Workflow and Processing Engineering Report
GEOSS Architecture Implementation Pilot, Phase 2

Version 1.0
Revision History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Editor and Content providers</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>13/July/09</td>
<td>Greg Yetman, Genong Yu, Brian Falk</td>
<td>First Draft</td>
</tr>
<tr>
<td>0.2</td>
<td>30/July/09</td>
<td>Gobe Hobona</td>
<td>General revision of the document</td>
</tr>
<tr>
<td>1.0</td>
<td>31/September/09</td>
<td>Greg Yetman</td>
<td>Final revisions.</td>
</tr>
</tbody>
</table>

Document Contact Information

If you have questions or comments regarding this document, you can contact:

<table>
<thead>
<tr>
<th>Name</th>
<th>Organization</th>
<th>Contact Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Greg Yetman</td>
<td>CIESIN, Columbia University</td>
<td><a href="mailto:gyetman@ciesin.columbia.edu">gyetman@ciesin.columbia.edu</a></td>
</tr>
</tbody>
</table>
Table of Contents

1. Introduction 4
   1.1 Scope of this document 4
   1.2 GEOSS AIP 4

2. Workflow and Processing Objectives 4

3. Construct and Deploy Workflow (Design and Execution) Use Case 4
   3.1 Actors 4
   3.2 Context and pre-conditions 5
   3.3 Scenario Events 6
   3.4 Post-Conditions 6

4. System Model of the Construct and Deploy Use Case 6
   4.1 Service Authoring Diagram 6

5. AIP-2 Implementation of Construct and Deploy Use Case 7
   5.1 BPEL Script 7
   5.2 Unbound WSDL Stub 8
   5.3 Default Input Parameters 8
   5.4 Deployment 8
      5.4.1 Deployment Procedure 8
      5.4.2 Bound WSDL 9
   5.5 Execution 9
      5.5.1 Request 9
      5.5.2 Response and Results 10

6. Publish Workflow Use Case 12
   6.1 Actors 12
   6.2 Context and Preconditions 12
   6.3 Scenario Events 12
   6.4 Post Conditions 12

7. System Model of the Publish Use Case 12
   7.1 Population Statistics WPS 12

8. AIP-2 Implementation of the Publish Use Case 13
   8.1 Publication in the GEOSS Registry 13
   8.2 Population Statistics WPS Capabilities 13

9. Next Steps 14

10. References 15
Workflow Authoring and Publishing Use Cases

1. Introduction

1.1 Scope of this document

This AIP-2 ER describes the workflow and processing group experience in implementing two use cases: authoring workflows and publishing workflows. Extracts from a detailed use case developed for registering workflows within the GEOSS registry are included, and issues related to the limits of the registry related to workflow and processing interoperability are discussed. Technical issues in processing services within GEOSS are presented.

1.2 GEOSS AIP

The GEOSS Architecture Implementation Pilot (AIP) leads the incorporation of contributed components consistent with the GEOSS Architecture using a GEO Web Portal and a Clearinghouse search facility to access services through GEOSS special arrangements in support of the GEOSS Societal Benefit Areas. AIP is a GEO task for elaborating the GEOSS Architecture under the purview of the GEO Architecture and Data Committee.

This Engineering Report (ER) is a key result of the second phase of AIP. AIP-2 was conducted from July 2008 to June 2009. A separate AIP-2 ER describes the overall process and results of AIP-2 and thereby provides a context for this Community SBA ER.¹

2. Workflow and Processing Objectives

Reusable workflows and processing services are intended to support multiple use cases that utilize Earth Observation and related data. Published workflows and processing services permit discovery and access (find and bind) to GEOSS components for reuse in service chaining.

ISO 19119 identifies three design patterns for workflows or service chains:

- user defined chaining in which the human user manages the workflow. This is also known as transparent chaining.
- workflow-managed chaining in which the human user invokes a workflow management service that controls the chain and the user is aware of the individual services. This is also known as translucent chaining.
- aggregate service in which the user invokes a service that carries out the chain, with the user having no awareness of the individual services. This is also known as opaque chaining.

The Construct and Deploy use case described in this ER applies workflow-managed chaining because of the use of a BPEL-based workflow engine. The Publish use case described in this ER applies an example of an aggregate service because the WPS acts a gateway to the service chain with participating services unexposed to the user.

3. Construct and Deploy Workflow (Design and Execution) Use Case

3.1 Actors

Three general types of actors are involved in Workflow and Processing: service providers, client applications, and GEOSS integrators. Service providers are those organizations or actors that publish workflows or processing services. Client applications bind to published workflows and processing services, normally directed by a GEOSS user (such as a scientist). GEOSS integrators may chain various processes or workflows to obtain a desired result.

¹ A listing of all AIP-2 Engineering Reports: http://www.ogcnetwork.net/AIP2ERs
The workflow authoring scenario involves the service provider and GEOSS integrator actors.

3.2 Context and pre-conditions

This use case describes designing a workflow, deploying a workflow, and executing a workflow. The workflow can be described in Business Process Execution Language (BPEL), Sensor Markup Language (SensorML), and any other script languages. The first case shows the workflow defined by using the Business Process Execution Language (BPEL). Two designing approaches were accounted for. One is the concrete BPEL workflow that can be designed using a standard BPEL designer, such as the Oracle BPEL Designer. Another is the abstract model which was exemplified by the design of GeoBrain Virtual Product Designer of the Center for Spatial Information Science and Systems, George Mason University. Semantic support was provided through geospatial ontology. The abstract model was then instantiated through an instantiation service to create the working workflow on the fly.

Data – Data or observations from sensors are often in a rudimentary format and projection. Further processing is required to generate useful information for decision-makers.

Algorithms – The algorithms may be abstract, thereby needing to be implemented as geographic processing services in order to re-use them in different workflows. ISO 19119 defines geographic processing services as services that perform large-scale computations involving substantial amounts of data.

Standard Web Services – Individual working Web services. For the specific use case of severe weather detection, we have a Web Feature Service that supports transactions, a Web Coverage Service for serving GOES data, a Web Processing Service for the severe weather event detection algorithms.

Workflow design tool – The use case requires a workflow design tool for authoring the workflow. BPEL is encoded in XML therefore an interactive design tool can hide the complexity of the XML describing the workflow and ensure that the defined workflow is in valid BPEL.

Workflow engine – a workflow engine is required for executing the designed workflow.
3.3 Scenario Events

Table 1 – Steps in the Workflow Authoring Scenario

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Trans. Tech Use Case</th>
<th>Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>GEOSS integrator designs the service using the BPEL designer, e.g. Oracle Designer.</td>
<td>Workflow authoring</td>
<td>Workflow authoring tool</td>
</tr>
<tr>
<td>1</td>
<td>GEOSS integrator deploys the workflow to a BPEL engine, e.g., GMU BPELPower BPEL engine.</td>
<td>Workflow authoring</td>
<td>Workflow execution engine</td>
</tr>
<tr>
<td>2</td>
<td>A GEOSS integrator or service provider provides appropriate inputs to the workflow.</td>
<td>Workflow authoring</td>
<td>Data (WFS, WCS, and WPS service endpoints)</td>
</tr>
<tr>
<td>3</td>
<td>A GEOSS integrator or service provider executes the workflow.</td>
<td>Workflow authoring</td>
<td>Workflow execution engine</td>
</tr>
<tr>
<td>4</td>
<td>Results are made available via standard service endpoints.</td>
<td>Workflow authoring</td>
<td>Data (WFS, WCS, and WPS service endpoints)</td>
</tr>
<tr>
<td>5</td>
<td>Alternatively, the execution fails and the GEOSS integrator or service provider is notified.</td>
<td>Workflow authoring</td>
<td>Workflow execution engine</td>
</tr>
</tbody>
</table>

3.4 Post-Conditions

1. Data are served through standard geospatial Web services. The data are managed by WCS and WFS.
2. Algorithms are wrapped into a standard process and managed under the Web Processing Service.
3. A composite service (or a workflow) can be executed.
4. The results are described in Geography Markup Language (GML) that can be shared across and supported widely by software platform.
5. Error case: the execution failed and the workflow engine reported the failure. Depending on the severity of the error, the composite service or workflow may not be ready for execution.

4. System Model of the Construct and Deploy Use Case

4.1 Service Authoring Diagram

In the figure, the orange squares represent the assigning activity. The blue squares are actual partners or component Web services’ operations. The individual processing algorithms are encapsulated as standard processes and managed in an implementation of the OGC Web Processing Service (WPS) standard. WPS offers a standardized interface of the inputs and outputs for geographic processing services in accordance with the OGC Web Service Reference system and ISO 19119. Two algorithms were implemented in such processes by extending an instance of the 52°North WPS. One is the severe weather detection algorithm which extracts the features of peaks, sinks, and couplets and uses these features to describe severe weather events. Another is the
severe weather tracking algorithm which links the severe weather events across time series. The tracking algorithm detects the traces of the severe weather events moved during a period. Data or sensor observations were served using WCS and/or WFS. The results can be fed back into WCS and/or WFS for persistent preservation and future analysis.

5. AIP-2 Implementation of Construct and Deploy Use Case

5.1 BPEL Script

The BPEL script describes the workflow of Severe Weather Event Detection and Tracking. BPEL uses XPath 1.0 to pass around parameters between the operations of each partner. The BPEL script used is included in appendix A of the full workflow design use case report.
5.2 Unbound WSDL Stub

The stub of the Web Service Description Language (WSDL) description for the workflow of Severe Weather Event Detection and Tracking is included in Appendix B of the full workflow design use case report. The WSDL describes the inputs and outputs of the workflow. This stub also prepares the workflow as a standard Web service when the workflow is finally deployed into a workflow engine.

5.3 Default Input Parameters

Default input parameters can be given in the suite configuration. Appendix C of the full report lists the content of the suite file for the workflow of Severe Weather Event Detection and Tracking. The suite file contains the default input parameters.

5.4 Deployment

5.4.1 Deployment Procedure

The BPEL execution engine, BPELPower, of George Mason University, was used as the base workflow engine in this demonstration of the case. The engine allows the uploading of a packaged BPEL project file, using the browser-based interface illustrated in the following figure.

Once the workflow is uploaded and deployed into the workflow engine, it can be seen in the process list. It can also be examined on debugging, activities, and external partners and their WSDLs. The following figure shows the deployed workflow and its activity diagram.
5.4.2 **Bound WSDL**

Once the workflow is deployed, its WSDL is actually bounded to the service endpoint. It becomes a standard Web Service. Appendix D in the full report shows the bound WSDL for the workflow of Severe Weather Event Detection and Tracking.

5.5 **Execution**

5.5.1 **Request**

The invocation of a workflow can be programmatic. It is also possible to invoke the service manually in a Web browser. The following figure shows the interface for invoking the severe weather event detection and tracking workflow.

![Invoke Workflow Interface]

The actual XML request is as follows.

```xml
<severeweatherDTProcessRequest xmlns="http://csiss.gmu.edu/severeweatherDT">
  <!-- For tracking, detection results for a series of three consecutive times are required. This is for time 1 -->
  <dataurl4detectorattime1>http://data.laits.gmu.edu:8080/cgi-bin/wcs110?service=WCS&amp;version=1.1.0&amp;request=GetCoverage&amp;BoundingBox=1496,916,1696,1116,urn:ogc:def:crs:OGC:0.0:imageCRS&amp;identifier=NETCDF:"/Data/G12R04D20060407184500.nc":Band4_TEMP&amp;format=image/netCDF</dataurl4detectorattime1>
  <offsetx4detectorattime1>0</offsetx4detectorattime1>
  <offsety4detectorattime1>0</offsety4detectorattime1>
  <width4detectorattime1>200</width4detectorattime1>
  <height4detectorattime1>200</height4detectorattime1>
  <mindist4detectorattime1>60</mindist4detectorattime1>
  <maxradius4detectorattime1>16</maxradius4detectorattime1>
  <maxtemp4detectorattime1>2170</maxtemp4detectorattime1>
  <minsinkarea4detectorattime1>8</minsinkarea4detectorattime1>
</severeweatherDTProcessRequest>
```
5.5.2 Response and Results

The following figure shows the response presented in the workflow engine.
The following XML lists the actual response.

```xml
<severeweatherDTProcessResponse xmlns="http://csiss.gmu.edu/severeweatherDT"
20" xmlns:tns="http://csiss.gmu.edu/severeweatherDT"
xmlns:ldap="http://schemas.oracle.com/xpath/extension/ldap"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:client="http://csiss.gmu.edu/severeweatherDT"
xmlns:targetNamespace="http://csiss.gmu.edu/severeweatherDT"
xmlns:ns1="http://www.opengeospatial.net/wps"
xmlns:ns2="http://www.opengeospatial.net/ows"
xmlns:bpelx="http://schemas.oracle.com/bpel/extension"
  <peakattime1url>http://data.laits.gmu.edu:9180/gmuwps040/RetrieveResultServlet?id=957048969peak</peakattime1url>
  <coupletattime1url>http://data.laits.gmu.edu:9180/gmuwps040/RetrieveResultServlet?id=957048969couplet</coupletattime1url>
  <sinkattime1url>http://data.laits.gmu.edu:9180/gmuwps040/RetrieveResultServlet?id=957048969sink</sinkattime1url>
  <trackvectorurl>http://data.laits.gmu.edu:9180/gmuwps040/RetrieveResultServlet?id=957048970trackvector</trackvectorurl>
  <peakattime2url>http://data.laits.gmu.edu:9180/gmuwps040/RetrieveResultServlet?id=957048967peak</peakattime2url>
  <sinkattime2url>http://data.laits.gmu.edu:9180/gmuwps040/RetrieveResultServlet?id=957048967sink</sinkattime2url>
  <peakattime3url>http://data.laits.gmu.edu:9180/gmuwps040/RetrieveResultServlet?id=957048968peak</peakattime3url>
```

6. Publish Workflow Use Case

6.1 Actors
The GEOSS service provider publishes a workflow to the GEOSS registry in this use case.

6.2 Context and Preconditions
A Web Processing Service (WPS) can act as an interface to a workflow on its own or be chained together with other processes to be part of a larger workflow. For workflow creation, WPS should be discoverable via the GEOSS registry, which requires that it be published in the registry. Workflow components, such as special arrangements, may also be published for use as part of a service chain in the same manner that is detailed here for a WPS.

Although WPS is a published OGC standard, it is not an explicit service endpoint choice for publication within the GEOSS registry. Special arrangement was chosen as the best fit for publishing authored WPS in the registry.

Preconditions:
• A deployed, findable, invokeable workflow must exist to be published.
• The GEOSS service provider, or their organization, must have an account with the GEOSS registry to publish records.

6.3 Scenario Events

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Trans. Tech Use Case</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Service provider logs into GEOSS registry portal</td>
<td>Construct and Deploy Workflow</td>
</tr>
<tr>
<td>1</td>
<td>Service provider works through service registry wizard and enters appropriate information.</td>
<td>Construct and Deploy Workflow</td>
</tr>
</tbody>
</table>

6.4 Post Conditions
The workflow is discoverable via portals that index the service registry.

7. System Model of the Publish Use Case

7.1 Population Statistics WPS
A GIS analysis of raster population data (persons, 2005) has been set up as a WPS. The WPS accepts simple feature (polygon) GML from a WFS or as a file upload and returns the same features with associated population statistics (parametric statistics, population totals, and data quality indicators). The WPS acts a proxy to an ArcGIS Server Geoprocessing task. The Population Statistics WPS is intended to be used for analysis of GIS data or as a single component contributing to a more complex workflow.
In contrast to the Construct and Deploy use case which uses a browser-based client to invoke the workflow, the Publish use case uses the uDig desktop GIS to invoke the WPS.

A demonstration of the Population Statistics WPS is available as a video. See the “Geoprocessing Services” video at: http://www.ogcnetwork.net/pub/ogcnetwork/GEOSS/AIP2/index.html.

8. AIP-2 Implementation of the Publish Use Case

8.1 Publication in the GEOSS Registry

The population statistics WPS was published in the GEOSS registry as a GEOSS service (special arrangement). The registry wizard interface was used to assign a unique name, id, and enter service details (URLs, contact information, description).

The information and interface URL for the WPS is: http://beta.sedac.ciesin.columbia.edu/wps/WebProcessingService?Request=GetCapabilities&Service=WPS

8.2 Population Statistics WPS Capabilities

<?xml version="1.0" encoding="UTF-8"?>
  <ows:ServiceIdentification>
    <ows:Title>CIESIN WPS</ows:Title>
    <ows:Abstract>WPS for GEOSS AIP Workflow WG</ows:Abstract>
    <ows:Keywords>
      <ows:Keyword>WPS</ows:Keyword>
      <ows:Keyword>AAFC</ows:Keyword>
      <ows:Keyword>geospatial</ows:Keyword>
      <ows:Keyword>geoprocessing</ows:Keyword>
    </ows:Keywords>
    <ows:ServiceType>WPS</ows:ServiceType>
    <ows:ServiceTypeVersion>1.0.0</ows:ServiceTypeVersion>
    <ows:ServiceTypeVersion>0.4.0</ows:ServiceTypeVersion>
    <ows:Fees>NONE</ows:Fees>
    <ows:AccessConstraints>NONE</ows:AccessConstraints>
  </ows:ServiceIdentification>
  <ows:ServiceProvider>
    <ows:ProviderName>CIESIN</ows:ProviderName>
  </ows:ServiceProvider>
  <ows:ServiceContact>
    <ows:IndividualName>Brian Falk</ows:IndividualName>
    <ows:PositionName>Programmer</ows:PositionName>
  </ows:ServiceContact>
  <ows:ContactInfo>
    <ows:Phone/>
    <ows:Voice/>
    <ows:Facsimile/>
  </ows:ContactInfo>
  <ows:Address/>
</wps:Capabilities>
9. Next Steps

One of the features of BPEL that has not been explored in the AIP is the use of correlation sets for creating parallel sub-processes. BPEL offers correlation sets for running inter-related processes that may need to share data. The processes may have separate service endpoints although being part of the same workflow. An example of a
geospatial workflow that uses correlation sets is presented by Hobona et al (2009). Such a feature could be applicable in scenarios where data is retrieved from different sensors and processed concurrently before being presented to a user.

10. References

