AstroGrid-D

Deliverable

Metadata Management

Providing Static Metadata of Robotic Telescopes to Stellaris[1]

Deliverable	D2.4
Authors	Working Group Metadata Management
Editors	Frank Breitling, Michael Braun
Date	May 15, 2007
Document Version	1.0.0
Current Version	1.0.0
Previous Versions	

A: Status of this Document

Deliverable 4 of working group 2.

B: Reference to project plan

Fourth deliverable of working group *Metadata Management*.

C: Abstract

Static metatdata of robotic telescopes as grid ressources is provided in RDF/XML to the information service Stellaris. The software components are discussed, and their usage is demonstrated by examples.

[1]This work is part of the AstroGrid-D project and D-Grid. The project is funded by the German Federal Ministry of Education and Resarch (BMBF).

D: Change History

Version	Date	Name	Brief summary
1.0.0	15.05.2007	Frank Breitling	Rephrased a few sentences.
0.9.6	05.04.2007	Michael Braun	Corrections to XML2RDF section.
0.9.5	05.04.2007	Frank Breitling	Spelling corrections according to R. Tucker.
0.9.4	04.04.2007	Frank Breitling	Corrections according to comments by H. Enke, added references.
0.9.3	02.04.2007	Frank Breitling	Corrected page width by removing preamble. Corrected table width.
0.9.2	30.03.2007	Frank Breitling	Corrected figure captions. Added order to SPARQL query.
0.9.1	26.03.2007	Frank Breitling	Added copyright of Apache license.
0.9	23.03.2007	Frank Breitling	Second draft.
0.8	21.03.2007	Frank Breitling	First draft.

E:

Contents

1 Introduction

The integration of robotic telescopes into the AstroGrid-D infrastructure aims at a global network of astronomic instruments based on grid technology. This integration foresees the usage of the information service Stellaris [6] [5] and the RTML format for the communication with telescopes. This document describes the exchange of static metadata of robotic telescopes described in RTML with Stellaris. The described methods are general enough to be applied to any metadata provided in XML format.

2 RTML and RDF as Metadata Formats

Metadata of robotic telescopes can be represented in a format established by the Heterogeneous Telescope Networks consortium called RTML (Remote Telescope Markup Language [4]). It is an XML format developed for the description of telescope resources and observation requests. It is defined through a schema, which specifies the information that can be contained. The information can be divided into static and dynamic metatdata. The static metatdata includes specifications regarding the hardware and setup of telescopes, calibration types and source catalogs. The dynamic metadata includes observation requests, schedules, observation histories as well as project and weather information. RTML was adopt in AstroGrid-D. An example RTML document representing static metadata of the location of the robotic telescope STELLA-I [14] is shown in the appendix (Listing 2).

A general representation of metadata is provided by the Resource Description Framework (RDF, [7]), which is used by the AstroGrid-D information service Stellaris [10]. RDF contains information in "subject - predicate - object" statements, called triples in RDF terminology. A collection of RDF triples represents a labeled, directed pseudo-graph. Two formats of RDF exist — RDF/XML and Notation 3. An example RDF/XML document which contains the same static metadata of Listing 2 of the robotic telescope STELLA-I is given in the appendix (Listing 3). Part of the corresponding RDF graph is shown in Fig. 1. The plot was obtained using the W3C validation service[2].

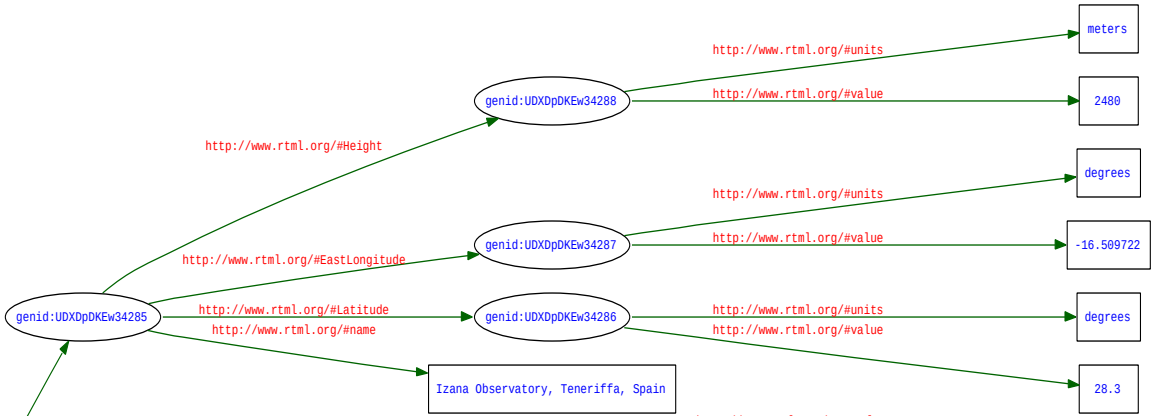

Figure 1: Partial RDF graph for the static metadata of the robotic telescope STELLA-I. The selected region shows the information regarding the telescope's location.

[2] http://www.w3.org/RDF/Validator/

3 Translation from RTML to RDF

Before information provided in RTML can be uploaded to Stellaris, a transformation into RDF is necessary. Two such transformations have been developed in AstroGrid-D. Both of them are discussed here, whereas the first represents the standard solution while the second became obsolete.

3.1 XSLT as Universal Transformation from XML into RDF

The XSL stylesheet [15] "xml2rdf.xsl" (Listing 4) was developed to transform RTML files into RDF/XML format [7] using an XSLT processor. The stylesheet is general enough to be applied to any XML document. Therefore it is also applied to metadata provided by the Monitor and Discovery Service (MDS) of the Globus Toolkit [9]. The "<oXygen/> XML Editor & XSLT Debugger" [11] for the open development platform "Eclipse" [2] has been very useful for the development.

The XSLT works as follows: first, all XML attributes of element nodes in the XML file are turned into subordinated element nodes. The name of an attribute will be used as the name of the new element node, while the value of the attribute will be represented as text node of the new element node. Next all text nodes are enclosed in a tag named "value". This conversion provides a uniform XML structure without mixed content and without attributes. All elements can now be converted into RDF nodes. The transformation can be accomplished with a single command line. For example, the following command line shows the conversion of the RTML document "STELLA-I.rtml" into the RDF/XML document "STELLA-I.rdf" using the XSL stylesheet "xml2rdf.xsl" and the XSL processor "xsltproc" [16]:

```
xsltproc -o STELLA-I.rdf xml2rdf.xsl STELLA-I.rtml
```

This solution is very appealing, since it provides a simple and universal solution for the translation of XML documents to RDF. The XSL stylesheet is available through the AstroGrid-D web site [10].

3.2 Transformation using OwlMap

An alternative method for the conversion from RTML to RDF was developed based on the Java package OwlMap [8], which consists of the two programs XS2DAMLOIL and XML2RDF. OwlMap was used, before the XSLT stylesheet had been developed. Although it provides identical results, it has some disadvantages for which the usage of the XSLT is now preferred. A disadvantage is the dependence on the RTML schema, which has to be translated using XS2DAMLOIL into an auxiliary dictionary before the actual RTML file can be transformed. Moreover, XS2DAMLOIL could only be applied after small modifications to the original RTML schema. This fact compromises the compatibility to new versions of the RTML schema. In addition XS2DAMLOIL had to be modified to include a newer version of the Jena library. Also a special treatment of the complex recursions inside the XML schema definition of RTML had to be included into the code. An additional problem was given by the co-existence of text nodes and attributes of the same element node in RTML. Since the XML2RDF could not handle such elements, a pre-processing of the RTML file by an XSL transformation was necessary to transforms all text nodes into an attribute named "value". Since attribute "value" is not defined in the RTML schema, the XML document obtained after pre-processing fails the standard RTML schema validation. However, the XML2RDF produces the same RDF/XML document which is obtained with the xml2rdf.xsl stylesheet described in Sec. 3.1.

4 Exchange of Telescope Metadata with Stellaris

4.1 Exchanging Metadata via HTTP

The RDF metadata of telescopes can be uploaded to and retrieved from Stellaris through the Stellaris HTTP interface. A few examples are given below using the command line tool cURL [1].

The following command demonstrates uploading of static metadata of the robotic telescope STELLA-I to the information service running on the host mintaka.aip.de. The information is stored individually for each telescope under the context "telescopes".

```
curl -vT STELLA-I.rdf http://mintaka.aip.de:24060/context/telescopes/STELLA-I
```

The next command retrieves stored information for telescope Robotel in Notation 3 [12].

```
curl http://mintaka.aip.de:24060/context/telescopes/Robotel?format=n3
```

The last command shows how information for the telescope Robotel is deleted.

```
curl -vX DELETE http://mintaka.aip.de:24060/context/telescopes/Robotel
```

4.2 Retrieving Metadata with SPARQL Queries

The metadata stored in Stellaris can be retrieved using SPARQL [13] queries via the Stellaris HTTP interface. SPARQL queries can be executed through Perl scripts using the corresponding Perl modules or they can be directly entered in the Stellaris web browser interface. An example of a SPARQL query for the geographic location of all available telescopes is shown in Listing 1. The response from Stellaris is shown in Tbl. 1. The list is sorted by the height as requested.

Listing 1: SPARQL query for retrieving a list of telescopes ranked by their height.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rtml: <http://www.rtml.org/v3.1a#>
SELECT ?telescope ?latitude ?longitude ?height
#FROM NAMED <http://stellaris.gac-grid.org/context/gridmap/rtel#context>
WHERE { ?telescope rtml:RTML ?n1 .
        ?n1 rtml:Telescope ?n2 .
        ?n2 rtml:Location ?loc .
        ?loc rtml:Latitude ?lat .
        ?lat rtml:value ?latitude .
        ?loc rtml:EastLongitude ?long .
        ?long rtml:value ?longitude .
        ?loc rtml:Height ?vheight .
        ?vheight rtml:value ?height .
} ORDER BY DESC(?height)
```

4.3 Retrieving Metadata through the Web Browser Interface

Also a graphical web browser user interface has been developed. It is called "Telescope Map" and based on the Grid Resource Map [10]. The Grid Resource Map includes the application interface of Google Maps [3] on which the telescope's location is displayed. If a telescope is selected by mouse click, additional information about the telescope's capabilities, such as filter types are displayed.

Table 1: A list of the geographic locations of registered telescopes sorted by their height is the result of the SPARQL query in Listing 1.

Telescope UID	Latitude	Longitude	Height
rtml://de.aip.STELLA-I	28.300000	-16.509722	2480
rtml://de.aip.STELLA-II	28.300000	-16.509722	2480
rtml://de.Uni-Goettingen.MONET	30.6717	-104.0283	2075
rtml://at.ac.univie.Wolfgang	31.386628	-110.69487	1600
rtml://at.ac.univie.Amadeus	31.386628	-110.69487	1600
rtml://de.aip.Robotel	52.404833	13.1016639	80

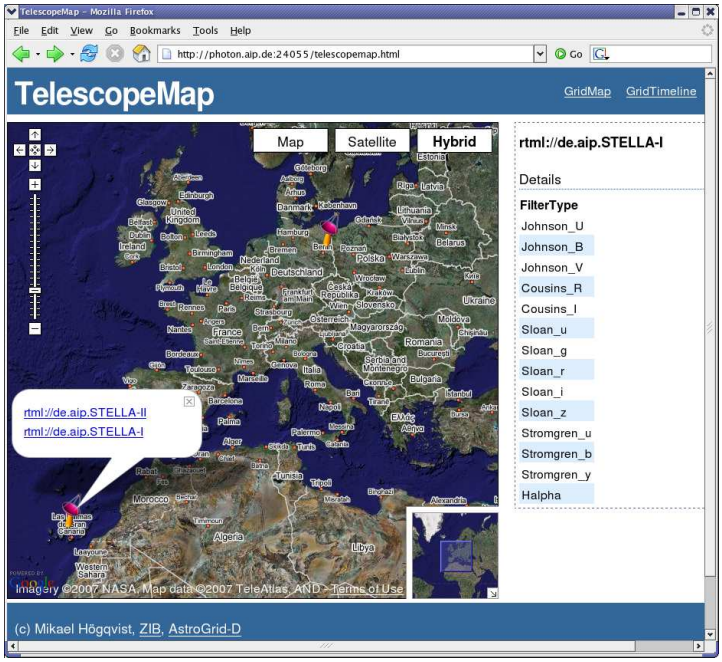

Figure 2: Screen shot of the "Telescope Map", which provides a user interface to metadata of registered telescopes. It is accessed through a web browser and shows the location and setup of telescopes.

5 Conclusion

For the integration of static metadata of robotic telescopes into the AstroGrid-D information service Stellaris, several software components have been developed and their usage has been described. They allow a general transformation of XML and in particular RTML metadata into RDF/XML format and the uploading to Stellaris. From Stellaris the information can be retrieved using SPARQL queries. Moreover a web browser user interface was developed to provide access to metadata of telescopes such as the location and capabilities of integrated telescopes.

A Listings

Listing 2: RTML description for the telescope STELLA-I.

```
<?xml version="1.0" encoding="UTF-8"?>
<RTML version="3.1a" mode="resource" uid="rtml://DE.aip.STELLA-I"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns="http://www.rtml.org/v3.1a"
    xsi:schemaLocation="http://www.rtml.org/v3.1a http://monet.uni-sw.gwdg.de/
        XMLSchema/RTML/schemas/RTML-nightly-target.xsd">
    <History>
        <Entry timeStamp="2006-09-13T16:51:29">
            <Agent name="rtmlServer" uri="http://www.rtml.org/cgi-bin/
                WebObjects/RTMLServer.woa/"/>
        </Entry>
    </History>
 <Telescope name="STELLA-I">
   <Aperture type="geometric" units="meters">1.2</Aperture>
   <SpectralRegion>optical</SpectralRegion>
   <Camera name="WIFSIP">
     <Description>WiFSIP</Description>
     <Detector>
       <NumColumns>4096</NumColumns>
       <NumRows>4096</NumRows>
     </Detector>
     <PlateScale>21.486</PlateScale>
     <FilterWheel>
       <Filter type="Johnson_U" name="U"/>
       <Filter type="Johnson_B" name="B"/>
       <Filter type="Johnson_V" name="V"/>
       <Filter type="Cousins_R" name="R"/>
       <Filter type="Cousins_I" name="I"/>
       <Filter type="Sloan_u" name="uprime"/>
       <Filter type="Sloan_g" name="gprime"/>
       <Filter type="Sloan_r" name="rprime"/>
       <Filter type="Sloan_i" name="iprime"/>
       <Filter type="Sloan_z" name="zprime"/>
       <Filter type="Stromgren_u" name="u"/>
       <Filter type="Stromgren_b" name="b"/>
       <Filter type="Stromgren_y" name="y"/>
       <Filter type="Halpha" name="Han"/>
     </FilterWheel>
   </Camera>
   <FocalLength units="meters">9.600</FocalLength>
   <FocalRatio>f/8</FocalRatio>
   <Location name="Izana Observatory, Teneriffa, Spain">
     <Latitude units="degrees">28.3</Latitude>
     <EastLongitude units="degrees">-16.509722</EastLongitude>
     <Height units="meters">2480</Height>
   </Location>
 </Telescope>
</RTML>
```

Listing 3: RDF/XML description of the telescope STELLA-I (truncated).

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <rdf:Description rdf:about="rtml://de.aip.STELLA-I">
    <ns1:RTML xmlns:ns1="http://www.rtml.org/v3.1a#" rdf:parseType="Resource">
      <ns1:version>3.1a</ns1:version>
      <ns1:mode>resource</ns1:mode>
      <ns1:uid>rtml://de.aip.STELLA-I</ns1:uid>
      <ns1:schemaLocation>http://www.rtml.org/v3.1a http://monet.uni-sw.gwdg.de
          /XMLSchema/RTML/schemas/RTML-nightly-target.xsd</ns1:schemaLocation>
      <ns1:History rdf:parseType="Resource">
        <ns1:Entry rdf:parseType="Resource">
          <ns1:timeStamp>2006-09-13T16:51:29</ns1:timeStamp>
          <ns1:Agent rdf:parseType="Resource">
            <ns1:name>rtmlServer</ns1:name>
            <ns1:uri>http://www.rtml.org/cgi-bin/WebObjects/RTMLServer.woa/</
                ns1:uri>
          </ns1:Agent>
        </ns1:Entry>
      </ns1:History>
      <ns1:Telescope rdf:parseType="Resource">
        <ns1:name>STELLA-I</ns1:name>
        <ns1:Aperture rdf:parseType="Resource">
          <ns1:type>geometric</ns1:type>
          <ns1:units>meters</ns1:units>
          <ns1:value>1.2</ns1:value>
        </ns1:Aperture>
        <ns1:SpectralRegion rdf:parseType="Resource">
          <ns1:value>optical</ns1:value>
        </ns1:SpectralRegion>
        <ns1:Camera rdf:parseType="Resource">
          <ns1:name>WIFSIP</ns1:name>
          <ns1:Description rdf:parseType="Resource">
            <ns1:value>WiFSIP</ns1:value>
          </ns1:Description>
```

Listing 4: XSLT stylesheet for transforming XML documents into RDF/XML.

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- xml2rdf.xsl          is the first freely available XSLT stylesheet for
                          transforming XML into RDF/XML documents. It is a
                          contribution of the AstroGrid-D to the public domain.

     Authors              Michael Braun and
                          Frank Breitling (fbreitling at aip.de)

     Version              1.0  (Mar 26, 2007)

     Example Usage        xsltproc xml2rdf.xsl file.rtml

     Copyright 2007       AstroGrid-D (http://www.gac-grid.org)

     Licensed under the Apache License, Version 2.0 (the "License");
     you may not use this file except in compliance with the License.
     You may obtain a copy of the License at

         http://www.apache.org/licenses/LICENSE-2.0
```

```
     Unless required by applicable law or agreed to in writing, software
     distributed under the License is distributed on an "AS IS" BASIS,
     WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
     See the License for the specific language governing permissions and
     limitations under the License.
 -->

<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
       xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">

       <xsl:strip-space elements="*"/>
       <xsl:output method="xml" indent="yes"/>

       <!-- Begin RDF document -->
       <xsl:template match="/">
              <xsl:element name="rdf:RDF">
                     <rdf:Description rdf:about="{*/@uid}">
                            <xsl:apply-templates/>
                     </rdf:Description>
              </xsl:element>
       </xsl:template>

       <!-- Turn nodes into blank nodes and keep their namespaces -->
       <xsl:template match="*">
              <xsl:element name="{name()}" namespace="{namespace-uri()}#">
                     <xsl:attribute name="rdf:parseType">Resource</
                         xsl:attribute>
                     <xsl:apply-templates select="node()|@*"/>
              </xsl:element>
       </xsl:template>

       <!-- Convert any attribute into an element and apply namespace-uri -->
       <xsl:template match="@*">
              <xsl:element name="{name()}" namespace="{namespace-uri(..)}#">
                     <xsl:value-of select="."/>
              </xsl:element>
       </xsl:template>

       <!-- Enclose text nodes in value tags -->
       <xsl:template match="text()">
          <!-- Retrieve the namespace prefix of the parent node and generate
               the name of the new element by appending 'value'
               If a prefix is not found, the name will be 'value' only -->
          <!-- Do not disrupt the next line !!! -->
          <xsl:variable name="valname"><xsl:if test="contains(name(..),':')"><
              xsl:value-of select="substring-before(name(..),':')"/>:</xsl:if>
              value</xsl:variable>
          <!-- Do not disrupt the previous line !!! -->

          <!-- Create a new element from the text node using
               $valname and the namespace of the parent node -->
          <xsl:element name="{$valname}" namespace="{namespace-uri(..)}#">
             <xsl:value-of select="."/>
          </xsl:element>
       </xsl:template>

</xsl:stylesheet>
```

F: References / Bibliography

References

[1] cURL. http://curl.haxx.se/, 2007.

[2] Eclipse - an open development platform. http://www.eclipse.org/, 2007.

[3] Google Maps. http://maps.google.com/, 2007.

[4] F. V. Hessman. Remote Telescope Markup Language (RTML). *Astronomische Nachrichten*, 327:751–+, September 2006.

[5] M. Högqvist, T. Röblitz, and A. Reinefeld. Stellaris: An rdf-based information service for astrogrid-d. In *German e-Science conference*, May 2007.

[6] M. Högqvist, T. Röblitz. AstroGrid-D Information Service Requirements Specification and Architectural Design. Technical Report D2.1, AstroGrid-D project, July 2006.

[7] F. Manola and E. Miller. RDF primer. http://www.w3.org/TR/rdf-primer/, February 2004.

[8] Matthias Ferdinand, Christian Zirpins, and David Trastour. Lifting XML Schema to OWL. http://vsis-www.informatik.uni-hamburg.de/getDoc.php/publications/204/f%zt-lxs-04.pdf, http://fresco-www.informatik.uni-hamburg.de/technology/index.php, March 2004.

[9] MDS - Monitoring and Discovery System. http://www.globus.org/mds/, 2007.

[10] Members of AstroGrid-D. AstroGrid-D Project Products. http://www.gac-grid.org/project-products/Software.html, 2007.

[11] <oXygen/> XML Editor & XSLT Debugger. http://www.oxygenxml.com, 2007.

[12] RDF, Notation 3 Primer. http://www.w3.org/2000/10/swap/Primer/, 2007.

[13] SPARQL Query Language for RDF, 2006.

[14] K. G. Strassmeier, T. Granzer, M. Weber, M. Woche, M. I. Andersen, J. Bartus, S.-M. Bauer, F. Dionies, E. Popow, T. Fechner, G. Hildebrandt, A. Washuettl, A. Ritter, A. Schwope, A. Staude, J. Paschke, P. A. Stolz, M. Serre-Ricart, T. de la Rosa, and R. Arnay. The STELLA robotic observatory. *Astronomische Nachrichten*, 325:527–+, October 2004.

[15] The Extensible Stylesheet Language Family (XSL). http://www.w3.org/Style/XSL/, 2007.

[16] xsltproc. http://www.xmlsoft.org/, 2007.