
GENI Network Stitching

Architecture Status

GEC11

July 27, 2011

Tom Lehman (USC/ISI)

Xi Yang (USC/ISI)

Stitching Architecture and Schema Review

Stitching Architecture Status

Stitching Architecture Reviewed at GEC10

- **Agreed to move forward with Stitching Architecture and work on details**
- **GEC11 Plan**
 - **review schema and stitching implementation status**
 - **identify and begin discussions on next set of stitching architecture components and issues**

Stitching Architecture

Five Stitching Architecture Components

- **Stitching Schema**
- **Stitching Topology Service**
- **Stitching Path Computation Function**
- **Stitching Workflow Function**
- **GENI AM API Stitching Extensions**

Architecture Components

- **Stitching Schema**

- provides mechanism to define and describe inter-aggregate touch points
- implemented as an extension to ProtoGENIv2 Schemas
- Intent is that Aggregate Managers will use this as part of Advertisement RSPEC
- also includes stitching path elements intended to be used as part of request and stitching operations

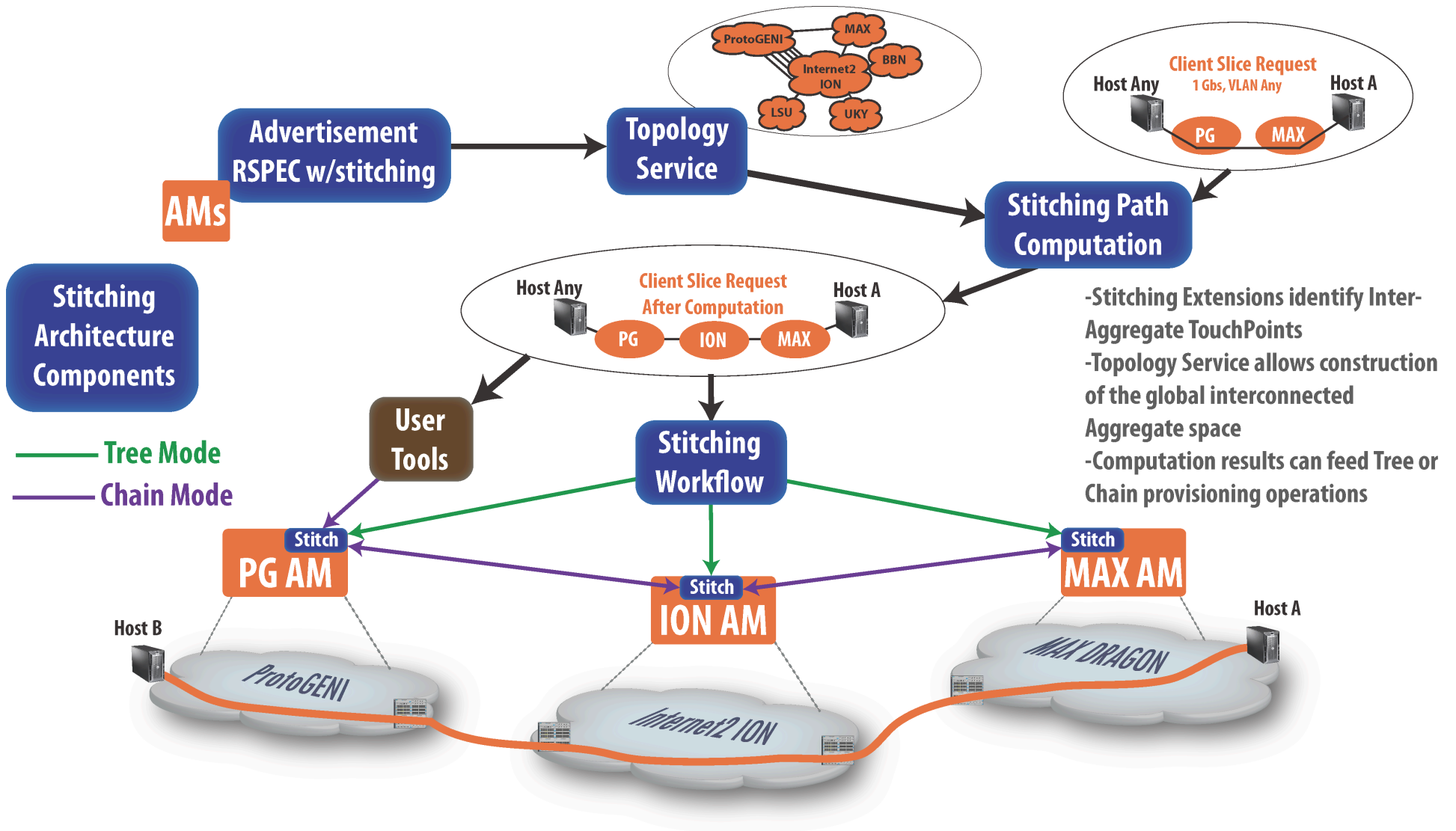
Architecture Components

- **Stitching Topology Service**
 - collects Stitching RSPECs from all GENI AMs
 - Allows the building of larger views (global if desired) of inter-connected GENI AM space
- **Stitching Path Computation Function**
 - has the ability to obtain the global network stitching topology view from the Stitching Topology Service, perform slice instantiation specific path computations

Architecture Components

- **Stitching Workflow Function**
 - has the ability to take the path computations results and execute the workflow steps required to accomplish network stitching. This includes interacting with the specific AM APIs and external network resources
- **GENI AM API Stitching Extensions**
 - extensions to the GENI AM API to support robust and globally scalable network stitching functions.

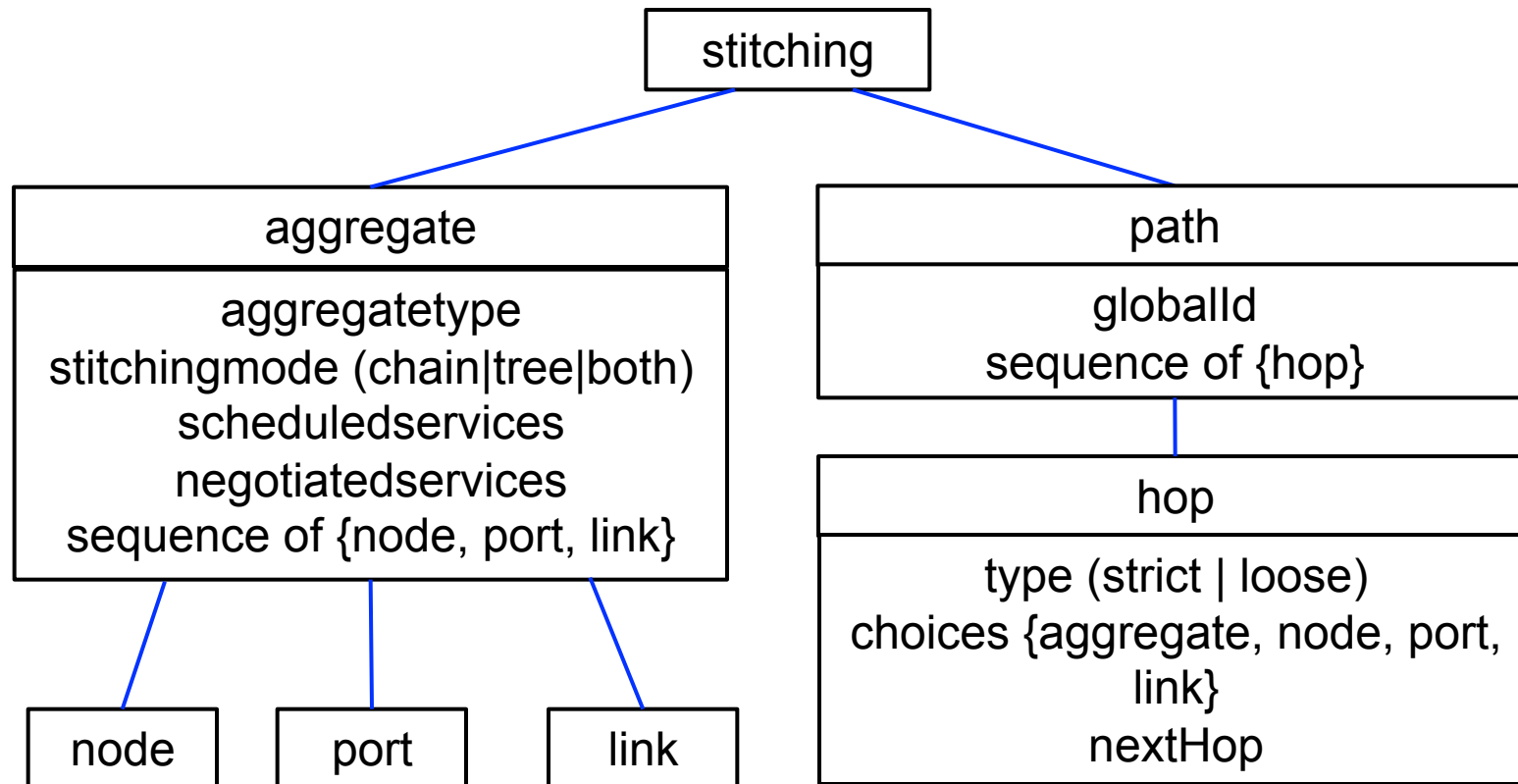
Stitching Architecture



Stitching Schema

- Stitching schema based on Dynamic Circuit Network (ION, IDC, OSCARS) schemas and perfSONAR Topology Service schemas
- Some adjustments for GENI Aggregate specific considerations
- Two goals are
 - maximize interoperation with existing and future dynamic networking deployments
 - provide opportunities to leverage existing perfSONAR systems and software

Stitching Schema



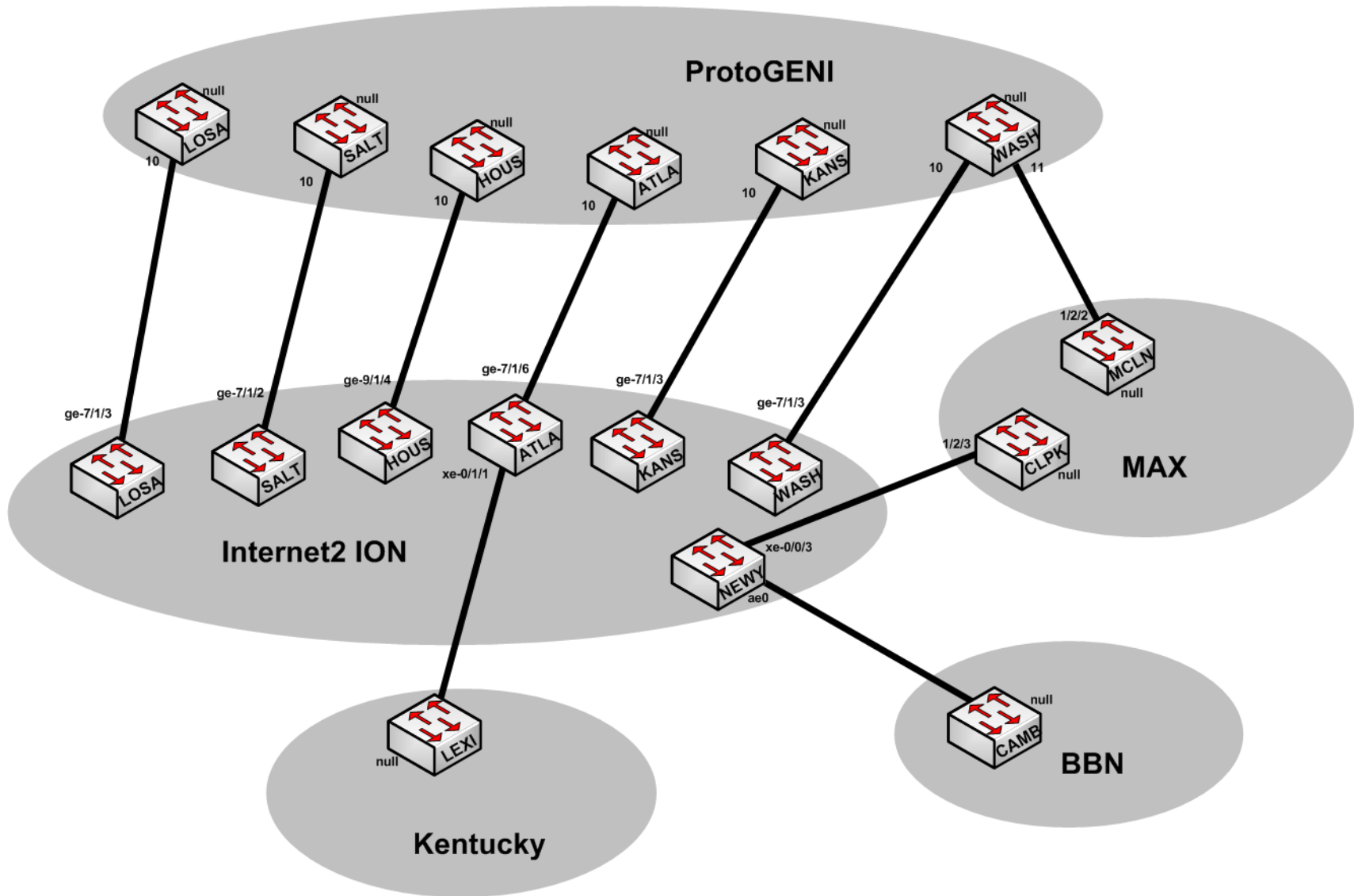
technology specific details here

<http://hpn.east.isi.edu/rspec/ext/stitch/0.1/stitch-schema.xsd>

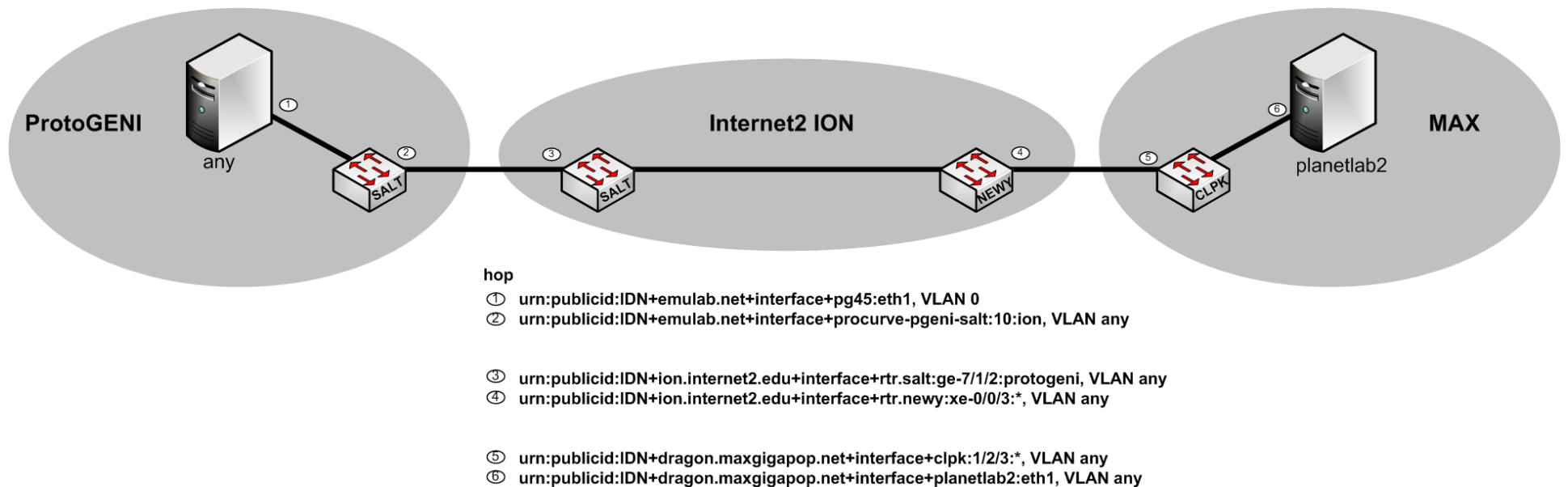
Stitching Schema

- aggregate element
 - used by Aggregate Managers in Advertisement RSpec to describe inter-aggregate touch points
- path element
 - used in Request to indicate stitching preferences and/or requirements
 - may be fed by path computation or simply client preferences
 - used in Manifest RSpec to describe stitching of instantiated slice
- Stitching schema available as an extension to ProtoGENIv2 RSpec Schemas

Advertisement (aggregate element) – Graph View

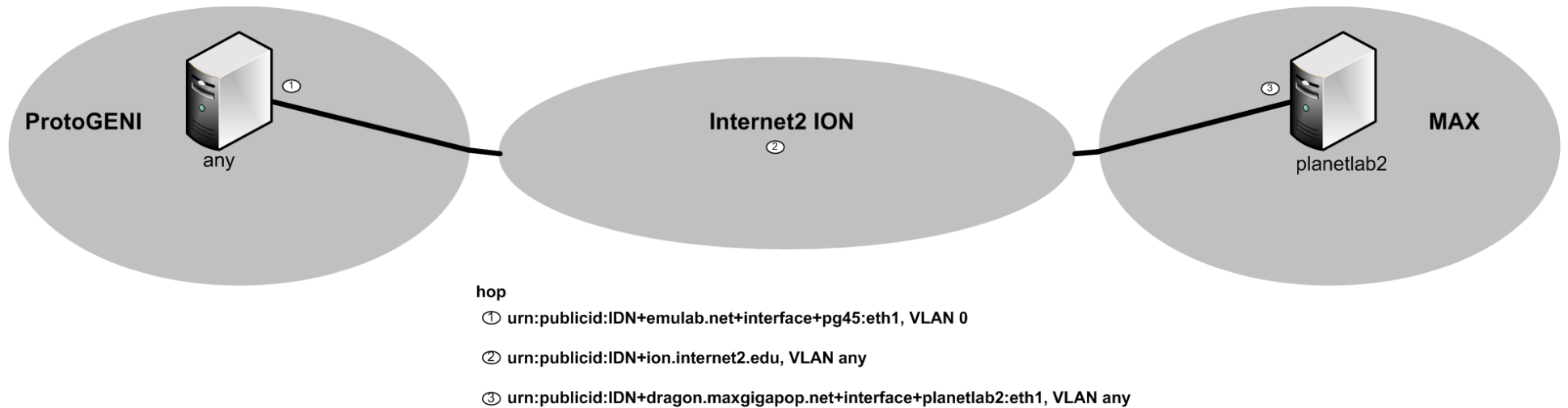


Request (path element) – Graph View



- Hops can be loose or strict
- Hops can be specified at the aggregate, node, port, or link level
- Manifests will look similar to Requests (annotated with instantiated info)

Request (path element) – Graph View



- Hops can be loose or strict
- Hops can be specified at the aggregate, node, port, or link level
- Manifests will look similar to Requests (annotated with instantiated info)

Request RSpec Example

<http://geni.maxgigapop.net/twiki/pub/GENI/NetworkStitchingRpecsandWorkFlow/request-from-client-after-computation-v4.xml>

```
- <rspec xsi:schemaLocation="http://www.protogeni.net/resources/rspec/2 http://www.protogeni.net/resources/rspec/2/request.xsd" type="request">
+ <node client_id="left" component_manager_id="urn:publicid:IDN+emulab.net+authority+cm" exclusive="true"></node>
+ <node client_id="right" component_id="urn:publicid:IDN+dragon.maxgigapop.net+node+planetlab2"
  component_manager_id="urn:publicid:IDN+dragon.maxgigapop.net+authority+cm" exclusive="true"></node>
+ <link client_id="mylink"></link>
- <stitching lastUpdateTime="20110220:09:30:21">
  - <path id="mylink">
    - <hop id="1" type="strict">
      + <link id="urn:publicid:IDN+emulab.net+interface+*:*"></link>
      <nextHop>2</nextHop>
    </hop>
    - <hop id="2" type="strict">
      + <link id="urn:publicid:IDN+emulab.net+interface+procurve-pgeni-salt:eth0:ion"></link>
      <nextHop>3</nextHop>
    </hop>
    - <hop id="3" type="strict">
      + <link id="urn:publicid:IDN+ion.internet2.edu+interface+rtr.salt:ge-7/1/2:protogeni"></link>
      <nextHop>4</nextHop>
    </hop>
    - <hop id="4" type="strict">
      + <link id="urn:publicid:IDN+ion.internet2.edu+interface+rtr.newy:xe-0/0/3:*"></link>
      <nextHop>5</nextHop>
    </hop>
    - <hop id="5" type="strict">
      + <link id="urn:publicid:IDN+dragon.maxgigapop.net+interface+clpk:1/2/3:*"></link>
      <nextHop>6</nextHop>
    </hop>
    - <hop id="6" type="strict">
      + <link id="urn:publicid:IDN+dragon.maxgigapop.net+interface+planetlab2:eth1"></link>
      <nextHop>null</nextHop>
    </hop>
  </path>
</stitching>
</rspec>
```

Stitching Architecture Information

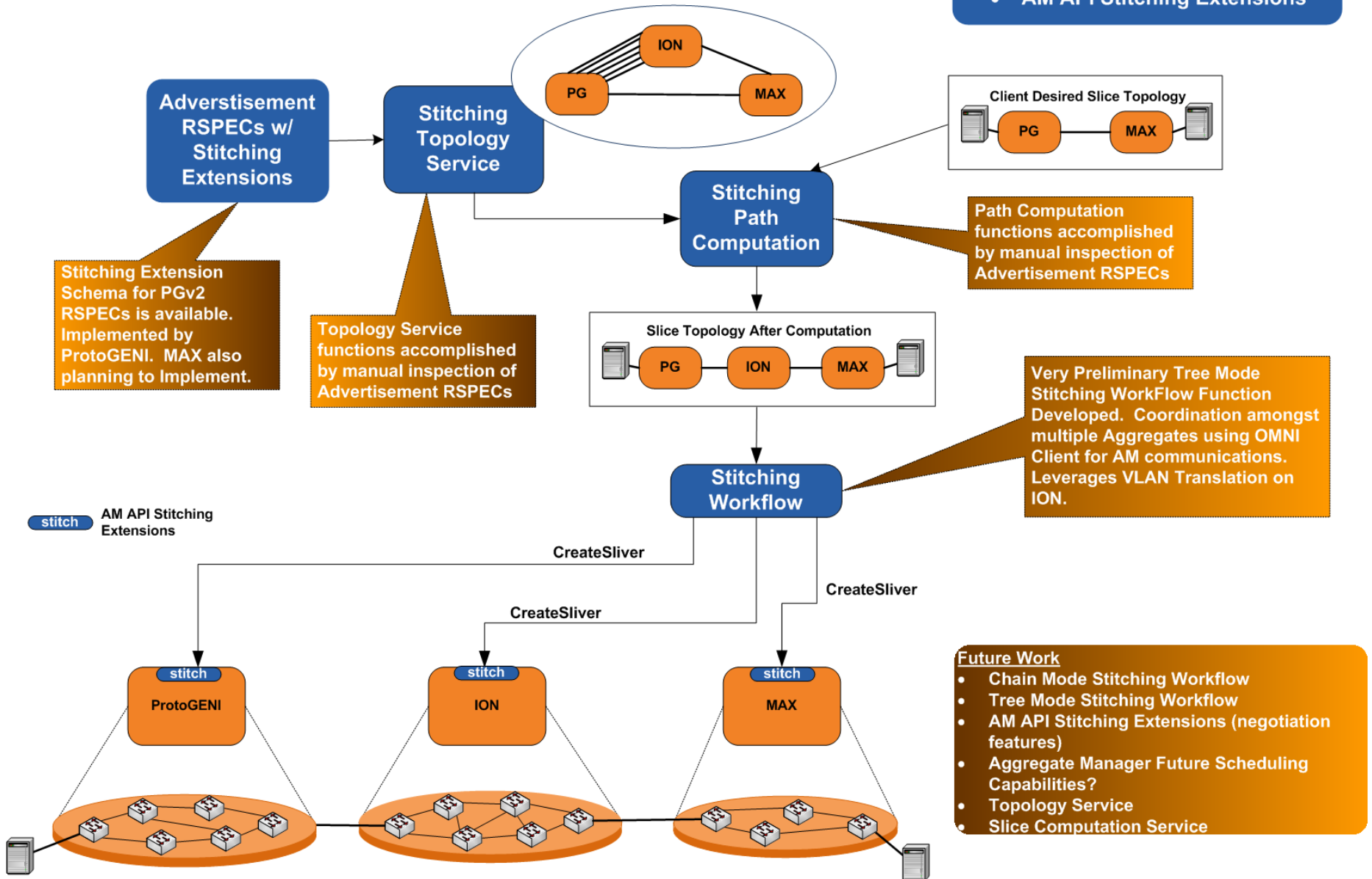
- **Overview:** geni.dragon.maxgigapop.net → Network Stitching
- **Architecture:** geni.maxgigapop.net/twiki/bin/view/GENI/NetworkStitching
- **Stitching Schema Examples:**
geni.maxgigapop.net/twiki/bin/view/GENI/NetworkStitchingRpecsandWorkFlow
- **WorkFlow Example:** geni.maxgigapop.net/twiki/bin/view/GENI/NetworkStitchingWorkFlowExamples
- **GENI Wiki:** <http://groups.geni.net/geni/wiki/GeniNetworkStitching>

Stitching Architecture Next Steps

How Does Current Implementation and Demonstration Capability Relate to Stitching Architecture?

Stitching Architecture Components

- RSPEC Stitching Extensions
- Stitching Topology Service
- Stitching Path Computation
- Stitching Workflow
- AM API Stitching Extensions



Stitching Extension Schema for PGv2 RSPECs is available. Implemented by ProtoGENI. MAX also planning to implement.

Topology Service functions accomplished by manual inspection of Advertisement RSPECs

Path Computation functions accomplished by manual inspection of Advertisement RSPECs

Very Preliminary Tree Mode Stitching WorkFlow Function Developed. Coordination amongst multiple Aggregates using OMNI Client for AM communications. Leverages VLAN Translation on ION.

Future Work

- Chain Mode Stitching Workflow
- Tree Mode Stitching Workflow
- AM API Stitching Extensions (negotiation features)
- Aggregate Manager Future Scheduling Capabilities?
- Topology Service
- Slice Computation Service

Stitching Next Steps

- Tree Mode Workflow
- Chain Mode Workflow
- AM API extensions (negotiation)
- Topology Service
- Stitching Path Computation Service
- AM Scheduled Services

Stitching Next Steps

- Tree Mode Stitching Workflow
 - the tree mode logic is built into the Stitching Workflow function
 - Stitching Workflow function can inspect AM Advertisements and Request RSpecs do a lot now using the existing AM API
 - However, adding a "negotiation" feature to the AM API would help in many situations

Stitching Next Steps

- Chain Mode Stitching Workflow
 - the chain mode logic is built into the Aggregate Manager
 - this will require extensions to AM API for AM to AM messaging.
 - This also includes a "negotiation" feature.
- Chain Mode and Tree Mode use the same Advertisement and Request RSpecs
 - can also use the same AM API extension for "negotiation"

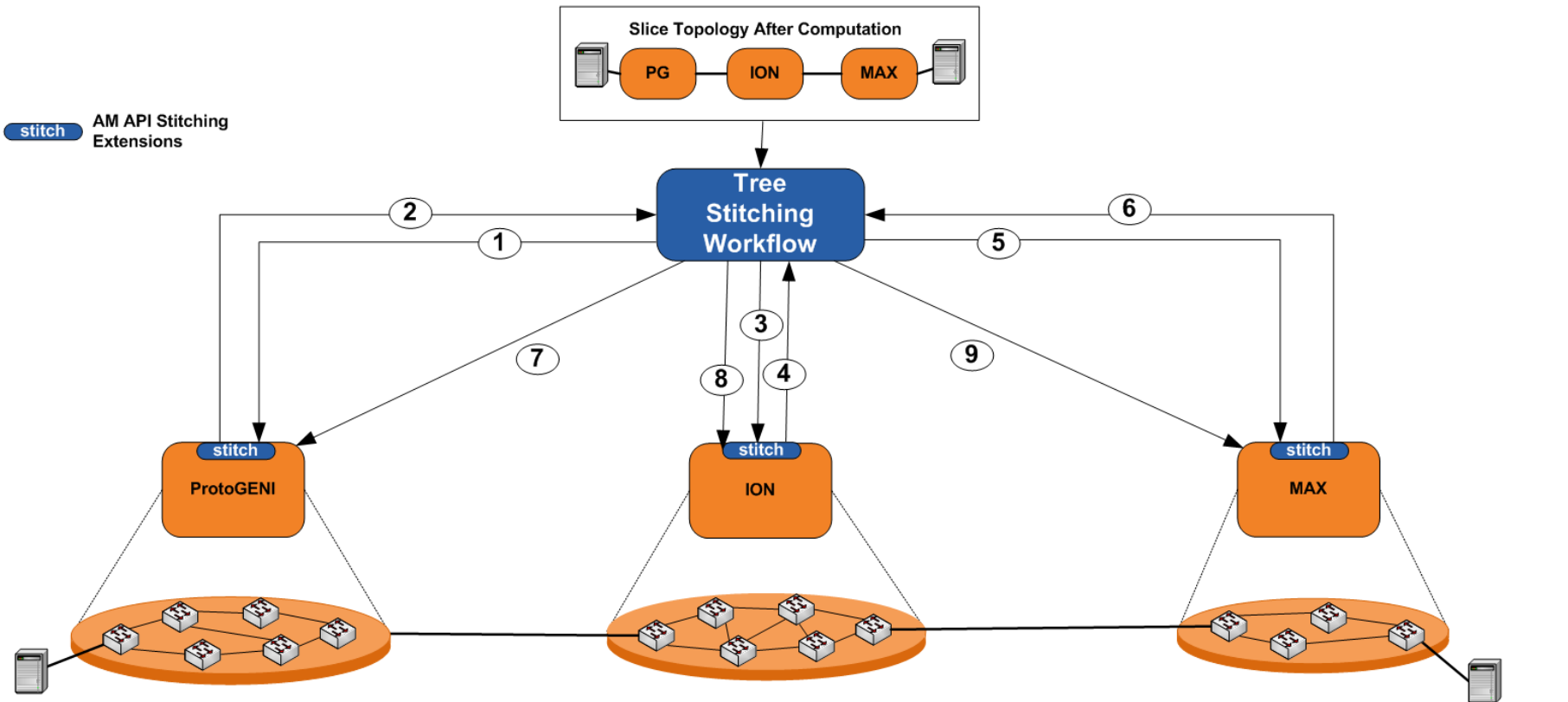
AM API Negotiation

- AM API Extension for Negotiation
 - a simple flag indicating if an AM supports negotiation in Advertisement RSpec ("negotiatedservices= true | false")
 - An AM which supports negotiation, may receive a "CreateSilver" request with "negotiatedservices=on"
 - This indicates that the AM should process the request as normal, except the return should include a "range of vlans that are available for use" and one "suggested vlan" which will be held for a specific amount of time.

AM API Negotiation

- AM API Extension for Negotiation (cont'd)
 - Stitching resources are not instantiated as a result of a request with "negotiatedservices=on"
 - Tree Mode Workflows will use this to tailor requests to subsequent AMs
 - Chain Mode Workflows will use this as part of AM to AM messaging which will work their way down and back up the chain as "vlan range" and "suggested vlan" fields are modified along the way
 - This is a vlan based description, can also be used for negotiation of any other resources in the schema as well

Tree Mode Stitching Workflow with Negotiation



1. CreateSliver Request, negotiatedservices=on, stitchingmode=tree, vlan=any

2. CreateSliver Reponse, vlanRangeAvailability=3003-3015, suggestedVLANRange=3005

3. CreateSliver Request, negotiatedservices=on, stitchingmode=tree, vlanRangeAvailability=3003-3015, suggestedVLANRange=3005

4. CreateSliver Response, vlanRangeAvailability=3004-3010, suggestedVLANRange=3005

5. CreateSliver Response, negotiatedservices=on, stitchingmode=chain, vlanRangeAvailability=3004-3010, suggestedVLANRange=3010

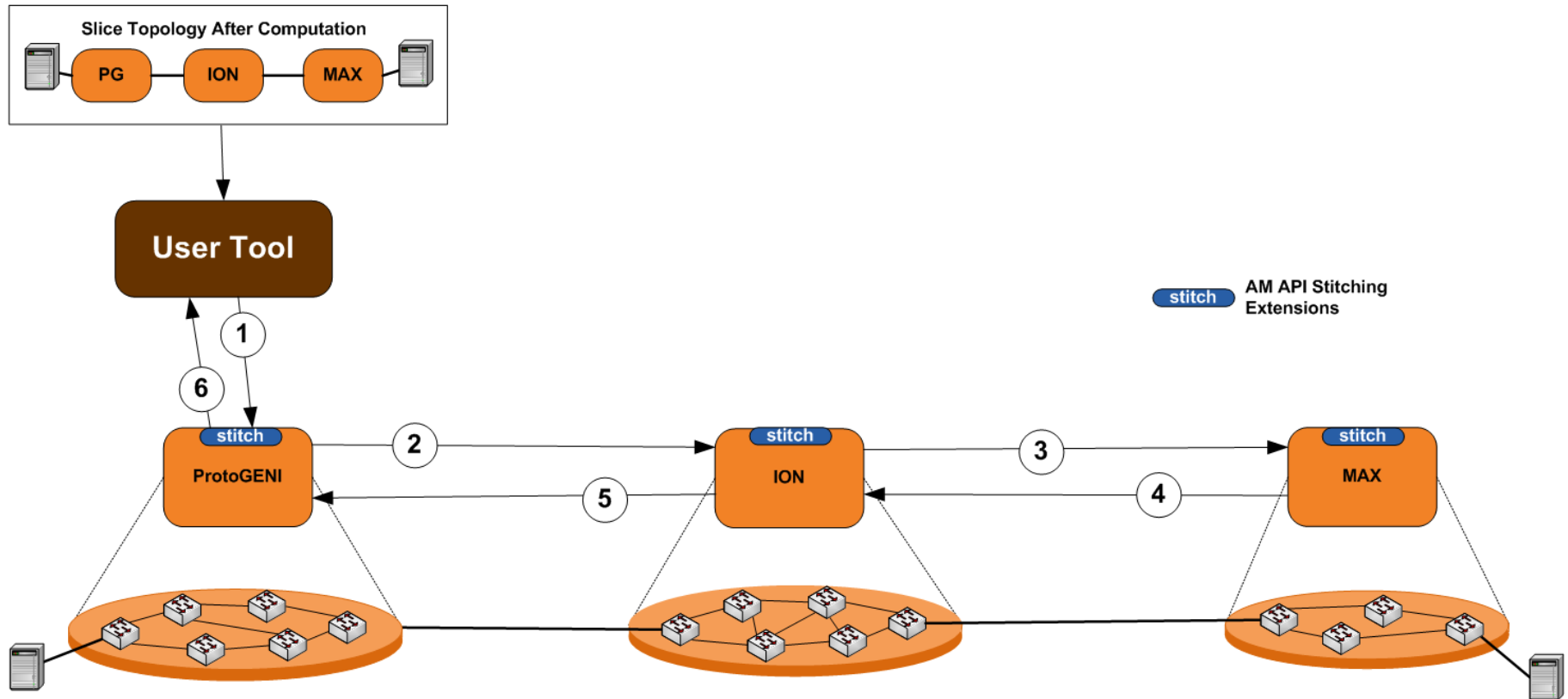
6. CreateSliver Response, vlanRangeAvailability=3007-3009, suggestedVLANRange=3009

7. CreateSliver Request, negotiatedservices=off, stitchingmode=tree, vlanRangeAvailability=3009, suggestedVLANRange=3009

8. CreateSliver Request, negotiatedservices=off, stitchingmode=tree, vlanRangeAvailability=3009, suggestedVLANRange=3009

9. CreateSliver Request, negotiatedservices=off, stitchingmode=tree, vlanRangeAvailability=3009, suggestedVLANRange=3009

Chain Mode Stitching Workflow with Negotiation



1. CreateSliver Request, negotiatedservices=off, stitchingmode=chain, vlan=any

2. CreateSliver Request, negotiatedservices=on, stitchingmode=chain, vlanRangeAvailability=3003-3015, suggestedVLANRange=3005

3. CreateSliver Request, negotiatedservices=on, stitchingmode=chain, vlanRangeAvailability=3004-3010, suggestedVLANRange=3005

4. CreateSliver Response, negotiatedservices=off, stitchingmode=chain, vlanRangeAvailability=3009, suggestedVLANRange=3009

5. CreateSliver Response, negotiatedservices=off, stitchingmode=chain, vlanRangeAvailability=3009, suggestedVLANRange=3009

6. CreateSliver Response, negotiatedservices=off, stitchingmode=chain, vlanRangeAvailability=3009, suggestedVLANRange=3009

Topology Service - Next Steps

- Topology Service
 - Deploying a "GENI Topology Service" as persistent infrastructure item would be convenient for user to explore the interconnected global Aggregate Space
 - could be based on modified perfSONAR topology service
 - populated by AM Advertisement RSpecs
 - provides basis for visualization of global GENI space
 - may not be necessary for large aggregates/control frameworks with their own advertisement aggregation and computation capabilities

Computation Service - Next Steps

- Computation Service
 - Deploying a "GENI Computation Service" as persistent infrastructure item would be convenient for user to get specific options to seed their Sliver Requests
 - could be based on modified IDC computation service
 - uses information from the Topology service
 - may not be necessary for large aggregates/control frameworks with their own advertisement aggregation and computation capabilities

Possible Activities before Next GEC

- Define AM API Extensions for negotiation
- Develop prototype Chain Mode Workflow
 - AM to AM messaging
 - Using AM API negotiation feature
- Develop enhanced Tree Mode Workflow using AM API negotiation feature
- Design and Develop prototype Topology Service

Possible Activities before Next GEC

- Design and Develop prototype Computation Service
- Add more Aggregates into the Stitching Capable AM infrastructure
- Deploy a "production" AM for ION

The End
Thank-you!

Stitching Path Global ID

- Stitching schema path element has a <globalId> element
- the intent of this element is to define an instance specific globally unique reference for a stitching path which cross multiple aggregates
- recommended format is aggregateid-serviceinstanceid, where
 - aggregateid (DNS style name)
 - serviceinstanceid: a number generated by the aggregate, which is unique within the scope of that aggregate.

Stitching Path Global ID

- construction of this element can be as follows:
 - chain mode: the first aggregate in the chain creates using its globally unique namespace
 - tree mode: the workflow manager needs to have globally unique namespace to manage (perhaps its domain id if has one) and can then create the globalId
 - hybrid mode: the workflow manager will create globalId for the tree sections of the path, and keep track of the globalId created by the chain sections, and associate them all with a single slice instantiation.