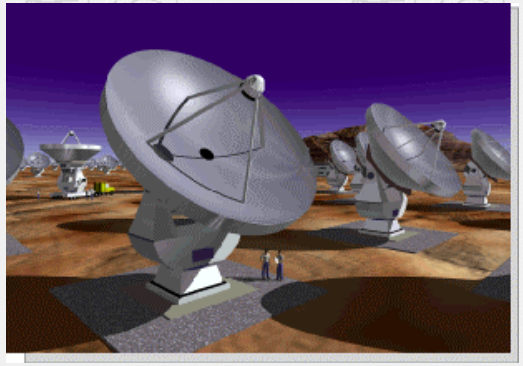


Atacama Large Millimeter Array

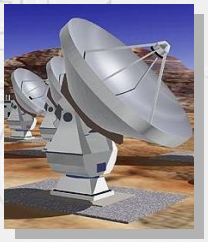
ACS

advanced Common Software for the ALMA Project

G.Chiozzi

European Southern Observatory

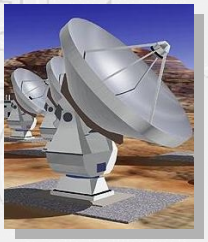
ESO Garching, 17 January 2001



ALMA Project

Summary

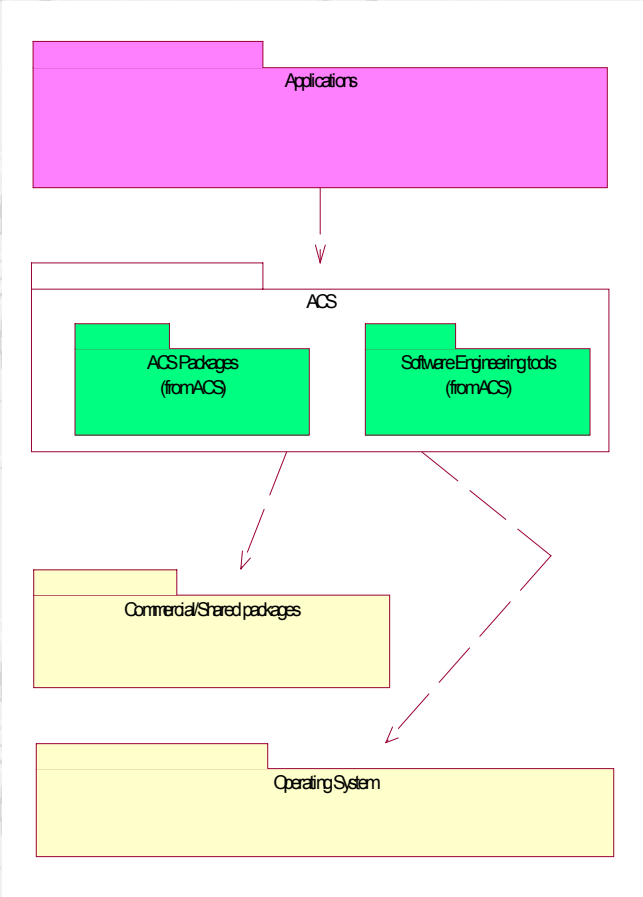
- Part 1 - ACS (60min)
  - What and why
  - Architecture
  - Impressions on new technologies used
- Part 2 - Kitt Peak 2000 test (30min)
  - What and why
  - System architecture
  - Demo
  - Conclusions
- Free discussion (30min)

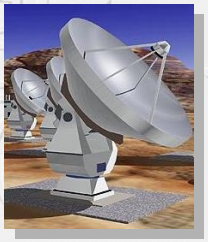


ALMA Project

What is ACS?

- ACS means:
  advanced Common Software for the **ALMA** Project
- software infrastructure located in between application software and commercial/shared software on top of the OS
- ACS includes SE tools
- ACS is **NOT** ALMA specific

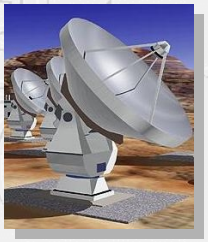




ALMA Project

A Common SW is necessary...

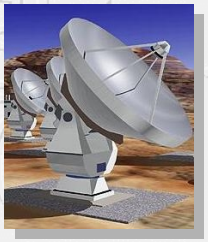
- The ESO Common SW is a key element for the success of VLT Telescope control and Instrumentation software.
- ALMA is very geographically distributed in terms of development. This poses even higher constraints and pressure on SW development.
- The Common Software should extend over the boundaries of the control system



ALMA Project

… why is it really necessary?

- Avoid duplication among development teams.
- Enforce and make transparent the standards.
- Synchronize development teams via Releases.
- Allow some degree of independence from operating system and basic SW protocols.



ALMA Project

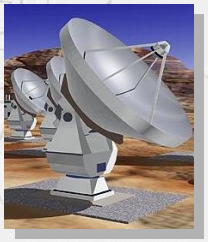
The technology under ACS

consolidated:

- Unix, SUN, VxWorks
- GNU tools
- C++, TCL
- ESO Common SW tools
- ESO VLT SW Engineering

new:

- Linux
- CORBA (ACE/TAO, Orbacus, MICO…)
- Java, JPython
- XML
- Visual Age Java IDE
- UML modeling

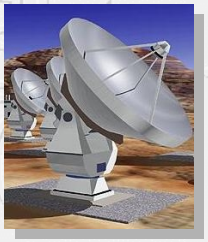


ALMA Project

Relations with VLT CCS

ACS

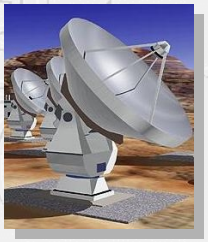
- Learns from the VLT experience
- Introduces new concepts and new technologies
- Is developed for ALMA and defined by the ALMA community
- Jumps over the “Control System” boundaries
- Reuses components from the VLT CCSLite
- Allows migrating from CCS to ACS and interoperability between the two worlds



ALMA Project

ACS Milestones and Status

- Feb 2000: ACS Requirements
- June 2000: ACS Architecture document
- Oct 2000: ACS v.0.0 (Prototype)
- Dec 2000: KP test
- ➤ Feb 2001: ACS Architecture review
- July 2001: ACS 1.0 Release
- Dec 2001: ACS 1.0 Update (support for Test Interferometer)
- An incremental release every 6 months (VLT style)

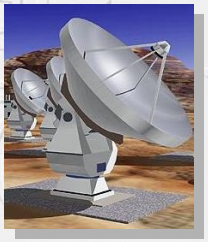


ALMA Project

Software Engineering

Based on the ESO experience, the following concepts have been included in ACS:

- standard environment
- directory structure
- acsMakefile
- CMM
- Software Problem Report

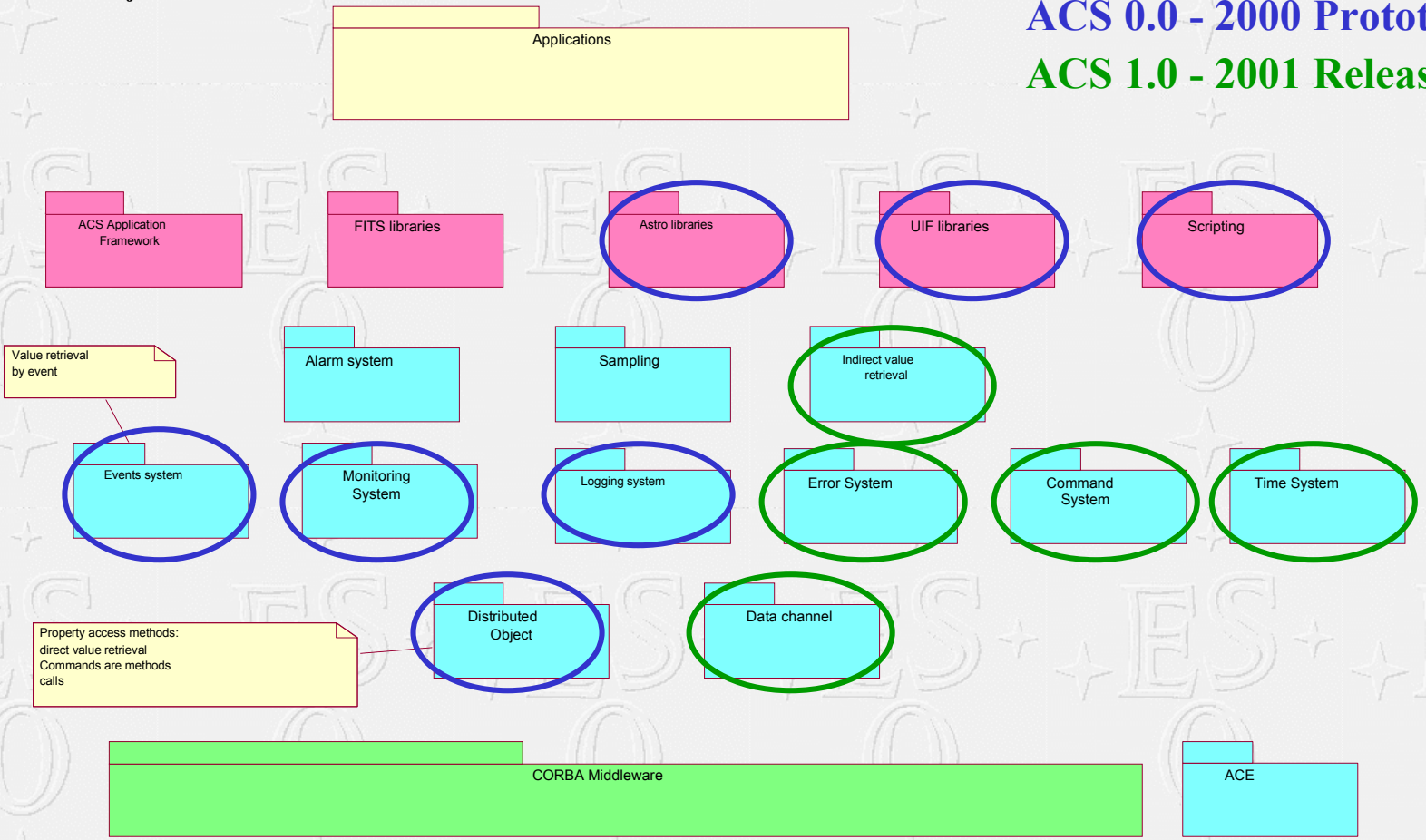


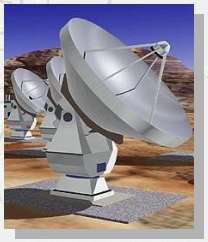
ALMA Project

ACS Main Packages

ACS 0.0 - 2000 Prototype

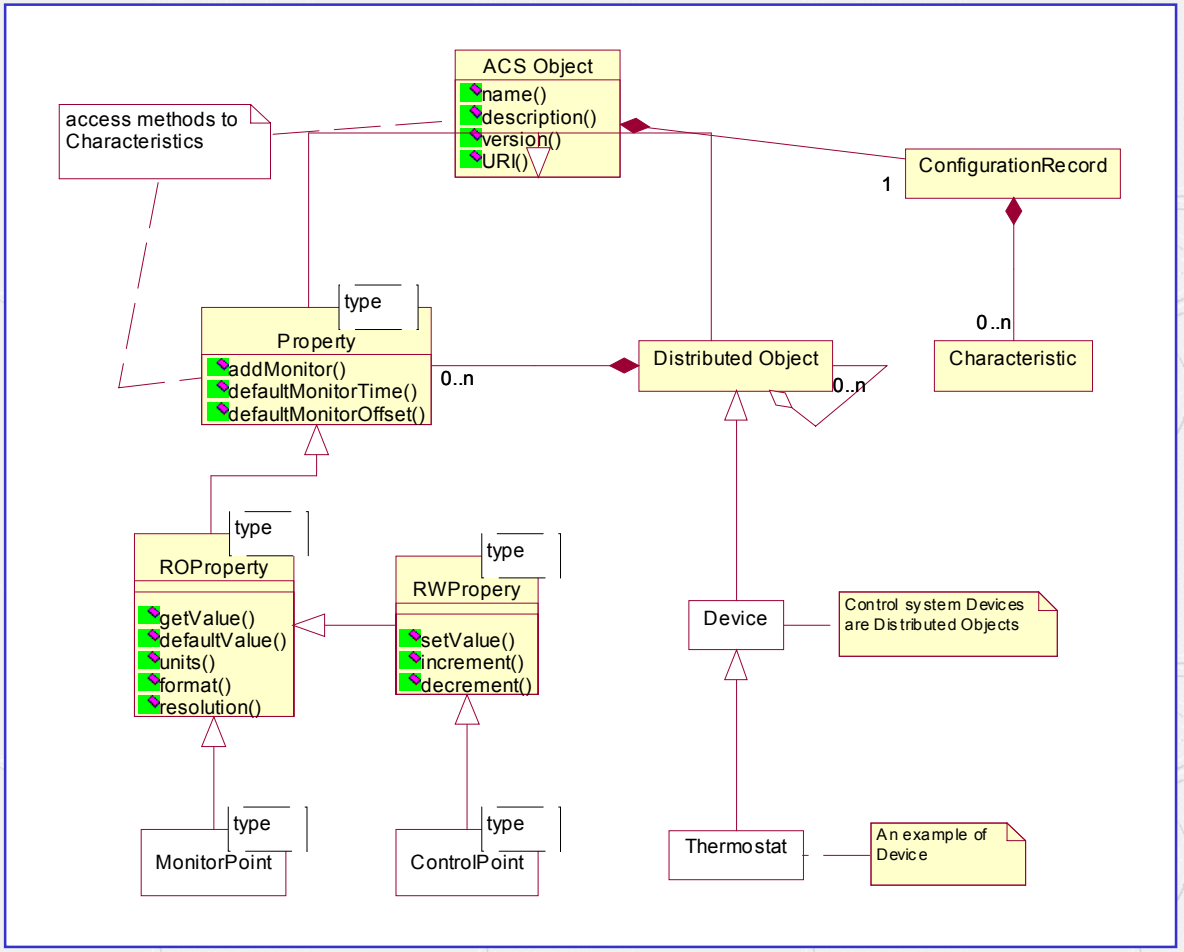
ACS 1.0 - 2001 Release

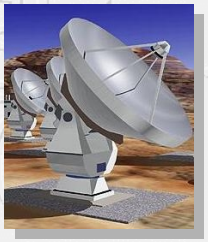




ALMA Project

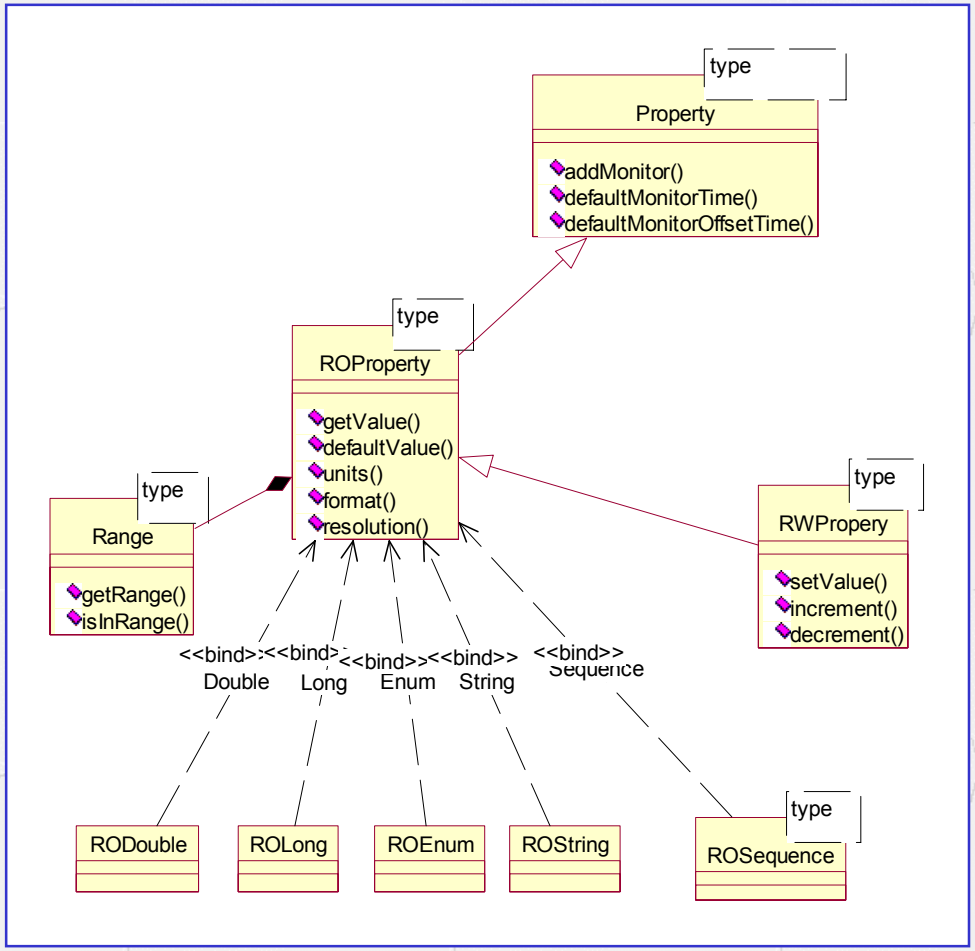
The core: Distributed Objects

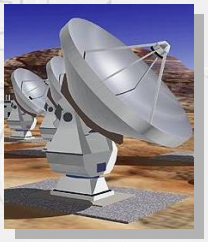




ALMA Project

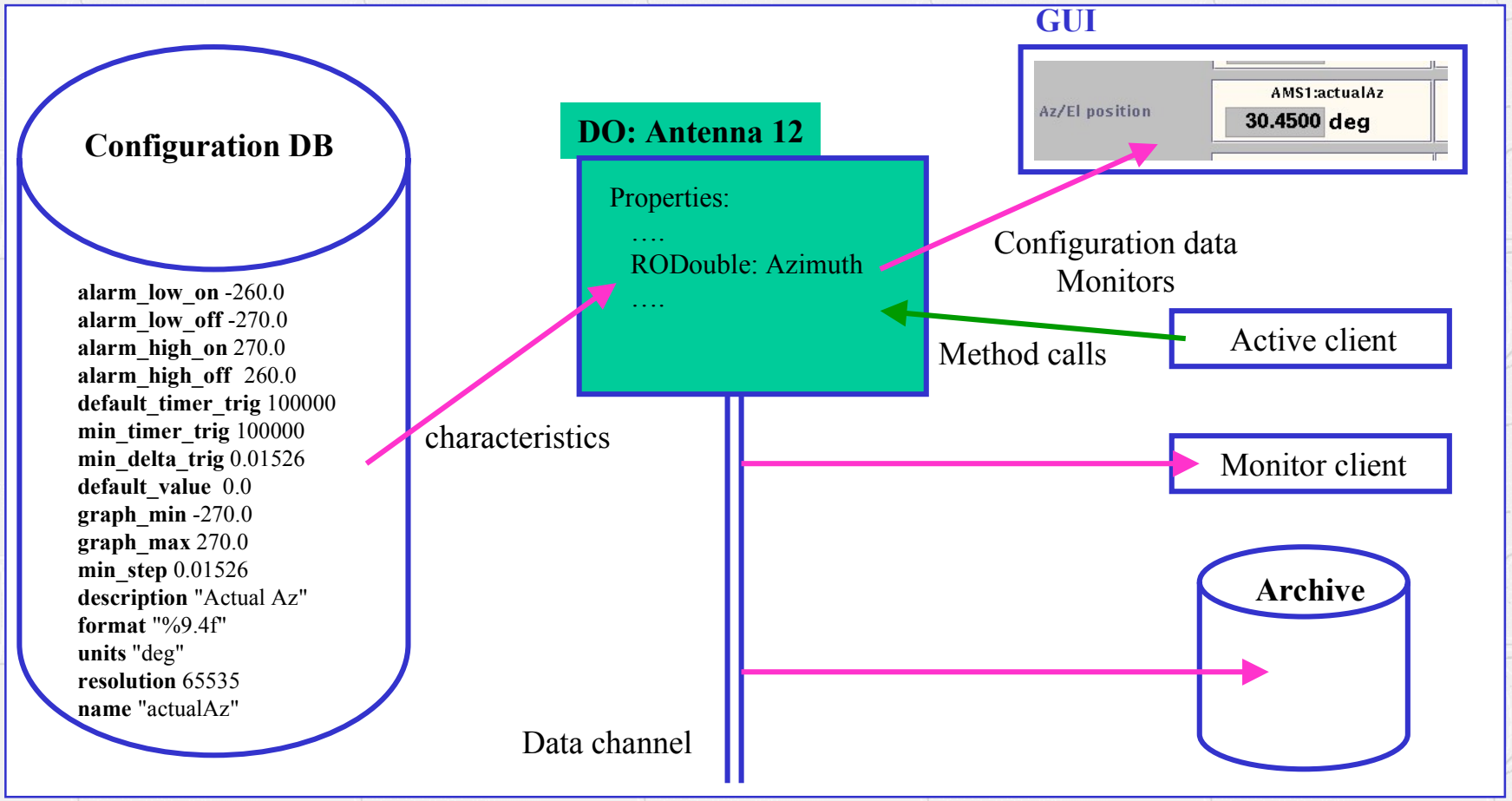
Properties

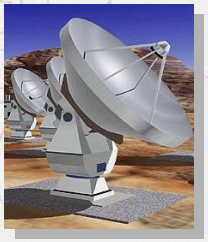




ALMA Project

example: Antenna Azimuth

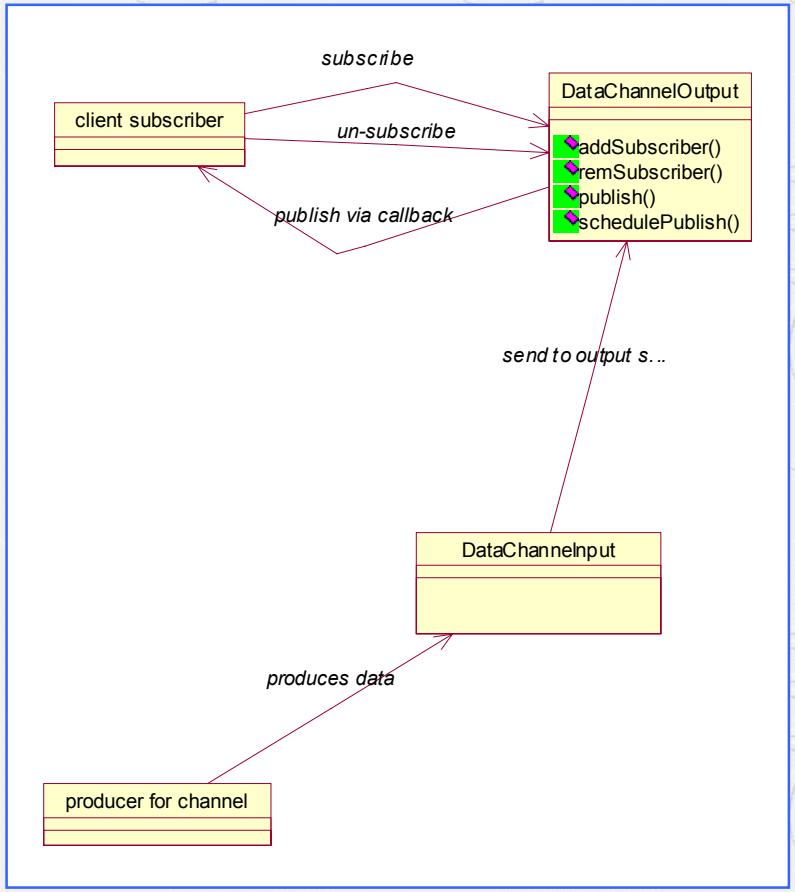


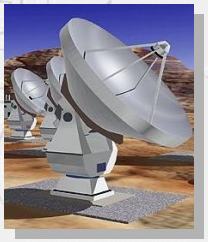


ALMA Project

Data channel

- Implements publisher/subscriber design pattern
- Optimizes transmission of data
- De-couples data producers and consumers
- Based on CORBA notification service





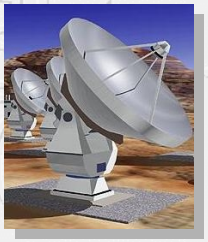
ALMA Project

New technologies: mature?

- Impressions after 1.5 years of ACS development:

  rather but not fully mature!

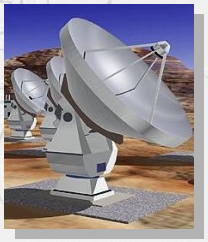
- Cannot be summarized in just a few slides (more on request in the open discussion)



ALMA Project

New technologies: CORBA

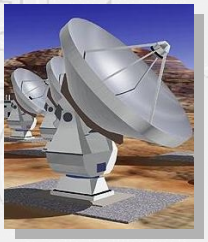
- Keeps its promises
- Very good interoperability between vendors
- Well documented
  (Advanced CORBA Programming bible)
- Big and complex
  (C++ code readability, debugging)
- Only major problem:
  memory allocation in C++



ALMA Project

New technologies: Java

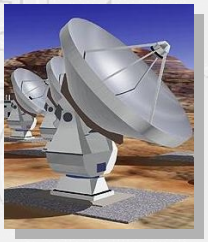
- Language and libraries very nice
- Visual Age introduces very nice visual programming concepts
- Clean CORBA code
- Many problems:
  - Java versions and tools support
  - Visual Age Linux implementation: slow, big and late
  - Memory footprint and memory leaks
  - …….



ALMA Project

New technologies: XML

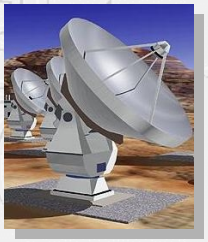
- Fully satisfied with the concept
- Perfect to replace small configuration files, result of queries and as a transport format
- Not good for large amounts of data
- Tools still limited, slow and poor



ALMA Project

New technologies: UML

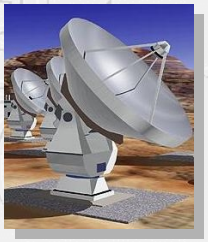
- Consistently used as modeling language. Everyone understands it now
- Tools (Rose) are always behind the definition of the standard and have limitations



ALMA Project

New technologies: Linux

- Works reasonably well for a single user
- PCs are cheap and easy to install
- Requires a lot of "hacker knowledge"
- The risks of "*personal* computer" are very high:
  - running after the "latest version"
  - not having two machines with the same configuration



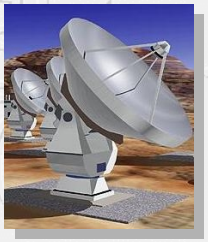
ALMA Project

Opportunities for other projects

MANY opportunities for reuse in other projects and for collaborations

Some examples:

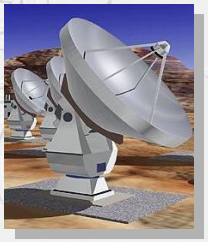
- The whole ACS or parts of it can be used in other projects
- Once DO/Property/Characteristic concept is adopted UIF components can be shared
- XML DTDs and tools can be shared
- Many packages, in particular in the high level and outside the control system, still have to be designed



ALMA Project

To know more

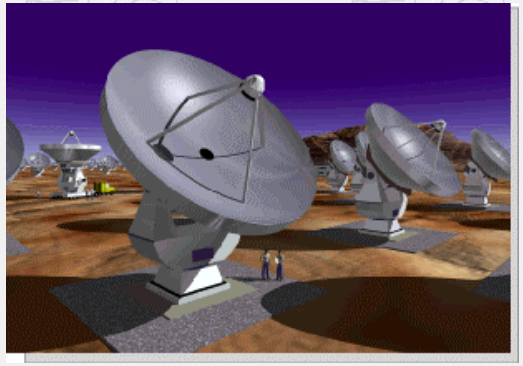
- The ALMA project: http://www.eso.org/projects/alma/
- SW Working group: http://www.mma.nrao.edu/development/computing/index.html
- Common SW (and ACS Architecture draft document): .../workinggroups/common_software/docs/index.html
- Java, CORBA and other technology links: http://www.eso.org/projects/vlt/sw-dev/oowg-forum/readings.html



ALMA Project

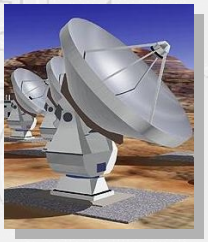
Acknowledgments

- ESO ACS team:
  - G.Chiozzi, B.Gustafsson, B.Jeram, P.Sivera + R.Karban
- ALMA ACS collaborators
  - R.Heald, R.Lemke, A.Perrigouard…. many others
- KGB, Ljubljiana team
  - http://kgb.ijs.si/KGB/



Atacama
Large
Millimeter
Array

Part 2
Kitt Peak 2000 test

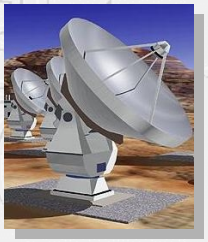


ALMA Project

The KP 2000 Test

Demonstrate ACS concepts by developing and Antenna Mount Software and testing basic functionality:

- Repeat (partially) 1999 test with ESO CCS/TCS
- Test antenna motion SW
- Test new tracking SW based on ACS
- Create Pointing Model (interoperability ACS/CCSLite)



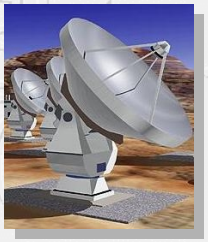
ALMA Project

The KP 2000 test (cont.)

- Where:

  Kitt Peak 12m Telescope
- When:

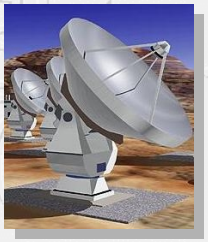
  01/08 December 2000
- Who:
  - R.Heald - NRAO
  - G.Chiozzi, B.Gustafsson, B.Jeram, R.Karban - ESO
  - A.Perrigouard - IRAM
  - with the collaboration of R.Lemke (MP) and M.Plesko team



ALMA Project

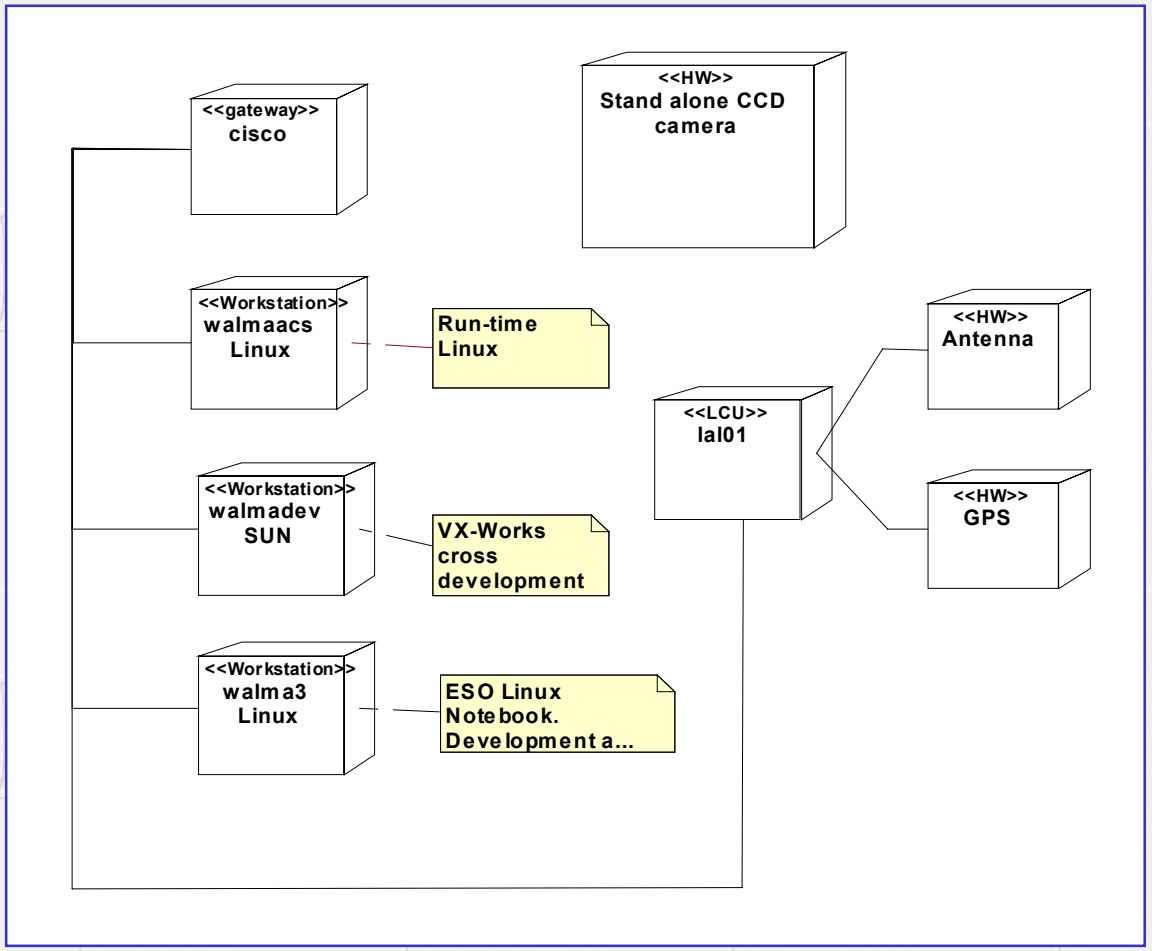
The Sites involved

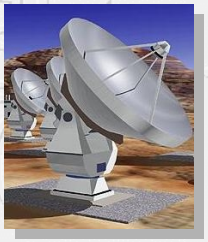
- Kitt Peak Observatory
  - Complete system with HW and SW
- NRAO Socorro
  - ACS on Linux and VxWorks + HW&SW simulator
- ESO Garching
  - ACS on Linux and VxWorks + SW simulator
- IRAM
- MP Bonn
- IJS Ljubljiana
  - ACS on Linux



ALMA Project

KP Computer configuration

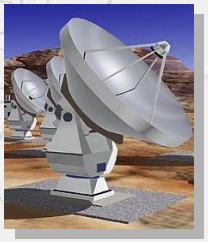




ALMA Project

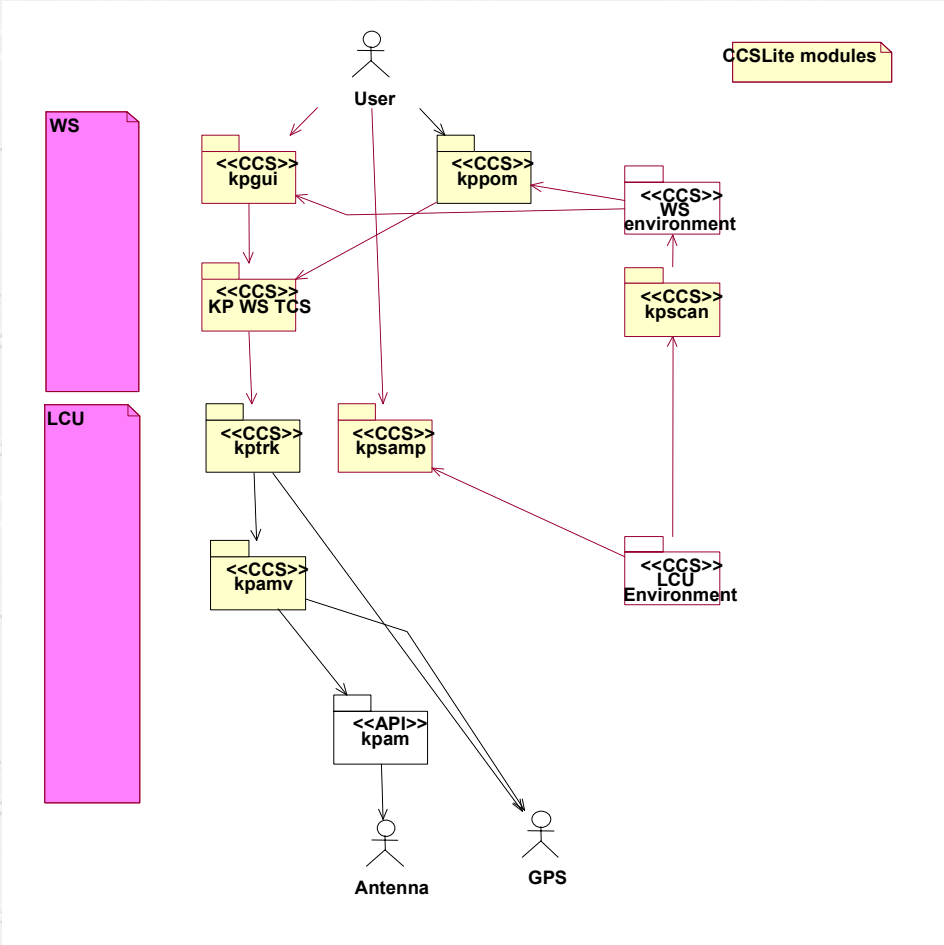
Demo Configuration

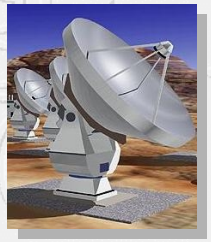
- Laptop with Win NT, 650MHz, 256Mb ram
- VmWare Linux Virtual Machine (CPU equivalent to 400MHz?, 96Mb ram)
- WS and LCU code running under Linux
- Xserver running under NT



ALMA Project

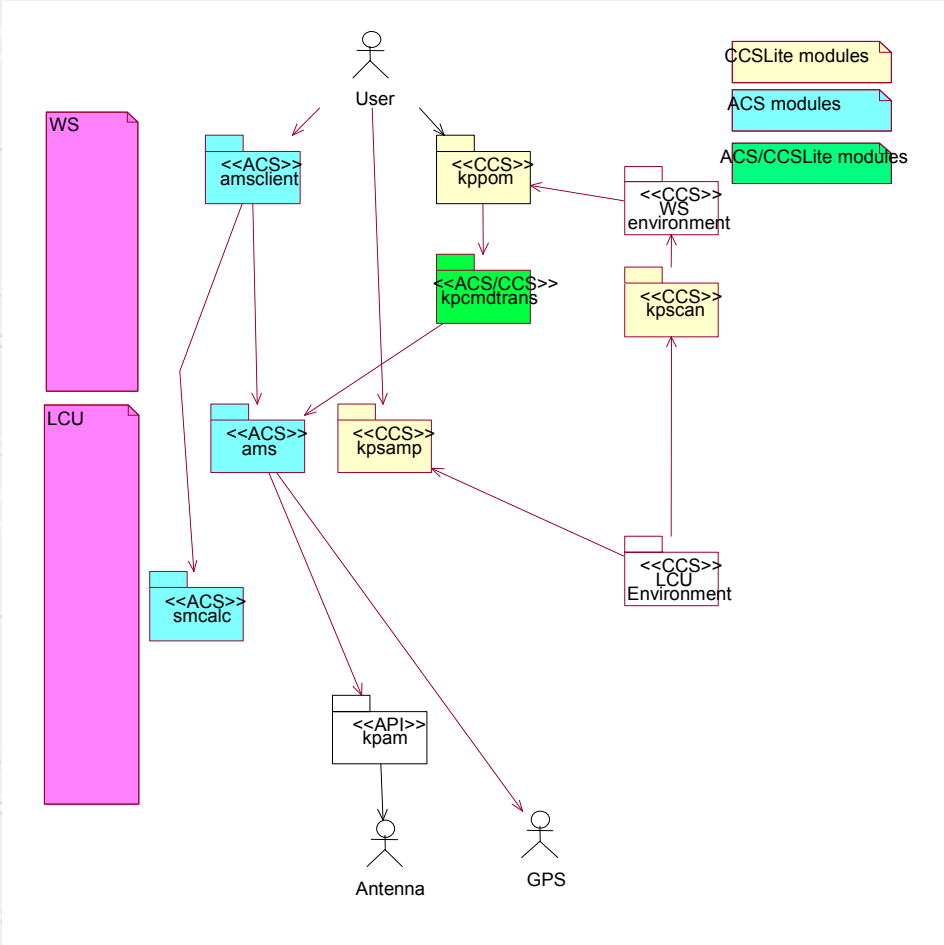
SW Architecture (VLT)

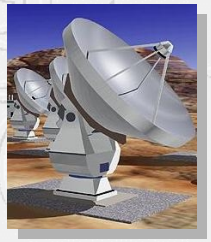




ALMA Project

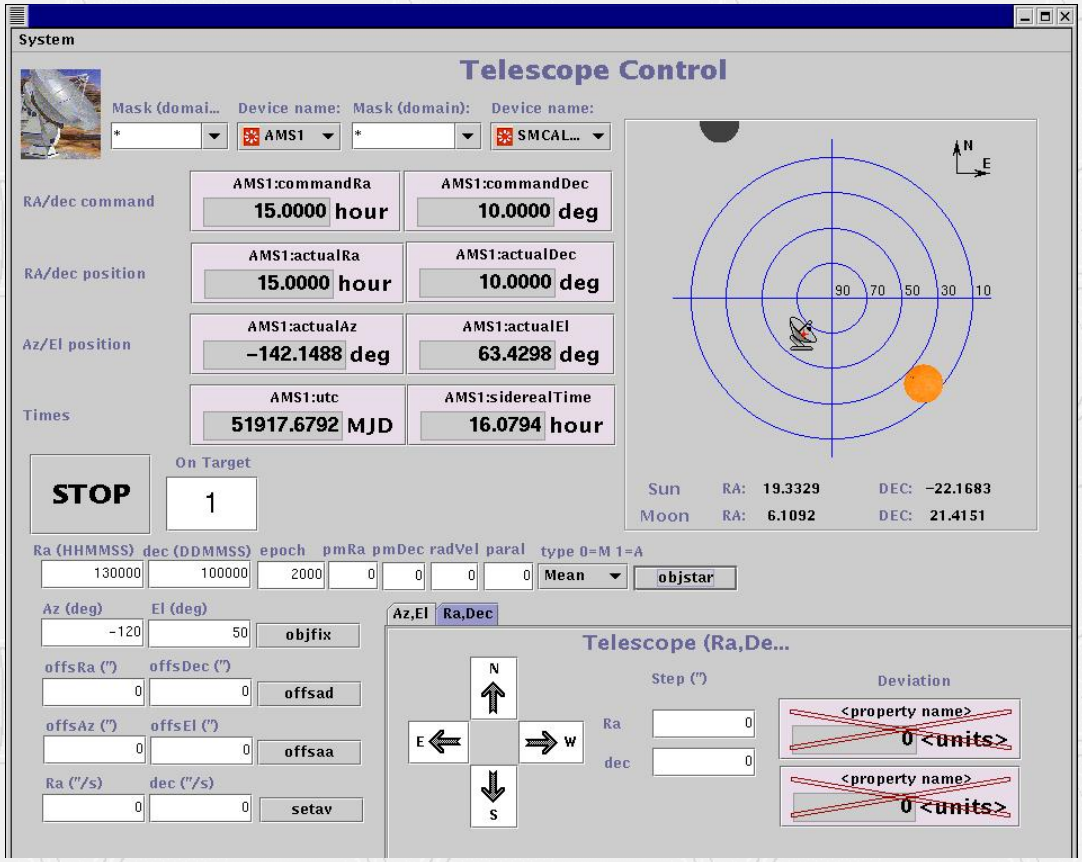
SW Architecture (ACS)

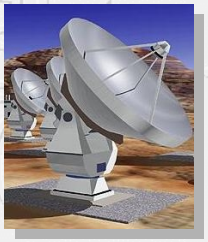




ALMA Project

TCS Control GUI





ALMA Project

Conclusions

- ACS concepts proofed to be working very well
- Weaknesses in the prototype implementation only
- Easy integration with CCSLite applications
- Documentation, tutorials and powerful framework for building applications are needed
- Selected new technologies are very promising, but not fully mature
- There is a lot of design and implementation work to be done
- There are plenty of opportunities for collaboration and reuse