PPARC

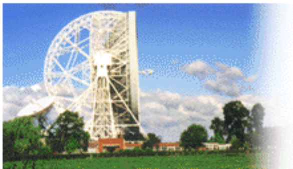
A PPARC funded project

AstroGrid

Astro Runtime as an Application Interface

Noel Winstanley
nw@jb.man.ac.uk
AstroGrid, Jodrell Bank, UK

The University of Edinburgh
University of Leicester
University of Bristol
University of Cambridge
University of Leeds
University of Exeter
CCLRC Rutherford Appleton Laboratory



Jodrell Bank Observatory
The University of Manchester
University of Portsmouth
Queen's University Belfast
UCL MSSL
AstroGrid
The University of Manchester

AstroGrid

- A nearly complete Virtual Observatory (VO) system
- http://software.astrogrid.org
- Built upon Web Services – callable via SOAP
- Conforms to international (IVOA) standards where defined, elsewhere proposes new standards
- Useful for real research
- Security -final part that needs to be done.
  - Authentication & Authorisation
- The presentation shows how AstroGrid provides access to remote applications and data collections – the underpinnings of workflow.

Registry

- A hierarchical database (XML).
- Contains records that describe
  - data collections (coverage, catalogue structure, access methods)
  - remote applications (purpose, parameters, invocation methods)
  - supporting web services – storage, security, etc
  - other useful resources – e.g. client-side applications, organizations.
- Accessed by Web Service
- Query using XQuery, Keywords, or ADQL
- Used to locate (resolve) all other VO Web Services
- Exchanges records with registries in other VO projects (harvesting)

Myspace

- Distributed, location transparent, file storage
  - Each user has a single folder hierarchy – maintained by a *filemanager* service
  - Files in the hierarchy may be stored at different locations – each location is a *filestore*
  - clients typically interact only with the filemanager.
- AstroGrid services can read / write to Myspace
  - Place to stage results of long-running queries & computations
  - Used as a buffer for intermediate products of workflows
- Enables data to be kept near processing tools
- Being standardized in IVOA as VOSpace / VOStore

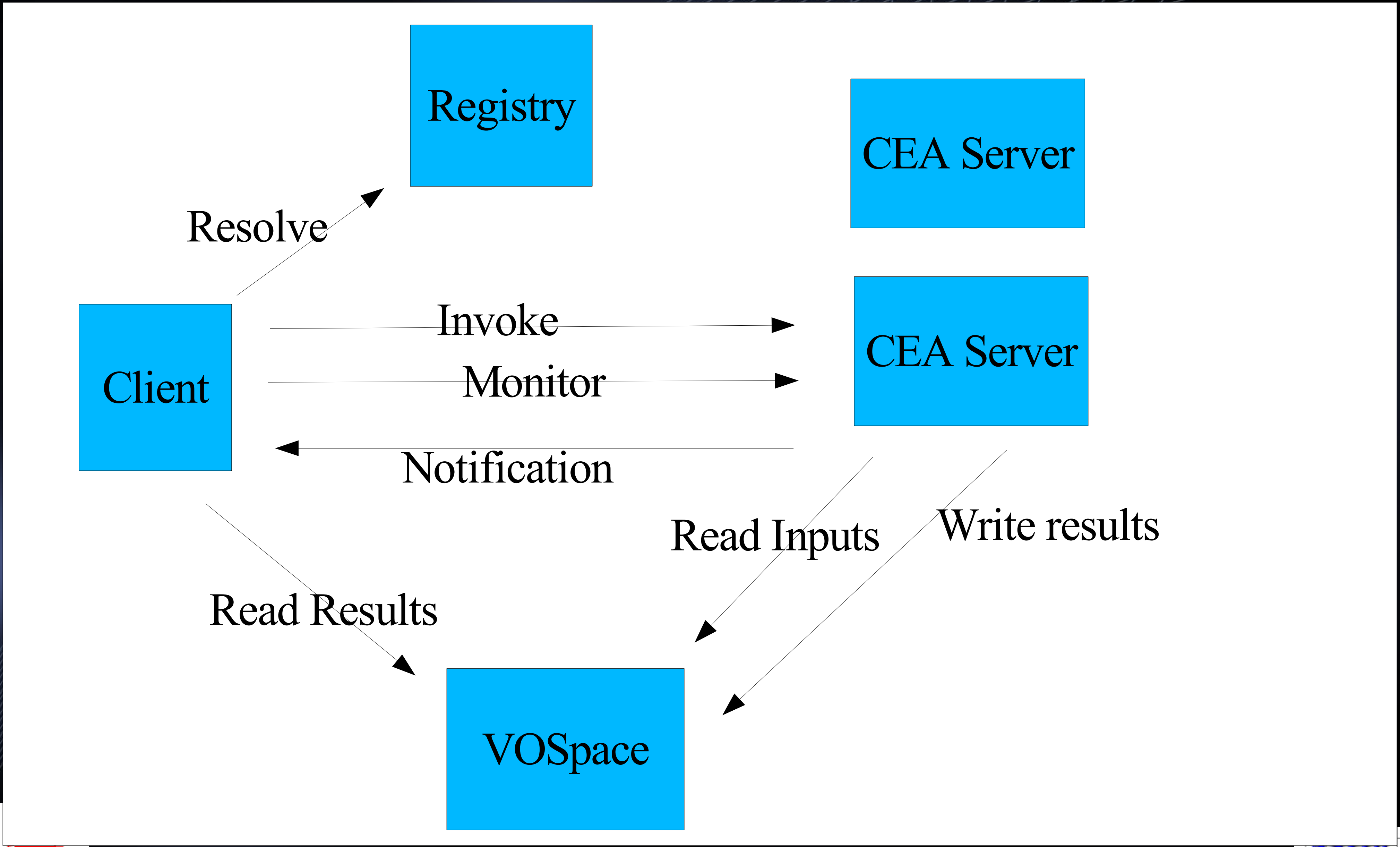
Remote Applications

- Common Execution Architecture - CEA
- Uniform method of describing and providing access to remote applications. Encompasses:
  - dataset access (e.g. Querying a catalogue database or image collection)
  - data processing – by VO-enabling popular legacy codes (e.g. X-matching, source extraction, simulations)
- CEA applications can be invoked from client scripts, UI, and server-side workflows
- Working System, with installed base.
- Described in IVOA Note
- Next generation IVOA proposal 'Universal Worker Service'

CEA Servers

- A CEA Server may provide one or more applications
- Asynchronous invocation
  - CEA server provides progress monitoring, notifications, and control
  - results can be retrieved from the service, or staged to myspace, ftp server, etc.
  - All interfaces are standard web services.
- Servers, and the applications they provide, are described in the registry.
- Clients can query registry to find applications, and servers that provide them.
- Applications can be replicated across multiple servers, for fault-tolerance.

Execution schematic



Installed Base - 2006-05-10

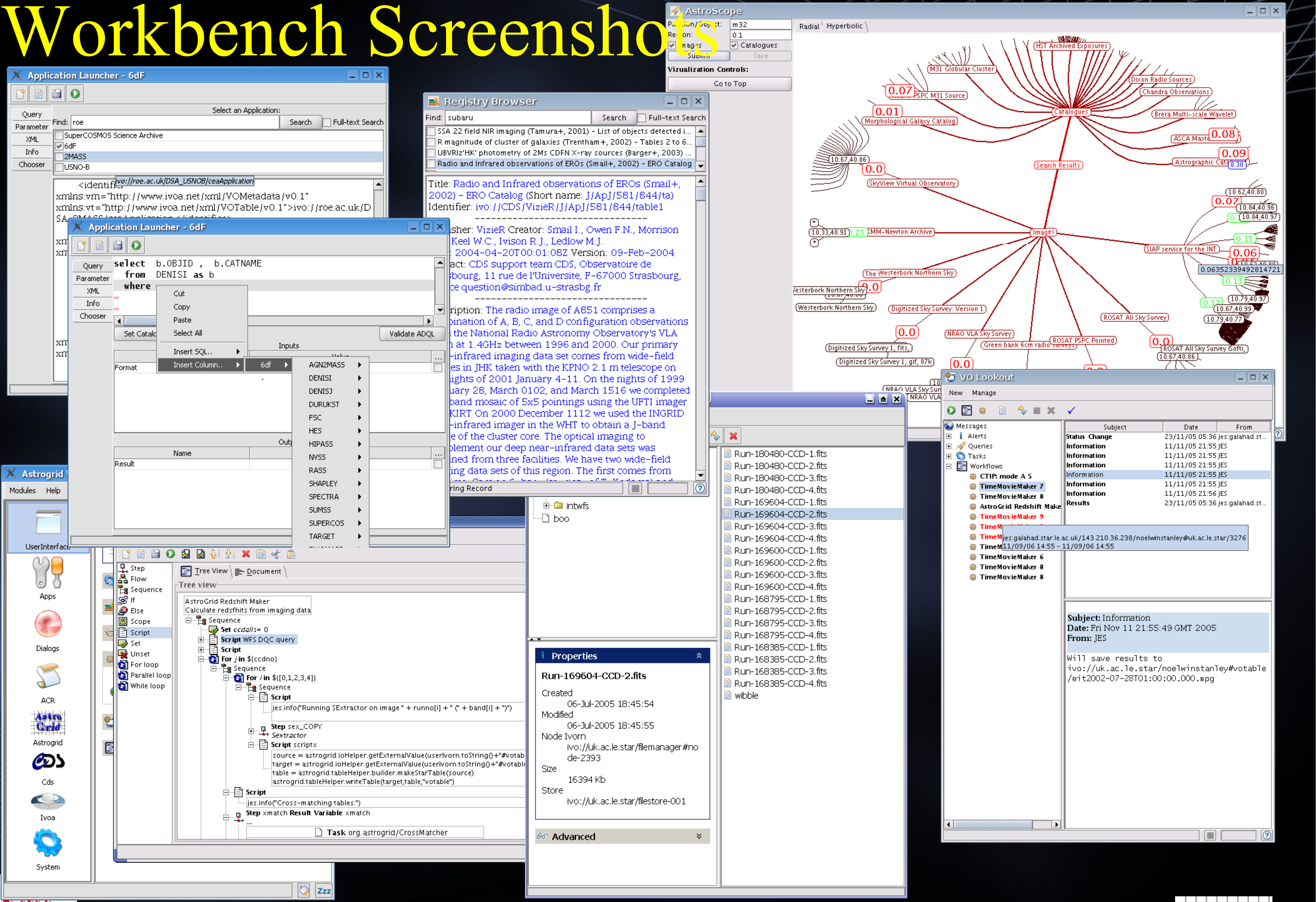
- 50 CEA servers,
  - UK – portsmouth, ral, mssl, edinburgh, cambridge, leiecester, jodrell
  - France, Russia,
- 70 applications available
  - Datasets – FIRST, INT-WFS, Ledas, SDSS,
  - Images – GOODS, HST-UDF, SDSS, XMM, Merlin
  - Existing Astro Apps – Sextractor, HyperZ, Galaxev, Pegase, BPZ, ACE, SWarp
  - New Apps – solar movie maker
  - General purpose – GNUPlot, R,

Accessing AstroGrid Services

User Perspecticve

- Two alternative user interfaces
- Portal – a web interface, accessed through browser
  - handy for occasional use
  - technical limitations of the web (pre-AJAX) make it awkward for advanced tasks
- Workbench – GUI client
  - Java WebStart Application
  - rich user applications
    - data discovery – astroscope & helioscope
    - application launcher – run a single CEA application
    - workflow builder – combine CEA applications
    - myspace browser
  - scripting access to VO
  - PLASTIC Hub – data exchange with Topcat, Aladin, Vospec, Visivo

Workbench Screenshots



```
select  b.OBJID ,  b.CATNAME
  from  DENISI as b
 where
```

<identifi ivo://roe.ac.uk/DSA_USNOB/ceaApplication
xmlns:vm="http://www.ivoa.net/xml/VOMetadata/v0.1"
xmlns:vt="http://www.ivoa.net/xml/VOTable/v0.1">ivo://roe.ac.uk/D

- SSA 22 field NIR imaging (Tamura+, 2001) - List of objects detected i...
- R magnitude of clusters of galaxies (Trentham+, 2002) - Tables 2 to 6...
- UBVRIz'HK' photometry of 2Ms CDFN X-ray sources (Barger+, 2003) ...
- Radio and Infrared observations of EROs (Smail+, 2002) - ERO Catalog

Title: Radio and Infrared observations of EROs (Smail+, 2002) - ERO Catalog (Short name: J/ApJ/581/844/ta)
Identifier: ivo://CDS/VizieR/J/ApJ/581/844/table1

Publisher: VizieR Creator: Smail I., Owen F.N., Morrison ... Keel W.C., Ivison R.J., Ledlow M.J. ... 2004-04-20T00:01:08Z Version: 09-Feb-2004 ... Contact: CDS support team CDS, Observatoire de Strasbourg, 11 rue de l'Universite, F-67000 Strasbourg, ... question@simbad.u-strasbg.fr

Description: The radio image of A851 comprises a combination of A, B, C, and D configuration observations ... the National Radio Astronomy Observatory's VLA ... at 1.4GHz between 1996 and 2000. Our primary ... near-infrared imaging data set comes from wide-field ... images in JHK taken with the KPNO 2.1 m telescope on ... nights of 2001 January 4-11. On the nights of 1999 ... January 28, March 0102, and March 1516 we completed ... a J-band mosaic of 5x5 pointings using the UFTI imager ... UKIRT On 2000 December 1112 we used the INGRID ... near-infrared imager in the WHT to obtain a J-band ... image of the cluster core. The optical imaging to ... complement our deep near-infrared data sets was ... obtained from three facilities. We have two wide-field ... imaging data sets of this region. The first comes from

```
AstroGrid Redshift Maker
Calculate redsfhits from imaging data
Sequence
  Set ccdall := 0
  Script WFS DQC query
  Script
  For j in ${ccdno}
    Sequence
      For i in ${[0,1,2,3,4]}
        Sequence
          Script
            jes.info("Running SExtractor on image" + runno[i] + " (" + band[i] + ")")
          Step sex_COPY
            Sextractor
          Script scriptx
            source = astrogrid.ioHelper.getExternalValue(userIvorn.toString()+"#votab
            target = astrogrid.ioHelper.getExternalValue(userIvorn.toString()+"#votabl
            table = astrogrid.tableHelper.builder.makeStarTable(source)
            astrogrid.tableHelper.writeTable(target,table,"votable")
    Script
      jes.info("Cross-matching tables.")
    Step xmatch Result Variable xmatch
      ...
      Task org.astrogrid/CrossMatcher
```

- Run-180480-CCD-1.fits
- Run-180480-CCD-2.fits
- Run-180480-CCD-3.fits
- Run-180480-CCD-4.fits
- Run-169604-CCD-1.fits
- Run-169604-CCD-2.fits
- Run-169604-CCD-3.fits
- Run-169604-CCD-4.fits
- Run-169600-CCD-1.fits
- Run-169600-CCD-2.fits
- Run-169600-CCD-3.fits
- Run-169600-CCD-4.fits
- Run-168795-CCD-1.fits
- Run-168795-CCD-2.fits
- Run-168795-CCD-3.fits
- Run-168795-CCD-4.fits
- Run-168385-CCD-1.fits
- Run-168385-CCD-2.fits
- Run-168385-CCD-3.fits
- Run-168385-CCD-4.fits
- wibble

Run-169604-CCD-2.fits
Created
06-Jul-2005 18:45:54
Modified
06-Jul-2005 18:45:55
Node Ivorn
ivo://uk.ac.le.star/filemanager#node-2393
Size
16394 Kb
Store
ivo://uk.ac.le.star/filestore-001

Subject	Date	From
Status Change	23/11/05 05:36	jes:galahad.st...
Information	11/11/05 21:55	JES
Information	11/11/05 21:55	JES
Information	11/11/05 21:55	JES
Information	11/11/05 21:55	JES
Information	11/11/05 21:55	JES
Information	11/11/05 21:56	JES
Results	23/11/05 05:36	jes:galahad.st...

jes:galahad.star.le.ac.uk/143.210.36.238/noelwinstanley@uk.ac.le.star/3276
11/09/06 14:55 - 11/09/06 14:55

Subject: Information
Date: Fri Nov 11 21:55:49 GMT 2005
From: JES

```
Will save results to
ivo://uk.ac.le.star/noelwinstanley#votable
/eit2002-07-28T01:00:00.000.mpg
```

THE UNIVERSITY of MANCHESTER

Noel Winstanley - nw@jb.man.ac.uk

Developer / Expert Perspective

- Three Alternatives
  - Call SOAP services directly, using WSDL
    - most basic – exposed to most complexity, necessary to understand services interact (e.g. resolution)
    - security – can require advanced SOAP handling.
  - Call AstroGrid delegate libraries
    - hides some complexity,
    - maybe not the cleanest or most reusable interface – developed for internal use.
    - Java-only – requires whole AstroGrid library stack
  - Use the Astro Runtime (AR)
    - Uniform facade interface to the VO.
    - Simpler to learn & provides extra functionality
    - Language Neutral

Astro Runtime (AR)

- A common facade library for virtual observatory services
  - aim to integrate all IVOA standards, popular ad-hoc services, and suitable helper functions.
- uniform abstraction level and types
  - cleaner API, less special cases, lower learning curve
- Shared component – single signon, configuration, cached registry entries, file trees
- Deployment alternatives
  - Webstartable desktop service / embeddable library / stand-along application
  - Workbench is implemented upon AR – an AR instance is available to other clients when Workbench is running.

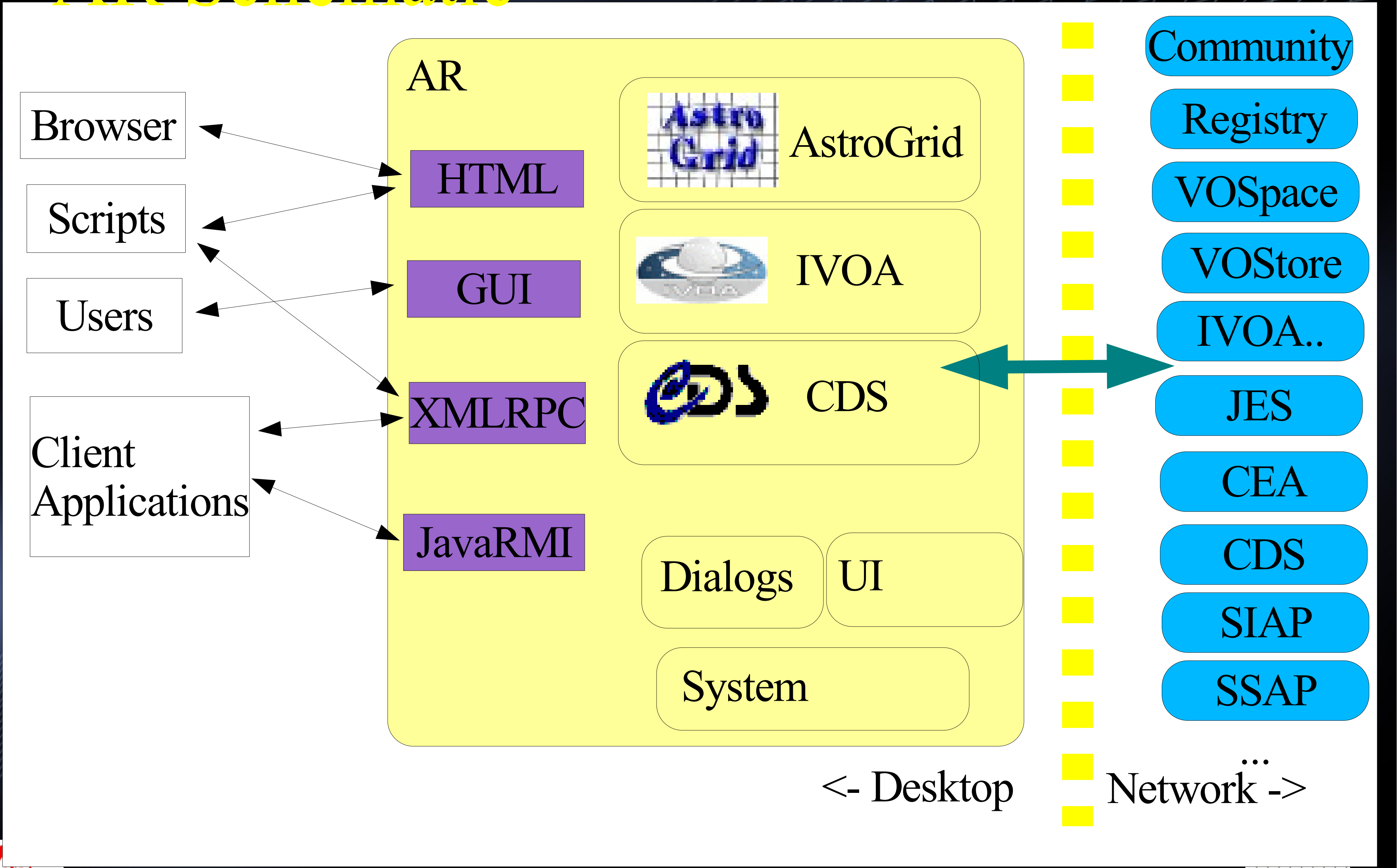
Design

- AR designed to be accessible from all programming languages
- Procedural design, rather than OO (astronomer friendly)
- Typical configuration - a service that runs on a user's desktop
    - accepts requests from other desktop applications
    - processes requests by calling webservices using the AstroGrid Java client libraries.
- Can also be embedded in a service implementation and called directly.

Access Methods

- At present, 3 ways to access AR functionality.
- JavaRMI (Java, Groovy, Jython)
  - JVM-only inter-process communication
  - strongly typed
  - requires a minimal set of libraries
  - allows remote event listeners to be registered
- XMLRPC (Python, Perl, C++, C# , Java)
  - Forerunner of SOAP: http://www.xmlrpc.com/
  - simpler types than SOAP
  - implementations for a wide range of languages
- HTTP-Get (Shell, R, IDL, Matlab)
  - rough-n-ready procedure call
  - fallback for other languages

AR Schematic



AR Abilities

- AstroGrid
  - CEA: query, build, execute, monitor
  - Workflow – submit, monitor.
  - MySpace: read, write, list, create, delete
  - Registry: query, xquery, resolve
- IVOA – SIAP, SSAP; SkyNode to follow.
- CDS – Simbad, Vizier, coordinates, UCD.
- NVO – cone; NED, other ad-hoc to follow
- PLASTIC – send data to other client-side applications
- UI - control workbench user interface, display dialogues to prompt for input

Connecting to AR - Code

Don't Panic!

Java: RMI

Import AR classes

Instantiate finder

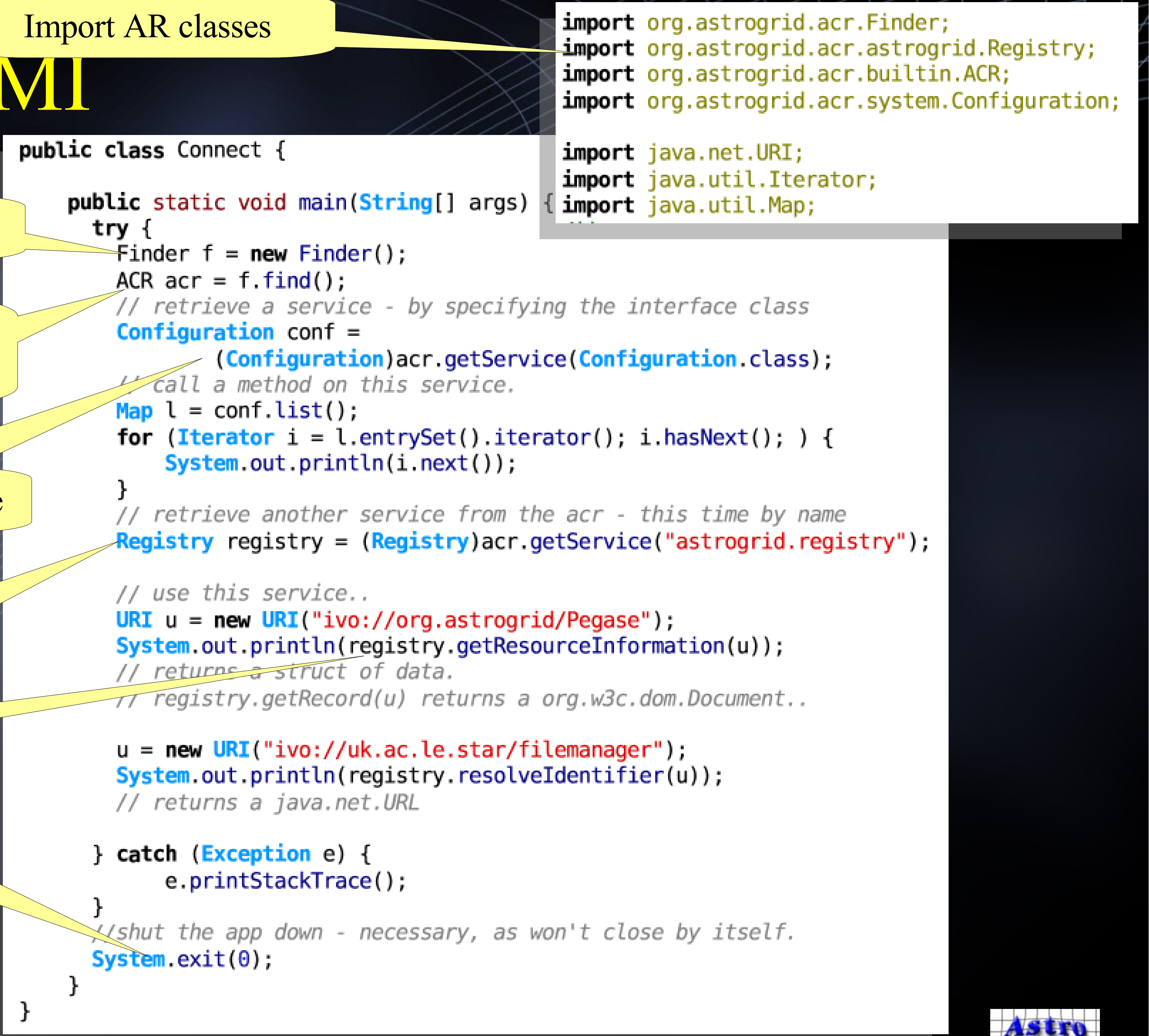
Find running AR, or execute new

Get reference to service

Alternative way to get service

Call service function

Tell program to exit



```
import org.astrogrid.acr.Finder;
import org.astrogrid.acr.astrogrid.Registry;
import org.astrogrid.acr.builtin.ACR;
import org.astrogrid.acr.system.Configuration;

import java.net.URI;
import java.util.Iterator;
import java.util.Map;
```

```
public class Connect {

    public static void main(String[] args) {
      try {
        Finder f = new Finder();
        ACR acr = f.find();
        // retrieve a service - by specifying the interface class
        Configuration conf =
                (Configuration)acr.getService(Configuration.class);
        // call a method on this service.
        Map l = conf.list();
        for (Iterator i = l.entrySet().iterator(); i.hasNext(); ) {
            System.out.println(i.next());
        }
        // retrieve another service from the acr - this time by name
        Registry registry = (Registry)acr.getService("astrogrid.registry");

        // use this service..
        URI u = new URI("ivo://org.astrogrid/Pegase");
        System.out.println(registry.getResourceInformation(u));
        // returns a struct of data.
        // registry.getRecord(u) returns a org.w3c.dom.Document..

        u = new URI("ivo://uk.ac.le.star/filemanager");
        System.out.println(registry.resolveIdentifier(u));
        // returns a java.net.URL

      } catch (Exception e) {
            e.printStackTrace();
      }
      //shut the app down - necessary, as won't close by itself.
      System.exit(0);
    }
}
```

Python: XML-RPC

Import xmlrpc library

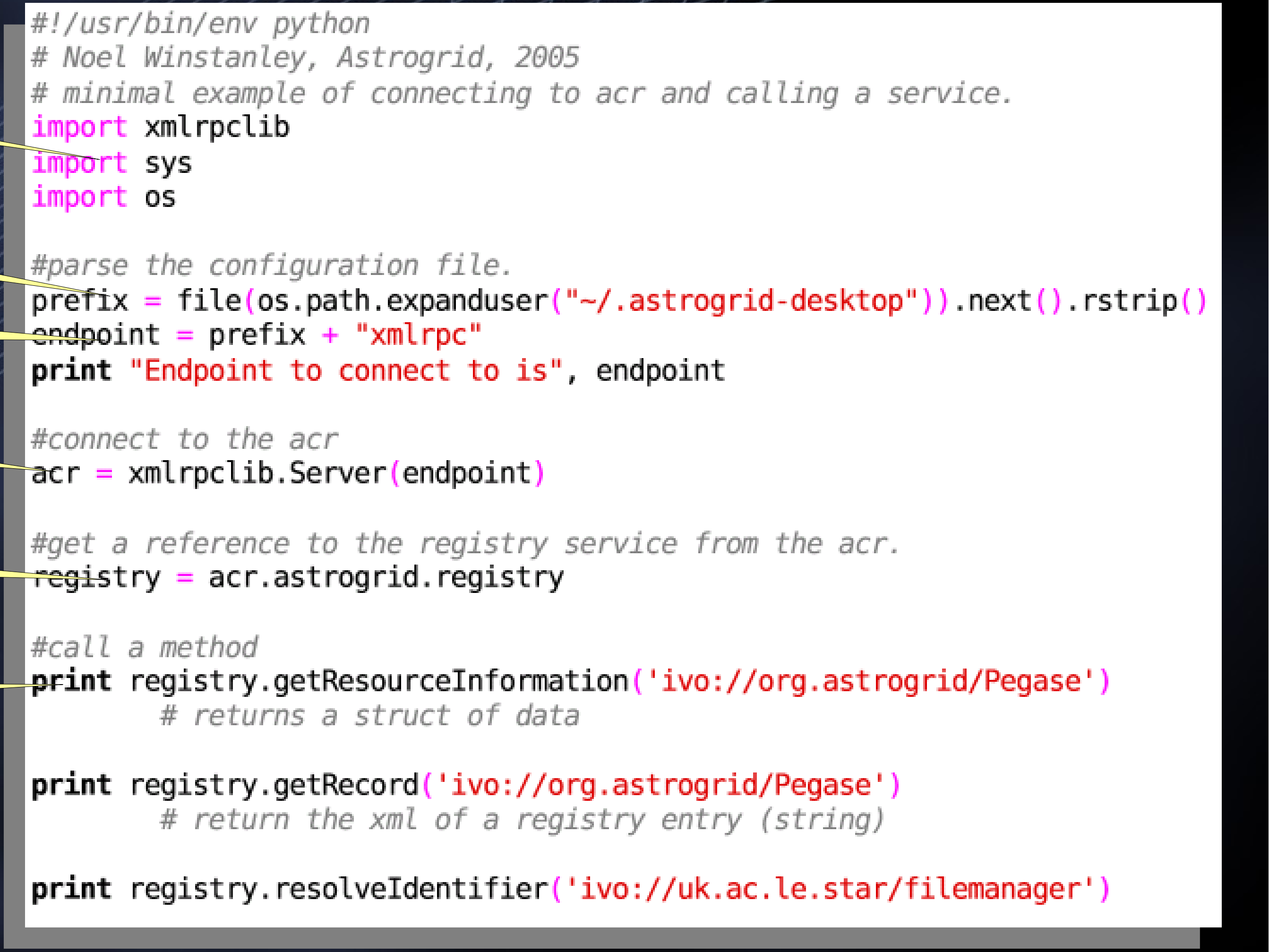
Read AR configuration file

Construct xmlrpc endpoint

Create client

Get reference to service

Call service function



```
#!/usr/bin/env python
# Noel Winstanley, Astrogrid, 2005
# minimal example of connecting to acr and calling a service.
import xmlrpclib
import sys
import os

#parse the configuration file.
prefix = file(os.path.expanduser("~/.astrogrid-desktop")).next().rstrip()
endpoint = prefix + "xmlrpc"
print "Endpoint to connect to is", endpoint

#connect to the acr
acr = xmlrpclib.Server(endpoint)

#get a reference to the registry service from the acr.
registry = acr.astrogrid.registry

#call a method
print registry.getResourceInformation('ivo://org.astrogrid/Pegase')
        # returns a struct of data

print registry.getRecord('ivo://org.astrogrid/Pegase')
        # return the xml of a registry entry (string)

print registry.resolveIdentifier('ivo://uk.ac.le.star/filemanager')
```

Uses for AR

- Environment for implementing VO Services
  - Workflow engines, VOEvent processors.
- VO-enable existing client-side applications
  - Topcat (.uk), Aladin (.fr), Visivo (.it), VoSpec (.es), ...
- New VO client-side Applications
  - AstroGrid Workbench, Sampo (.fi)
- Commandline utilities for VO
- Client-side Scripting (expert users) – (python, perl)
  - Script contains control flow
  - performs work by accessing CEA applications via AR
  - more interactive operation than batch Workflows
- Access VO in existing wrapper environments. - e.g.PyRaf, Parceltongue

Future Plans

- Will maintain backwards compatibility
- To add
  - missing service types - SkyNode
  - other ad-hoc astronomy webservices
  - VOTable, Xpath, XML helpers
  - make workbench UI components more controllable
- Track developing standards (VOSpace, SSO)
- Multi-session ACR for server-side deployment.
- Improve error reporting.
- Submit AR interfaces to IVOA in some way

References

- CEA
  - http://www.ivoa.net/Documents/latest/CEA.html
- Workbench Launch
  - http://software.astrogrid.org/userdocs/workbench.html
- AR Site – docs & download
  - http://software.astrogrid.org/beta/ar
- AR Wikipage
  - http://wiki.astrogrid.org/bin/view/Astrogrid/AstroClientRuntime
- Plastic
  - http://plastic.sourceforge.net