

MAX GENI Aggregate Manager Control Framework Usage Overview Version 2

1.0 MAX GENI Overview

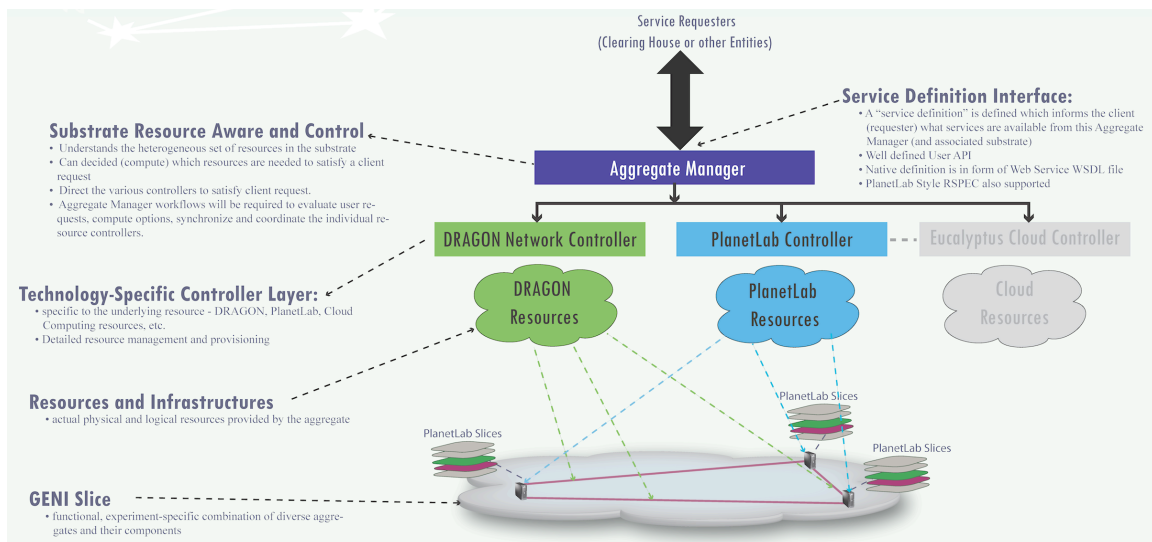
The MAX project is building a GENI facility known as the "MAX GENI Facility". This is based on the MAX network research infrastructure that includes a dedicated experimental fiber-based regional network in the Washington DC metro area. As part of this effort, MAX GENI will provide the GENI community with access to a regional optical network consisting of wavelength-selectable switches, 10 Gbps Ethernet switches, virtual machines, and other host based resources. The network infrastructure is building upon the NSF-funded DRAGON research network, which enables dynamic provisioning of deterministic, high-speed, dedicated network resources via a standardized Web Services interface and a distributed control plane. These dynamic network technologies and software are also deployed on multiple research and education networks throughout the world. This wide deployment will be utilized to facilitate access to the MAX GENI capabilities by remotely located researchers.

A key effort of the MAX GENI project is the development of the MAX GENI Aggregate Manager. This is the component of the GENI Architecture that enables researchers to instantiate GENI slices/slivers. For the GENI MAX substrate, a GENI sliver includes the integration of dynamic network provision with host based resource provisioning in the form of a experimenter specified topology. The host based resources currently include the PlanetLab nodes and slices. Work is underway to also include other host based resources including NetFPGA hosts, OpenFlow enabled systems, and Eucalyptus clusters. The network based resources include the ability to dynamically provision high bandwidth dedicated network resources across the MAX DRAGON research network. The project web site is located here: <http://geni.maxgigapop.net>.

This document provides an overview of how to access and use the MAX GENI Aggregate Manager.

2.0 MAX GENI Aggregate Manager Overview

The MAX Aggregate Manager is responsible for management of the MAX GENI substrate resources and the client interfaced. In this context the client may be a GENI Clearinghouse agent, another type of slice manager, another aggregate manager, or an individual experimenter. Our current focus is on the Aggregate Manager Interface and usage independent of the nature of the client requesting these services. The basic architecture of this Aggregate Manager is as shown below:



The key architectural features to note regarding this Aggregate Manager are:

-Standard External Interface

The external interface is based on web services with a service definition in the form of a WSDL file (AggregateGENI.wsdl).

-Modularization

Since the resources in any given substrate are expected to be variable and change over time, this architecture accommodates this via a modular approach to resource controllers. For the initial MAX Substrate, the primary resources are DRAGON provisioned network paths, PlanetLab provisioned user slices, and NetFPGA hosts. The DRAGON resource controller is the existing dynamic provisioning system that is deployed on multiple research and education networks throughout the world. This resource controller is known as an InterDomain Controller (IDC) and is based on the combination of OSCARS and DRAGON software. The PlanetLab controller is MyPLC PlanetLab Central/SFA system that allows slices to be created via human web interface or via application access. The MAX aggregate manager is responsible for integrating the functions across these multiple resource controllers in response to requests presented at the common external interface.

-Extensibility

Another natural benefit of this architecture is extensibility in terms of accommodation of new resources (and associated controllers) as they become available. As shown in the above figure, planned future activities include the addition of Eucalyptus resources to this capability set. As a result a slice topology can be instantiated that may span multiple resource types within the substrate.

3.0 MAX GENI Aggregate Manager Installation Notes

The focus of this document is a description of the service interface and an overview of how to access the services and capabilities of the MAX GENI Aggregate Manager and associated substrate resources. Therefore, the details of installation and deployment of a MAX GENI Aggregate Manager are not covered in this document. However, this section has a few notes for those interested in more information on the software implementation and/or deployment of an aggregate manager of this type.

The MAX Aggregate Manager The MAX Aggregate Manager software is a Java-based implementation that provides a Web Services API (WSDL) to clients. It is intended to be deployed in Apache Tomcat as an Axis2 service. The MAX GENI web site has the Aggregate manager documentation here:

- <http://geni.maxgigapop.net/twiki/bin/view/GENI/Software>

The latest version of the MAX AggregateGENI software distribution will also allow the compile and build process to be completed wholly at the Linux terminal if the use of NetBeans is not desired as currently described on the web site. The ANT build script has been modified and all dependency libraries are now included in the distribution and the configuration structure is streamlined. The svn repository contains README and sample configure files which provides detailed instructions on installation and configuration. The effort has greatly improved deployment experience. This build procedure is documented in the repository \$AGGREGATE_HOME/AggregateWS/README file.

The MAX-DRAGON-GENI Aggregate Manager software repository is located here:

- <svn://svn.maxgigapop.net/geni-aggregate>

Below is a listing of the locations of some of other critical software components for the various substrate resources:

PlanetLab Control Framework MAX RSPEC:

<http://svn.planet-lab.org/svn/sfa/trunk/sfa/rspecs/aggregates/max.xml>

PlanetLab Control Framework SFA:
<http://svn.planet-lab.org/svn/sfa/trunk/>

DRAGON Software Repositories:
<svn://cvs.maxgigapop.net/dragon-sw>
<http://sands225.east.isi.edu/narb-sw>

DRAGON/OSCARS/IDC Software Package:
<https://wiki.internet2.edu/confluence/display/DCNSS>

4.0 MAX GENI Aggregate Manager Service Usage Overview

The MAX GENI Aggregate Manager Service Interface is available here:

- MAX Aggregate Manager Service Interface
 - Human Readable Service Definition via Web Browser (FireFox) or SOAP Client (ditchnet.org/soapclient)
<http://geni.dragon.maxgigapop.net:8080/axis2/services/AggregateGENI?wsdl>
 - SOAP API Interface
<http://geni.dragon.maxgigapop.net:8080/axis2/services/AggregateGENI>

As indicated, the application interface is defined in terms of a standard SOAP based web services mechanisms. For each of the service descriptions below, an example client program is available in the software repository. An overview of usage for this is provided in Section 5.0 Example GENI MAX Aggregate Manager Workflow. References to some of the components in the GENI MAX Aggregate Manager software distribution (<svn://svn.maxgigapop.net/geni-aggregate>) are utilized in the service descriptions that follow.

The GENI MAX Aggregate Manager services are structured into four groups:

- *Compute Slice Services* group includes:
 - CreateSlice, DeleteSlice, QuerySlice, UpdateSlice, StartSlice and StopSlice.
- *VLAN Networking Services* group includes:
 - CreateSliceVlan, DeleteSliceVlan and QuerySliceVlan
- *RSpec-Based Slice Services* group includes:
 - CreateSliceNetwork, DeleteSliceNetwork and QuerySliceNetwork.
- *Resource Information and Topology Services* group includes:
 - ListCapabilities, ListNodes ListSlices and GetResourceTopology.

4.1 Compute Slice Services

These services allow for the creation of host resource based slices. At the current time this is limited to PlanetLab slices. In the future this will be extended to include NetFPGA hosts, OpenFlow enabled systems, and Eucalyptus clusters.

The typical use case for the services is to request a slice with CreateSlice. Parameters include sliceName, user (email, name or ID) and node names. A slice is created with a “SUCCESSFUL” status returned or “FAILED” when error occurs. A “message” field is also included in the response to provide extra information such as failure reasons.

The slice can then be modified with UpdateSlice. Parameters include sliceName, new user list, new node list and new expiration time (epoch timestamp). The slice can be terminated via DeleteSlice. Similarly, “SUCCESSFUL” or “FAILED” status and message fields are replied.

One may also query the slice details with the QuerySlice service. This service can query more than one slice at a time. Example SOAP request messages for the above services are shown in createSlice.xml, updateSlice.xml, deleteSlice.xml and querySlice.xml under:

- \$AGGREGATE_HOME/AggregateAttic/soap/

Corresponding example SOAP replies messages are under:

- \$AGGREGATE_HOME/AggregateAttic/soap/responses/.

4.2. VLAN Networking Services

If a compute slice has already been created, a point-to-point (P2P) VLAN connection can be added between any pair of nodes/slivers by using CreateSliceVlan. The parameters include sliceName, vlanTag, sourceNodeName, destinationNodeName. When source and/or destination interface (e.g. "eth1") and IP address/mask are also provided, the service will configure the tagged Ethernet interface on the node/sliver (e.g. "eth1.3000").

The QuerySliceVlan and DeleteSliceVlan can then be utilized to query and terminate the P2P VLAN connection. Note that the CreateSliceVlan service only returns a 'ACCEPTED' or 'FAILED' status. Use of the QuerySliceVlan service will be required to check whether the connection is up into 'ACTIVE' status. Transition to the ACTIVE state may take from ten seconds to a few minutes depending on how many domains are involved in the circuit instantiation. Example SOAP request messages are shown in the createSliceVlan-*.xml, deleteSliceVlan-*.xml and querySliceVlan-*.xml files in \$AGGREGATE_HOME/AggregateAttic/soap/ location.

It should be noted that the point-to-point (P2P) VLAN connections can be made between PlanetLab slices which are in different substrates. However, the initial creation of the other substrate the PlanetLab slices, and subsequent IP address configuration, will be limited until the PlanetLab Federation capabilities are available. The other solution for this issue, is for a higher level slice manager (clearinghouse or some other slice manager application) to utilize the MAX GENI Aggregate Manager to instantiate the MAX substrate PlanetLab Slices, and inter-domain (substrate) networking. Then the same higher level slice manager could instantiate the PlanetLab slice on the other substrate via the remote aggregate manager.

4.3 RSpec-Based Slice Services

Another method to instantiate resources within the substrate is to describe the request in a RSpec description. The CreateSliceNetwork is the service which provides for this. An example SOAP request message is shown in createSliceNetwork.xml. As can be seen from this example, a RSpec XML description is wrapped in the <statement></statement> element. It describes a networked compute slice that includes three PlanetLab nodes/slivers and three P2P VLANs that connect them into a triangle topology. The RSpec XML description is validated against the XSD schema files under \$AGGREGATE_HOME/AggregateAttic/conf/schema/.

Similarly one can query and terminate a RSpec slice via the QuerySliceNetwork and DeleteSliceNetwork services. The example SOAP messages are in querySliceNetwork.xml and deleteSliceNetwork.xml. Note that both CreateSliceNetwork and DeleteSliceNetwork receives instant response from the Aggregate Manager that only indicates whether the creation/deletion request is acceptable. A subsequent use of QuerySliceNetwork service should be utilized to obtain latest resource allocation states.

4.4 Resource Information and Topology Services

The Resource Information and Topology Services group is intended to allow experimenters to discover the resources available in the substrate. The ListCapabilities service replies with a list of capabilities the current aggregate supports. The "filter" parameter is not enforced at the current time, a value of "all" should be place in there to obtain a full listing of substrate resources. An example is shown in the caps.xml file located in \$AGGREGATE_HOME/AggregateAttic/soap/ location.

The ListNodes service replies with a list of compute nodes that have ALL the capabilities whose URNs are listed in the request. An example is shown in the nodes-*.xml files located in \$AGGREGATE_HOME/AggregateAttic/soap/ location.

The ListSlices service replies with a list of compute slices that have been created by the aggregate. An example is shown in the slices.xml files located in \$AGGREGATE_HOME/AggregateAttic/soap/ location.

The GetResourceTopology service provides a topology and resource description of the entire substrate or of a specific experiment topology. Use of this service without a "rspec" field returns the physical topology of the entire substrate infrastructure including both compute nodes and network topology. If the "rspec" field contains a RSpec name (specified by the 'id' attribute of a RSpec description), the resource topology of the requested slice will be returned.

4.5 Fault Messages

For service requests that are admitted by the Aggregate Manager, an operation fault is indicated by a "FAILED" value of the "status" field in the response message. For service requests that are refused by the Aggregate Manager, for example a duplicate request, a SOAP Fault/Exception message will be replied.

5.0 GENI MAX Aggregate Manager Work Flow Example

Use of the GENI MAX RSpec-Based Slice Services is recommended baseline mechanism for the provision of integrated compute and network resources. However, the more atomic Compute Slice and VLAN Networking services may be desired to fulfill logic of a designated workflow. In addition, it may be desired to utilize one of the VLAN Networking services to create and manage P2P VLAN connections between nodes in separate aggregates.

Below is a sequence of GENI MAX Aggregate Manager service calls that may be implemented in a typical experiment workflow. The following description includes the client side command line examples for the ws_client.pl located in \$AGGREGATE_HOME/AggregateAttic/soap/ws_client.pl. The request file formats, and resulting response outputs are also included in the source tree soap directory, so they will not be repeated here.

i) Run ListCapabilities service to see what capabilities are available in the MAX substrate:

```
./ws_client.pl -s geni.dragon.maxgigapop.net:8080 -x caps --pretty
```

ii) Run GetResourceTopology service to see what the MAX physical network and host resource topology looks like.

```
./ws_client.pl -s geni.dragon.maxgigapop.net:8080 -x getResourceTopology --pretty
```

iii) From the above call, it will be possible to create a specific Rspec request, as shown in the createSliceNetwork.xml file in the \$AGGREGATE_HOME/AggregateAttic/soap/ directory. The following command will utilize the CreateSliceNetwork service to instantiate that topology:

```
./ws_client.pl -s geni.dragon.maxgigapop.net:8080 -x createSliceNetwork --pretty
```

iv) The next call will utilize the QuerySliceNetwork service to query the status of the requested topology:

```
./ws_client.pl -s geni.dragon.maxgigapop.net:8080 -x querySliceNetwork --pretty
```

vi) When the experiment is complete, the DeleteSliceNetworkService can be utilized to release the reserved topology:

```
./ws_client.pl -s geni.dragon.maxgigapop.net:8080 -x deleteSliceNetwork --pretty
```

6.0 MAX GENI Substrate and Aggregate Manager Access Control

This early version of the MAX GENI Aggregate Manager relies on the username field in all the service request to control access to the substrate resources. This requires the set up of accounts for the MAX MyPLC in advance of submitting service requests via the MAX GENI Aggregate Manager. Please check the web site (geni.maxgigapop.net) for information on setting up these accounts.

7.0 Service Interface Locations

The currently deployed capabilities are available at the following locations.

- MAX Aggregate Manager Service Interface
 - Human Readable Service Definition via Web Browser (FireFox) or SOAP Client (ditchnet.org/soapclient)
<http://geni.dragon.maxgigapop.net:8080/axis2/services/AggregateGENI?wsdl>
 - SOAP API Interface
<http://geni.dragon.maxgigapop.net:8080/axis2/services/AggregateGENI>
- MAX MyPLC Service Interface
 - Human Web Browser Interface
<https://max-myplc.dragon.maxgigapop.net/>
 - XML-RPC/SOAP Interface
<https://max-myplc.dragon.maxgigapop.net/PLCAPI/>
- DRAGON Network Provisioning Interface
 - Web Browser Provisioning Interface:
<https://idc.dragon.maxgigapop.net:8443/OSCARS/>
 - Human Readable Service Definition via Web Browser (FireFox)
<https://idc.dragon.maxgigapop.net:8443/axis2/services/OSCARS?wsdl>
 - SOAP API Interface
<https://idc.dragon.maxgigapop.net:8443/axis2/services/OSCARS>