

Appendix A

LONI Pipeline Overview

1. LONI Pipeline Server Module

The LONI Pipeline server module handles the authentication of users, execution of pipelines and facilitates the progress tracking. It is primarily configured via the preferences.xml.

1.1 Preferences.xml

This is the principal file to configure the Loni Pipeline server module, and outlining components of the file help us get a significant view over the internals of LONI Pipeline. The following significant items are specified in the file.

1.1.1 Temporary folder location for dumping intermediary output , pipeline and status data

The following things are stored in the temporary location

1. Directories are created for each executing workflows
2. Output and error stream of executed applications (shown in the interface to track progress)
3. Sequence of commands to be executed (the pipeline)

Directory structure

```
irfan-mbp:2008April15_10h33m54s750ms irfan$ ls
commands.xml  streams
```

Figure A1-1: Directory Listing

Figure A1-1 shows the directory listing of 2008April15_10h33m54s750ms. This folder was created for a pipeline executed on 15th, 10:33:54.750 am. LONI Pipeline does not identify users, and server administrators have open access to the execution state of the pipeline. It contains two items, one file commands.xml which represents the sequence of commands of the pipeline in XML and streams, which are the outputs/error streams of the individual applications.

```
<?xml version="1.0" encoding="UTF-8"?>
<commandLog version="1.0" cacheFileDir="/Users/irfan/tmp/irfan/2008April15_10h33m54s750ms/">
  <module name="CompressImg_0" rootModule="true">
    <command instance="0" command="/usr/bin/tar -czf /Users/irfan/desktop.ini -o /Users/irfan/tmp/irfan/2008April15_10h33m54s750ms/CompressImg.Newparameter1_0-0"/>
    <postProcessing from="pipeline://localhost//Users/irfan/tmp/irfan/2008April15_10h33m54s750ms/CompressImg.Newparameter1_0-0" to="pipeline://localhost//Users/irfan/Desktop/compressed.tar.gz"/>
  </module>
</commandLog>
```

Figure A1-2: Sample commands.xml file

As we can observe from commands.xml file (shown in Figure A1-2), we can see that that sequences of the pipeline are represented by <command> tags, the command which is executed is outlined in the command option to the tag. In this case both input and output are produced in the server. After the command executes the output is moved to the appropriate client node. This task is done via the <postProcessing>

Tag.

Hence one of the tasks of the LONI Pipeline Interface is to create an xml file which translates the user created visual pipeline into a sequence of commands to be executed at the server. We call this component the “Pipeline-Command Translator”.

Other items outlined in the preferences.xml file:

6.2.1 Authentication, which is JAAS based

LONI Pipeline uses a simple login/password scheme.

• Command line string to access Grid Queue

LONI Pipeline supports submission to a Cluster service which supports both DRMAA and DRMAA Java bindings, which includes engines such as PBS, Sun Grid Engine and Gridway, and notably others like Condor lack Java bindings hence are not supported.

The basic steps of how LONI Pipeline server uses the bindings is as follows:

- Obtain an object representing a DRMAA session, form the library such as in the case of Sun Grid engine.
- Open the session.
- Create a job template object describing various aspects of the job
- Submit the Job to a Queue
- Wait till execution is complete (synchronous wait)
- Close the session and exit.

1.1.4. Location to the Server library

The Server library

To make the execution of a task available over a workflow, a module for the task has to be defined. A module is the same as an executable process. A module specification outlines the inputs, the switches to be passed to the process and the output. For example the process MINCMask, which is a part of MNI-BIC tools is described by the following module, depicted in Figure A1-3:



Figure A1-3: MINCMask Specification

There are numerous tags which need to be specified but the most important are the `<module>`, `<input>` and `<output>` tags. The module tag identifies the location of the executable, in the above figure we can see that it is:

```
'pipeline://cranium.loni.ucla.edu//usr/local/mni_64bit/bin/mincmask
```

The first part outlines on which server the executable resides. It is possible to create workflows of executables which reside in different servers. There is no migration of processes rather the servers execute only their local executables.

The `<input>` tag identified the switches, which are parsed to the executable. The first two outlined inputs are a file, hence the LONI Pipeline interface will expect file inputs. The other inputs, are not files, rather switches, hence those switches will be appended to the command. The `<output>` tag identifies what kind of output will be results from the process. Both the metadata about the input and output are used for validating the pipeline. The MINCMask process is displayed as the following way in the interface.



Figure A1-4: MINCMask Icon

The top two circles are connectors for inputs, and the triangle is the output. XML descriptions are written for each process which is to be supported and put into a folder, which is pointed to by the preferences.xml file.

2. LONI Pipeline Client

The LONI Pipeline client supports these basic functions:

- Ability to connect to a server
- Visualize module descriptions into a user-friendly format
- Ability to define input data sources
- Ability to define output location
- Monitor the execution of the pipeline

- **Ability to connect to a server**

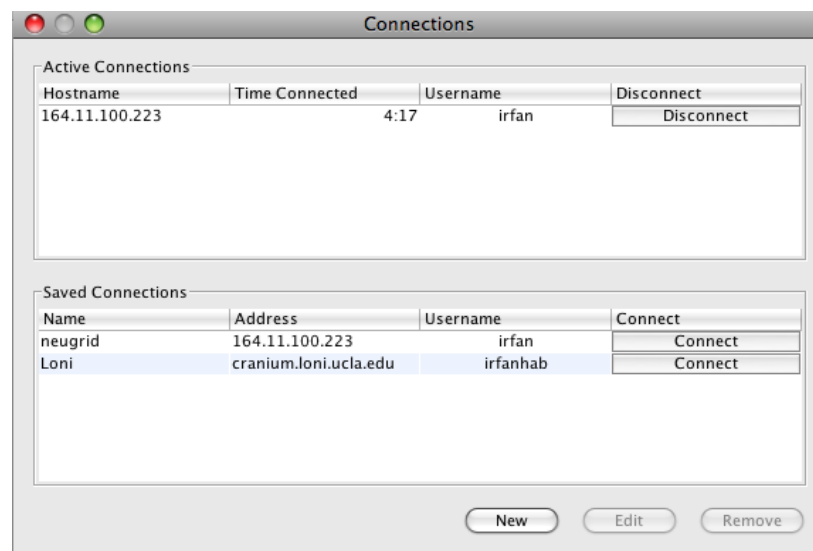


Figure A1-5: Connecting to a LONI Server

The LONI Pipeline client is capable of connecting to multiple servers at once (Connections interface is depicted in Figure A1-5), and users can create pipelines of processes of multiple servers, the servers only execute their local pipelines.

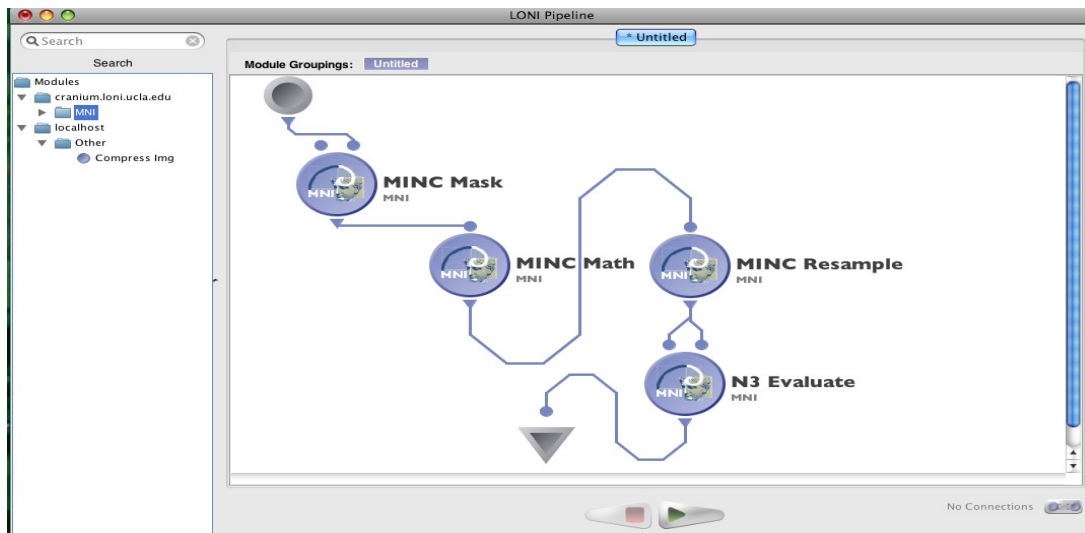


Figure A1-6: LONI Interface

The modules of the servers are listed to the right of the interface, as shown in Figure A1-6, and a pipeline can be designed using the modules.

2.2 Execution

Once an execution of a pipeline is invoked, the pipeline is first validated from the module xml definitions and finally it is executed on the server.

Pipeline can be tracked from the Execution Information Window. From here the output and error streams of the processes and the current command being executed are displayed. Output files are listed

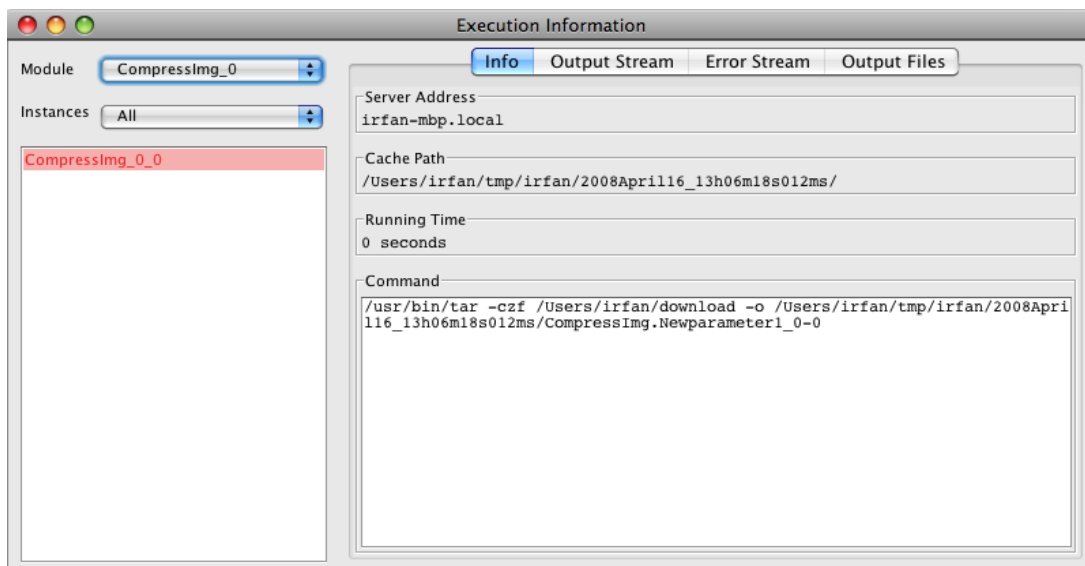


Figure A1-7: Execution Output

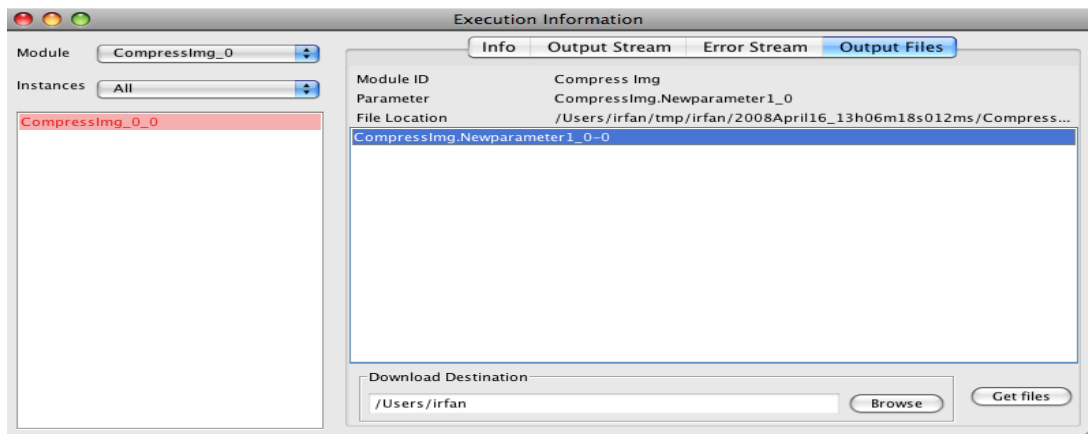


Figure A1-8: Execution Output

3.0 Architectural Interaction

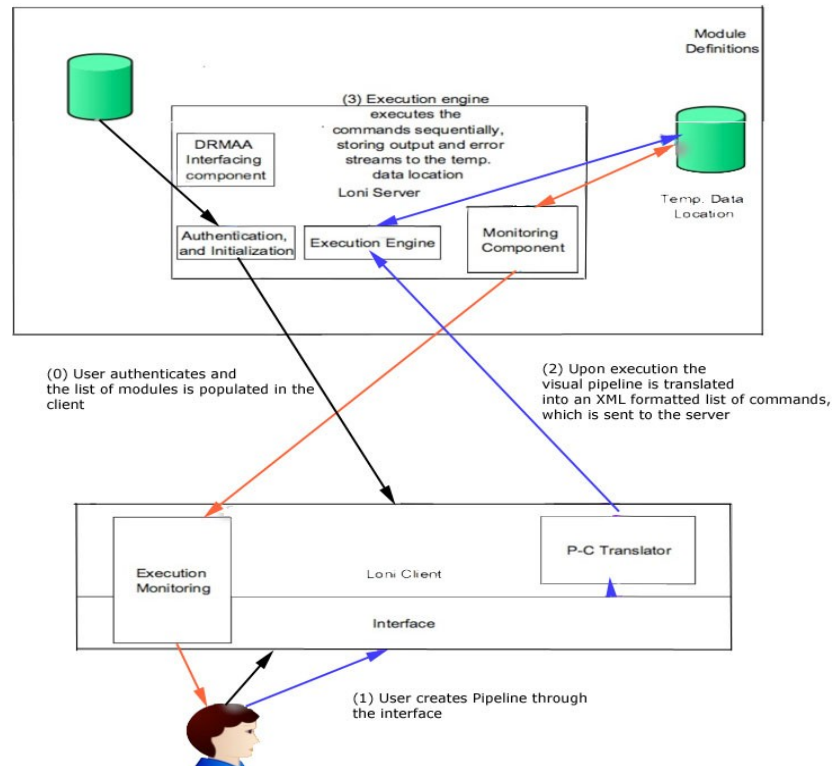


Figure A1-9: LONI Architecture

In Figure A1-9, the architecture of LONI Pipeline is presented. Architecturally the whole package can perform three tasks:

- Authentication and Initialization (Shown by black lines in Figure A1-9)

This involves the password-based authentication of a user against a server. Initialization involves populating the available modules list on the client with the modules available on the server

- Pipe Line creation and Execution (Shown by blue lines in Figure A1-9)

This primarily involves the ability to graphically create pipeline. The client validates this user created pipeline and converts it into an XML formatted list for the server to execute. The server executes this pipeline sequentially, storing all output, as well dumps of the output and error stream into the temp. directory created for the pipeline.

- Execution Monitoring (show by orange lines in Figure A1-9)

1. This component primarily allows the user to monitor the execution of the pipeline. This simply involves displaying the output, error streams and related data in an organized fashion on the interface.

Appendix B

Workflow in Kepler MoML format

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE entity PUBLIC "-//UC Berkeley//DTD MoML 1//EN"
"http://ptolemy.eecs.berkeley.edu/xml/dtd/MoML_1.dtd">
<entity name="demomodel" class="ptolemy.actor.TypedCompositeActor">
  <property name="_createdBy" class="ptolemy.kernel.attributes.VersionAttribute"
value="7.1.devel">
  </property>
  <property name="_windowProperties"
class="ptolemy.actor.gui.WindowPropertiesAttribute" value="{1551, bounds=38, 883,
802), maximized=false}">
  </property>
  <property name="_vergilSize" class="ptolemy.actor.gui.SizeAttribute" value="[600,
681]">
  </property>
  <property name="_vergilZoomFactor" class="ptolemy.data.expr.ExpertParameter"
value="1.0">
  </property>
  <property name="_vergilCenter" class="ptolemy.data.expr.ExpertParameter" value="">
  </property>
  <entity name="MincDefrag" class="eu.neuGrid.pipelineservice.mincdefrag">
    <property name="entityId" class="org.kepler.moml.NamedObjId" value="urn:lsid:kepler-
project.org:actor:545:1">
    </property>
    <property name="class" class="ptolemy.kernel.util.StringAttribute"
value="eu.neuGrid.pipelineservice.mincdefrag">
    <property name="id" class="ptolemy.kernel.util.StringAttribute" value="urn:lsid:kepler-
project.org:actor:545:1">
    </property>
    </property>
    <property name="semanticType00" class="org.kepler.sms.SemanticType"
value="urn:lsid:localhost:onto:2:1#Constant">
    </property>
    <property name="_location" class="ptolemy.kernel.util.Location" value="(155, 265)">
    </property>
    <port name="ClassifyFile" class="ptolemy.actor.TypedIOPort">
    <property name="input"/>
    </port>
```



```

<port name="value" class="ptolemy.actor.TypedIOPort">
<property name="input"/>
</port>
<port name="value2" class="ptolemy.actor.TypedIOPort">
<property name="input"/>
</port>
<port name="output" class="ptolemy.actor.TypedIOPort">
<property name="output"/>
<property name="multiport"/>
</port>
</entity>
<entity name="MincDefrag2" class="eu.neuGrid.pipelineservice.mincdefrag">
<property name="entityId" class="org.kepler.moml.NamedObjId" value="urn:lsid:kepler-
project.org:actor:545:1">
</property>
<property name="class" class="ptolemy.kernel.util.StringAttribute"
value="eu.neuGrid.pipelineservice.mincdefrag">
<property name="id" class="ptolemy.kernel.util.StringAttribute" value="urn:lsid:kepler-
project.org:class:545:1">
</property>
</property>
<property name="semanticType00" class="org.kepler.sms.SemanticType"
value="urn:lsid:localhost:onto:2:1#Constant">
</property>
<property name="_location" class="ptolemy.kernel.util.Location" value="(340, 350)">
</property>
<port name="ClassifyFile" class="ptolemy.actor.TypedIOPort">
<property name="input"/>
</port>
<port name="value" class="ptolemy.actor.TypedIOPort">
<property name="input"/>
</port>
<port name="value2" class="ptolemy.actor.TypedIOPort">
<property name="input"/>
</port>
<port name="output" class="ptolemy.actor.TypedIOPort">
<property name="output"/>
<property name="multiport"/>
</port>
</entity>
<entity name="CorticalSurface" class="eu.neuGrid.pipelineservice.cortical_surface">

```

```

<property name="entityId" class="org.kepler.moml.NamedObjId" value="urn:lsid:kepler-
project.org:actor:445:1">
</property>
<property name="class" class="ptolemy.kernel.util.StringAttribute"
value="eu.neuGrid.pipelineservice.cortical_surface">
<property name="id" class="ptolemy.kernel.util.StringAttribute" value="urn:lsid:kepler-
project.org:actor:445:1">
</property>
</property>
<property name="semanticType00" class="org.kepler.sms.SemanticType"
value="urn:lsid:localhost:onto:2:1#Constant">
</property>
<property name="_location" class="ptolemy.kernel.util.Location" value="(490, 445)">
</property>
<port name="ClassifyFile" class="ptolemy.actor.TypedIOPort">
<property name="input"/>
</port>
<port name="value" class="ptolemy.actor.TypedIOPort">
<property name="input"/>
</port>
<port name="output" class="ptolemy.actor.TypedIOPort">
<property name="output"/>
<property name="multiport"/>
</port>
</entity>
<entity name="String Constant" class="ptolemy.actor.lib.StringConst">
<property name="firingCountLimit" class="ptolemy.data.expr.Parameter" value="NONE">
</property>
<property name="NONE" class="ptolemy.data.expr.Parameter" value="0">
</property>
<property name="value" class="ptolemy.data.expr.Parameter"
value="/tmp/test_0001_classify.mnc">
</property>
<property name="KeplerDocumentation"
class="ptolemy.vergil.basic.KeplerDocumentationAttribute">
<property name="description"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>null</configure></property>
<property name="author"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>Edward Lee</configure></
property>
<property name="version"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>null</configure></property>

```

>

```
<property name="userLevelDocumentation"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>&lt;p&gt;The
StringConstant actor outputs a string specified via the actor's value
parameter.&lt;/p&gt;&#10;&#10;&lt;p&gt;Specifying strings with the StringConstant actor
is convenient, as the actor does not require that strings be surrounded by quotes. The actor
is often used to specify file paths, which can be selected using the Browse button available
in the actor's parameters.&lt;/p&gt;&#10;&#10;&lt;p&gt;Specified string values can
include references to parameters within scope (i.e., parameters defined at the same level of
the hierarchy or higher). &lt;/p&gt;&#10;&#10;&lt;p&gt;NOTE: If using a PN Director,
the 'firingCountLimit' parameter is often set to a finite integer (e.g. '1') so that the workflow
will terminate. &lt;/p&gt;&#10;&#10;</configure></property>
```

```
<property name="port:output"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>An output port that
broadcasts a string constant specified by the value parameter. </configure></property>
```

```
<property name="port:trigger"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>A multiport that has no
declared type (in other words, the port can accept any data type: double, int, array, etc.) If
the port is connected, the actor will not fire until the trigger port receives an input token.
Connecting the port is optional, but useful when scheduling the actor to perform at a certain
time. </configure></property>
```

```
<property name="prop:firingCountLimit"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>The limit on the number of
times the actor will fire. The default value is 'NONE', meaning there is no limit on the
number of time the constant will be provided to the output port. Any integer can be
provided as a value for this parameter.</configure></property>
```

```
<property name="prop:value"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>The value produced by the
actor. Specified strings do not require enclosing quotes. (To include a '$' sign in the string,
enter '$$'.)</configure></property>
```

```
</property> <property name="entityId" class="org.kepler.moml.NamedObjId"
value="urn:lsid:kepler-project.org:actor:204:1">
```

```
</property>
```

```
<property name="class" class="ptolemy.kernel.util.StringAttribute"
value="ptolemy.actor.lib.StringConst">
```

```
<property name="id" class="ptolemy.kernel.util.StringAttribute" value="urn:lsid:kepler-
project.org:class:1052:1">
```

```
</property>
```

```
</property>
```

```
<property name="semanticType00" class="org.kepler.sms.SemanticType"
value="urn:lsid:localhost:onto:1:1#StringFunctionActor">
```

```
</property>
```

```
<property name="semanticType11" class="org.kepler.sms.SemanticType"
value="urn:lsid:localhost:onto:2:1#Constant">
```

```
</property>
```

```
<property name="_icon" class="ptolemy.vergil.icon.BoxedValueIcon">
```

```
<property name="attributeName" class="ptolemy.kernel.util.StringAttribute"
value="value">
```

```
</property>
```

```

<property name="displayWidth" class="ptolemy.data.expr.Parameter" value="60">
</property>
</property>
<property name="_location" class="ptolemy.kernel.util.Location" value="">
</property>
</entity>
<entity name="Constant" class="ptolemy.actor.lib.Const">
<property name="firingCountLimit" class="ptolemy.data.expr.Parameter" value="NONE">
</property>
<property name="NONE" class="ptolemy.data.expr.Parameter" value="0">
</property>
<property name="value" class="ptolemy.data.expr.Parameter" value="3">
</property>
<property
                                name="KeplerDocumentation"
class="ptolemy.vergil.basic.KeplerDocumentationAttribute">
<property
                                name="description"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>null</configure></property>
>
<property
                                name="author"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>null</configure></property>
>
<property
                                name="version"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>null</configure></property>
>
<property
                                name="userLevelDocumentation"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>&lt;p&gt;The Constant
actor outputs a constant, which is specified by the value parameter. By default, the actor
outputs the integer 1.&lt;/p&gt;&#10;&#10;&lt;p&gt;The actor can be used to output other
types of values, e.g., a string (such as &quot;Hello&quot;) or a double (such as 1.2). The
actor's output type matches the type of the specified
value.&lt;/p&gt;&#10;&#10;&lt;p&gt;NOTE: If using a PN Director, the
'firingCountLimit' parameter is often set to a finite integer (e.g. '1') so that the workflow will
terminate. &lt;/p&gt;&#10;</configure></property>
<property
                                name="port:output"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>An output port that
broadcasts the specified constant. By default, the output is 1.</configure></property>
<property
                                name="port:trigger"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>A multiport that has no
declared type (in other words, the port can accept any data type: double, int, array, etc.) If
the port is connected, the actor will not fire until the trigger port receives an input token.
Connecting the port is optional, but useful when scheduling the actor to perform at a certain
time.</configure></property>
<property
                                name="prop:firingCountLimit"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>The limit on the number of
times the actor will fire. The default value is 'NONE', meaning there is no limit on the
number of time the constant will be provided to the output port. Any integer can be
provided as a value for this parameter.</configure></property>
<property
                                name="prop:value"

```

```

class="ptolemy.kernel.util.ConfigurableAttribute"><configure>The value produced by the
Constant actor. By default, the value is the integer token 1. The value can be set to another
type, e.g., a string (such as &quot;Hello&quot;,) or a double (such as 1.2). The output type
matches the type of the value specified here.</configure></property>
</property>    <property    name="entityId"        class="org.kepler.moml.NamedObjId"
value="urn:lsid:kepler-project.org:actor:1:1">
</property>
<property          name="class"                class="ptolemy.kernel.util.StringAttribute"
value="ptolemy.actor.lib.Const">
<property name="id" class="ptolemy.kernel.util.StringAttribute" value="urn:lsid:kepler-
project.org:class:877:1">
</property>
</property>
<property          name="semanticType00"        class="org.kepler.sms.SemanticType"
value="urn:lsid:localhost:onto:1:1#ConstantActor">
</property>
<property          name="semanticType11"        class="org.kepler.sms.SemanticType"
value="urn:lsid:localhost:onto:2:1#Constant">
</property>
<property name="kar" class="ptolemy.kernel.util.StringAttribute" value="urn:lsid:kepler-
project.org:kar:57:1">
</property>
<property name="_icon" class="ptolemy.vergil.icon.BoxedValueIcon">
<property          name="attributeName"        class="ptolemy.kernel.util.StringAttribute"
value="value">
</property>
<property name="displayWidth" class="ptolemy.data.expr.Parameter" value="40">
</property>
</property>
<property name="_location" class="ptolemy.kernel.util.Location" value="(50, 280)">
</property>
</entity>
<entity name="Constant2" class="ptolemy.actor.lib.Const">
<property name="firingCountLimit" class="ptolemy.data.expr.Parameter" value="NONE">
</property>
<property name="NONE" class="ptolemy.data.expr.Parameter" value="0">
</property>
<property name="value" class="ptolemy.data.expr.Parameter" value="6">
</property>
<property          name="KeplerDocumentation"
class="ptolemy.vergil.basic.KeplerDocumentationAttribute">
<property          name="description"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>null</configure></property>

```

```

>
<property name="author"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>null</configure></property>
>

<property name="version"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>null</configure></property>
>

<property name="userLevelDocumentation"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>&lt;p&gt;The Constant
actor outputs a constant, which is specified by the value parameter. By default, the actor
outputs the integer 1.&lt;/p&gt;&#10;&#10;&lt;p&gt;The actor can be used to output other
types of values, e.g., a string (such as &quot;Hello&quot;) or a double (such as 1.2). The
actor's output type matches the type of the specified
value.&lt;/p&gt;&#10;&#10;&lt;p&gt;NOTE: If using a PN Director, the
'firingCountLimit' parameter is often set to a finite integer (e.g. '1') so that the workflow will
terminate. &lt;/p&gt;&#10;</configure></property>

<property name="port:output"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>An output port that
broadcasts the specified constant. By default, the output is 1.</configure></property>

<property name="port:trigger"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>A multiport that has no
declared type (in other words, the port can accept any data type: double, int, array, etc.) If
the port is connected, the actor will not fire until the trigger port receives an input token.
Connecting the port is optional, but useful when scheduling the actor to perform at a certain
time.</configure></property>

<property name="prop:firingCountLimit"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>The limit on the number of
times the actor will fire. The default value is 'NONE', meaning there is no limit on the
number of time the constant will be provided to the output port. Any integer can be
provided as a value for this parameter.</configure></property>

<property name="prop:value"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>The value produced by the
Constant actor. By default, the value is the integer token 1. The value can be set to another
type, e.g., a string (such as &quot;Hello&quot;) or a double (such as 1.2). The output type
matches the type of the value specified here.</configure></property>

</property> <property name="entityId" class="org.kepler.moml.NamedObjId"
value="urn:lsid:kepler-project.org:actor:1:1">
</property>

<property name="class" class="ptolemy.kernel.util.StringAttribute"
value="ptolemy.actor.lib.Const">
<property name="id" class="ptolemy.kernel.util.StringAttribute" value="urn:lsid:kepler-
project.org:class:877:1">
</property>
</property>

<property name="semanticType00" class="org.kepler.sms.SemanticType"
value="urn:lsid:localhost:onto:1:1#ConstantActor">
</property>

<property name="semanticType11" class="org.kepler.sms.SemanticType"
value="urn:lsid:localhost:onto:2:1#Constant">

```

```

</property>
<property name="kar" class="ptolemy.kernel.util.StringAttribute" value="urn:lsid:kepler-
project.org:kar:57:1">
</property>
<property name="_icon" class="ptolemy.vergil.icon.BoxedValueIcon">
<property      name="attributeName"      class="ptolemy.kernel.util.StringAttribute"
value="value">
</property>
<property name="displayWidth" class="ptolemy.data.expr.Parameter" value="40">
</property>
</property>
<property name="_location" class="ptolemy.kernel.util.Location" value="(105, 345)">
</property>
</entity>
<entity name="Constant3" class="ptolemy.actor.lib.Const">
<property name="firingCountLimit" class="ptolemy.data.expr.Parameter" value="NONE">
</property>
<property name="NONE" class="ptolemy.data.expr.Parameter" value="0">
</property>
<property name="value" class="ptolemy.data.expr.Parameter" value="2">
</property>
<property                                name="KeplerDocumentation"
class="ptolemy.vergil.basic.KeplerDocumentationAttribute">
<property                                name="description"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>null</configure></property>
>
<property                                name="author"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>null</configure></property>
>
<property                                name="version"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>null</configure></property>
>
<property                                name="userLevelDocumentation"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>&lt;p&gt;The      Constant
actor outputs a constant, which is specified by the value parameter. By default, the actor
outputs the integer 1.&lt;/p&gt;&#10;&#10;&lt;p&gt;The actor can be used to output other
types of values, e.g., a string (such as &quot;Hello&quot;) or a double (such as 1.2). The
actor'&#10;s      output      type      matches      the      type      of      the      specified
value.&lt;/p&gt;&#10;&#10;&lt;p&gt;NOTE:  If using a PN Director, the
'firingCountLimit' parameter is often set to a finite integer (e.g. '1') so that the workflow will
terminate. &lt;/p&gt;&#10;</configure></property>
<property                                name="port:output"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>An  output  port  that
broadcasts the specified constant. By default, the output is 1.</configure></property>
<property                                name="port:trigger"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>A  multiport  that  has  no

```

declared type (in other words, the port can accept any data type: double, int, array, etc.) If the port is connected, the actor will not fire until the trigger port receives an input token. Connecting the port is optional, but useful when scheduling the actor to perform at a certain time.</configure></property>

<property name="prop:firingCountLimit" class="ptolemy.kernel.util.ConfigurableAttribute"><configure>The limit on the number of times the actor will fire. The default value is 'NONE', meaning there is no limit on the number of time the constant will be provided to the output port. Any integer can be provided as a value for this parameter.</configure></property>

<property name="prop:value" class="ptolemy.kernel.util.ConfigurableAttribute"><configure>The value produced by the Constant actor. By default, the value is the integer token 1. The value can be set to another type, e.g., a string (such as "Hello") or a double (such as 1.2). The output type matches the type of the value specified here.</configure></property>

</property> <property name="entityId" class="org.kepler.moml.NamedObjId" value="urn:lsid:kepler-project.org:actor:1:1">

</property>

<property name="class" class="ptolemy.kernel.util.StringAttribute" value="ptolemy.actor.lib.Const">

<property name="id" class="ptolemy.kernel.util.StringAttribute" value="urn:lsid:kepler-project.org:class:877:1">

</property>

</property>

<property name="semanticType00" class="org.kepler.sms.SemanticType" value="urn:lsid:localhost:onto:1:1#ConstantActor">

</property>

<property name="semanticType11" class="org.kepler.sms.SemanticType" value="urn:lsid:localhost:onto:2:1#Constant">

</property>

<property name="kar" class="ptolemy.kernel.util.StringAttribute" value="urn:lsid:kepler-project.org:kar:57:1">

</property>

<property name="_icon" class="ptolemy.vergil.icon.BoxedValueIcon">

<property name="attributeName" class="ptolemy.kernel.util.StringAttribute" value="value">

</property>

<property name="displayWidth" class="ptolemy.data.expr.Parameter" value="40">

</property>

</property>

<property name="_location" class="ptolemy.kernel.util.Location" value="(265, 395)">

</property>

</entity>

<entity name="Constant4" class="ptolemy.actor.lib.Const">

<property name="firingCountLimit" class="ptolemy.data.expr.Parameter" value="NONE">

</property>


```

<property name="NONE" class="ptolemy.data.expr.Parameter" value="0">
</property>
<property name="value" class="ptolemy.data.expr.Parameter" value="6">
</property>
<property name="KeplerDocumentation"
class="ptolemy.vergil.basic.KeplerDocumentationAttribute">
<property name="description"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>null</configure></property>
>
<property name="author"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>null</configure></property>
>
<property name="version"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>null</configure></property>
>
<property name="userLevelDocumentation"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>&lt;p&gt;The Constant
actor outputs a constant, which is specified by the value parameter. By default, the actor
outputs the integer 1.&lt;p&gt;&#10;&#10;&lt;p&gt;The actor can be used to output other
types of values, e.g., a string (such as &quot;Hello&quot;) or a double (such as 1.2). The
actor's output type matches the type of the specified
value.&lt;p&gt;&#10;&#10;&lt;p&gt;NOTE: If using a PN Director, the
'firingCountLimit' parameter is often set to a finite integer (e.g. '1') so that the workflow will
terminate. &lt;p&gt;&#10;</configure></property>
<property name="port:output"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>An output port that
broadcasts the specified constant. By default, the output is 1.</configure></property>
<property name="port:trigger"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>A multiport that has no
declared type (in other words, the port can accept any data type: double, int, array, etc.) If
the port is connected, the actor will not fire until the trigger port receives an input token.
Connecting the port is optional, but useful when scheduling the actor to perform at a certain
time.</configure></property>
<property name="prop:firingCountLimit"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>The limit on the number of
times the actor will fire. The default value is 'NONE', meaning there is no limit on the
number of time the constant will be provided to the output port. Any integer can be
provided as a value for this parameter.</configure></property>
<property name="prop:value"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>The value produced by the
Constant actor. By default, the value is the integer token 1. The value can be set to another
type, e.g., a string (such as &quot;Hello&quot;) or a double (such as 1.2). The output type
matches the type of the value specified here.</configure></property>
</property> <property name="entityId" class="org.kepler.moml.NamedObjId"
value="urn:lsid:kepler-project.org:actor:1:1">
</property>
<property name="class" class="ptolemy.kernel.util.StringAttribute"
value="ptolemy.actor.lib.Const">
<property name="id" class="ptolemy.kernel.util.StringAttribute" value="urn:lsid:kepler-
project.org:class:877:1">

```

```

</property>
</property>
<property      name="semanticType00"      class="org.kepler.sms.SemanticType"
value="urn:lsid:localhost:onto:1:1#ConstantActor">
</property>
<property      name="semanticType11"      class="org.kepler.sms.SemanticType"
value="urn:lsid:localhost:onto:2:1#Constant">
</property>
<property name="kar" class="ptolemy.kernel.util.StringAttribute" value="urn:lsid:kepler-
project.org:kar:57:1">
</property>
<property name="_icon" class="ptolemy.vergil.icon.BoxedValueIcon">
<property      name="attributeName"      class="ptolemy.kernel.util.StringAttribute"
value="value">
</property>
<property name="displayWidth" class="ptolemy.data.expr.Parameter" value="40">
</property>
</property>
<property name="_location" class="ptolemy.kernel.util.Location" value="(285, 440)">
</property>
</entity>
<entity name="Constant5" class="ptolemy.actor.lib.Const">
<property name="firingCountLimit" class="ptolemy.data.expr.Parameter" value="NONE">
</property>
<property name="NONE" class="ptolemy.data.expr.Parameter" value="0">
</property>
<property name="value" class="ptolemy.data.expr.Parameter" value="1.5">
</property>
<property      name="KeplerDocumentation"
class="ptolemy.vergil.basic.KeplerDocumentationAttribute">
<property      name="description"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>null</configure></property
>
<property      name="author"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>null</configure></property
>
<property      name="version"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>null</configure></property
>
<property      name="userLevelDocumentation"
class="ptolemy.kernel.util.ConfigurableAttribute"><configure>&lt;p&gt;The      Constant
actor outputs a constant, which is specified by the value parameter. By default, the actor
outputs the integer 1.&lt;/p&gt;&#10;&#10;&lt;p&gt;The actor can be used to output other
types of values, e.g., a string (such as &quot;Hello&quot;) or a double (such as 1.2). The

```

actor's output type matches the type of the specified value.</p><p>NOTE: If using a PN Director, the 'firingCountLimit' parameter is often set to a finite integer (e.g. '1') so that the workflow will terminate. </p></configure></property>

<property name="port:output" class="ptolemy.kernel.util.ConfigurableAttribute"><configure>An output port that broadcasts the specified constant. By default, the output is 1.</configure></property>

<property name="port:trigger" class="ptolemy.kernel.util.ConfigurableAttribute"><configure>A multiport that has no declared type (in other words, the port can accept any data type: double, int, array, etc.) If the port is connected, the actor will not fire until the trigger port receives an input token. Connecting the port is optional, but useful when scheduling the actor to perform at a certain time.</configure></property>

<property name="prop:firingCountLimit" class="ptolemy.kernel.util.ConfigurableAttribute"><configure>The limit on the number of times the actor will fire. The default value is 'NONE', meaning there is no limit on the number of time the constant will be provided to the output port. Any integer can be provided as a value for this parameter.</configure></property>

<property name="prop:value" class="ptolemy.kernel.util.ConfigurableAttribute"><configure>The value produced by the Constant actor. By default, the value is the integer token 1. The value can be set to another type, e.g., a string (such as 'Hello') or a double (such as 1.2). The output type matches the type of the value specified here.</configure></property>

</property> <property name="entityId" class="org.kepler.moml.NamedObjId" value="urn:lsid:kepler-project.org:actor:1:1">

</property>

<property name="class" class="ptolemy.kernel.util.StringAttribute" value="ptolemy.actor.lib.Const">

<property name="id" class="ptolemy.kernel.util.StringAttribute" value="urn:lsid:kepler-project.org:class:877:1">

</property>

</property>

<property name="semanticType00" class="org.kepler.sms.SemanticType" value="urn:lsid:localhost:onto:1:1#ConstantActor">

</property>

<property name="semanticType11" class="org.kepler.sms.SemanticType" value="urn:lsid:localhost:onto:2:1#Constant">

</property>

<property name="kar" class="ptolemy.kernel.util.StringAttribute" value="urn:lsid:kepler-project.org:kar:57:1">

</property>

<property name="_icon" class="ptolemy.vergil.icon.BoxedValueIcon">

<property name="attributeName" class="ptolemy.kernel.util.StringAttribute" value="value">

</property>

<property name="displayWidth" class="ptolemy.data.expr.Parameter" value="40">

</property>

```

</property>
<property name="_location" class="ptolemy.kernel.util.Location" value="">
</property>
</entity>
<relation name="relation" class="ptolemy.actor.TypedIORelation">
<property name="width" class="ptolemy.data.expr.Parameter" value="1">
</property>
</relation>
<relation name="relation2" class="ptolemy.actor.TypedIORelation">
<property name="width" class="ptolemy.data.expr.Parameter" value="1">
</property>
</relation>
<relation name="relation3" class="ptolemy.actor.TypedIORelation">
<property name="width" class="ptolemy.data.expr.Parameter" value="1">
</property>
</relation>
<relation name="relation4" class="ptolemy.actor.TypedIORelation">
<property name="width" class="ptolemy.data.expr.Parameter" value="1">
</property>
</relation>
<relation name="relation5" class="ptolemy.actor.TypedIORelation">
<property name="width" class="ptolemy.data.expr.Parameter" value="1">
</property>
</relation>
<relation name="relation6" class="ptolemy.actor.TypedIORelation">
<property name="width" class="ptolemy.data.expr.Parameter" value="1">
</property>
</relation>
<relation name="relation7" class="ptolemy.actor.TypedIORelation">
<property name="width" class="ptolemy.data.expr.Parameter" value="1">
</property>
</relation>
<relation name="relation8" class="ptolemy.actor.TypedIORelation">
<property name="width" class="ptolemy.data.expr.Parameter" value="1">
</property>
</relation>
<link port="MincDefrag.ClassifyFile" relation="relation"/>
<link port="MincDefrag.value" relation="relation2"/>
<link port="MincDefrag.value2" relation="relation3"/>

```

<link port="MincDefrag.output" relation="relation4"/>
<link port="MincDefrag2.ClassifyFile" relation="relation4"/>
<link port="MincDefrag2.value" relation="relation5"/>
<link port="MincDefrag2.value2" relation="relation6"/>
<link port="MincDefrag2.output" relation="relation7"/>
<link port="CorticalSurface.ClassifyFile" relation="relation7"/>
<link port="CorticalSurface.value" relation="relation8"/>
<link port="String Constant.output" relation="relation"/>
<link port="Constant.output" relation="relation2"/>
<link port="Constant2.output" relation="relation3"/>
<link port="Constant3.output" relation="relation5"/>
<link port="Constant4.output" relation="relation6"/>
<link port="Constant5.output" relation="relation8"/>
</entity>

Appendix C

LONI Pipeline Workflow

```
<?xml version="1.0" encoding="UTF-8"?>
<pipeline version=".1">
  <connections>
    <connection source=".Output_1" sink="MINCDefrag.Newparameter1_0" />
    <connection source=".Output_0" sink="MINCDefrag.file_0" />
    <connection source=".Output_2" sink="MINCDefrag.Newparameter2_0" />
    <connection source=".Output_1" sink="MINCDefrag.Constant_0" />
    <connection source=".Output_3" sink="MINCDefrag.Newparameter1_1" />
    <connection source="MINCDefrag.output2_0" sink="MINCDefrag.file_1" />
    <connection source=".Output_4" sink="CorticalSurface.Constant_0" />
    <connection
      source="MINCDefrag.output2_1"
      sink="CorticalSurface.Newparameter1_0" />
    <connection source="CorticalSurface.output2_0" sink=".Input_0" />
  </connections>
  <moduleGroup name="Untitled" description="" id="Untitled_0" posX="0"
    posY="0" rotation="0">
    <authors />
    <citations />
    <module
      name="MINC Defrag" description=""
      location="pipeline://localhost//Users/irfan/Desktop/FairGame.app"
      id="MINCDefrag_0" posX="103" posY="295" rotation="0">
      <authors />
      <executableAuthors />
      <citations />
      <tag />
      <metadata>
        <data key="__creationDateKey" value="Mon Dec 29 11:18:38 GMT 2008" />
      </metadata>
      <input name="New parameter 1" id="MINCDefrag.Newparameter1_0"
        enabled="true" required="true" order="0">
        <format type="File" cardinality="1">
          <fileTypes>
            <filetype name="File" extension="" description="Any type of data file" />
          </fileTypes>
```

```

</format>
</input>
<input name="file" id="MINCDefrag.file_0" enabled="true" required="true"
order="1">
<format type="File" cardinality="1">
<fileTypes>
<filetype name="File" extension="" description="Any type of data file" />
</fileTypes>
</format>
</input>
<output name="output2" id="MINCDefrag.output2_0" enabled="true"
required="true" order="2">
<format type="File" cardinality="1">
<fileTypes>
<filetype name="File" extension="" description="Any type of data file" />
</fileTypes>
</format>
</output>
<input name="New parameter 2" id="MINCDefrag.Newparameter2_0"
enabled="true" required="true" order="3">
<format type="String" cardinality="1" />
</input>
</module>
<module name="MINC Defrag" description=""
location="pipeline://localhost/Users/irfan/Desktop/FairGame.app"
id="MINCDefrag_1" posX="248" posY="431" rotation="0">
<authors />
<executableAuthors />
<citations />
<tag />
<metadata>
<data key="__creationDateKey" value="Mon Dec 29 11:18:38 GMT 2008" />
</metadata>
<input name="Constant" id="MINCDefrag.Constant_0" enabled="true"
required="true" order="0">
<format type="String" cardinality="1" />
</input>
<input name="New parameter 1" id="MINCDefrag.Newparameter1_1"
enabled="true" required="true" order="1">

```

```

<format type="String" cardinality="1" />
</input>
<input name="file" id="MINCDefrag.file_1" enabled="true" required="true"
order="2">
<format type="File" cardinality="1">
<fileTypes>
<filetype name="File" extension="" description="Any type of data file" />
</fileTypes>
</format>
</input>
<output name="output2" id="MINCDefrag.output2_1" enabled="true"
required="true" order="3">
<format type="File" cardinality="1">
<fileTypes>
<filetype name="File" extension="" description="Any type of data file" />
</fileTypes>
</format>
</output>
</module>
<module name="Cortical Surface" description=""
location="pipeline://localhost//Users/irfan/Desktop/FairGame.app"
id="CorticalSurface_0" posX="390" posY="584" rotation="0">
<authors />
<executableAuthors />
<citations />
<tag />
<metadata>
<data key="__creationDateKey" value="Mon Dec 29 11:18:38 GMT 2008" />
</metadata>
<input name="Constant" id="CorticalSurface.Constant_0" enabled="true"
required="true" order="0">
<format type="String" cardinality="1" />
</input>
<input name="New parameter 1" id="CorticalSurface.Newparameter1_0"
enabled="true" required="true" order="1">
<format type="File" cardinality="1">
<fileTypes>
<filetype name="File" extension="" description="Any type of data file" />
</fileTypes>

```



```

</format>
</input>
<output name="output2" id="CorticalSurface.output2_0" enabled="true"
required="true" order="2">
<format type="File" cardinality="1">
<fileTypes>
<filetype name="File" extension="" description="Any type of data file" />
</fileTypes>
</format>
</output>
</module>
<dataModule name="" description="" id="_0" posX="94" posY="197" rotation="0"
dirDump="false" type="File" source="true">
<authors />
<citations />
<tag />
<fileTypes>
<filetype name="File" extension="" description="Any type of data file" />
</fileTypes>
<metadata>
<data key="__creationDateKey" value="Mon Dec 29 11:25:40 GMT 2008" />
</metadata>
<output name="Output" id=".Output_0" enabled="true" required="true" order="-
1">
<format type="File" cardinality="1">
<fileTypes>
<filetype name="File" extension="" description="Any type of data file" />
</fileTypes>
</format>
</output>
</dataModule>
<dataModule name="" description="" id="_1" posX="21" posY="380" rotation="0"
dirDump="false" type="String" source="true">
<authors />
<citations />
<tag />
<metadata>
<data key="__creationDateKey" value="Mon Dec 29 11:28:30 GMT 2008" />

```

```
</metadata>
<output name="Output" id=".Output_1" enabled="true" required="true" order="-1">
<format type="String" cardinality="1" />
</output>
</dataModule>
<dataModule name="" description="" id="_2" posX="158" posY="204"
rotation="0" dirDump="false" type="String" source="true">
<authors />
<citations />
<tag />
<metadata>
<data key="__creationDateKey" value="Mon Dec 29 11:29:14 GMT 2008" />
</metadata>
<output name="Output" id=".Output_2" enabled="true" required="true" order="-1">
<format type="String" cardinality="1" />
</output>
</dataModule>
<dataModule name="" description="" id="_3" posX="338" posY="305"
rotation="0" dirDump="false" type="String" source="true">
<authors />
<citations />
<tag />
<metadata>
<data key="__creationDateKey" value="Mon Dec 29 11:32:06 GMT 2008" />
</metadata>
<output name="Output" id=".Output_3" enabled="true" required="true" order="-1">
<format type="String" cardinality="1" />
</output>
</dataModule>
<dataModule name="" description="" id="_4" posX="298" posY="592"
rotation="0" dirDump="false" type="String" source="true">
<authors />
<citations />
<tag />
<metadata>
<data key="__creationDateKey" value="Mon Dec 29 11:32:21 GMT 2008" />
```

```
</metadata>
<output name="Output" id=".Output_4" enabled="true" required="true" order="-1">
<format type="String" cardinality="1" />
</output>
</dataModule>
<dataModule name="" description="" id="_5" posX="403" posY="705"
rotation="0" dirDump="false" type="File" source="false">
<authors />
<citations />
<tag />
<fileTypes>
<filetype name="File" extension="" description="Any type of data file" />
</fileTypes>
<metadata>
<data key="__creationDateKey" value="Mon Dec 29 11:32:39 GMT 2008" />
</metadata>
<input name="Input" id=".Input_0" enabled="true" required="true" order="-1">
<format type="File" cardinality="1">
<fileTypes>
<filetype name="File" extension="" description="Any type of data file" />
</fileTypes>
</format>
</input>
</dataModule>
<variables>
<variable name="Var 2">Value</variable>
<variable name="Var 1">Value</variable>
</variables>
</moduleGroup>
</pipeline>
```

Appendix D

WSDL of the Glueing Service Interface

A sample WSDL of the glueing service is as follows:

```
<wsdl:definitions targetNamespace="https://localhost:18443/axis/saga/saga-impl-1.0rc2/GlueingService.jws">

  <!--
  WSDL created by Apache Axis version: 1.2.1-itinnov-1
  Built on Nov 09, 2007 (12:33:49 GMT)
  -->

  <wsdl:types>

    <schema targetNamespace="https://localhost:18443/axis/saga/saga-impl-1.0rc2/GlueingService.jws">
      <import namespace="http://monitoring.saga.ogf.org"/>
      <import namespace="http://job.saga.ogf.org"/>
      <import namespace="http://buffer.saga.ogf.org"/>
      <import namespace="http://task.saga.ogf.org"/>
      <import namespace="http://url.saga.ogf.org"/>
      <import namespace="http://file.saga.ogf.org"/>
      <import namespace="http://logicalfile.saga.ogf.org"/>
      <import namespace="http://io.java"/>
      <import namespace="http://schemas.xmlsoap.org/soap/encoding"/>

      <complexType name="ArrayOf_xsd_string">

        <complexContent>

          <restriction base="soapenc:Array">
            <attribute ref="soapenc:arrayType" wsdl:arrayType="xsd:string[]"/>
          </restriction>
        </complexContent>
      </complexType>
    </schema>
  </wsdl:types>
```

```
<wsdl:message name="getJobDescriptionRequest">
  <wsdl:part name="in0" type="tns4:Job"/>
</wsdl:message>
```

```
<wsdl:message name="openDirectoryRequest">
  <wsdl:part name="in0" type="tns3:Directory"/>
  <wsdl:part name="in1" type="tns2:URL"/>
</wsdl:message>
<wsdl:message name="addLocationResponse">
```

```
</wsdl:message>
```

```
<wsdl:message name="getStdoutRequest">
  <wsdl:part name="in0" type="tns4:Job"/>
</wsdl:message>
<wsdl:message name="updateLocationResponse">
```

```
</wsdl:message>
<wsdl:message name="removeLocationResponse">
</wsdl:message>
<wsdl:message name="runResponse">
```

```
</wsdl:message>
```

```
<wsdl:message name="replicateRequest">
  <wsdl:part name="in0" type="tns1:LogicalFile"/>
  <wsdl:part name="in1" type="tns2:URL"/>
</wsdl:message>
```

```
<wsdl:message name="createJobServiceResponse">
  <wsdl:part name="createJobServiceReturn" type="xsd:anyType"/>
</wsdl:message>
```

```
<wsdl:message name="openFileResponse">
  <wsdl:part name="openFileReturn" type="xsd:anyType"/>
</wsdl:message>
<wsdl:message name="migrateResponse">
```

```
</wsdl:message>
```

```
<wsdl:message name="getJobDescriptionResponse">
<wsdl:part name="getJobDescriptionReturn" type="tns4:JobDescription"/>
</wsdl:message>
<wsdl:message name="setAttributeResponse">

</wsdl:message>
<wsdl:message name="monitorResponse">

</wsdl:message>

<wsdl:message name="getIdRequest">
<wsdl:part name="in0" type="tns4:Job"/>
</wsdl:message>

<wsdl:message name="getIdResponse">
<wsdl:part name="getIdReturn" type="xsd:string"/>
</wsdl:message>
<wsdl:message name="replicateResponse">

</wsdl:message>

<wsdl:message name="createJobResponse">
<wsdl:part name="createJobReturn" type="xsd:anyType"/>
</wsdl:message>

<wsdl:message name="writeResponse">
<wsdl:part name="writeReturn" type="xsd:int"/>
</wsdl:message>

<wsdl:message name="listMetricsResponse">
<wsdl:part name="listMetricsReturn" type="xsd:string"/>
</wsdl:message>

<wsdl:message name="openFileRequest">
<wsdl:part name="in0" type="xsd:anyType"/>
<wsdl:part name="in1" type="tns2:URL"/>
</wsdl:message>

<wsdl:message name="openDirectoryResponse">
<wsdl:part name="openDirectoryReturn" type="tns3:Directory"/>
```

</wsdl:message>

<wsdl:message name="createJobDescriptionRequest">

<wsdl:part name="in0" type="tns2:URL"/>

</wsdl:message>

<wsdl:message name="migrateRequest">

<wsdl:part name="in0" type="tns4:Job"/>

<wsdl:part name="in1" type="tns4:JobDescription"/>

</wsdl:message>

<wsdl:message name="writeRequest">

<wsdl:part name="in0" type="tns3:File"/>

<wsdl:part name="in1" type="tns7:Buffer"/>

</wsdl:message>

<wsdl:message name="getStdoutResponse">

<wsdl:part name="getStdoutReturn" type="xsd:anyType"/>

</wsdl:message>

<wsdl:message name="readResponse">

<wsdl:part name="readReturn" type="xsd:int"/>

</wsdl:message>

<wsdl:message name="readPResponse">

<wsdl:part name="readPReturn" type="xsd:int"/>

</wsdl:message>

<wsdl:message name="readPRequest">

<wsdl:part name="in0" type="tns3:File"/>

<wsdl:part name="in1" type="xsd:string"/>

<wsdl:part name="in2" type="xsd:anyType"/>

</wsdl:message>

<wsdl:message name="removeLocationRequest">

<wsdl:part name="in0" type="tns1:LogicalFile"/>

<wsdl:part name="in1" type="tns2:URL"/>

</wsdl:message>

<wsdl:message name="listMetricsRequest">

```
<wsdl:part name="in0" type="tns4:Job"/>
</wsdl:message>
```

```
<wsdl:message name="updateLocationRequest">
<wsdl:part name="in0" type="tns1:LogicalFile"/>
<wsdl:part name="in1" type="tns2:URL"/>
<wsdl:part name="in2" type="tns2:URL"/>
</wsdl:message>
```

```
<wsdl:message name="monitorRequest">
<wsdl:part name="in0" type="xsd:string"/>
</wsdl:message>
<wsdl:message name="setVectorAttributeResponse">
```

```
</wsdl:message>
```

```
<wsdl:message name="runRequest">
<wsdl:part name="in0" type="tns4:Job"/>
</wsdl:message>
<wsdl:message name="createJobDescriptionResponse">
<wsdl:part name="createJobDescriptionReturn"
type="tns4:JobDescription"/>
</wsdl:message>
```

```
<wsdl:message name="getMetricResponse">
<wsdl:part name="getMetricReturn" type="xsd:anyType"/>
</wsdl:message>
<wsdl:message name="suspendResponse">
```

```
</wsdl:message>
```

```
<wsdl:message name="getStateRequest">
<wsdl:part name="in0" type="tns4:Job"/>
</wsdl:message>
```

```
<wsdl:message name="setVectorAttributeRequest">
<wsdl:part name="in0" type="xsd:anyType"/>
<wsdl:part name="in1" type="xsd:string"/>
<wsdl:part name="in2" type="impl:ArrayOf_xsd_string"/>
</wsdl:message>
```



```

<wsdl:message name="readRequest">
  <wsdl:part name="in0" type="tns3:File"/>
  <wsdl:part name="in1" type="tns7:Buffer"/>
</wsdl:message>

<wsdl:message name="getStateResponse">
  <wsdl:part name="getStateReturn" type="xsd:anyType"/>
</wsdl:message>

<wsdl:message name="addLocationRequest">
  <wsdl:part name="in0" type="xsd:anyType"/>
  <wsdl:part name="in1" type="xsd:anyType"/>
</wsdl:message>

<wsdl:message name="setAttributeRequest">
  <wsdl:part name="in0" type="tns4:JobDescription"/>
  <wsdl:part name="in1" type="xsd:string"/>
  <wsdl:part name="in2" type="xsd:string"/>
</wsdl:message>
<wsdl:message name="createJobServiceRequest">

</wsdl:message>

<wsdl:message name="createJobRequest">
  <wsdl:part name="in0" type="tns4:JobService"/>
  <wsdl:part name="in1" type="tns4:JobDescription"/>
</wsdl:message>

<wsdl:message name="getMetricRequest">
  <wsdl:part name="in0" type="tns4:Job"/>
  <wsdl:part name="in1" type="xsd:string"/>
</wsdl:message>

<wsdl:message name="suspendRequest">
  <wsdl:part name="in0" type="tns4:Job"/>
</wsdl:message>

<wsdl:portType name="GlueingService">
  <wsdl:operation name="monitor" parameterOrder="in0">

```

```
<wsdl:input message="impl:monitorRequest" name="monitorRequest"/>
<wsdl:output message="impl:monitorResponse" name="monitorResponse"/>
</wsdl:operation>
```

```
<wsdl:operation name="addLocation" parameterOrder="in0 in1">
<wsdl:input message="impl:addLocationRequest"
name="addLocationRequest"/>
<wsdl:output message="impl:addLocationResponse"
name="addLocationResponse"/>
</wsdl:operation>
```

```
<wsdl:operation name="replicate" parameterOrder="in0 in1">
<wsdl:input message="impl:replicateRequest" name="replicateRequest"/>
<wsdl:output message="impl:replicateResponse"
name="replicateResponse"/>
</wsdl:operation>
```

```
<wsdl:operation name="openFile" parameterOrder="in0 in1">
<wsdl:input message="impl:openFileRequest" name="openFileRequest"/>
<wsdl:output message="impl:openFileResponse"
name="openFileResponse"/>
</wsdl:operation>
```

```
<wsdl:operation name="createJobService">
<wsdl:input message="impl:createJobServiceRequest"
name="createJobServiceRequest"/>
<wsdl:output message="impl:createJobServiceResponse"
name="createJobServiceResponse"/>
</wsdl:operation>
```

```
<wsdl:operation name="setVectorAttribute" parameterOrder="in0 in1 in2">
<wsdl:input message="impl:setVectorAttributeRequest"
name="setVectorAttributeRequest"/>
<wsdl:output message="impl:setVectorAttributeResponse"
name="setVectorAttributeResponse"/>
</wsdl:operation>
```

```
<wsdl:operation name="createJob" parameterOrder="in0 in1">
<wsdl:input message="impl:createJobRequest" name="createJobRequest"/>
<wsdl:output message="impl:createJobResponse"
```

```
name="createJobResponse"/>
</wsdl:operation>
```

```
<wsdl:operation name="getJobDescription" parameterOrder="in0">
<wsdl:input message="impl:getJobDescriptionRequest"
name="getJobDescriptionRequest"/>
<wsdl:output message="impl:getJobDescriptionResponse"
name="getJobDescriptionResponse"/>
</wsdl:operation>
```

```
<wsdl:operation name="getStdout" parameterOrder="in0">
<wsdl:input message="impl:getStdoutRequest" name="getStdoutRequest"/
>
<wsdl:output message="impl:getStdoutResponse"
name="getStdoutResponse"/>
</wsdl:operation>
```

```
<wsdl:operation name="migrate" parameterOrder="in0 in1">
<wsdl:input message="impl:migrateRequest" name="migrateRequest"/>
<wsdl:output message="impl:migrateResponse" name="migrateResponse"/
>
</wsdl:operation>
```

```
<wsdl:operation name="listMetrics" parameterOrder="in0">
<wsdl:input message="impl:listMetricsRequest"
name="listMetricsRequest"/>
<wsdl:output message="impl:listMetricsResponse"
name="listMetricsResponse"/>
</wsdl:operation>
```

```
<wsdl:operation name="getMetric" parameterOrder="in0 in1">
<wsdl:input message="impl:getMetricRequest" name="getMetricRequest"/
>
<wsdl:output message="impl:getMetricResponse"
name="getMetricResponse"/>
</wsdl:operation>
```

```
<wsdl:operation name="createJobDescription" parameterOrder="in0">
<wsdl:input message="impl:createJobDescriptionRequest"
name="createJobDescriptionRequest"/>
<wsdl:output message="impl:createJobDescriptionResponse"
```

```
name="createJobDescriptionResponse"/>
</wsdl:operation>
```

```
<wsdl:operation name="openDirectory" parameterOrder="in0 in1">
<wsdl:input message="impl:openDirectoryRequest"
name="openDirectoryRequest"/>
<wsdl:output message="impl:openDirectoryResponse"
name="openDirectoryResponse"/>
</wsdl:operation>
```

```
<wsdl:operation name="readP" parameterOrder="in0 in1 in2">
<wsdl:input message="impl:readPRequest" name="readPRequest"/>
<wsdl:output message="impl:readPResponse" name="readPResponse"/>
</wsdl:operation>
```

```
<wsdl:operation name="removeLocation" parameterOrder="in0 in1">
<wsdl:input message="impl:removeLocationRequest"
name="removeLocationRequest"/>
<wsdl:output message="impl:removeLocationResponse"
name="removeLocationResponse"/>
</wsdl:operation>
```

```
<wsdl:operation name="run" parameterOrder="in0">
<wsdl:input message="impl:runRequest" name="runRequest"/>
<wsdl:output message="impl:runResponse" name="runResponse"/>
</wsdl:operation>
```

```
<wsdl:operation name="write" parameterOrder="in0 in1">
<wsdl:input message="impl:writeRequest" name="writeRequest"/>
<wsdl:output message="impl:writeResponse" name="writeResponse"/>
</wsdl:operation>
```

```
<wsdl:operation name="getId" parameterOrder="in0">
<wsdl:input message="impl:getIdRequest" name="getIdRequest"/>
<wsdl:output message="impl:getIdResponse" name="getIdResponse"/>
</wsdl:operation>
```

```
<wsdl:operation name="getState" parameterOrder="in0">
<wsdl:input message="impl:getStateRequest" name="getStateRequest"/>
<wsdl:output message="impl:getStateResponse" name="getStateResponse"/>
>
```

</wsdl:operation>

<wsdl:operation name="suspend" parameterOrder="in0">
<wsdl:input message="impl:suspendRequest" name="suspendRequest"/>
<wsdl:output message="impl:suspendResponse" name="suspendResponse"/>
>

</wsdl:operation>

<wsdl:operation name="read" parameterOrder="in0 in1">
<wsdl:input message="impl:readRequest" name="readRequest"/>
<wsdl:output message="impl:readResponse" name="readResponse"/>
</wsdl:operation>

<wsdl:operation name="updateLocation" parameterOrder="in0 in1 in2">
<wsdl:input message="impl:updateLocationRequest" name="updateLocationRequest"/>
<wsdl:output message="impl:updateLocationResponse" name="updateLocationResponse"/>
</wsdl:operation>

<wsdl:operation name="setAttribute" parameterOrder="in0 in1 in2">
<wsdl:input message="impl:setAttributeRequest" name="setAttributeRequest"/>
<wsdl:output message="impl:setAttributeResponse" name="setAttributeResponse"/>
</wsdl:operation>
</wsdl:portType>

<wsdl:binding name="GlueingServiceSoapBinding" type="impl:GlueingService">
<wsdlsoap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>

<wsdl:operation name="monitor">
<wsdlsoap:operation soapAction="">

<wsdl:input name="monitorRequest">
<wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" namespace="http://DefaultNamespace" use="encoded"/>
</wsdl:input>

```
<wsdl:output name="monitorResponse">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="https://localhost:18443/axis/saga/saga-impl-
1.0rc2/GlueingService.jws" use="encoded"/>
</wsdl:output>
</wsdl:operation>
```

```
<wsdl:operation name="addLocation">
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="addLocationRequest">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://DefaultNamespace" use="encoded"/>
</wsdl:input>
```

```
<wsdl:output name="addLocationResponse">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="https://localhost:18443/axis/saga/saga-impl-
1.0rc2/GlueingService.jws" use="encoded"/>
</wsdl:output>
</wsdl:operation>
```

```
<wsdl:operation name="replicate">
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="replicateRequest">
<wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://DefaultNamespace" use="encoded"/>
</wsdl:input>
```

```
<wsdl:output name="replicateResponse">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="https://localhost:18443/axis/saga/saga-impl-
1.0rc2/GlueingService.jws" use="encoded"/>
</wsdl:output>
</wsdl:operation>
```

```
<wsdl:operation name="openFile">  
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="openFileRequest">  
<wsdlsoap:body  
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"  
namespace="http://DefaultNamespace" use="encoded"/>  
</wsdl:input>
```

```
<wsdl:output name="openFileResponse">  
<wsdlsoap:body  
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"  
namespace="https://localhost:18443/axis/saga/saga-impl-  
1.0rc2/GlueingService.jws" use="encoded"/>  
</wsdl:output>  
</wsdl:operation>
```

```
<wsdl:operation name="createJobService">  
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="createJobServiceRequest">  
<wsdlsoap:body  
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"  
namespace="http://DefaultNamespace" use="encoded"/>  
</wsdl:input>
```

```
<wsdl:output name="createJobServiceResponse">  
<wsdlsoap:body  
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"  
namespace="https://localhost:18443/axis/saga/saga-impl-  
1.0rc2/GlueingService.jws" use="encoded"/>  
</wsdl:output>  
</wsdl:operation>
```

```
<wsdl:operation name="setVectorAttribute">  
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="setVectorAttributeRequest">  
<wsdlsoap:body  
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"  
namespace="http://DefaultNamespace" use="encoded"/>
```

</wsdl:input>

<wsdl:output name="setVectorAttributeResponse">

<wsdlsoap:body

encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"

namespace="https://localhost:18443/axis/saga/saga-impl-1.0rc2/GlueingService.jws" use="encoded"/>

</wsdl:output>

</wsdl:operation>

<wsdl:operation name="createJob">

<wsdlsoap:operation soapAction=""/>

<wsdl:input name="createJobRequest">

<wsdlsoap:body

encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"

namespace="http://DefaultNamespace" use="encoded"/>

</wsdl:input>

<wsdl:output name="createJobResponse">

<wsdlsoap:body

encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"

namespace="https://localhost:18443/axis/saga/saga-impl-1.0rc2/GlueingService.jws" use="encoded"/>

</wsdl:output>

</wsdl:operation>

<wsdl:operation name="getJobDescription">

<wsdlsoap:operation soapAction=""/>

<wsdl:input name="getJobDescriptionRequest">

<wsdlsoap:body

encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"

namespace="http://DefaultNamespace" use="encoded"/>

</wsdl:input>

<wsdl:output name="getJobDescriptionResponse">

<wsdlsoap:body

encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"

namespace="https://localhost:18443/axis/saga/saga-impl-1.0rc2/GlueingService.jws" use="encoded"/>


```
</wsdl:output>
</wsdl:operation>
```

```
<wsdl:operation name="getStdout">
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="getStdoutRequest">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://DefaultNamespace" use="encoded"/>
</wsdl:input>
```

```
<wsdl:output name="getStdoutResponse">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="https://localhost:18443/axis/saga/saga-impl-
1.0rc2/GlueingService.jws" use="encoded"/>
</wsdl:output>
</wsdl:operation>
```

```
<wsdl:operation name="migrate">
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="migrateRequest">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://DefaultNamespace" use="encoded"/>
</wsdl:input>
```

```
<wsdl:output name="migrateResponse">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="https://localhost:18443/axis/saga/saga-impl-
1.0rc2/GlueingService.jws" use="encoded"/>
</wsdl:output>
</wsdl:operation>
```

```
<wsdl:operation name="listMetrics">
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="listMetricsRequest">
```

```
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://DefaultNamespace" use="encoded"/>
</wsdl:input>
```

```
<wsdl:output name="listMetricsResponse">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="https://localhost:18443/axis/saga/saga-impl-
1.0rc2/GlueingService.jws" use="encoded"/>
</wsdl:output>
</wsdl:operation>
```

```
<wsdl:operation name="getMetric">
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="getMetricRequest">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://DefaultNamespace" use="encoded"/>
</wsdl:input>
```

```
<wsdl:output name="getMetricResponse">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="https://localhost:18443/axis/saga/saga-impl-
1.0rc2/GlueingService.jws" use="encoded"/>
</wsdl:output>
</wsdl:operation>
```

```
<wsdl:operation name="createJobDescription">
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="createJobDescriptionRequest">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://DefaultNamespace" use="encoded"/>
</wsdl:input>
```

```
<wsdl:output name="createJobDescriptionResponse">
<wsdlsoap:body
```

```
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="https://localhost:18443/axis/saga/saga-impl-
1.0rc2/GlueingService.jws" use="encoded"/>
</wsdl:output>
</wsdl:operation>
```

```
<wsdl:operation name="openDirectory">
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="openDirectoryRequest">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://DefaultNamespace" use="encoded"/>
</wsdl:input>
```

```
<wsdl:output name="openDirectoryResponse">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="https://localhost:18443/axis/saga/saga-impl-
1.0rc2/GlueingService.jws" use="encoded"/>
</wsdl:output>
</wsdl:operation>
```

```
<wsdl:operation name="readP">
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="readPRequest">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://DefaultNamespace" use="encoded"/>
</wsdl:input>
```

```
<wsdl:output name="readPResponse">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="https://localhost:18443/axis/saga/saga-impl-
1.0rc2/GlueingService.jws" use="encoded"/>
</wsdl:output>
</wsdl:operation>
```

```
<wsdl:operation name="removeLocation">
```

```
<wsdl:operation soapAction=""/>
```

```
<wsdl:input name="removeLocationRequest">
```

```
<wsdlsoap:body
```

```
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
```

```
namespace="http://DefaultNamespace" use="encoded"/>
```

```
</wsdl:input>
```

```
<wsdl:output name="removeLocationResponse">
```

```
<wsdlsoap:body
```

```
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
```

```
namespace="https://localhost:18443/axis/saga/saga-impl-1.0rc2/GlueingService.jws" use="encoded"/>
```

```
</wsdl:output>
```

```
</wsdl:operation>
```

```
<wsdl:operation name="run">
```

```
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="runRequest">
```

```
<wsdlsoap:body
```

```
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
```

```
namespace="http://DefaultNamespace" use="encoded"/>
```

```
</wsdl:input>
```

```
<wsdl:output name="runResponse">
```

```
<wsdlsoap:body
```

```
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
```

```
namespace="https://localhost:18443/axis/saga/saga-impl-1.0rc2/GlueingService.jws" use="encoded"/>
```

```
</wsdl:output>
```

```
</wsdl:operation>
```

```
<wsdl:operation name="write">
```

```
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="writeRequest">
```

```
<wsdlsoap:body
```

```
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
```

```
namespace="http://DefaultNamespace" use="encoded"/>
```

```
</wsdl:input>
```

```
<wsdl:output name="writeResponse">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="https://localhost:18443/axis/saga/saga-impl-
1.0rc2/GlueingService.jws" use="encoded"/>
</wsdl:output>
</wsdl:operation>
```

```
<wsdl:operation name="getId">
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="getIdRequest">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://DefaultNamespace" use="encoded"/>
</wsdl:input>
```

```
<wsdl:output name="getIdResponse">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="https://localhost:18443/axis/saga/saga-impl-
1.0rc2/GlueingService.jws" use="encoded"/>
</wsdl:output>
</wsdl:operation>
```

```
<wsdl:operation name="getState">
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="getStateRequest">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://DefaultNamespace" use="encoded"/>
</wsdl:input>
```

```
<wsdl:output name="getStateResponse">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="https://localhost:18443/axis/saga/saga-impl-
1.0rc2/GlueingService.jws" use="encoded"/>
</wsdl:output>
```

```
</wsdl:operation>
```

```
<wsdl:operation name="suspend">  
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="suspendRequest">  
<wsdlsoap:body  
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"  
namespace="http://DefaultNamespace" use="encoded"/>  
</wsdl:input>
```

```
<wsdl:output name="suspendResponse">  
<wsdlsoap:body  
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"  
namespace="https://localhost:18443/axis/saga/saga-impl-  
1.0rc2/GlueingService.jws" use="encoded"/>  
</wsdl:output>  
</wsdl:operation>
```

```
<wsdl:operation name="read">  
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="readRequest">  
<wsdlsoap:body  
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"  
namespace="http://DefaultNamespace" use="encoded"/>  
</wsdl:input>
```

```
<wsdl:output name="readResponse">  
<wsdlsoap:body  
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"  
namespace="https://localhost:18443/axis/saga/saga-impl-  
1.0rc2/GlueingService.jws" use="encoded"/>  
</wsdl:output>  
</wsdl:operation>
```

```
<wsdl:operation name="updateLocation">  
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="updateLocationRequest">  
<wsdlsoap:body
```

```
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://DefaultNamespace" use="encoded"/>
</wsdl:input>
```

```
<wsdl:output name="updateLocationResponse">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="https://localhost:18443/axis/saga/saga-impl-
1.0rc2/GlueingService.jws" use="encoded"/>
</wsdl:output>
</wsdl:operation>
```

```
<wsdl:operation name="setAttribute">
<wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input name="setAttributeRequest">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://DefaultNamespace" use="encoded"/>
</wsdl:input>
```

```
<wsdl:output name="setAttributeResponse">
<wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="https://localhost:18443/axis/saga/saga-impl-
1.0rc2/GlueingService.jws" use="encoded"/>
</wsdl:output>
</wsdl:operation>
</wsdl:binding>
```

```
<wsdl:service name="GlueingServiceService">
```

```
<wsdl:port binding="impl:GlueingServiceSoapBinding"
name="GlueingService">
<wsdlsoap:address location="https://localhost:18443/axis/saga/saga-impl-
1.0rc2/GlueingService.jws"/>
</wsdl:port>
</wsdl:service>
</wsdl:definitions>
```

Appendix E Provenance API

Provenance Service Recording Interface

Following is a list of recording API functions with a small description of each

WorkflowID pipelineDescription(WorkflowID id, String description);
//stores the description of a workflow and returns a unique workflow ID to refer it in further API calls.

void versionInfo(WorkflowID id, Version version);
//associate versioning information with a workflow
HeadNodeId <Node> rootNode(WorkflowID id);
//head node of a workflow is logged and its ID is returned
Vector getInputs(WorkflowId id, Node workflowNode);
//returns all the inputs associated with a workflow node/actor

void rootNodeInputs(WorkflowID id, HeadNodeId hn, Vector inputs);

//logs an array of inputs, associated the head node

Process getScript(WorkflowId id, Node workflowNode);

//returns a script associated with a workflow node

Void logScriptCredentials(WorkflowId id, Node workflowNode, Process script);

//logs the script credentials associated with a particular actor in a WF

Boolean actorType (WorkflowId id, Node workflowNode);

//returns true if an actor/node in a WF is composite

Object[] getCompositeEntities(WorkflowId id, Node workflowNode, Boolean flag);

//return all the entities (sub actors) of a composite node.

Void logCompositeEntities(WorkflowId id, Node workflowNode);

// stores all the information related to the entities in a composite node

Successor[] <node> getSuccessor(WorkflowId id, Node workflowNode);

// returns successor nodes of the current workflow node.

Void logSuccessor(WorkflowId id, Node workflowNode, Successor[] successors);

//stores the successors associated with a particular node

Object[] getInputs(WorkflowId id, Node workflowNode);

//returns inputs (local variables, files etc.) to a workflow node


```

Predecessor[] <node> getPredecessor(WorkflowId id, Node workflowNode);

//returns predecessors associated with a particular node

void logInputsPredecessors(WorkflowId id, Node workflowNode, Object[]
inputs, Predecessor[] predecessor);

//stores inputs and predecessors associated with a workflow node

Vector getMetaData(WorkflowId id, Node workflowNode);

//returns the associated metadata with a workflow node

```

Provenance Service Querying Interface

Following is a list of recording API functions with a small description of each

```

ResultSet query(SQL query);

// retrieve a dataset from provenance database, based on the query

Version getVersion(int WfID);

// get the version of a workflow, based on its ID.

Workflow <Document> getWorkflow(int WfID, Version version);

// retrieve complete workflow description from the database, based on the id of
workflow and its version.

ResultSet executionResult(int WfID, Version version);

// retrieve execution results of a particular version of workflow

Boolean validateWf(ResultSet rs, ResultSet refResultSet);

// validate the results of workflow/node with a reference workflow/node

NodeId <int> getNodeId(SQL query);

// get ID of a workflow node, based on appropriate SQL query

Relation <Document> getRelation(NodeId [] ids, int WfID, Version version);

//Retrieve a part of workflow and all the associated inputs and relations

ResultSet executionResults(NodeId id, int WfID, Version version);

// retrieve execution results of a particular workflow node (executing a script)

```


Appendix F LORIS Schema

A schema diagram of LORIS database is shown in the diagram below.

