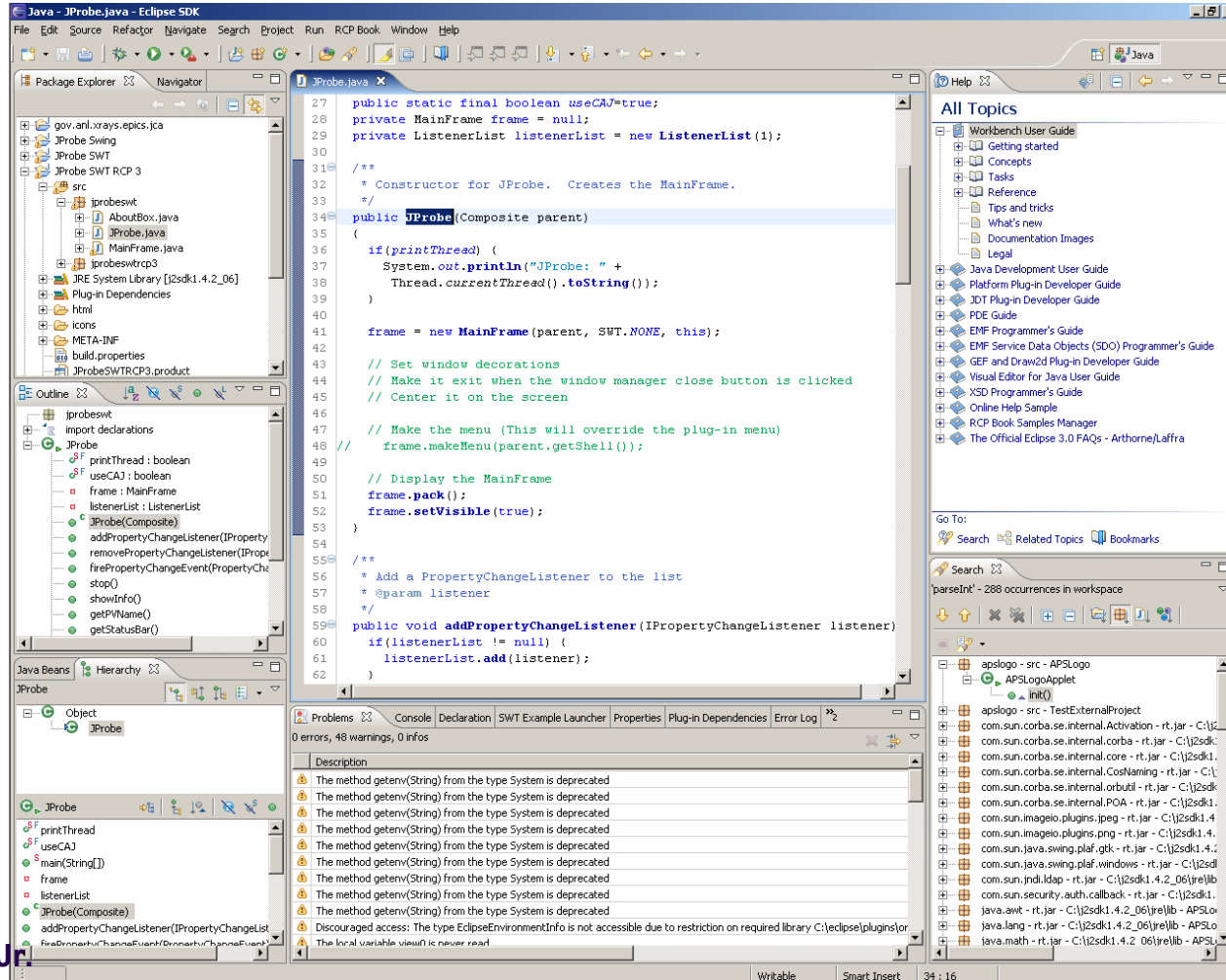
Eclipse As a Rich Client Platform

Looks like an application, not an IDE

Inherits a lot of functionality

- Persistence (Properties and Preferences)
- Help
- Featured About dialog (like Eclipse's)
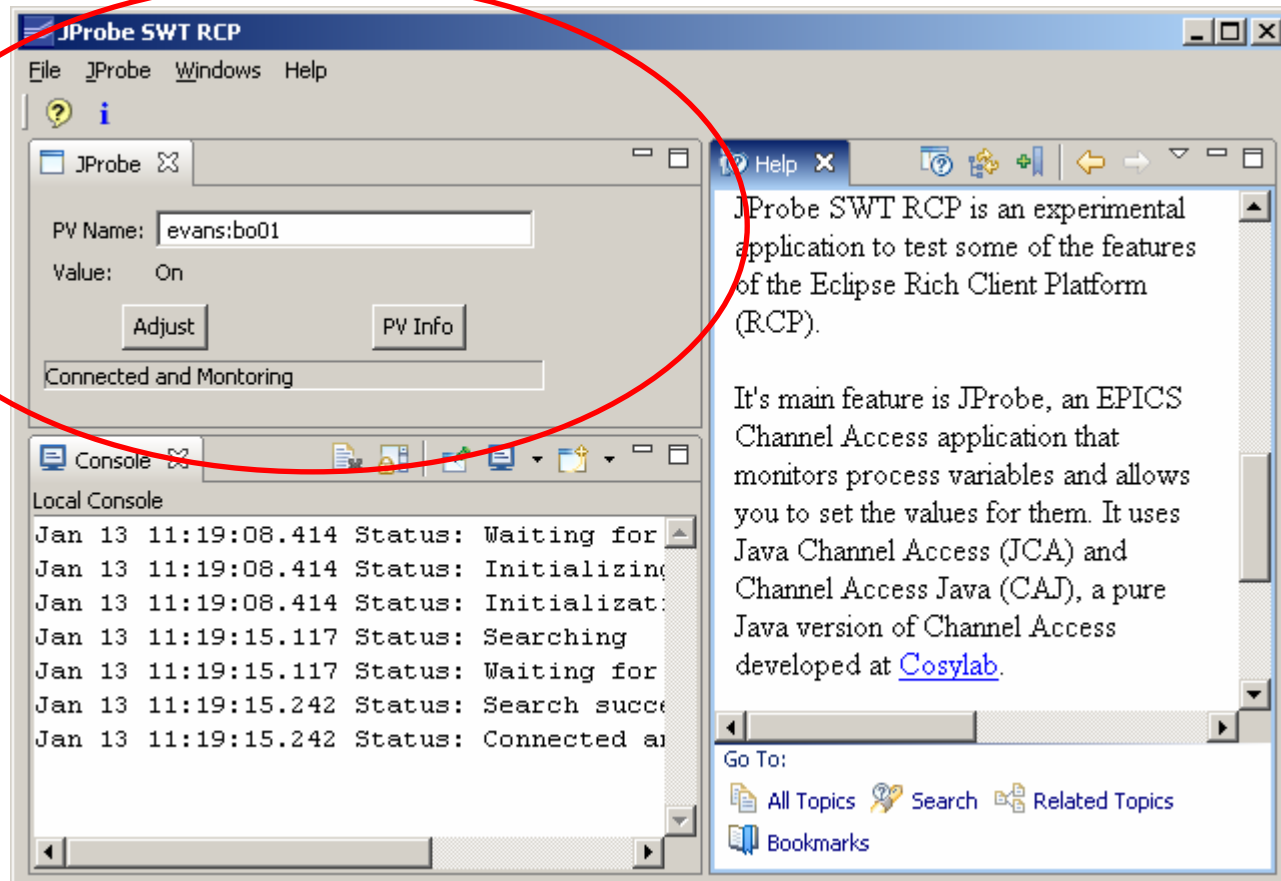- Splash screen
- Dockable windows, and much more
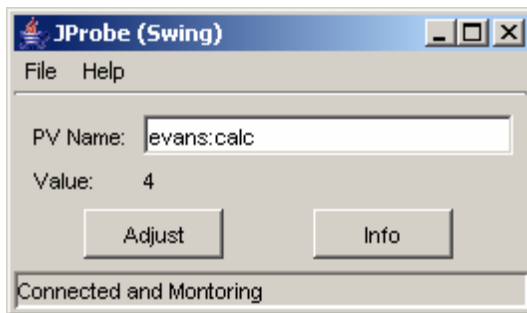
Java Application

RCP Application

10

Matthias Clausen, DESY
CSS – Control System Studio
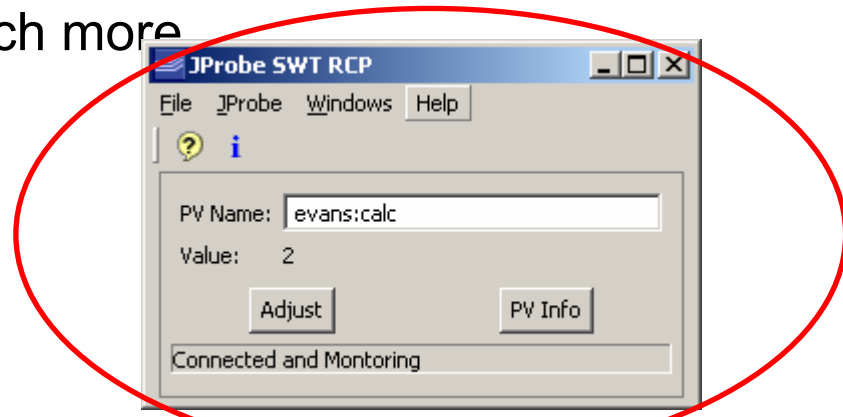
Probe on Steroids
Leveraging the Eclipse Framework

© Kenneth Evans, Jr.

Matthias Clausen, DESY
CSS – Control System Studio

An RCP Application is Also a Plug-In

© Kenneth Evans, Jr.

Matthias Clausen, DESY
CSS – Control System Studio

CSS-Core

Eclipse Platform

Eclipse Project

CSS Core

Your Plugin

CSS Libraries

RCP Plug-in in the CSS context

Matthias Clausen, DESY
CSS – Control System Studio

CSS Design II

Design Criteria:

- Common Look and Feel
- 'Information on your Fingertips'
  Find related applications using <MB3>
- Drag and drop Java objects between (CSS compliant) applications
- Management of installed CSS installations
- Talking to control systems through interfaces rather than directly through control system specific API's
- Authentication/ Authorization
- Internationalization
- Do not preclude other implementations

CSS Design II

Design Criteria:

- Common Look and Feel

- 'Information on your Fingertips'
  Find related applications using <MB3>

- Drag and drop Java objects between (CSS compliant) applications

- Management of installed CSS installations

- Talking to control systems through interfaces rather than directly through control system specific API's

- Authentication/ Authorization

- Internationalization

- Do not preclude other implementations

Matthias Clausen, DESY
CSS – Control System Studio

Common Look and Feel

CSS is running as an (Eclipse) RCP

- Inherits the Eclipse look and feel
- Adds CSS specific menus and actions

Matthias Clausen, DESY
CSS – Control System Studio

CSS Design II

Design Criteria:

- Common Look and Feel
- 'Information on your Fingertips'
  Find related applications using <MB3>
- Drag and drop Java objects between (CSS compliant) applications
- Management of installed CSS installations
- Talking to control systems through interfaces rather than directly through control system specific API's
- Authentication/ Authorization
- Internationalization
- Do not preclude other implementations

Matthias Clausen, DESY
CSS – Control System Studio

Information on your Fingertip
Accessing Object Aspect Data

Plug-in started with item's value (record name)

Click on item in LDAP tree

Select 'related' plug-in from list

Data from EPICS-Ora database

Data from EPICS-IOC

Matthias Clausen, DESY
CSS – Control System Studio

Object Aspect Management

GUI 1
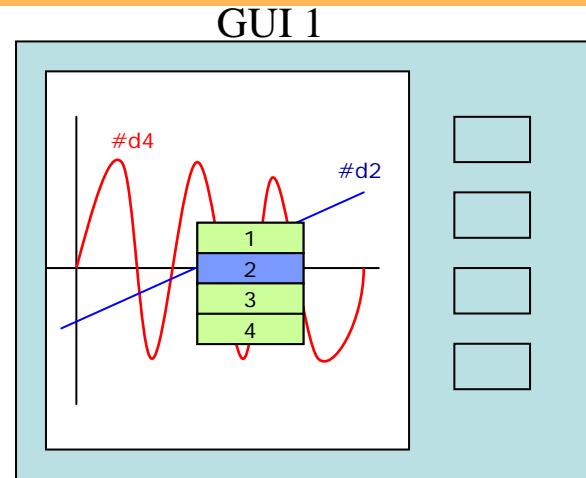
GUI 2 (another View / Plugin)

#d4

#d2

1
2
3
4

#	#d2	#d4
1	55	2334
2	43	34
3	544	4434
4	2323	621
5	5455	53

<< use >>

CORE

<< extensions >>

CSS.DA   VIDER

CSS
Data

C
C
CSS.DATA.RECEIVER

20
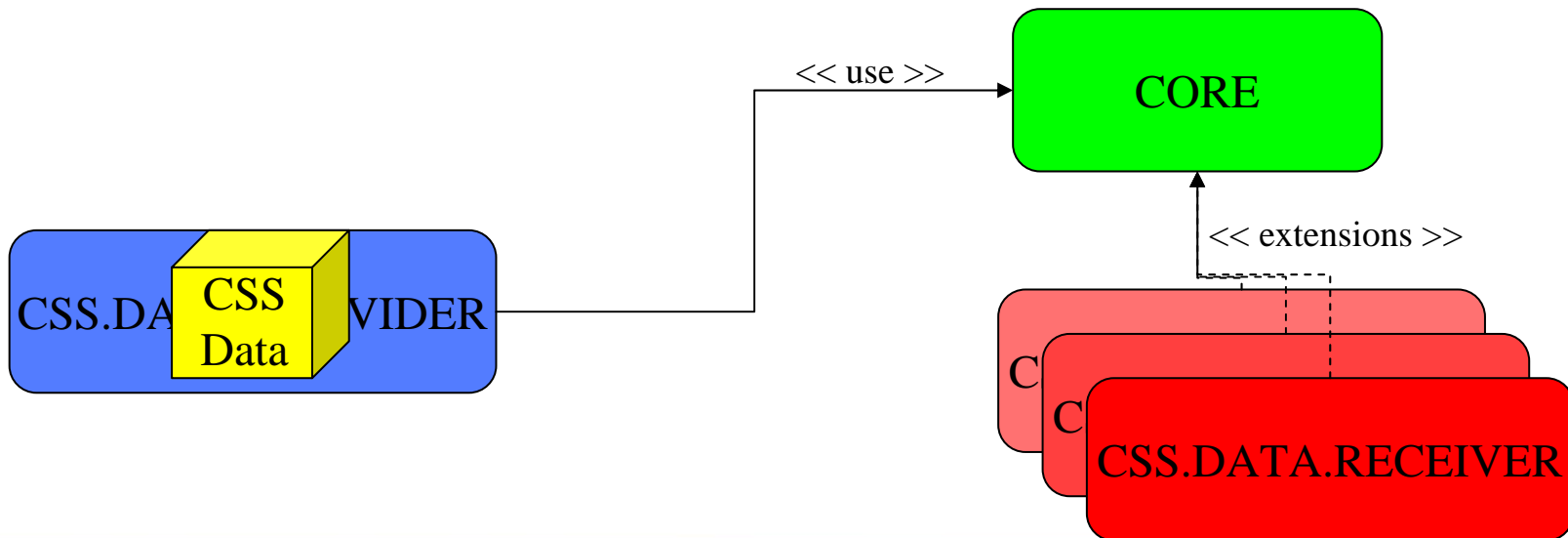
Object Aspect Applications/ Data

Applications:
- **(EPICS) Probe**
- **Device Properties ***
- **PV-Table**
- **StripTool/ Trend**

Data:
- **Configuration**
  - EPICS Ora DB *
  - Device DB
  - Field-bus
- **Maintenance DB**
- **Asset DB**
- **Purchase DB**

Matthias Clausen, DESY
CSS – Control System Studio

CSS Design II

Design Criteria:

- Common Look and Feel
- 'Information on your Fingertips'
  Find related applications using <MB3>
- **Drag and drop Java objects between (CSS compliant) applications**
- Management of installed CSS installations
- Talking to control systems through interfaces rather than directly through control system specific API's
- Authentication/ Authorization
- Internationalization
- Do not preclude other implementations

Matthias Clausen, DESY
CSS – Control System Studio

Drag and Drop

An extendible list of DaD data dypes is defined in org.csstudio.data.exchange

- 'Process Variable'
- 'Process Front-end'
- 'Archive Data'
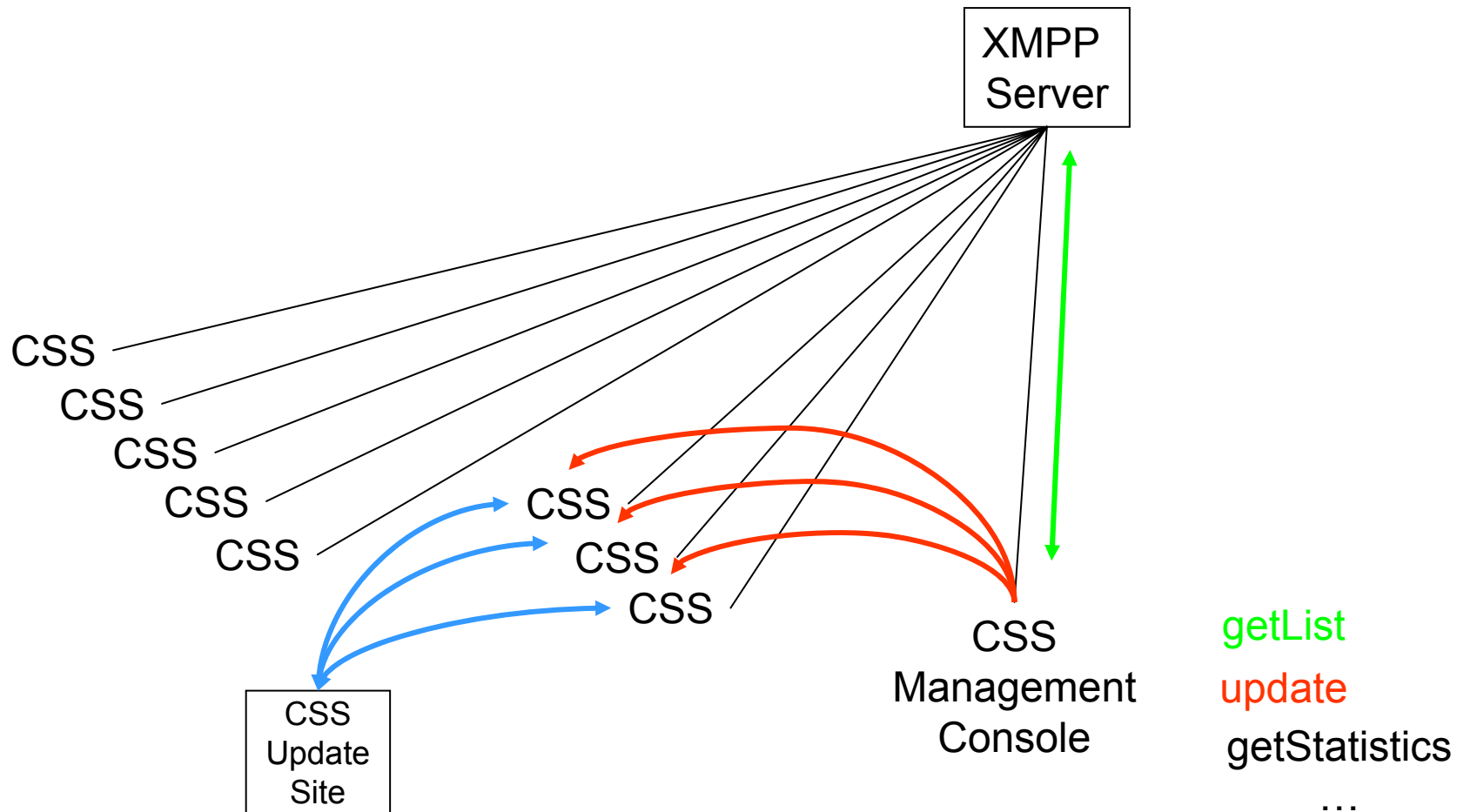- …

CSS Design II

Design Criteria:

- Common Look and Feel
- 'Information on your Fingertips'
  Find related applications using <MB3>
- Drag and drop Java objects between (CSS compliant) applications
- **Management of installed CSS installations**
- Talking to control systems through interfaces rather than directly through control system specific API's
- Authentication/ Authorization
- Internationalization
- Do not preclude other implementations

Matthias Clausen, DESY
CSS – Control System Studio

CSS Management

- **Query for existing CSS instances**
- **Control if and when CSS instances get updated from Eclipse Update Site**
- **Get statistical information from CSS instances**
- **Stop CSS instance in case of problems**

Matthias Clausen, DESY
CSS – Control System Studio

CSS Management

XMPP
Server

CSS

CSS

CSS

CSS

CSS

CSS

CSS

CSS

CSS
Management
Console

CSS
Update
Site

getList

update

getStatistics

…

Matthias Clausen, DESY
CSS – Control System Studio
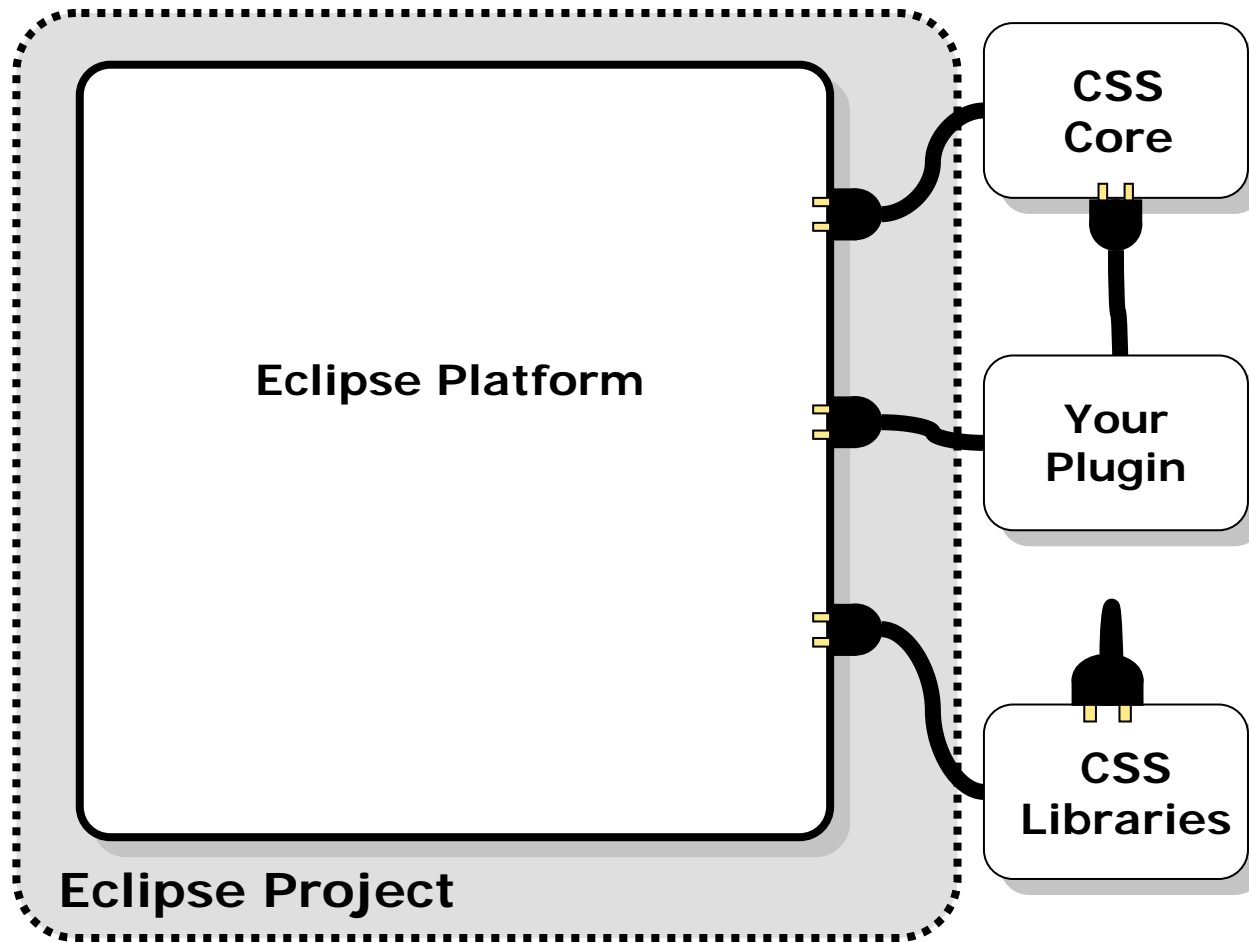
CSS Design II

Design Criteria:

- Common Look and Feel
- 'Information on your Fingertips'
  Find related applications using <MB3>
- Drag and drop Java objects between (CSS compliant) applications
- Management of installed CSS installations
- **Talking to control systems through interfaces rather than directly through control system specific API's**
- Authentication/ Authorization
- Internationalization
- Do not preclude other implementations

Matthias Clausen, DESY
CSS – Control System Studio

Interfaces - Interfaces

- **One prominent Interface-Family are the Eclipse extension-points**

CSS-Core

Eclipse Project

Eclipse Platform

CSS Core

Your Plugin

CSS Libraries

Matthias Clausen, DESY
CSS – Control System Studio

Interfaces - Interfaces

- **One prominent Interface-Family are the Eclipse extension-points**

- **Management Interfaces:**
  - Authentication
  - Authorization
  - Update/ Manage (XMPP)

- **Name Server (JNDI)**

- **Archive Access Layer (AAL)**

- **Data Access Layer (DAL)**

Matthias Clausen, DESY
CSS – Control System Studio

Data Access Library Layers

© Igor Kriznar

Wide style API with many methods and interfaces for programming convenience

CSS and other DAL applications

DAL API

Common DAL glue code

Collaborating set of common helper classes. Can be replaced by case or completely by DAL implementation.

Simple API concentrated on data transfer and simplicity when implementing DAL .

Proxy API

Proxy implementation (communication specific)

Communication API

Communication libraries

When common glue code does all the work, then this layer is thin, simple and oriented around communication particularities.

So far supported Simulator (internal), EPICS-CA and GSI.

31

CSS Design II

Design Criteria:

- Common Look and Feel
- 'Information on your Fingertips'
  Find related applications using <MB3>
- Drag and drop Java objects between (CSS compliant) applications
- Management of installed CSS installations
- Talking to control systems through interfaces rather than directly through control system specific API's
- **Authentication/ Authorization**
- Internationalization
- Do not preclude other implementations

Authentication/ Authorization

CSS-Core provides the means to keep (access) control over graphical widgets, (graphically) issued commands or even individual classes.

To use this feature the user must be authenticated to get authorized to perform a specifically configured action.

Authentication
- CSS-Core provides an interfaces to verify Username/ Passwords entered into Login Window
- Login Window can be enabled/ disabled (if you are authorized)

Authorization
- CSS-Core provides an interface to verify the username for it's actual role

Implementation:

This action can be performed by a user who has the role <u>Manager</u> for the Component <u>RF-System</u>

33

Authorization Roles

Roles defined for user: claus

Role^ / Component >	CSS	Cryo	RF	Water/ FLASH	
Guest			*	*	...
Operator					
Engineer		*			
Manager	*		🟥		

.
.
·   … 'claus' can not perform the action…

Matthias Clausen, DESY
CSS – Control System Studio

CSS Design II

Design Criteria:

- Common Look and Feel
- 'Information on your Fingertips'
  Find related applications using <MB3>
- Drag and drop Java objects between (CSS compliant) applications
- Management of installed CSS installations
- Talking to control systems through interfaces rather than directly through control system specific API's
- Authentication/ Authorization
- **Internationalization**
- Do not preclude other implementations

International Language Support

Fortunately Eclipse has a built in multi language support.

Developers are encouraged to make use of multi language support.

Matthias Clausen, DESY
CSS – Control System Studio

CSS Design II

Design Criteria:

- Common Look and Feel
- 'Information on your Fingertips'
  Find related applications using <MB3>
- Drag and drop Java objects between (CSS compliant) applications
- Management of installed CSS installations
- Talking to control systems through interfaces rather than directly through control system specific API's
- Authentication/ Authorization
- Internationalization
- **Do not preclude other implementations**

Matthias Clausen, DESY
CSS – Control System Studio

Do not preclude non-CSS implementations

The consequence of not using CSS interfaces is limited CSS functionality

Plug-ins which do not use all CSS interfaces can still run in the CSS context.

In fact: Any Eclipse plug-in can run in CSS

38

Applications

- **Synoptic Display**
- **Alarm Display**
- **Archive Viewer**
- **Diagnostic Tools**

Applications

- **Synoptic Display**
- **Alarm Display**
- **Archive Viewer**
- **Diagnostic Tools**

Synoptic Display Studio (SDS)

Designing and implementing a new Synoptic Display for the CSS is the main task of a new contract between DESY and the University of Hamburg.

The company C1-WPS is a key partner in a scientific project supported by German government. And – our partner through the University of Hamburg. The SDS is the proof of concept for the CoEUD – 'Component Based End User Development'.

This way the two partners share the same interest:

-> To successfully implement a new Synoptic Display based on – and integrated in - CSS.

Matthias Clausen, DESY
CSS – Control System Studio

SDS basic requirements (subset)

- **Graphics can be created by non professional programmers and thus are configured – or graphically programmed as one might call it.**

- **The persistent store of configuration data is kept in XML files.**

- **New graphic elements can easily be added to the CSS graphic framework. Ideally new elements have only to implement two extension points for the edit mode and the runtime mode.**

- **Display call up must be accomplished within the time frame of a second.**

- **Extended edit features. As a result of a survey of editing frameworks - GEF has been chosen.**

- **Support for zooming, panning and multiple graphic layers.**

Matthias Clausen, DESY
CSS – Control System Studio

Applications

- **Synoptic Display**
- **Alarm Display**
- **Archive Viewer**
- **Diagnostic Tools**

JMS based Alarm System
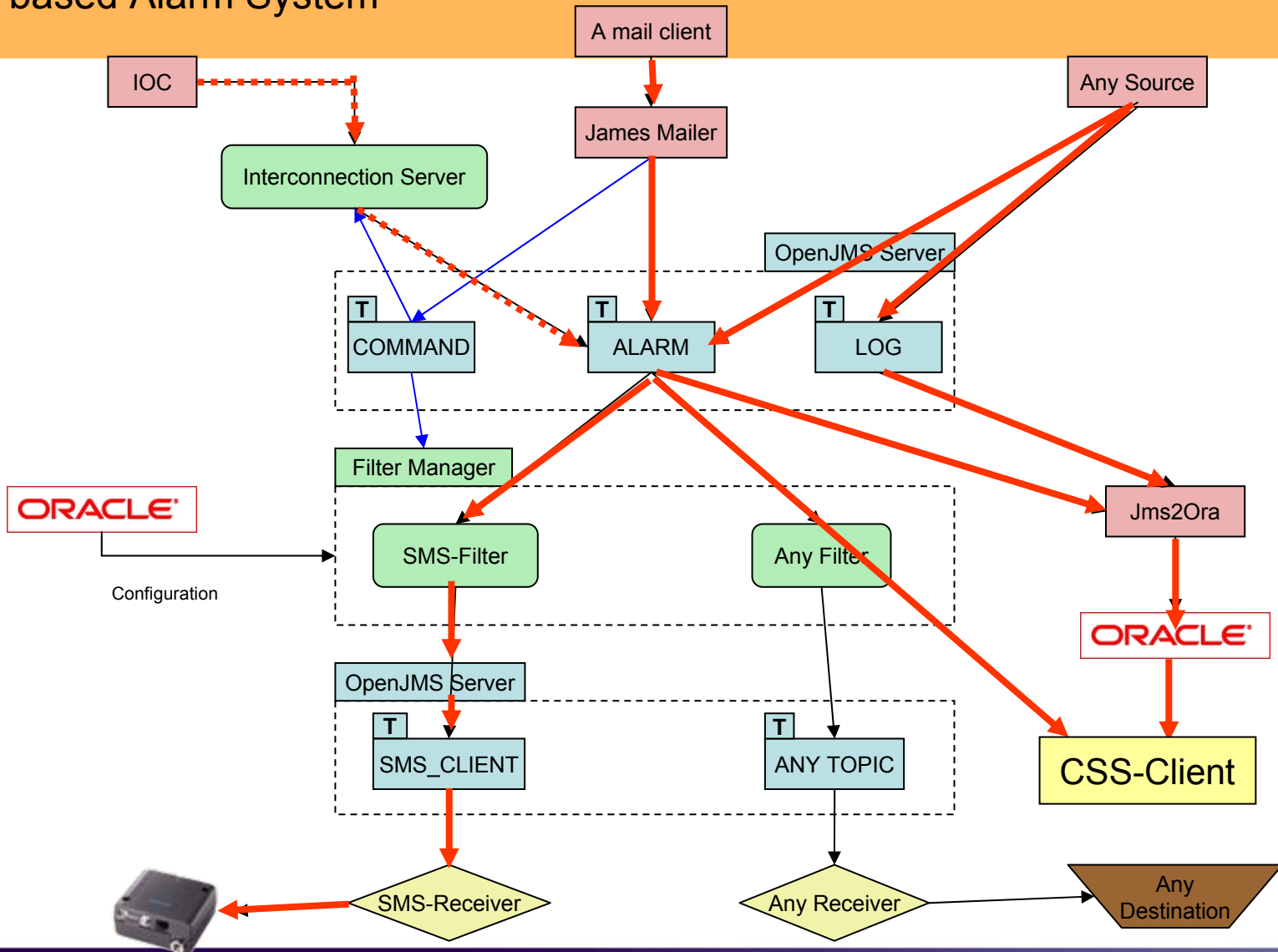
Alarm handling (CSS point of view)

Matthias Clausen, DESY
CSS – Control System Studio
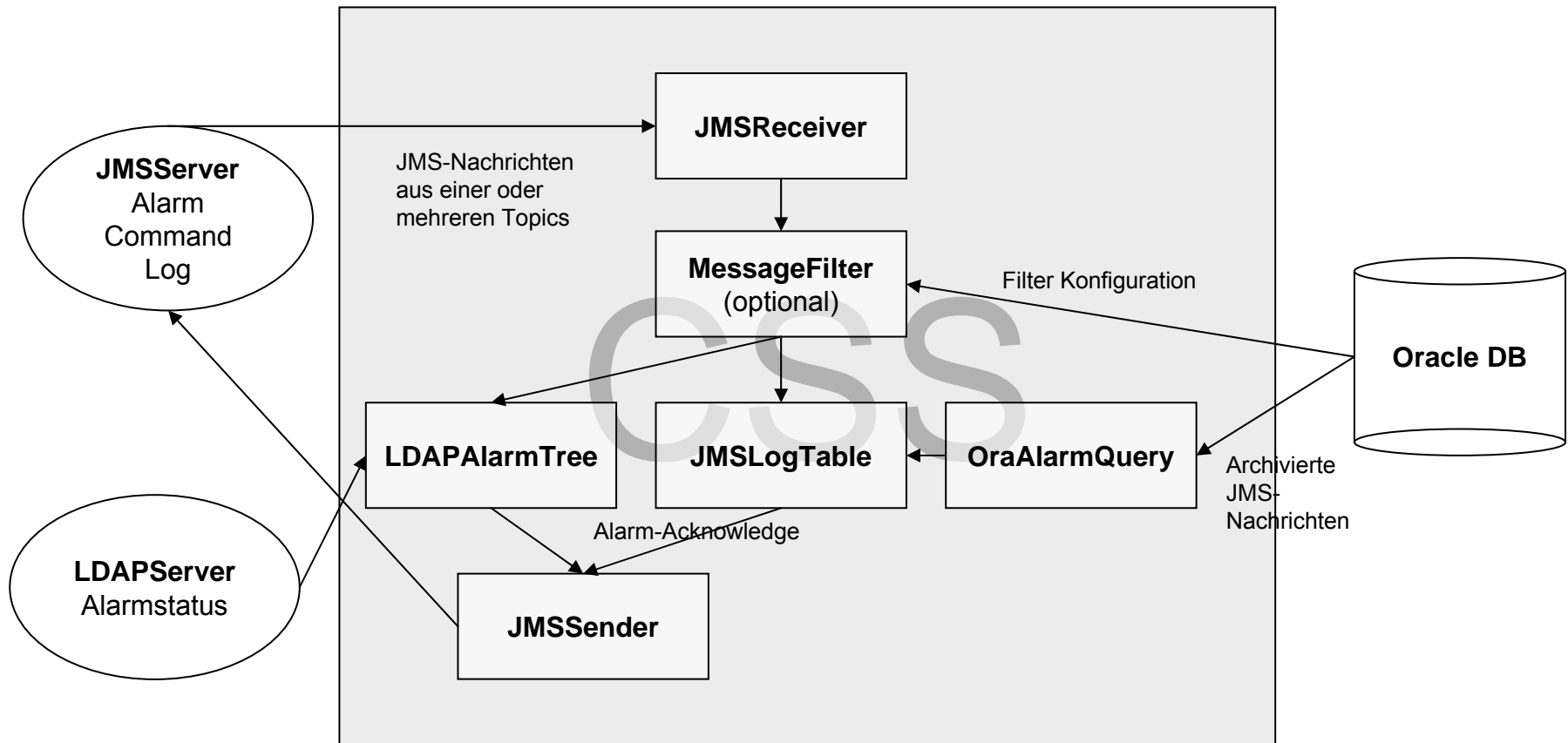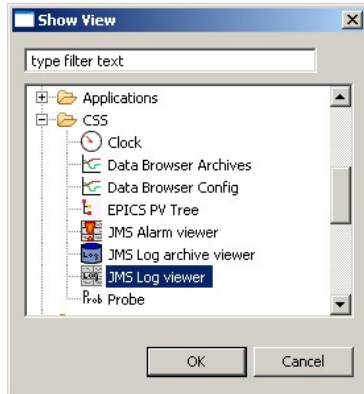
Log viewer plug-in 1

- Log viewer plugin offers three views:
  - JMS Log view
  - JMS Log archive view
  - JMS Alarm view
- All views use the same data model

JMS Log view

- Displays every new message on top of the table
- Color rule for the severity of messages
- Context menus for css-datatypes (obejct contribution)

Log viewer plugin 2

JMS Alarm view

- Displays only messages of type „Alarm"
- First sort criteria is „Severity", second is „Eventtime"
- Alarm acknowledge (not yet implemented)

JMS Log archive view

- Displays log messages archived in the Oracle database
- Offers buttons with predefined time periods
- Popup window to set special time periods

Applications

- Synoptic Display

- Alarm Display

- **Archive Viewer**

- Diagnostic Tools

Matthias Clausen, DESY
CSS – Control System Studio

Archive Viewer

Archive Viewer by Sergei

- Existing Swing based viewer will be modified to support also Eclipse
- JFreeChart /Swing) -> draw to memory -> display image in Eclipse
- Status: ready by 5'2007

Archive Viewer by Kay Kasemir/ Albert Kagarmanov

- Supports AAPI
- Supports Eclipse contributions and CSS DaD Data Types
- Status: prototype – end of 2006

Archive Analyser

- Query for archive data with special conditions
- Status: collecting requirements

Applications

- **Synoptic Display**
- **Alarm Display**
- **Archive Viewer**
- **Diagnostic Tools**

Existing CSS Tools (subset)

Matthias Clausen, DESY
CSS – Control System Studio
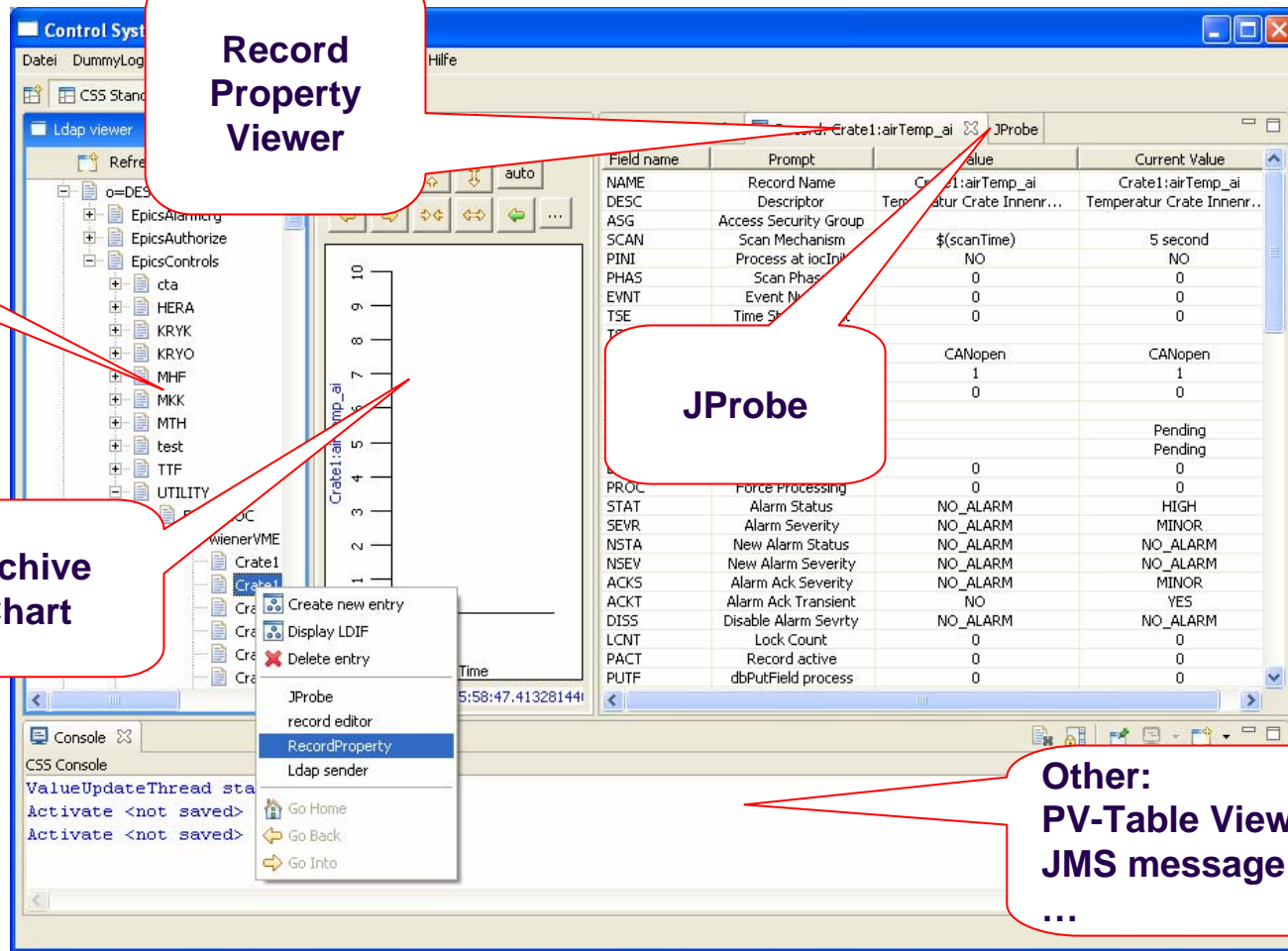
CSS Development/ Collaborating

Our Partners and their tasks:

Individuals:

- **Kay Kasemir, Sergei Chevtsov, Emma Shepherd, …**
- **Basic design, tools (archive viewer …)**

JSI (Josef Stefan Institute - Slovenia):

- **Students in Collaboration with Cosylab**
- **Basic design, graphic libraries, management tools, login, secure password store**

Cosylab:

- **Data Access Layer (DAL), coodination of students work**

C1-WPS (spin-off from University of Hamburg):

- **Basic design, CSS-Core refactoring, CSS-Wizard, synoptic display studio (SDS)**

University of Hamburg:

- **Students writing their diploma thesis**
- **Authorization in CSS**
- **CSS record/ playback**

DESY:

- **Logging, alarming, database integration, managing CSS**

Matthias Clausen, DESY
CSS – Control System Studio

Coordinating CSS Developments

Code repository:
- CVS repository at DESY
- 'Open' for registered users with DESY account (~15 external)

Mailing List:
- css-core@desy.de

Web Page:
- http://css.desy.de
- http://www.cs-studio.org

Collaborative Tools (bug tracker…)
- CodeBeamer -> http://elogbook.desy.de:8181

Collaborative Software Development
-> Documentation

Matthias Clausen, DESY
CSS – Control System Studio

Collaborative Software Development
-> Tracker

Matthias Clausen, DESY
CSS – Control System Studio

Collaborative Software Development
-> Tracker

Matthias Clausen, DESY
CSS – Control System Studio

Collaborative Software Development
-> (Tracker) Story Cards (Agile Programming)

Matthias Clausen, DESY
CSS – Control System Studio

Collaborative Software Development
-> Tracker / Story-Cards

Exporting Tracker for
Agile Project Meetings

Set Priorities
To
'Story-Cards'

ID	P	Summary	Status	Owner	Submitted at	Assigned	Submitted	Start Date	End Date	Estimated	Spent Hou	% Spent /	Category	Severity
1068	High	Model Editor			25.10.2006 18:47		awill			4		0,00%		Critical
1076	Normal	Dynamic behaviour of			25.10.2006 18:50		awill							
1075	Normal	Dynamic behaviour of			25.10.2006 18:50		awill							
1074	Normal	Dynamic behaviour of			25.10.2006 18:49		awill							
1073	Normal	Rendering mode			25.10.2006 18:49		awill							
1072	Normal	Introduce a line model			25.10.2006 18:49		awill							
1071	Normal	Introduce an ellipse m			25.10.2006 18:49		awill							
1070	Normal	Introduce a rectangle			25.10.2006 18:49		awill							
1069	Normal	Direct Manipulation of			25.10.2006 18:48		awill							
1067	Normal	DAL-Extension of the			25.10.2006 18:47		awill							
1066	Normal	Integration into the CS			25.10.2006 18:47		awill							
1065	Normal	Simple file browser vie			25.10.2006 18:47		awill							
1064	Normal	Introduce workspace			25.10.2006 18:46		awill							
1063	Normal	Load/store the data m			25.10.2006 18:46		awill							
1062	Normal	Extensibility of the da			25.10.2006 18:46		awill							
1061	Normal	Design a simple data			25.10.2006 18:45		awill							

Enabling Collaboration

Free access to cvs repository

Basic design criteria

- No hard coded configuration parameter

Matthias Clausen, DESY
CSS – Control System Studio

CSS Configuration

Eclipse preference pages are used to define current values of CSS plug-ins. (example: EPICS CA-Settings)

Preference pages allow the flexible implementation of plug-ins independent from the local runtime settings.
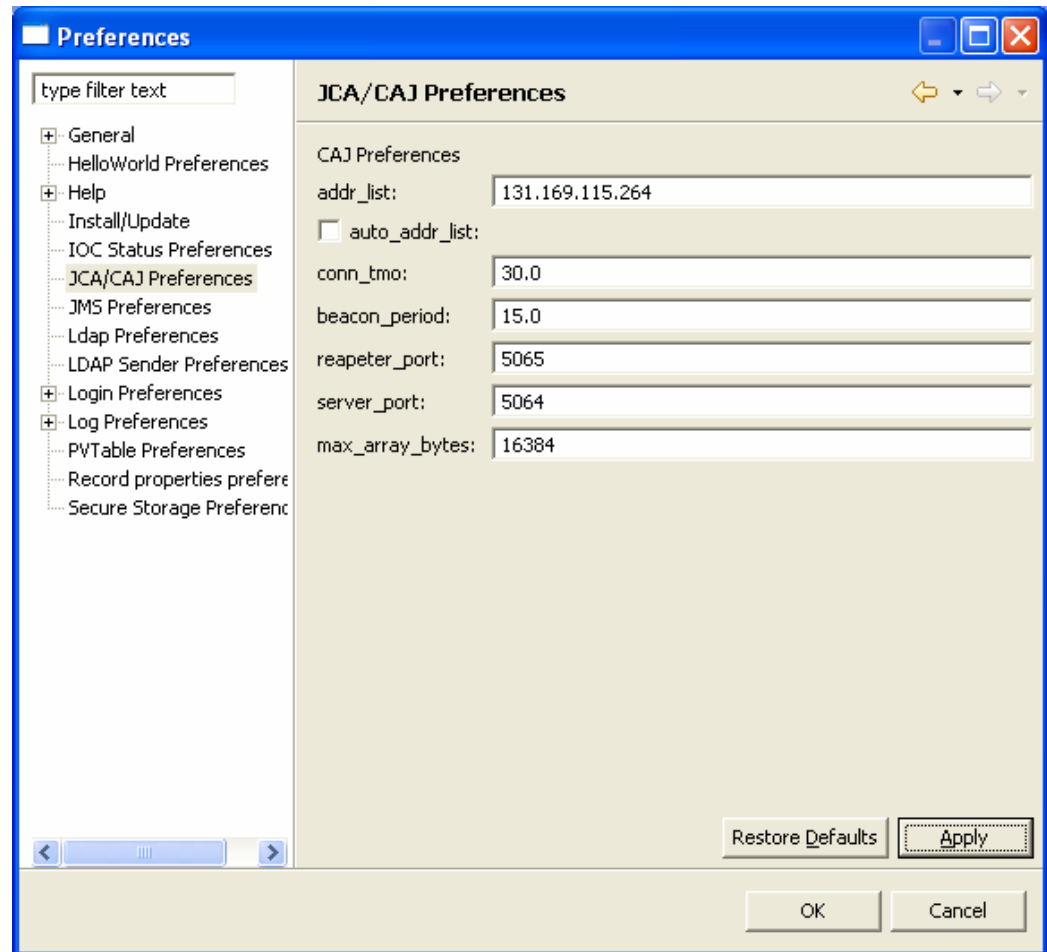
Enabling Collaboration

Free access to cvs repository

Basic design criteria

- No hard coded configuration parameter
- Do not preclude implementations with limited (no) CSS usage

CSS-Wizard

- Quick Starter for CSS plug-in developments
- Individual wizards to describe specific functionalities
  - ⇒ **Drag and drop**
  - ⇒ **Contributions …**
- 'Encourage' the developer to use CSS tools, to enter help pages, to fill the multi language files …

Matthias Clausen, DESY
CSS – Control System Studio

Schedule

10' 2006
- CSS-core design finished (C1-WPS)
- CSS refactoring started (C1-WPS)
- CSS Wizard started (C1-WPS)
- Authorization configuration ready (DESY – IT)

11' 2006 (end)
- CSS refactoring, wizard - beta ready to publish (C1-WPS)
- DAL V 1.1 finished (incl. CA plug) (Cosylab)
- Alarm viewer (incl. tree view) beta ready (DESY)
- Kay's utility tools (Kay Kasemir)

12' 2006
- SDS first working version (reduced functionality) (C1-WPS)
- CSS update site open to public (DESY / C1-WPS)

1st Q 2007
- Management plug-in (JSI-Students)
- Authorization beta (Univ. Hamburg-Students)

2nd Q 2007
- Sergei's archive viewer (Sergei)

Matthias Clausen, DESY
CSS – Control System Studio

Outlook

CSS is leaving the 'vapour ware' cloud

CSS is getting real

First test version(s) this year

Hope it will attract others to join this initiative

… or just give us input to improve the product

Matthias Clausen, DESY
CSS – Control System Studio

What is Missing?

What did I forget to mention?

CSS itself is independent from

EPICS

(Therefore it's not called EPICS-Office any more)
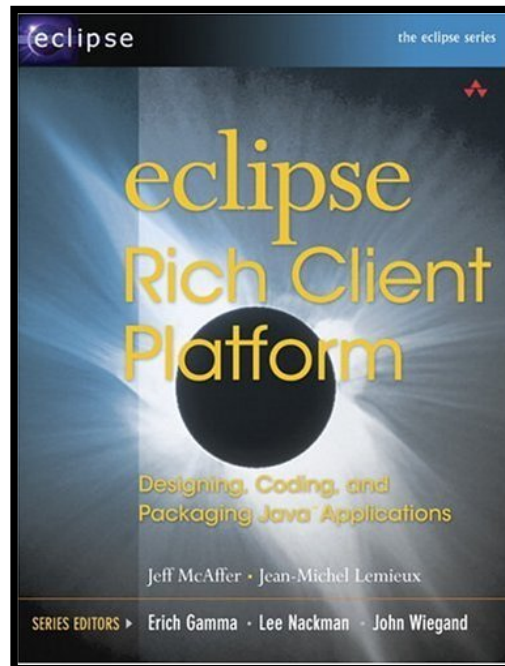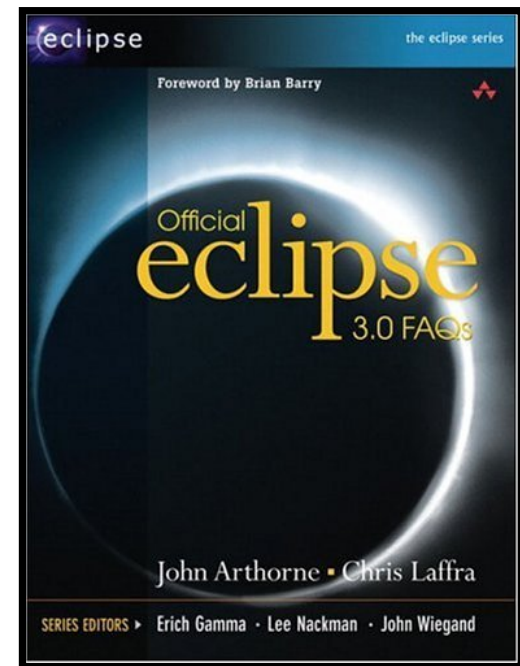
Any idea for a CSS-Logo?

Matthias Clausen, DESY
CSS – Control System Studio

Useful Books

Excellent, Must have
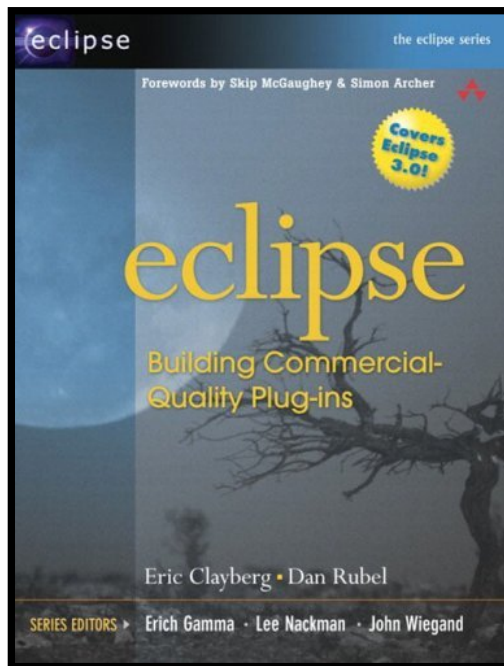
Only RCP book

For the Help Plug-in

http://www.cs-studio.org

Matthias Clausen, DESY
CSS – Control System Studio