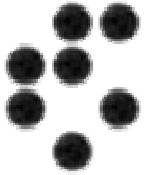


Jožef Stefan Institute

cosylab
CONTROL SYSTEM LABORATORY

ACS - the Advanced Control System

J.Dovc, M.Kadunc, I.Kriznar, M.Plesko, A.Pucelj, M.Sekoranja, G.Tkacik, I.Verstovsek, D.Vitas, K.Zagar
(IJS and Cosylab)

G.Chiozzi, B.Gustafsson, B.Jeram (ESO)

Oct. 14-17, 2002

PCaPAC 2002 Workshop

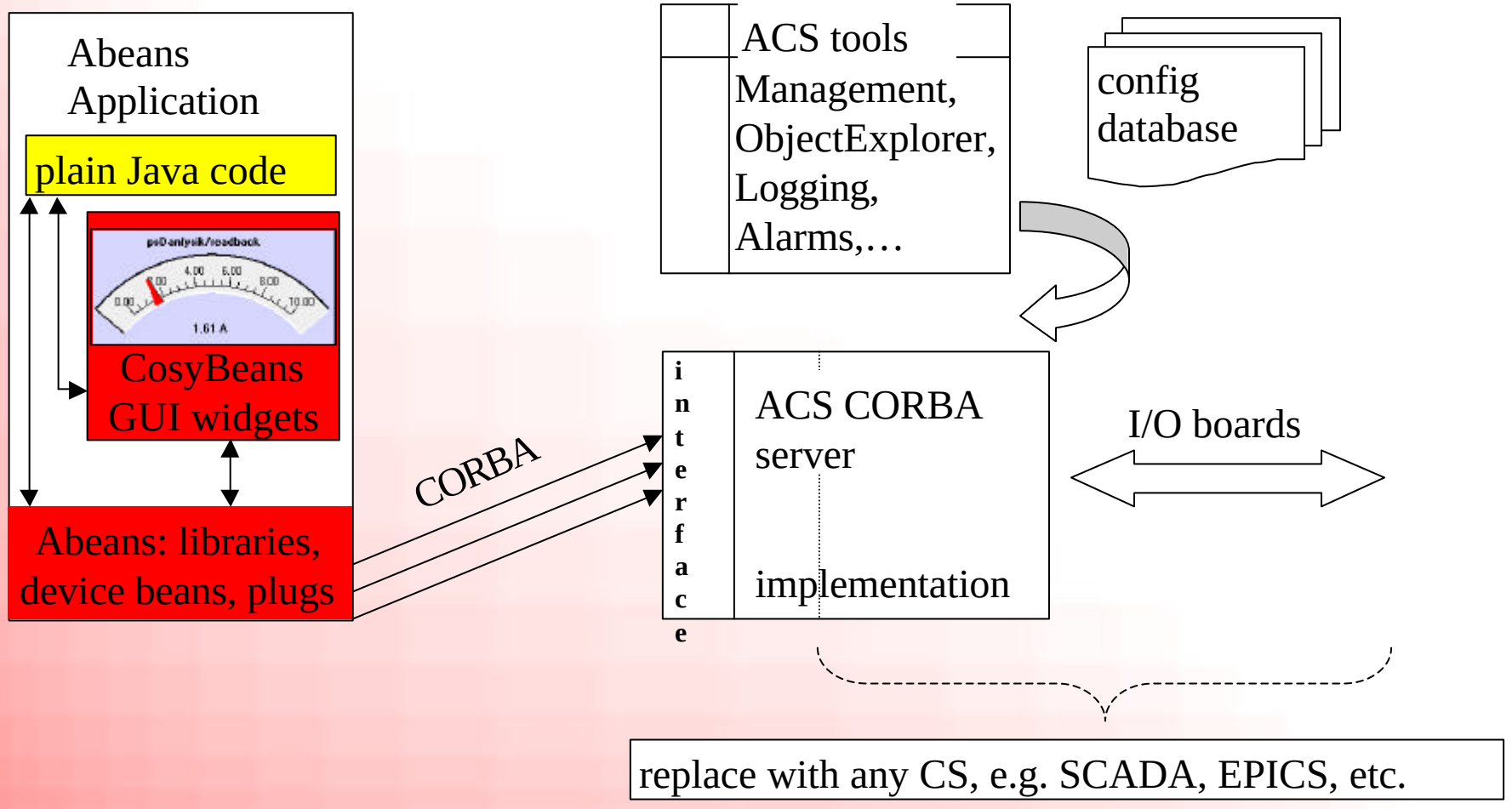
Contents

- Who are we?
- What is ACS?
- Key ideas of ACS: saying CORBA is not enough
- How are we going to go on?

Cosylab Facts

- Spin-off company from JSI (largest Slovenian institute)
  - 40 man-years experience in control systems
  - students start during undergraduate study, then join company
- Customer service oriented
  - develop, install and support complete control systems
  - custom development (software, device integration,...)
  - consulting
- Open Source Business Model
  - existing software is free to research labs
  - we charge only for the adaptation we do

Abeans and ACS

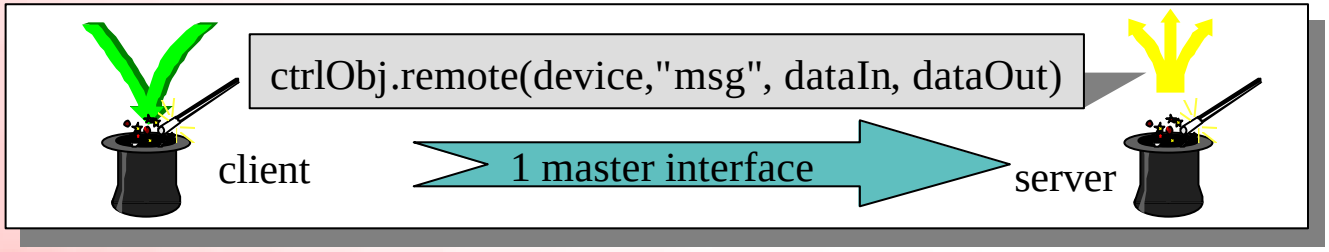


Main ACS Features

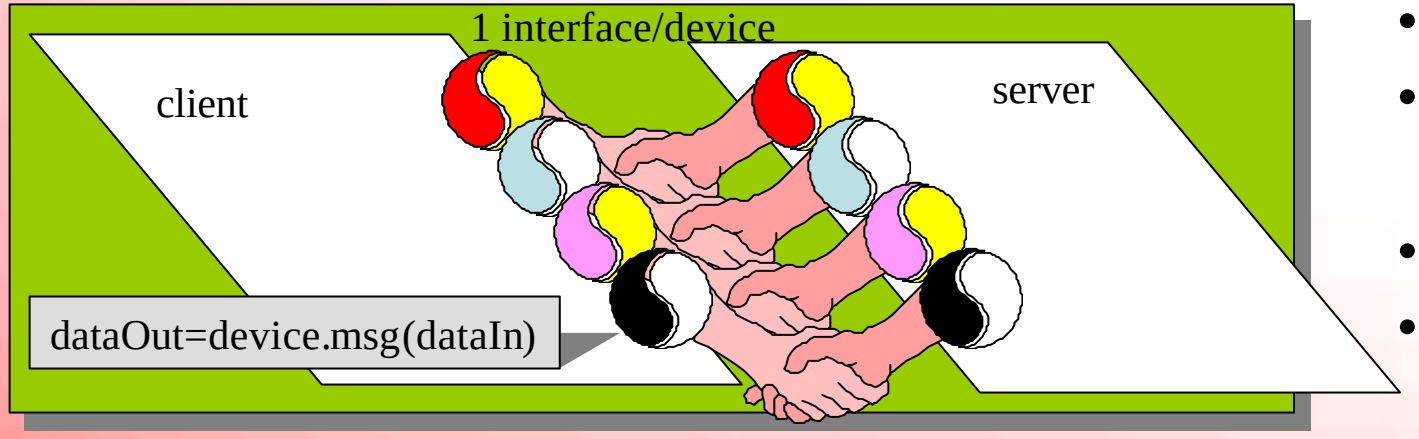
- Synchronous and asynchronous get/set
- Monitors (one to many)
- Data Notfication Channel (many to many)
- Logging
- Error Handling (exceptions, ErrStack)
- Alarm System
- Archiving System
- Management and Access Control

Basic Control Interface (BACI), or: How Do We Define the Device Interfaces

- "Narrow" (EPICS, TINE, TANGO,...)



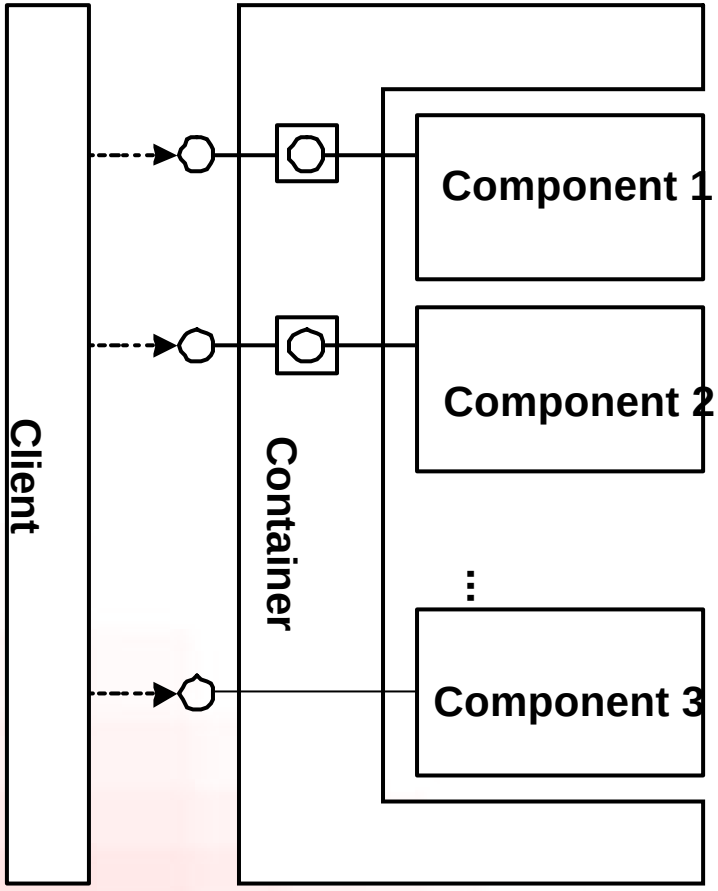
- "Wide" (ACS)



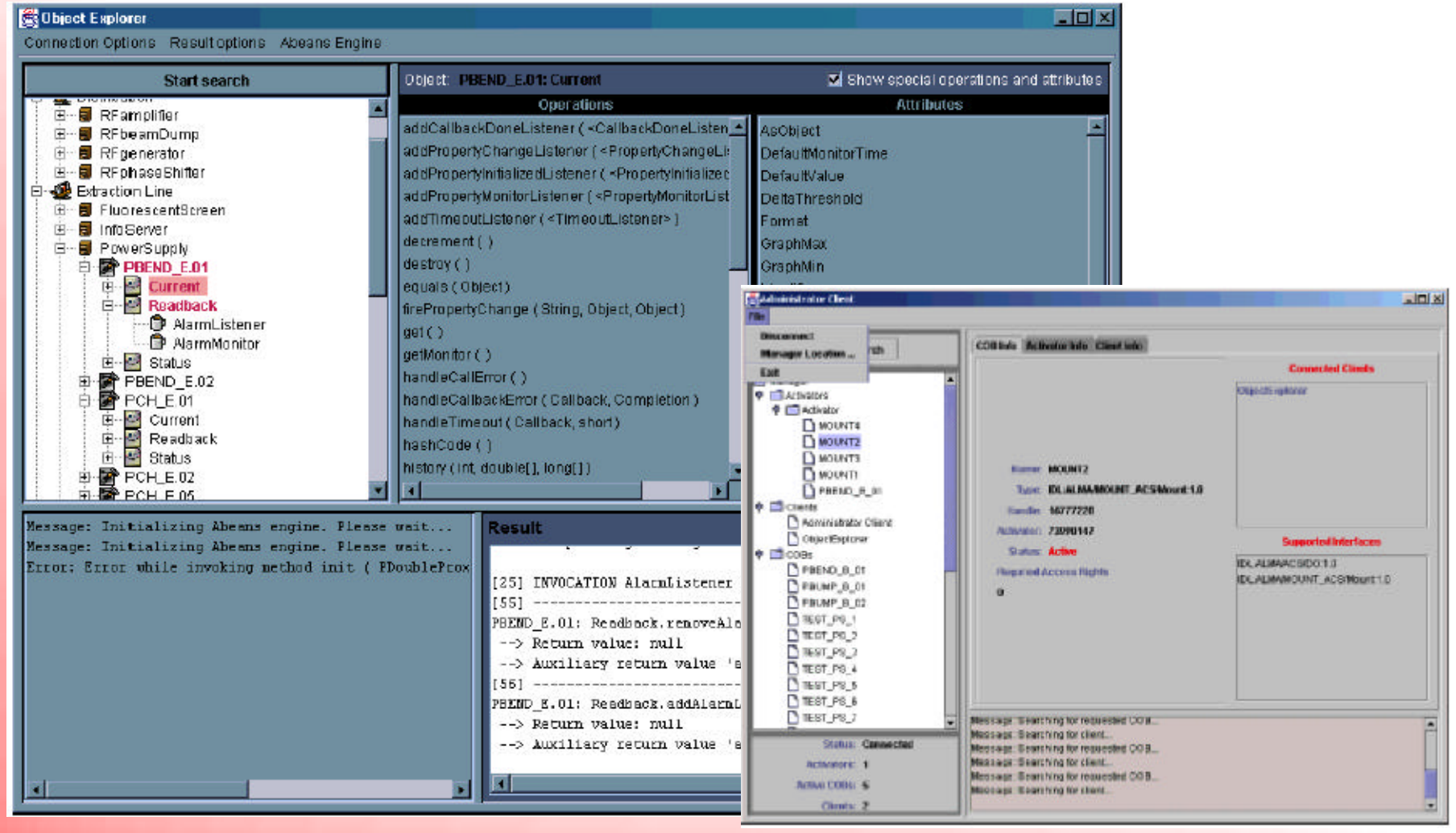
- devices
- properties
  - typed
- events
- commands

Management and Access Control Interfaces (MACI)

- Uses the Component-Container Model (CCM)
  - Same concept as in Sun's J2EE, and Microsoft's COM/COM+ and .NET
  - All added-value components live within their *containers*
  - Containers take care of management-related issues
- Generic user interface
  - Administration Client
  - Object Explorer



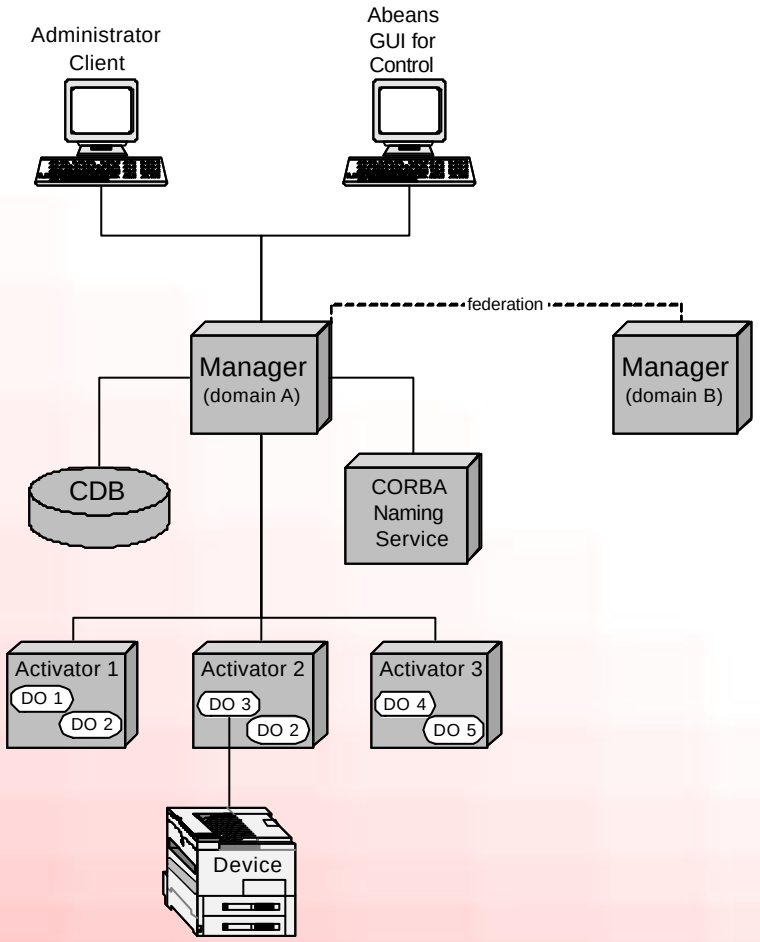
Object Explorer and Administration Client



Management Services (Components and Containers)

- Distributed object:
  - Application-specific (e.g., a model of a device in a control system)
  - The *component* in the CCM
- The Activator:
  - Takes care of distributed object lifecycle
  - Provides distributed objects with everything they need
  - The *container* in the CCM
- The Manager:
  - Implicitly takes care of control system lifecycle
  - The directory
  - One manager per *domain*

ACS Deployment diagram

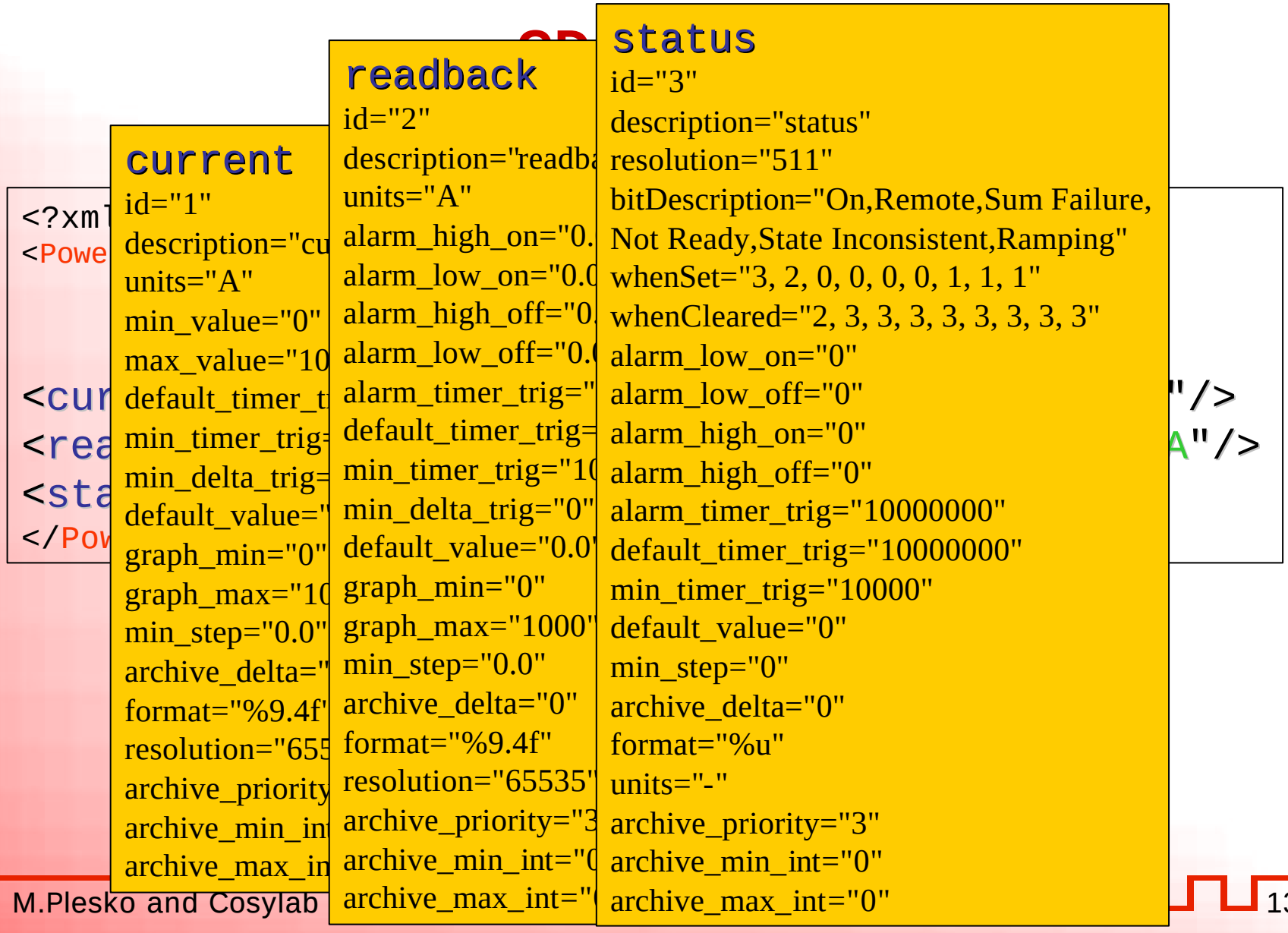


Management Services (Lifecycle and Access)

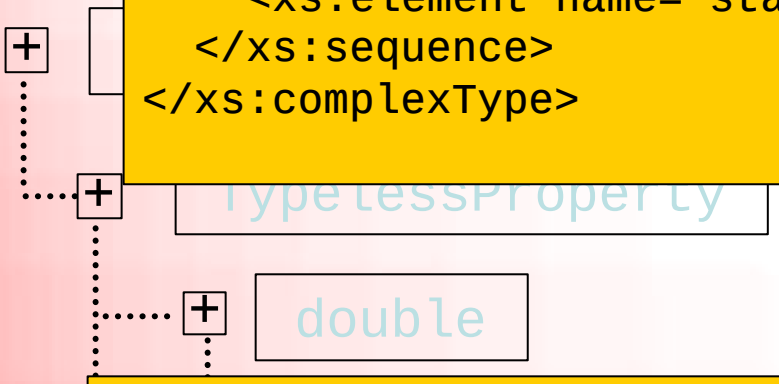
- Control system lifecycle:
  - Bringing the entire system online and offline
  - Triggering *ad hoc* activation of individual distributed objects
  - Startup sequences need not be explicitly specified
- Distributed object lifecycle:
  - Timely (*when needed*) activation of a distributed object
  - Deactivation of a distributed object that is no longer needed (*garbage collection*)
- Access to other distributed objects
- Access to the Configuration Database
- Directory of all distributed objects
  - Authorization

Configuration Database

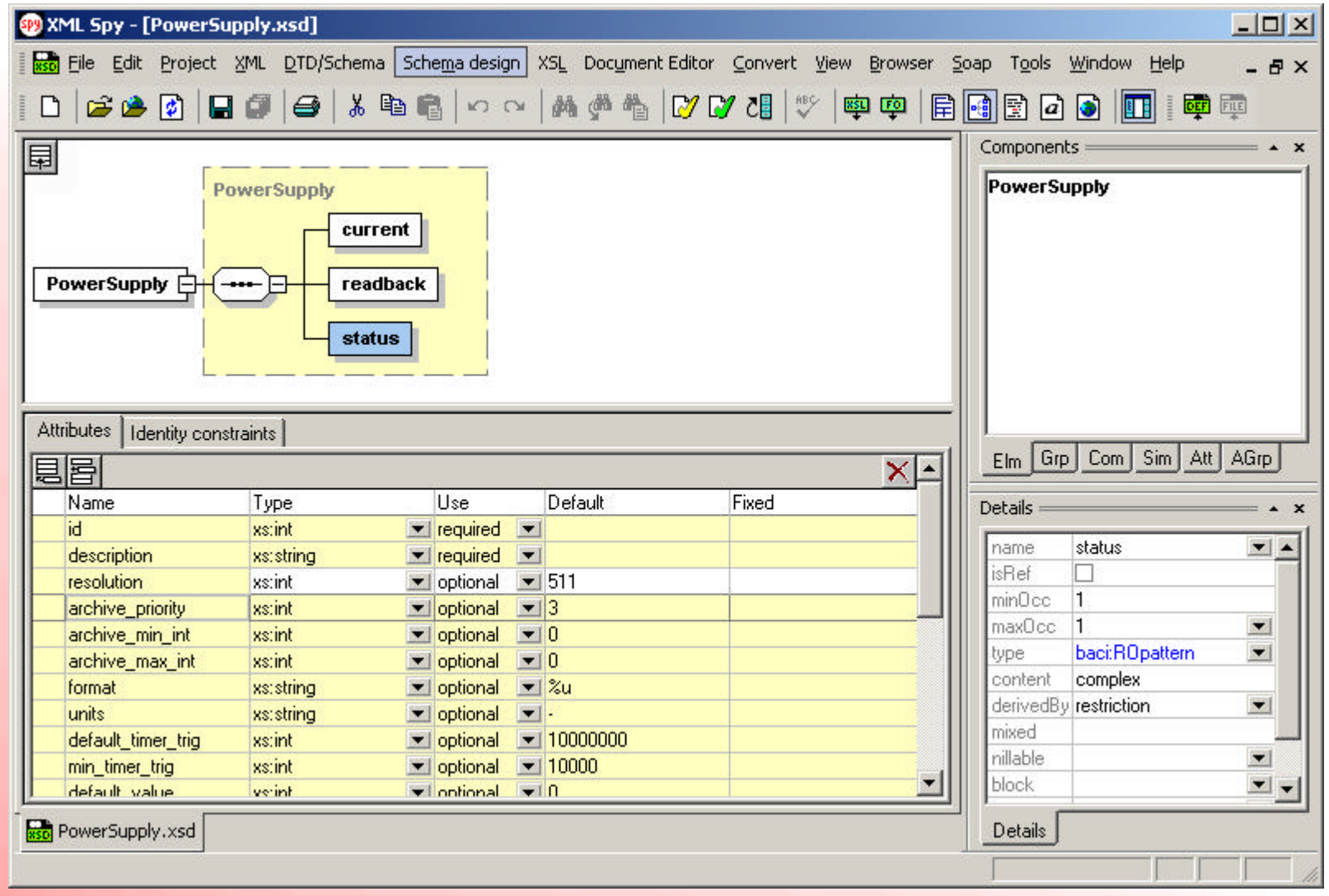
- Records form a hierarchy
  - Model of the entire system's structure
  - Addressable through Unique Resource Identifiers (URI)
- Records have a well-defined structure
  - All data in the record have assigned a data type and default value
  - Record is actually an Extensible Markup Language (XML) document associated with an XML schema (XSD)


<?xm
<Powe
<cur
<rea
<sta
</Pow
current
id="1"
description="cu
units="A"
min_value="0"
max_value="10
default_timer_t
min_timer_trig=
min_delta_trig=
default_value="
graph_min="0"
graph_max="1(
min_step="0.0"
archive_delta="
format="%9.4f"
resolution="655
archive_priority
archive_min_in
archive_max_in
readback
id="2"
description="readba
units="A"
alarm_high_on="0.
alarm_low_on="0.0
alarm_high_off="0.
alarm_low_off="0.(
alarm_timer_trig="
default_timer_trig=
min_timer_trig="1(
min_delta_trig="0"
default_value="0.0"
graph_min="0"
graph_max="1000"
min_step="0.0"
archive_delta="0"
format="%9.4f"
resolution="65535"
archive_priority="3
archive_min_int="0
archive_max_int="(
status
id="3"
description="status"
resolution="511"
bitDescription="On,Remote,Sum Failure,
Not Ready,State Inconsistent,Ramping"
whenSet="3, 2, 0, 0, 0, 0, 1, 1, 1"
whenCleared="2, 3, 3, 3, 3, 3, 3, 3, 3"
alarm_low_on="0"
alarm_low_off="0"
alarm_high_on="0"
alarm_high_off="0"
alarm_timer_trig="10000000"
default_timer_trig="10000000"
min_timer_trig="10000"
default_value="0"
min_step="0"
archive_delta="0"
format="%u"
units="-"
archive_priority="3"
archive_min_int="0"
archive_max_int="0"
"/>
A"/>

```
<xs:complexType name="PowerSupply">
  <xs:sequence>
    <xs:element name="current" type="baci:RWdouble"/>
    <xs:element name="readback" type="baci:RODouble"/>
    <xs:element name="status" type="baci:ROpattern"/>>
  </xs:sequence>
</xs:complexType>
```



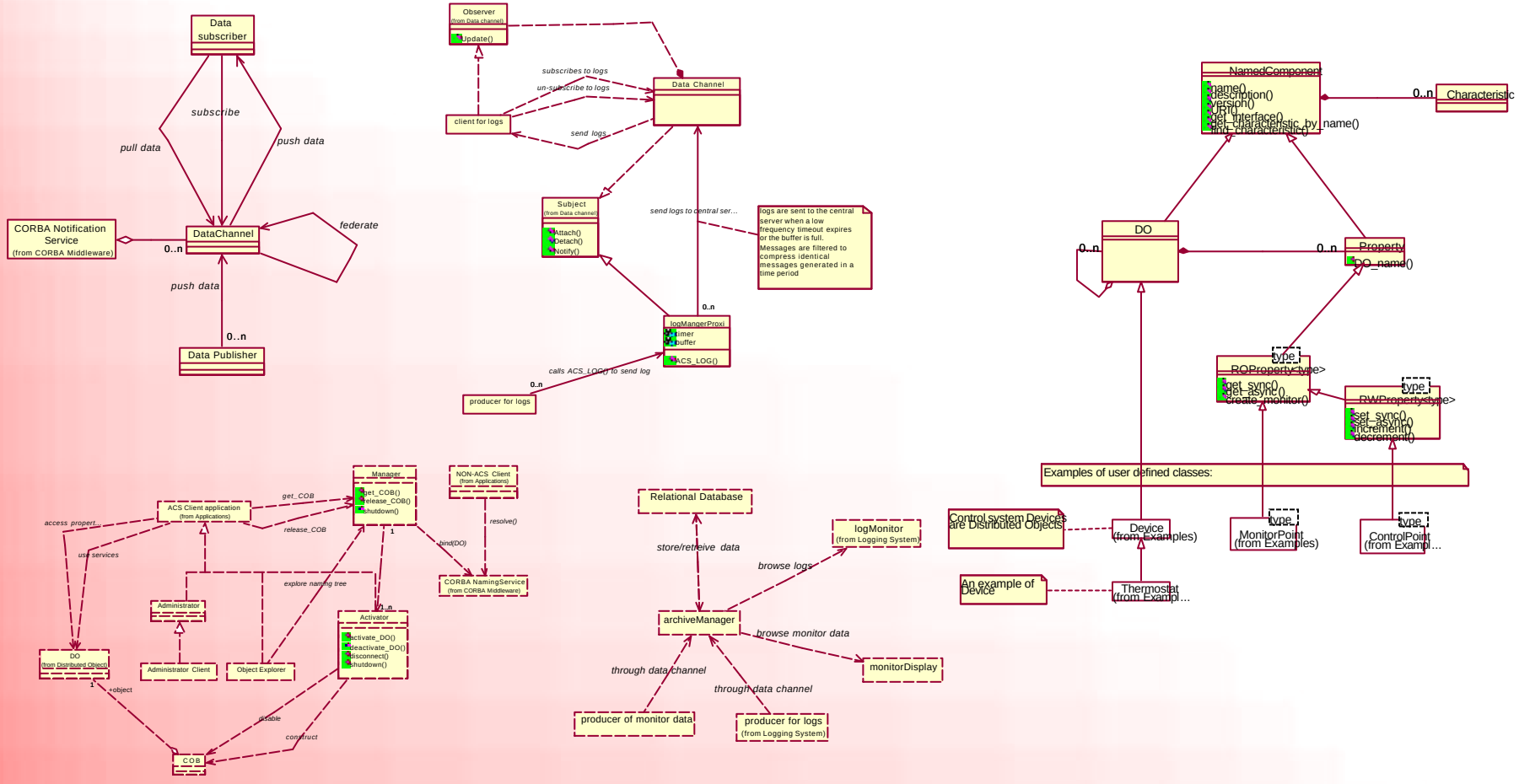

```
<xs:complexType name="RODouble">
<xs:complexContent>
  <xs:extension base="double">
    <xs:attribute name="alarm_high_on" type="xs:double" default="0.0"/>
    <xs:attribute name="alarm_low_on" type="xs:double" default="0.0"/>
    <xs:attribute name="alarm_high_off" type="xs:double" default="0.0"/>
    <xs:attribute name="alarm_low_off" type="xs:double" default="0.0"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
```


XML Spy - [PowerSupply.xsd]
File Edit Project XML DTD/Schema Schema design XSL Document Editor Convert View Browser Soap Tools Window Help
PowerSupply
current
readback
status
PowerSupply
Components
PowerSupply
Elm Grp Com Sim Att AGrp
Attributes
Identity constraints
Name Type Use Default Fixed
id xs:int required
description xs:string required
resolution xs:int optional 511
archive_priority xs:int optional 3
archive_min_int xs:int optional 0
archive_max_int xs:int optional 0
format xs:string optional %u
units xs:string optional -
default_timer_trig xs:int optional 10000000
min_timer_trig xs:int optional 10000
Details
name status
isRef
minOcc 1
maxOcc 1
type baci:ROpattern
content complex
derivedBy restriction
mixed
nillable
block
Details
PowerSupply.xsd

CORBA Services Used

- Naming service (built into manager)
- Telecom log service
- Notification service (successor of Event)
- Interface repository
- Property Service
- Implementation repository (activator)

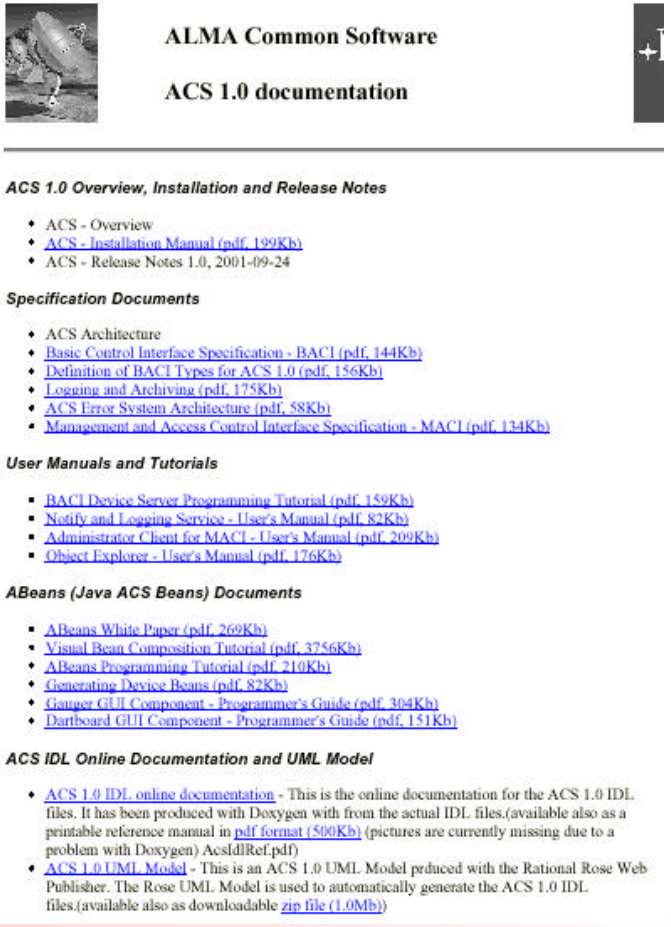
UML for Professional Quality Design



ACS 1.0 Documentation - 1

2000+ pages of printable design/reference docs

- Introductory documents (Overview, Installation, Rel Notes)
- Specification Documents
- User Manuals and Tutorials
- Abeans
- ACS IDL Online documentation



ALMA Common Software

ACS 1.0 documentation

ACS 1.0 Overview, Installation and Release Notes

- ACS - Overview
- ACS - Installation Manual (pdf, 199Kb)
- ACS - Release Notes 1.0, 2001-09-24

Specification Documents

- ACS Architecture
- Basic Control Interface Specification - BACI (pdf, 144Kb)
- Definition of BACI Types for ACS 1.0 (pdf, 156Kb)
- Logging and Archiving (pdf, 175Kb)
- ACS Error System Architecture (pdf, 58Kb)
- Management and Access Control Interface Specification - MACI (pdf, 134Kb)

User Manuals and Tutorials

- BACI Device Server Programming Tutorial (pdf, 159Kb)
- Notify and Logging Service - User's Manual (pdf, 82Kb)
- Administrator Client for MACI - User's Manual (pdf, 209Kb)
- Object Explorer - User's Manual (pdf, 176Kb)

ABeans (Java ACS Beans) Documents

- ABeans White Paper (pdf, 269Kb)
- Visual Bean Composition Tutorial (pdf, 3756Kb)
- ABeans Programming Tutorial (pdf, 210Kb)
- Generating Device Beans (pdf, 82Kb)
- Gauger GUI Component - Programmer's Guide (pdf, 304Kb)
- Dartboard GUI Component - Programmer's Guide (pdf, 151Kb)

ACS IDL Online Documentation and UML Model

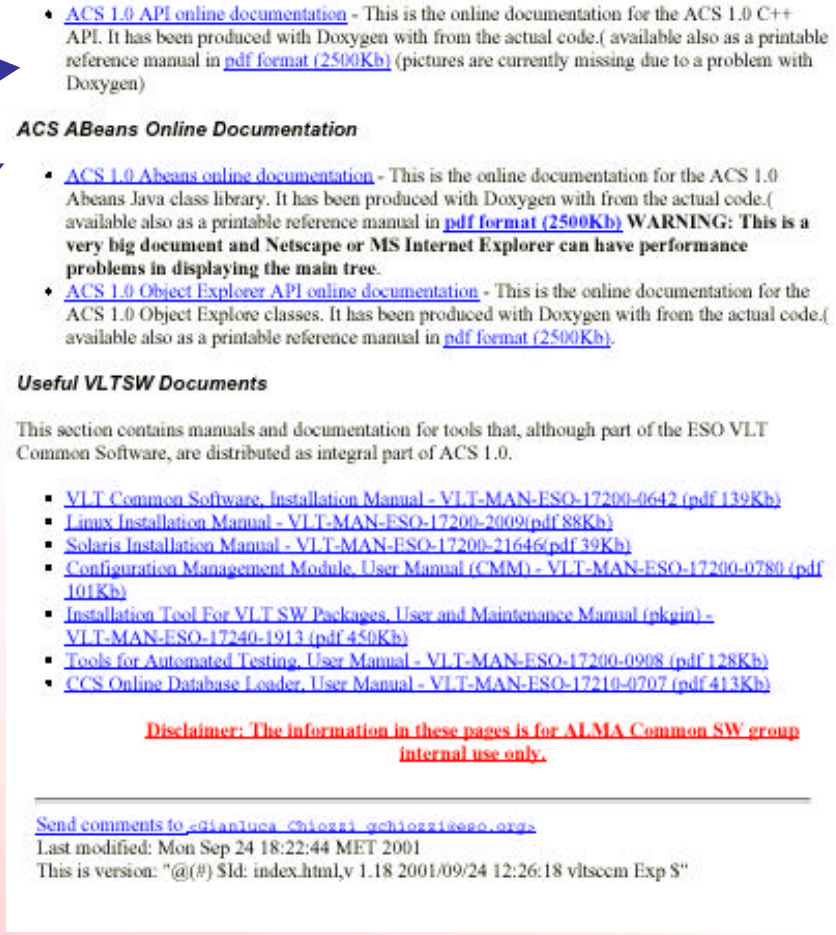
- ACS 1.0 IDL online documentation - This is the online documentation for the ACS 1.0 IDL files. It has been produced with Doxygen with from the actual IDL files.(available also as a printable reference manual in pdf format (500Kb) (pictures are currently missing due to a problem with Doxygen) AcsIdlRef.pdf)
- ACS 1.0 UML Model - This is an ACS 1.0 UML Model prduced with the Rational Rose Web Publisher. The Rose UML Model is used to automatically generate the ACS 1.0 IDL files.(available also as downloadable zip file (1.0Mb))

ACS 1.0 Documentation - 2

- ACS API Online Documentation
- ACS Abeans Online Documentation

See:

- ACS_1_0_Linux-CD/Docs
- $ACSROOT/man (online reference, html and man)



- ACS 1.0 API online documentation - This is the online documentation for the ACS 1.0 C++ API. It has been produced with Doxygen with from the actual code.(available also as a printable reference manual in pdf format (2500Kb) (pictures are currently missing due to a problem with Doxygen)

ACS ABeans Online Documentation

- ACS 1.0 Abeans online documentation - This is the online documentation for the ACS 1.0 Abeans Java class library. It has been produced with Doxygen with from the actual code.(available also as a printable reference manual in **pdf format (2500Kb) WARNING: This is a very big document and Netscape or MS Internet Explorer can have performance problems in displaying the main tree**.
- ACS 1.0 Object Explorer API online documentation - This is the online documentation for the ACS 1.0 Object Explore classes. It has been produced with Doxygen with from the actual code.(available also as a printable reference manual in pdf format (2500Kb).

Useful VLTSW Documents

This section contains manuals and documentation for tools that, although part of the ESO VLT Common Software, are distributed as integral part of ACS 1.0.

- VLT Common Software, Installation Manual - VLT-MAN-ESO-17200-0642 (pdf 139Kb)
- Linux Installation Manual - VLT-MAN-ESO-17200-2009(pdf 88Kb)
- Solaris Installation Manual - VLT-MAN-ESO-17200-21646(pdf 39Kb)
- Configuration Management Module, User Manual (CMM) - VLT-MAN-ESO-17200-0780 (pdf 101Kb)
- Installation Tool For VLT SW Packages, User and Maintenance Manual (pkgin) - VLT-MAN-ESO-17240-1913 (pdf 450Kb)
- Tools for Automated Testing, User Manual - VLT-MAN-ESO-17200-0908 (pdf 128Kb)
- CCS Online Database Loader, User Manual - VLT-MAN-ESO-17210-0707 (pdf 413Kb)

Disclaimer: The information in these pages is for ALMA Common SW group internal use only.

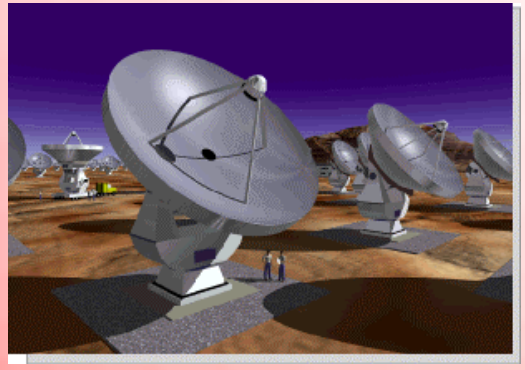
Send comments to <gianluca.chiozzi@chiozziesso.org>
Last modified: Mon Sep 24 18:22:44 MET 2001
This is version: "@(#) $Id: index.html,v 1.18 2001/09/24 12:26:18 vltsccm Exp $"

Supported Platforms

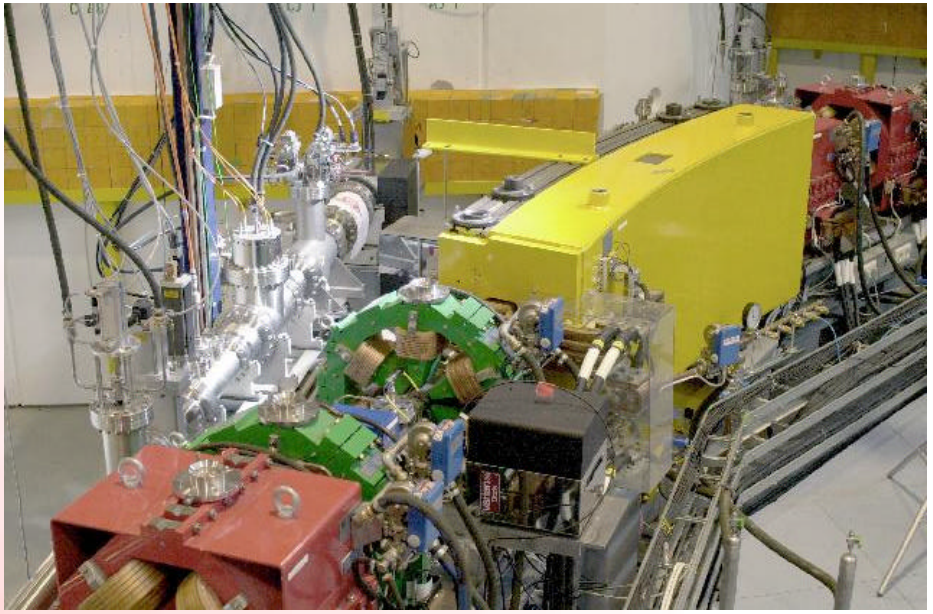
- Non-realtime operating system: Linux, SUN OS, MS-Windows
- Real-time: VME,VxWorks (Phase1), CAN bus
- Languages: C++, JAVA, Python
- CORBA middleware: TAO (& ACE) (C++), Omniorb (Python)

Two Major Installations

- ALMA Prototype
  - array of 64 radiotelescopes (EU-USA-Japan project)
  - Joint project between astronomical organisations in Europe and North America (and Japan)



- ANKA
  - Karlsruhe, Germany
  - synchrotron light source



Conclusion

- Developed based on the experience of both astronomical and accelerator control projects
- optimized for low total cost of ownership (TCO)
  - the implementation is efficient and clean
  - essential tools and services with many features
  - can easily run on many platforms
- Open source (GPL licence)
  - free development tools and ORBs

Several other projects are interested in ACS

A wider user's base will provide valuable feedback