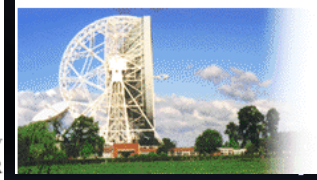
PPARC

A PPARC funded project

Astro Runtime
An API for the Virtual Observatory

Noel Winstanley - Jodrell Bank Observatory



Jodrell Bank Observatory

THE UNIVERSITY of MANCHESTER

THE UNIVERSITY of MANCHESTER

University of Portsmouth

Queen's University Belfast

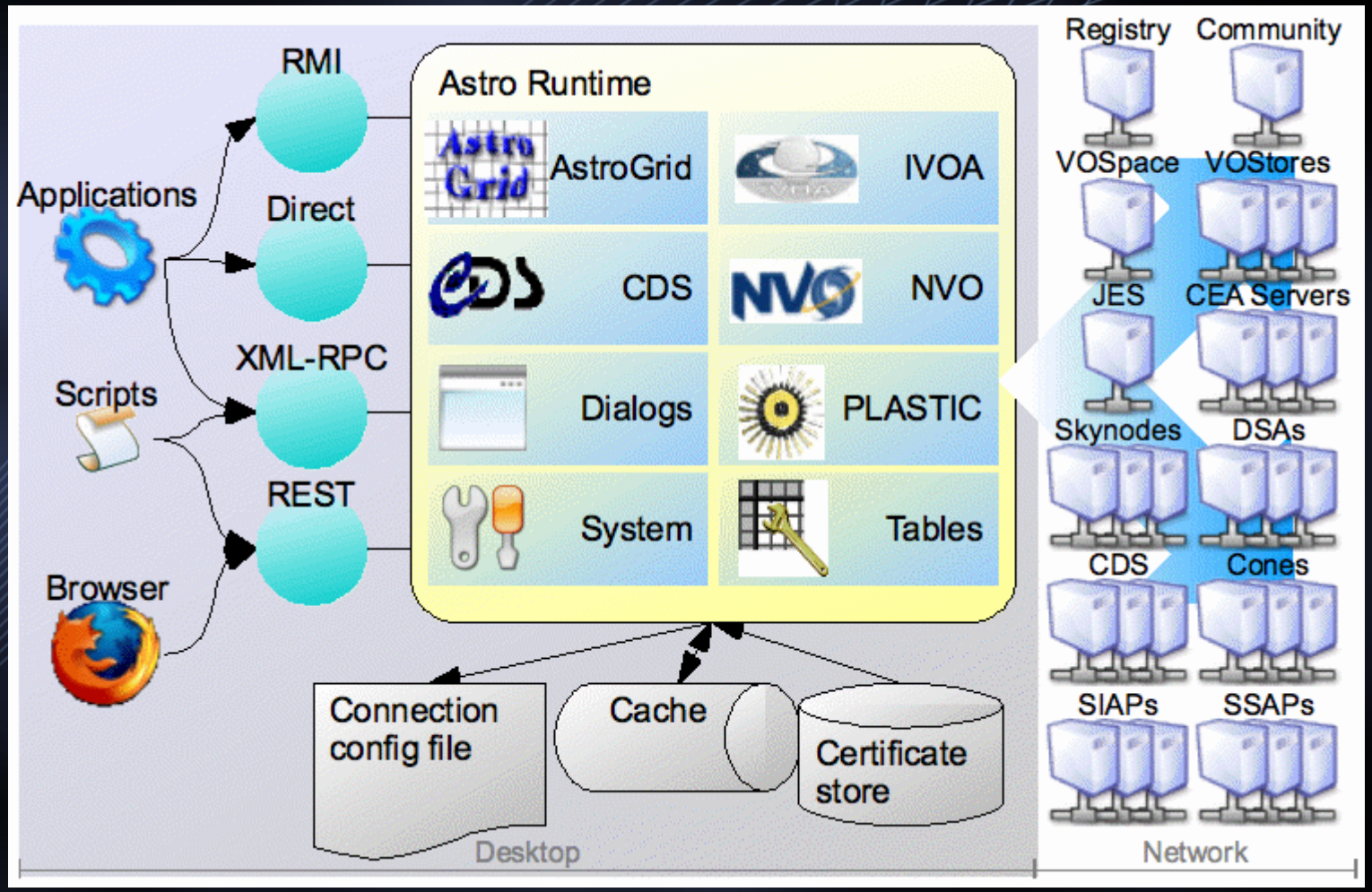
UCL MSSL

Astro Grid

Astro Runtime

- A library of virtual-observatory functions and clients.
  - integrates VO standards, popular ad-hoc services, useful helper functions.
- API design uses consistent abstractions and types
  - shallow learning curve, procedural design.
  - Insulates from the detail and change of VO specs.
- The API is exposed as a desktop service
  - accessible from almost all programming languages through either XMLRPC, RMI or REST.
  - trivial to install: installer, WebStart, or embedded.
  - minimal setup – no compilation, native libraries, linking
- Other benefits of a shared service
  - single signon, single configuration, single cache.

AR – Just the Middle Tier of the VO



AR simplifies access to..

- IVOA standard services
  - Registry: Retrieve records, XQuery, ADQL Query.
    - Full VOResource object model.
  - DAL: Cone, SIAP, SSAP. SIAP data model.
  - Single sign-on & user community
- CDS services – Simbad object resolver, Vizier, Coordinate conversion, UCD dictionary.
- PLASTIC – message exchange for client-side applications.
- Dialogues - display UI components to capture user input.
- Tables – Convert between VOTable, FITS, CSV, HTML, ...
- AstroGrid infrastructure
  - MySpace: list, read, write, create, delete.
  - CEA Tasks & ADQL Queries - build, execute, manage.

Why you might use AR...

- A quick way to VO-enable existing client-side applications. e.g.
  - Call DAL services from IDL
  - load and save into VoSpace from Emacs
- Infrastructure for new VO applications.
  - Concentrate on what's novel.
- Environment for implementing VO services
  - Workflow engines, Portals, VOEvent processors.
- Science scripting for the VO
- Access VO in existing wrapper environments.
  - PyRaf, ParselTongue

Applications using Astro Runtime

Application	Uses	Notes
Topcat	Myspace	Xmlrpc
Aladin	Myspace, CEA Tasks	Rmi
Visivo	CEA tasks to HPC	C++
VODA	Registry, DAL	
SED Builder	Myspace, Registry	Python Service
Vast	Registry, DAL, Myspace	
Workbench	Registry, DAL, Myspace, CEA, Dialogs	Direct
AstroWeka	Registry, DAL, Dialogs	

Scripting with Astro Runtime

- VO Commandline
  - unix-style small composable commandline programs.
  - Cone, siap, vols, voget, reg-query, ls-jobs ...
  - implemented as Python scripts calling the AR
  - Distributed as part of the development kit.
- Scripted Science Workflows
  - script contains control flow
  - performs work by querying DSA services, running CEA applications, calling DAL services.
    - Can use MySpace to stage data.
  - Can integrate existing non-VO programs
  - Possible in Python / Perl / IDL / Tcl / ..

Example: Calling SIAP services.

```
pos = ar.cds.sesame.resolve("crab")  # resolve object name

#build SIA query
s = 'ivo://adil.ncsa/targeted/SIA'
q = ar.ivoa.siap.constructQuery(s,pos['ra'],pos['dec'],1.0)

#execute query
resp = ar.ivoa.siap.execute(q)
print "Rows returned: ", len(resp)
print "Column Names: ", resp.keys()

# display the SIA response in a PLASTIC viewer
myId = ar.plastic.hub.registerNoCallBack("example script")
msg = 'ivo://votech.org/votable/loadFromURL'
ar.plastic.hub.requestAsynch(myId,msg,[q,q])

# save images (to myspace)
home = ar.astrogrid.myspace.getHome()
resultsDir = ar.astrogrid.myspace.createChildFolder(home,"results")
ar.ivoa.siap.saveDatasets(q,resultsDir)

# save an ascii version of the SIA response
ar.util.tables.convertFiles(q, "votable",resultsDir + "/response.txt","ascii")
```

We're doing the dull stuff so you don't have to..

- Hides a lot of nasty technology
- Masks differences between standard versions
- Masks varying implementations of the standards
- Masks interactions and dependencies between services.

- Makes writing code for the VO quick and easy
- Lets you concentrate on the interesting bits.

Status

- Maturing.
  - Nearing stable release.
  - Has been available for download for over a year
  - First presented at IVOA Interop, Kyoto, 2005
  - IVOA Note planned.
- Used in >10 applications (+ science scripting)
- Planned Improvements:
  - C client library binding (wraps XMLRPC interface)
  - Improve API – add more helper functions.
  - Track existing IVOA standards (Registry, DAL)
  - Support for new IVOA standards – e.g. TAP, VoSpace
  - Maintain backwards compatability.

AR is a publishing mechanism.

- Exposes standard Java libraries to a wider range of users.
- Some constraints on API design
  - Parameters need to be datastructures, not objects.
  - Interface <-> Implementation
- Currently published
  - Plastic – John Taylor et al. - plastic.sourceforge.net
  - STIL – Mark Taylor - www.star.bris.ac.uk/~mbt/stil/
- Upcoming
  - VOEvent – Alasdair Allan
  - VoSpace - t.b.d.
  - Your library here.

Try it out.

- Astro Runtime Site: http://www.astrogrid.org/desktop/astro-runtime
- Download – 2 things required
  - AR Development Kit - libs, documentation, examples.
  - A Runtime: Either Astro Runtime or Workbench
    - http://www.astrogrid.org/desktop
- Contact, Questions, Feedback :
  - Noel Winstanley - nw@jb.man.ac.uk