

The CSS Story

PCaPAC

Kolkata, December 5th, 2012

Matthias Clausen, DESY Cryogenic Control Group



Overview

- What is Control System Studio?
- What started this project?
- How we started
- New Technologies
- New development strategies
- Building Collaboration
- Collaborative Tools
- Collaboration: The real life
- CSS Users
- Outlook



What is Control System Studio (CSS)?

- A Control System User Interface
- A Configuration Tool
- A Toolbox
- An Architecture
- A Collaboration
- A Set of Java Programs
- Consists of Eclipse RCP (Rich Client Platform) Plugins

• Yes – all of this – and more





Some CSS Examples





The European

CSS apps: Watching the running facility





CSS apps: Watching trends



The European X-Ray Laser Project

CSS apps: Monitoring alarms

🛰 Control System Studio	
File CSS Quickstart Window Help	
i 📬 🚽 🔚 i 🖢 = 🖗 = i 🚾 i 💽 i 🍫 i 🚳 📄	
🖹 🖪 CSS Standard	
Alarm Tree X N I X X Z I I	Individual and aggregated alarms: Current state and previous unnoticed state
	Sorted according to severity
	Contea according to seventy
📕 Alarm Table 🔀 🛛 😨 🗖 🗖	
Alarme bestaetigen Best. INVALID Alarm sound Gestartet an Invalid Invalid Inv	
A. COUNT TYPE EVENTTIME N	
181 event 2010-11-16 13:34:49.632 Sat r 182 event 2010-11-16 13:34:50.631 Satv r 28 event 2010-11-16 13:34:38.632 Satv r 29 event 2010-11-16 13:34:43.632 Satv	

Matthias Clausen, DESY The CSS Story, 5th December 2012



8

The European X-Ray Laser Project

CSS apps: Configuring EPICS databases





CSS apps: Programming state machines





The European X-Ray Laser Project

CSS apps: Monitoring servers and services





The European X-Ray Laser Project

CSS How did we get there? → **European XFEL**







The XFEL Tunnel







The real XFEL Tunnel





Experimental Area



Matthias Clausen, DESY The CSS Story, 5th December 2012



15

The European X-Ray Laser Project XFEL

A Basis for a new Control System User Interface must be found. It shall last for the next two decades.

The ,old' EPICS view



Many individual client applications



The new CSS view:

- Not bound to a single control system
- Consistent Look&Feel
- Integration of tools while staying focused on the task at hand
- Interoperability of tools from different sites
- Interface to different control systems
- Concentration on one technology enabling our developers to do the job



The CSS Story starts:

• 2001 JoiMint

Java Operator Interface and Management Integration Tool

- Created during sabatical @ SLAC
- Proof of concept using Java as a platform
- Still using channel access ,C' implementation through JNI
- 2005 '08 Control System Office Proposal (Cosylab*)
- 2006 '02 DAL** API Proposal (Cosylab*)
- 2006 '08 Control System Studio Workshop at DESY



The CSS Story starts:

• 2001 JoiMint

Java Operator Interface and Management Integration Tool

- Created during sabatical @ SLAC
- Proof of concept using Java as a platform
- Still using channel access ,C' implementation through JNI
- 2005 '08 Control System Office Proposal (Cosylab*)
- 2006 '02 DAL** API Proposal (Cosylab*)
- 2006 '08 Control System Studio Workshop at DESY



CSS Workshop Participants



Jan Judy Ken Emma Matthias Kay (Marlies)

The European X-Ray Laser Project

Matthias Clausen, DESY The CSS Story, 5th December 2012



19

Workshop ToDo List

CSS-Core Meeting, overview and next steps

CSS-Introduction	(C1 WPS): Idea and functionality of the main CSS plugins core, core.ui,
log, manager, startu	p.
Tasks (C1 WPS):	- more (java doc) comments in source code.
	 set up an update site for new plugin releases.

- create a real standalone CSS application ("css.exe")
- DAL (Cosylab): Architecture and interfaces are complete.

 Tasks (Cosylab):
 - complete the implementation of DAL interfaces.

 Synoptic Display: Discussion of requirements. Presentation of a test implementation in GEF (Cosylab students).

 Tasks (C1-WPS):
 - should we use GEF or SWT for synoptic display?

 - concept for the architecture of synoptic display.

Diagnostic and Configuration: Current state of development of VDCT (Emma Shepherd). Tasks (DESY, Diamond): - design paper for VDCT.

EPICS Application: Presentation of an EPICS Application plugin (Kenneth Evans). Tasks : - integration of the EPICS Application plugin in CSS.

Data Types for DnD, MB3: We want to use special CSS data types. DnD with external applications should also be possible. Compatibility with GumTree and XAL data types? Tasks (Kay Kasemir, - list of CSS data types. Jan Hatje) - example: "How to use DnD and MB3"

Alarms: Presentation of the new alarm handling and filtering (Markus Moeller, Jan Hatje). The approach should be as generic as possible (Interfaces). Tasks (Cosylab students): - Tree view for alarms in CSS

- Connection to LDAP server (performance).

Data Viewer: Collecting requirements for data viewer.

Tasks (Kay Kasemir,	- Connection to "data crawler" interface
Cosylab students)	- research for graphic library.

Record Viewer: Browser for EPICS records. Tasks (Cosylab students): - Development of the CSS plugin

Authorization / Authentication: Roles AND groups are necessary. Tasks (C1 WPS): - implementation of the new requirements.

Management: Controlling CSS instances via eclipse communication framework (ECF). Tasks (Cosylab students): - test implementations with ECF.

Many new technologies need to be explored !

- Eclipse for CSS
- GEF* / SWT for SDS**
- Data Access Layer (DAL)
- VDCT as a CSS plugin?
- Data Types compatibility issues (XAL?/ EPICS!)
- Common Alarm issues
- Authentication/ Authorization



We try to finish the prototype of CSS by the end of this year.

Workshop ToDo List

CSS-Core Meeting, overview and next steps

CSS-Introduction (CI	WPS): Idea and functionality of the main CSS plugins core, core.ui,
log, manager, startup.	
Tasks (C1 WPS):	more (java doc) comments in source code.
	set up an update site for new plugin releases.

- create a real standalone CSS application ("css.exe")
- DAL (Cosylab): Architecture and interfaces are complete.

 Tasks (Cosylab):
 - complete the implementation of DAL interfaces.

 Synoptic Display: Discussion of requirements. Presentation of a test implementation in GEF (Cosylab students).

 Tasks (C1-WPS):
 - should we use GEF or SWT for synoptic display?

 - concept for the architecture of synoptic display.

Diagnostic and Configuration: Current state of development of VDCT (Emma Shepherd). Tasks (DESY, Diamond): - design paper for VDCT.

EPICS Application: Presentation of an EPICS Application plugin (Kenneth Evans). Tasks : - integration of the EPICS Application plugin in CSS.

Data Types for DnD, MB3: We want to use special CSS data types. DnD with external applications should also be possible. Compatibility with GumTree and XAL data types? Tasks (Kay Kasemir, - list of CSS data types. Jan Hatje) - example: "How to use DnD and MB3"

Alarms: Presentation of the new alarm handling and filtering (Markus Moeller, Jan Hatje). The approach should be as generic as possible (Interfaces). Tasks (Cosylab students): - Tree view for alarms in CSS

- Connection to LDAP server (performance).

Data Viewer: Collecting requirements for data viewer.

Tasks (Kay Kasemir,	- Connection to "data crawler" interface
Cosylab students)	- research for graphic library.

Record Viewer: Browser for EPICS records. Tasks (Cosylab students): - Development of the CSS plugin

Many new technologies need to be explored !

- Eclipse for CSS
- GEF* / SWT for SDS**
- Data Access Layer (DAL)
- VDCT as a CSS plugin?
- Data Types compatibility issues (XAL?/ EPICS!)
- Common Alarm issues
- Authentication/ Authorization
- * Graphical Editing Framework
- ** Synoptic Display Studio



Authorization / Authentication: Roles AND groups are necessary. Tasks (C1 WPS): - implementation of the new requirements.

Management: Controlling CSS instances via eclipse communication framework (ECF). Tasks (Cosylab students): - test implementations with ECF.



Collaboration: Initial Partners for CSS





DESY HELEMPROLIZY

New Technologies: Example – GEF Graphical Editing Framework

From the ToDo list (Workshop 2006)

Presentation for Synoptic Display: Discussion of requirements. Presentation of a test implementation in GEF (Cosylab students).

New Tasks (C1-WPS'):

- should we use GEF or SWT for synoptic display?
- concept for the architecture of synoptic display.

* Commercial Partner through University of Hamburg





Matthias Clausen, DESY The CSS Story, 5th December 2012 24





Methodology: Waterfall



- How can we track the external development?
- How and when will we be able to provide feedback?
- How and when do we notice that the project goals will not be achieved in time?





Methodology: Agile software development



HELMHOLIZ REMENSIONAT

Methodology: Agile software development



Using agile practices as building blocks

- Feedback as core element
- Evaluate working software
- Employ frequent delivery
- Iteration-based planning



Methodology used during SDS development

Prototype developed in 10 weeks

- Working demonstration prototype
- Feasibility checked (GEF, Throughput)
- Concept of run- vs. edit-mode verified
- Gained confidence in external developers

Implemented in about 1 year

- Progress was demonstrated in regular intervals (monthly)
- Allowed for tracking and feedback



SDS: Example display



HELMHOLTZ

Working in a Collaboration

Give and Take between equal Partners

- Try hard to agree on changes and jointly improve existing code
- Works perfectly fine for the CSS core components
- Works fine for most of the applications



Working in a collaboration / **Developing open Source (CSS is published under EPL)**



X-Ray Laser Project

The European



Developing open Source – In a collaboration

CSS Core - Development





DESY

Code Quality in a Collaboration

The overall quality depends on the weakest part of the product

Code quality is key for a successful collaborative development

Test code (using JUnit) is mendatory

• Write test code first is ideal - but wishful thinking

Continuous integration is permanently checking whether all code still compiles ok

Lessons learned: Code quality requirements add positive pressure on collaborators



Code Repository – I CVS Overview



Concurrent Version System (CVS)

- Revision Control for CSS sources since 2006
- Client Server repository model
- One server for all developers
- Bad performance for external developers
- Branching and tagging takes a lot of time
- Exchange of experimental developments to individual developer is complex
- Integrated in Eclipse standard distribution
- Maintained but development is stopped



Code Repository – II Git Overview



Git

- Revision Control for CSS sources since 2012
- Distributed repository model
- Each developer has his own repository
- Branching and tagging is easy and fast
- Each developer can exchange code with any other developer directly
- Different workflows are possible
- Integrated in Eclipse 4.X standard distribution
- Actively developed, large community



The European X-Ray Laser Project

CSS Source Repository

2006 - 2012



- Developers has to use DESY accounts
- CVS Server maintained by DESY
- Slow for external developers

since 2012



- Hosting service for projects using git
- Developers from different institutes can be administrators
- Provides wiki, bug-tracker, ...



Workflow @DESY



- Synchronization of DESY- and github-repository is done by developers
- DESY repository holds additional local branches (e.g. release branch)
- Exchange with external developers via github
- Build server "Jenkins" builds CSS from HEAD (Continuous Integration)
- Jenkins builds CSS Releases and Update Site



Continuous Integration with Jenkins

← → C ☆ krykhudsona	desy.de:808	2/job/0	CSS CI/				
🗀 Bearbeiten 🦳 IT 🦳 Infos 🎼 Leo	🗀 Orga 🧲	DESY	8 GCaldr	🔣 Gmaps	REOLINGUS	5 🕒 Bis bald	Barmbek!
Jenkins							
Jenkins CSS_CI							
A Back to Dachboard							
Chature	Proj	ect	css_	CI			
Status	CSS Cor	ntinuous	Integratio	n + JUnit ui	plugin/headle:	ss/unit tests	
Changes							Disable Proje
Workspace		_					
Build Now		l w	orkspace				
S Delete Project	V						
X Configure	4	La	st Successf	ul Artifacts			
Checkstyle Warnings			CSS-WI	<u>n32.win32.</u> ;	(86.21 <u>0</u> 101	,87 MB 📖	
· · · · · · · · · · · · · · · · · · ·	0000000) 	cent Chanc	100			
Build History (tren		<u>IXC</u>	cont ondite	103			
#350 Nov 21, 2012 1:56:51 PM #349 Nov 20, 2012 1:26:51 PM	Unch		Droiod	+-			
#348 Nov 19, 2012 7:57:55 AM	opsu	cam	FIOJEC				
#347 Nov 16, 2012 11:10:01 AM		GetCS	StudioG				
#346 Nov 16, 2012 11:05:17 AM	Perm	alink	s				
#345 Nov 16, 2012 10:43:35 AM	• <u>La</u>	ast build	1 (#350), 5	days 0 hr a	<u>100</u>		
#344 Nov 16, 2012 10:27:18 AM	• <u>La</u> • <u>La</u>	ast stab ast succ	le build (#3 essful build	<u>350), 5 daγ:</u> I (#350), 5	<u>s O hr ago</u> days O hr ago		
#343 Nov 16, 2012 10:17:58 AM	• <u>La</u>	<u>ast faile</u> ast unsu	d build (#3 accessful bu	<u>45), 10 day</u> jild (#346),	<u>s aqo</u> 10 days aqo		
#342 Nov 16, 2012 9:21:38 AM							
#341 Nov 15, 2012 2:41:02 PM							
#337 <u>Nov 14, 2012 8:17:36 AM</u>							
#336 Nov 13, 2012 4:25:09 PM							
#335 Nov 13, 2012 1:51:06 PM							
#334 Nov 12, 2012 3:58:44 PM							
#333 Nov 12, 2012 3:39:29 PM							
#331 Nov 8, 2012 4:30:05 PM							
#330 Nov 8, 2012 8:39:27 AM							
#329 Oct 31, 2012 4:27:15 PM							
#328 Oct 30, 2012 1:35:23 PM							
🔊 RSS for all 🔊 RSS for failur	35						
-							

- Build server "Jenkins" is triggered by changes in DESY repository
- Notifies developers of build results
- Check source code format with Checkstyle
- Build artifact is available on web interface
- With Eclipse project "Tycho" Jenkins executes in the future Java tests
- Jenkins can easily managed with web browser



- We build one CSS DESY Product with all plugins for all fields of application
- Build and product management is easy
- All applications are available (cryo operators need sometimes information from development tools)
- CSS Updates are distributed via DESY update site
- Individual settings and configuration kept over update (just restart of CSS is necessary)



X-Ray Laser Project

CSS DESY Product Configuration

CSS Configuration for different environments (cryo controls, office, ...) is done with a plugin-customization file, authorization roles and workspace







What is CSS?

For Operators:

A Toolbox for:

- Operator Interface
- Alarms
- Trends
- Diagnostics

For Engineers:

A Toolbox for:

- Graphic Design
- Database Configuration
- I/O Configuration
- Writing SNL* programs

* SNL: State Notation Language



What is CSS?

For Developers:

- A Toolkit to create control system applications
- An Architecture
- A Collaboration

The Manager:

- Platform to easily add/ integrate new applications
- After a steep learning curve it pays back in multiple respekts
- An environment which is supported by multiple industrial partners
 - →Outsourcing is actually an option



Outlook

What did we achieve?

- Satisfied users
- Stable CSS core functionality
- A large set of CSS plugins (applications)
- An actively working collaboration

What we want to achieve:

- Prepare CSS for the next Eclipse Release (4.3)
 - Making use of the new Eclipse features (Web enabled)
- Continue the collaboration find new partners





The CSS Collaboration: Growing Success





the way to new energy



china eu india japan korea russia usa









Spares



The European X-Ray Laser Project

Eclipse: Typical layers



Eclipse Runtime

Java Virtual Machine

Matthias Clausen, DESY The CSS Story, 5th December 2012



46

Collaboration: What's in Core

- Domain model
- Access layer for control systems
- Logging
- Rights management
- Startup helper
- Help facility
- Menu facility
- • •

