Remote Visualization System

Anil Chandra

Malte Marquarding

ATNF/CSIRO

Overview

- Requirements
- Architecture
- Components
- Features
- Possible extensions
- Demo

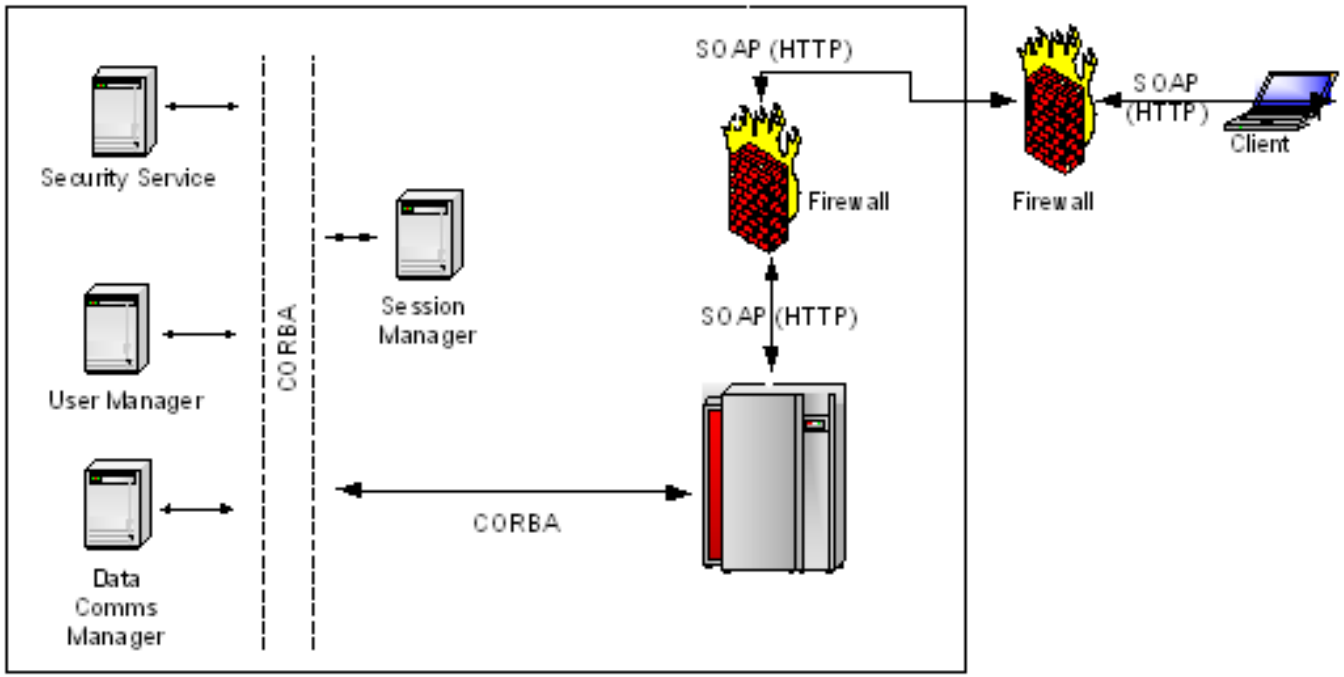
RVS - Outline

- Project started as a thesis project to explore technologies
- 2d visualization of and interaction with (large) n-dim astronomical images
  - Reuse of prexisting software
- scalable, maintainable
- image access without the need of high bandwith
  - Server-side rendering
  - co-location with data storage servers
- Customizable clients
  - webservice

RVS - framework

- RVS is predominantly a framework
  - designed because grid services were immature at the time of conception
- It provides
  - Inter-language communication (CORBA)
  - Distributed computing (CORBA)
  - Webservice definition (WSDL)

RVS - Architecture



RVS – Architecture (cont'd)

- Data Comms Manager [java]
  - External Data comms (HTTP,FTP, etc)
- User Manager [java]
  - Management of user specific data
- Security Manager [java]
  - User/Session key management
- Session Manager [c++]
  - manages the session specific data
  - holds the display library canvas
- Webserver [java]
  - SOAP interface, user access point

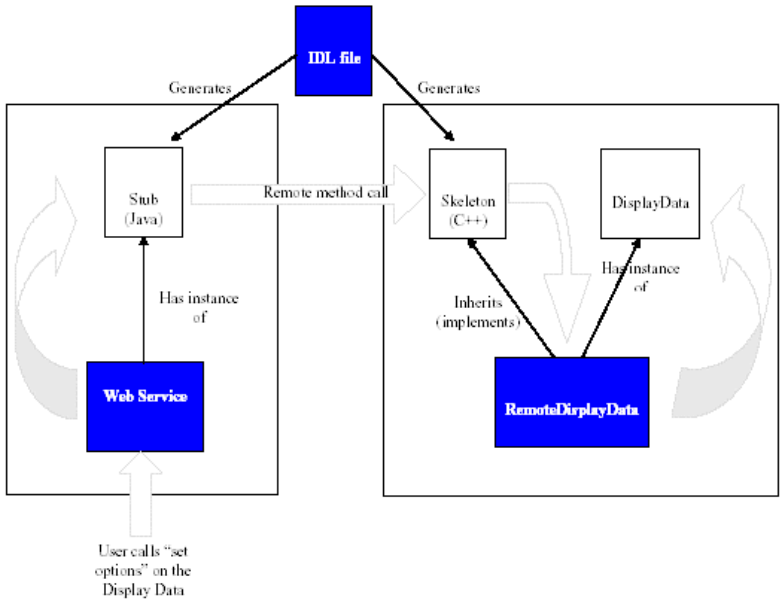
CSIRO

RVS – Software

- Software Components
  - C++
    - aips++ display library (rendering backend)
    - TAO
  - Java
    - jacorb
    - Tomcat, axis (soap)
    - Webstart
  - XML
    - xerces

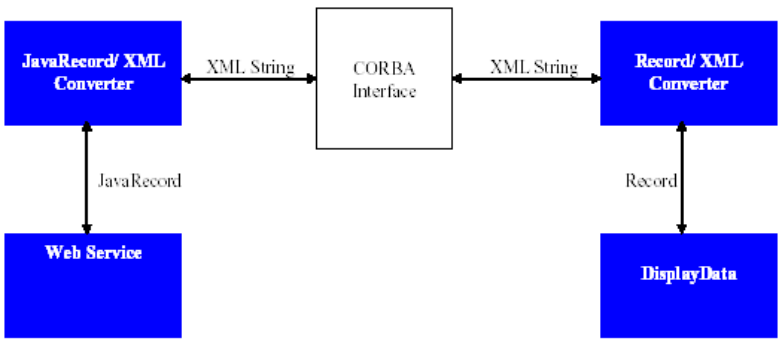
RVS - CORBA

- CORBA – the glue
  - CORBA is the acronym for Common Object Request Broker Architecture, a vendor independent architecture and infrastructure that computer applications use to work together over networks.



RVS - Communication

- Communication
  - Display Library settings as records which is easily converted to xml
  - Simple interface setOption(Record&)
  - Lean interface
  - Well suited to CORBA/SOAP communication



RVS – Display Library

- Abstract drawing classes
  - Implementions in Java, X11, PS
  - PixelCanvas – ProxyCanvas(Corba) – JavaPixelCanvas
- Rendering
  - Images (FITS, miriad, aips++)
  - Position tables (non-gridded)
  - n-dimensional
  - WCS
  - raster, contour, marker, vector drawing
  - on-the fly re-gridding
  - zooming

RVS – Clients

- postage stamping [java/html]
- A “traditional” viewer – rvsviewer [java]
- session sharing/conferencing [java]
- Mpeg generation (concept) [python]
- multi-axis view (prototype) [java]

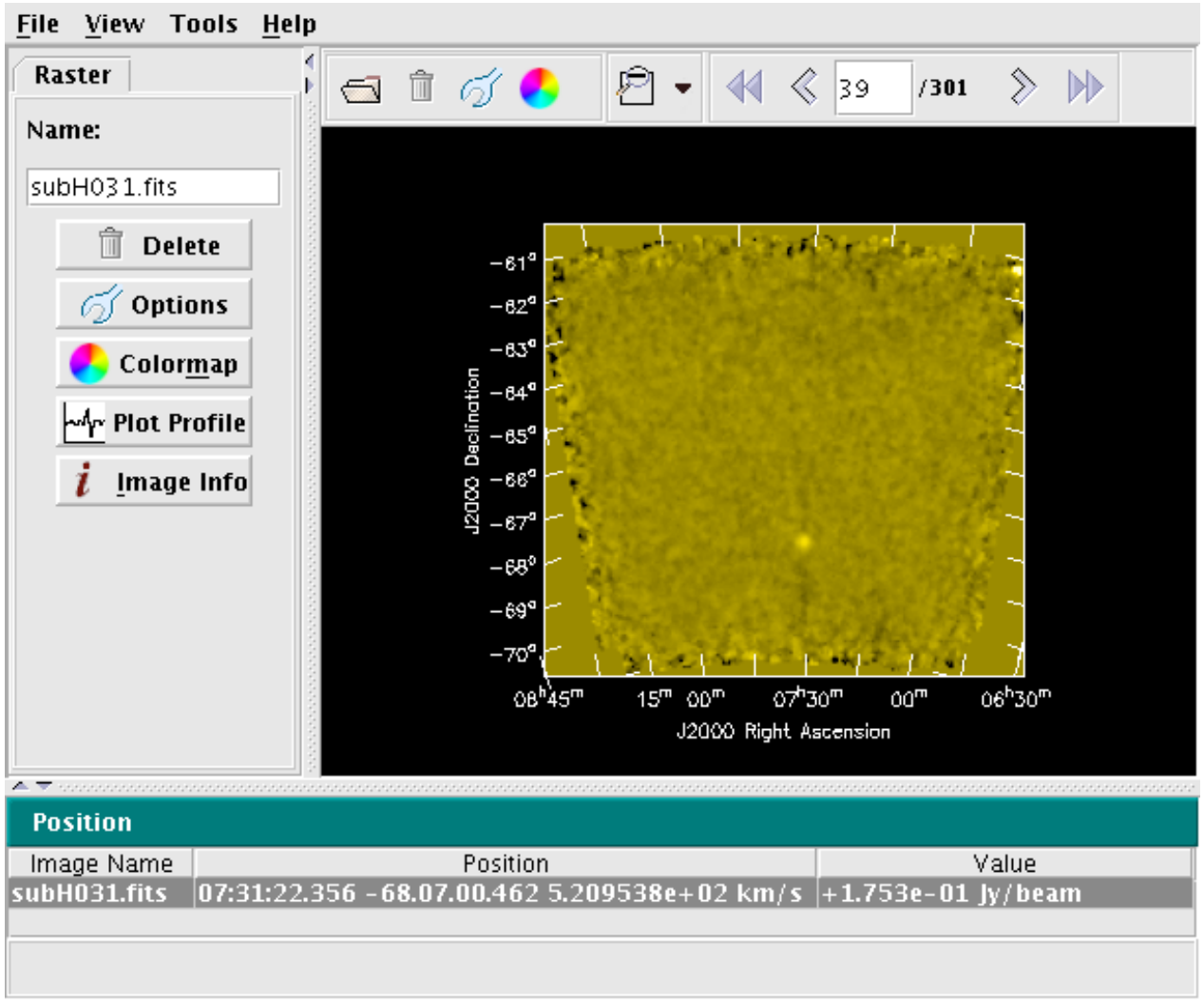
RVS – possible future additions

- Volume rendering
- Outreach client
- Reuse of architecture to bind other “legacy” systems or non-traditional visualization

RVS - Summary

- server-side visualization system
- Distributed architecture
- component based
- Reuse of existing software to do specialized application
- Multi-dimensional image display, profile plotting
- Custom clients via WSDL
- see http://www.atnf.csiro.au/vo/

RVS - Client



http://www.atnf.csiro.au/people/Malte.Marquarding/rvs