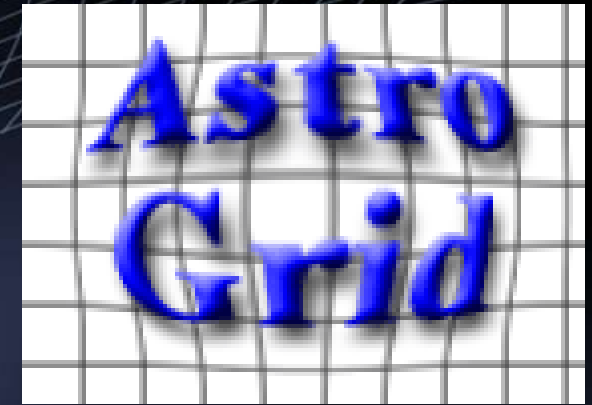


Workbench & ACR

- A VO Client Implementation



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



What is it?

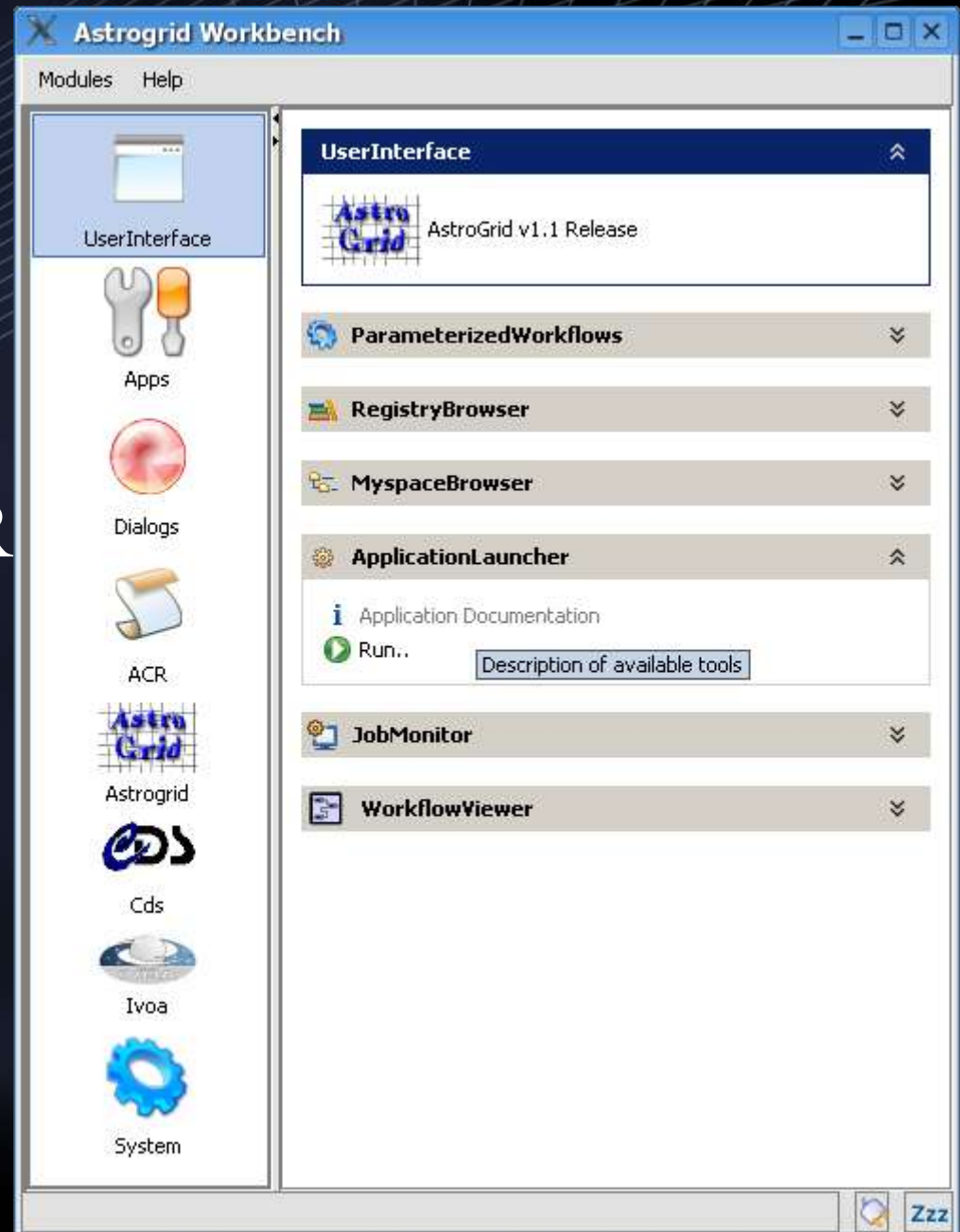
- ACR (Astro Client Runtime) is a desktop service that makes it simple for other programs to access VO services.
- Workbench is a suite of GUI applications built upon the ACR
- <http://software.astrogrid.org/userdocs/workbench.html>
 - Single-click launch using Java WebStart
 - choose 'Launch Workbench (Stable Version)'
 - try it now :)

Background: Java WebStart

- Available for Windows, Linux, Solaris, Mac
- The application is downloaded the first time the webstart link is clicked
 - May take a little time over wireless.
- Cached so that it starts instantly the next time.
- When new features / bug fixes are released, the cached version of the application is updated
 - Users always get the most up-to-date version, with no maintenance worries.
- Webstart handles native libs, different OS, desktop integration.
- Different parts of the application are signed by different organizations – users need to accept the trust dialogs

Workbench

- Main Menu Window
- Launch GUI applications
- Exposes primitive operations of the ACR



Workflow Editor

The screenshot displays the Workflow Builder interface with a workflow tree on the left and a detailed view of a task on the right. The workflow includes several steps:

- For / in** $\$(ccdno)$: A loop over CCD numbers.
- Sequence**: A sub-workflow containing:
 - For / in** $\$\{[0,1,2,3,4]\}$: A loop over bands.
 - Script**: `jes.info("Running SExtractor on image " + runno[i] + " (" + band[i] + ")")`
 - Step sex_COPY**: A task from `Taskorg.astrogrid/SExtractor Interface`.

Taskorg.astrogrid/SExtractor Interface	
Inputs	
Parameter File	<code>http://apm14.ast.cam.ac.uk/local/astr...</code> <input checked="" type="checkbox"/>
Configuration File	<code>http://apm14.ast.cam.ac.uk/local/astr...</code> <input checked="" type="checkbox"/>
SeeinName: AstroGrid Redshift Maker NG	<input type="checkbox"/>
DetectDescription: Calculate redshifts from imaging data	<input checked="" type="checkbox"/>
Input image band	<code>\\${band[i]}</code> <input type="checkbox"/>
detection filter	<code>http://apm14.ast.cam.ac.uk/local/astr...</code> <input checked="" type="checkbox"/>
magnitude zero-point	<code>\\${magzpt[i]}</code> <input type="checkbox"/>
Measurement Image	<code>\\${fileno[i]+j}</code> <input checked="" type="checkbox"/>
Analysis Threshold	1.5 <input type="checkbox"/>
Outputs	
Derived output catalogue	<code>\\${userIvorn}#votable/sex_\\\${band[i]}...</code> <input checked="" type="checkbox"/>
 - Script scriptx**:

```
source = astrogrid.ioHelper.getExternalValue(userIvorn.toString()+"#votable/sex_"+band[i]+"_"+j+".vot")
target = astrogrid.ioHelper.getExternalValue(userIvorn.toString()+"#votable/sex_"+band[i]+"_"+j+".vot")
table = astrogrid.tableHelper.builder.makeStarTable(source)
astrogrid.tableHelper.writeTable(target,table,"votable")
```
 - Script**: `jes.info("Cross-matching tables.")`
 - Step xmatchResult Variable xmatch**: A task from `Taskorg.astrogrid/CrossMatcher Interface`.

Taskorg.astrogrid/CrossMatcher Interface	
Inputs	
Match VOTable	<code>\\${userIvorn}#votable/sex_U_\\\${j}.vot</code> <input checked="" type="checkbox"/>
Target votable	<code>\\${userIvorn}#votable/sex_R_\\\${j}.vot</code> <input checked="" type="checkbox"/>
Match Radius	1.2 <input type="checkbox"/>
Maximum search radius	3.0 <input type="checkbox"/>
Match VOTable	<code>\\${userIvorn}#votable/sex_G_\\\${j}.vot</code> <input checked="" type="checkbox"/>
Match VOTable	<code>\\${userIvorn}#votable/sex_I_\\\${j}.vot</code> <input checked="" type="checkbox"/>
Match VOTable	<code>\\${userIvorn}#votable/sex_Z_\\\${j}.vot</code> <input checked="" type="checkbox"/>
Match VOTable	<input type="checkbox"/>
Outputs	
Resulting VOTable	<code>\\${userIvorn}#votable/xmatch_\\\${j}.vot</code> <input checked="" type="checkbox"/>
 - If** $\$(Integer.valueOf(do_hyperz) == 1)$:
 - Then**:
 - Sequence**:
 - Script**: `jes.info("Running HyperZ : "+userIvorn.toString()+"#votable/zphot_"+j+".vot")`
 - Step zphot_COPY**: A task from `Taskorg.astrogrid/HyperZ Interface`.

Taskorg.astrogrid/HyperZ Interface	
Inputs	
Configuration File	<code>http://apm14.ast.cam.ac.uk/local/a...</code> <input checked="" type="checkbox"/>
Band ordering	UGRIZ <input type="checkbox"/>

Application Launcher

Application Launcher - 6dF

Query: Find: roe

Parameter: Select an Application:

- SuperCOSMOS Science Archive
- 6dF
- 2MASS
- USNO-B

XML

Info

Chooser

```
<identifier xmlns:vm="http://www.ivoa.net/xml/VOMetadata/v0.1" xmlns:vt="http://www.ivoa.net/xml/VOTable/v0.1" iSA_2MASS/ceaApplication/>
<curator xmlns:vm="http://www.ivoa.net/xml/VOMetadata/v0.1" xmlns:vt="http://www.ivoa.net/xml/VOTable/v0.1">
  <publisher>Royal Observatory Edinburgh</publisher>
  <contact>
    <name>Martin Hill</name>
    <email>mch@roe.ac.uk</email>
  </contact>
</curator>
<content xmlns:vm="http://www.ivoa.net/xml/VOMetadata/v0.1" xmlns:vt="http://www.ivoa.net/xml/VOTable/v0.1">
  <description />
  <referenceURL>http://astrogrid.roe.ac.uk:8080</referenceURL>
</content>
</identifier>
```

Application Launcher - 6dF

Query: `select b.OBJID , b.CATNAME`

Parameter: `from DENISI as b`

XML

Info

Chooser

Set Catalog

Validate ADQL

Inputs

Format

Output

Result

- Cut
- Copy
- Paste
- Select All
- Insert SQL...
- Insert Column...

- 6df
- AGN2MASS
- DENISI
- DENISJ
- DURUKST
- FSC
- HES
- HIPASS
- NWSS
- RASS
- SHAPLEY
- SPECTRA
- SUMSS
- SUPERCOS
- TARGET

Myspace Browser

The screenshot shows a web browser window titled "VoSpace Browser". The main content area displays a file manager interface. On the left, a tree view shows the directory structure: home, fred, wibble, votable, intwfs, and boo. The main pane on the right lists files in a list view, including "Run-180480-CCD-1.fits" through "Run-168385-CCD-4.fits", and a folder named "wibble". The file "Run-169604-CCD-2.fits" is selected. Below the file list, a "Properties" window is open for the selected file, showing the following details:

Properties	
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

At the bottom of the browser window, there is an "Advanced" tab.

ACR – Purpose

- A uniform way to access VO components..
 - remote: web services – SOAP, REST, etc
 - client side: GUI components; dialogues; helper libraries
- ... from any programming, scripting or shell language
- ... on any platform

Aside: Ways to Consume VO Services

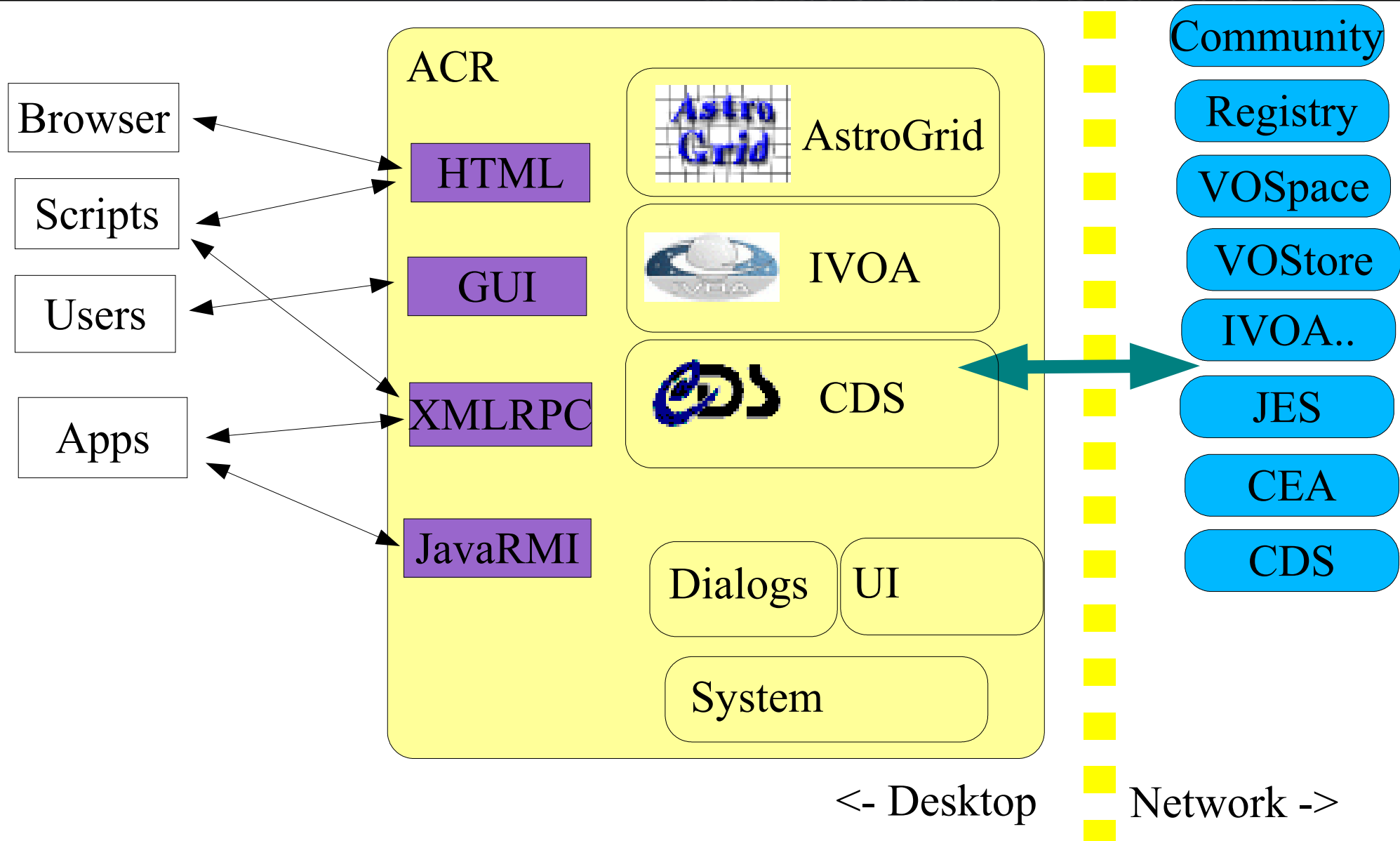
- Use published WSDL to generate own SOAP client, call services directly
 - need to understand how VO services interrelate
 - security – needs advanced SOAP handling.
- Use existing (AG) delegate libraries
 - Java-only
 - Maybe not the cleanest / most reusable
- Use ACR
 - Clean Facade Interface to AstroGrid (and soon all VO)
 - Provides extra benefits
- more info: <http://software.astrogrid.org/developerdocs/>

Astro Client Runtime

- *Application that runs on the user's desktop*
 - *accepts requests from other desktop applications*
 - *processes requests by calling VO Services using existing client libraries.*
- **Advantages**
 - **Shared state**
 - Single sign-on, configuration, preferences
 - **Common cache**
 - optimizes registry resolving, myspace filestructures
 - **Part of the Workbench – painless deployment & upgrade**
 - **uniform API accessible to all programming languages**
 - **easily extensible with new interfaces - PLASTIC**

Noel Winstanley nw@jb.man.ac.uk

ACR Schematic



Access Methods

- JavaRMI
 - Java-only inter-process communication
 - strongly typed
 - requires a minimal set of libs.
- XMLRPC
 - Forerunner of SOAP: <http://www.xmlrpc.com/>
 - simpler types than SOAP
 - implementations for a wide range of languages
- HTTP-Get
 - rough-n-ready procedure call
 - supported by every (useful) language

Functionality

- ACR provides a large suite of components; each has procedures that can be called via any of the access methods
- related components organized into modules.
- api documentation with examples and recipies
 - follow link from <http://software.astrogrid.org/developerdocs/>

astrogrid

- interface to AstroGrid system
 - handles authentication, service resolution, etc.
- registry – query, xquery, resolve service
- myspace – list, move, copy, create, upload, download
- applications – (CEA) list, build, execute, query, monitor, control
- workflows – (JES) build, execute, monitor, control

Background: Making Work Remote..

- CEA - Common Execution Architecture
 - Uniform method of registering and deploying remote applications and dataset access.
 - Application parameters can be given inline or as references to external resources (http / ftp / myspace)
- CEA applications can be executed directly, but are also the building blocks of *Workflows*
 - distributed execution of a series of CEA applications.
- JES – Job Execution System
 - service that schedules & executes *Workflow Documents*
 - decides on which CEA servers to place application executions.
 - records log & results of application execution
 - evaluates workflow control structures
 - executes workflow scripts.

more modules...

- dialogs - user interface components that prompt user for input
 - myspace / file / url chooser – 'file save dialogue'
 - registry resource chooser – 'registry google'
 - CEA application parameter editor
 - ADQL query editor
- ui – programming interface to workbench GUI

and more modules...

- cds
 - simple interface to existing CDS web services:
 - UCD resolver/ information
 - Simbad name resolver
 - VizieR search
- system
 - configuration – persistent store for settings
 - browser – control system web browser
 - help viewer

ivoa

- an interface to ivoa standard services.
- work in progress
- translators from ADQL/s <-> ADQL/x
- components that simplify calling SIAP, SSAP, Cone, etc will be added in next month
- afterwards will add a NVO module providing access to NVO-specific services.
 - SkyNode?
 - Datascope?

Uses of the ACR

- VO Commandline
 - unix-ey small composable commandline programs.
 - vols, voget, voput, reg-query, ls-jobs ...
 - implemented as Python scripts calling the ACR
- Python workflows
 - script contains control flow
 - performs work by querying DSA servers and running CEA applications via ACR
 - more interactive / iterative development than batch JES workflows
 - same could be done in Perl / IDL / ...

Other projects using ACR

- Aladin – Thomas Boch
 - MySpace, query registry, CEA X-match app
 - Java/RMI
- Topcat – Mark Taylor
 - MySpace
 - Java/XMLRPC
- VOTech – Richard Holbrey & John Taylor
 - connecting xmdv_light to MySpace
 - Java/RMI, C++/XMLRPC
- AstroNeural – Giuseppe Longo et al
 - upcoming

Testimonial from Thomas :)

- Reading from MySpace : easy to implement, easy win for VO applications

```
// recherche d'une instance d'ACR existante
Finder f = new Finder();
acr = f.find();
// ...
resChooser = (ResourceChooser)acr.getService(ResourceChooser.class);
URI uri = resChooser.fullChooseResource("MySpace Resource Chooser",
                                        true, false, false);
URL u = myspace.getReadContentURL(uri);
```

Future Plans

- Collaborate! (preferably somewhere warm)
- Add missing service types
 - SIAP, SSAP, SkyNode
- Track developing standards (VOStore, SSO)
- Get ACR interfaces standardized / approved by IVOA
 - mechanism?
- Help tool authors / scripty users get started.
- Refine Workbench applications

References

- Homepage & WebStart link:
<http://software.astrogrid.org/userdocs/workbench.html>
- Developer Documentation:
<http://software.astrogrid.org/developerdocs/>
 - API documentation
 - Getting started instructions for various access methods
 - Code examples in Java, C++, Python, Perl
- Experience of using ACR (Thomas Boch):
 - http://wiki.eurovotech.org/pub/VOTech/DS3PlanningStage02/ACR_in_action.pdf
 - <http://wiki.eurovotech.org/bin/view/VOTech/UsageOfAcrApiInAladin>