

# OnTimeMeasure Capabilities for GENI Experimenters

Prasad Calyam, Ph.D. (PI)

Paul Schopis, (Co-PI), Tony Zhu (Software Programmer), Yingxiao Xu (Software Programmer), Alex Berryman (REU Student)

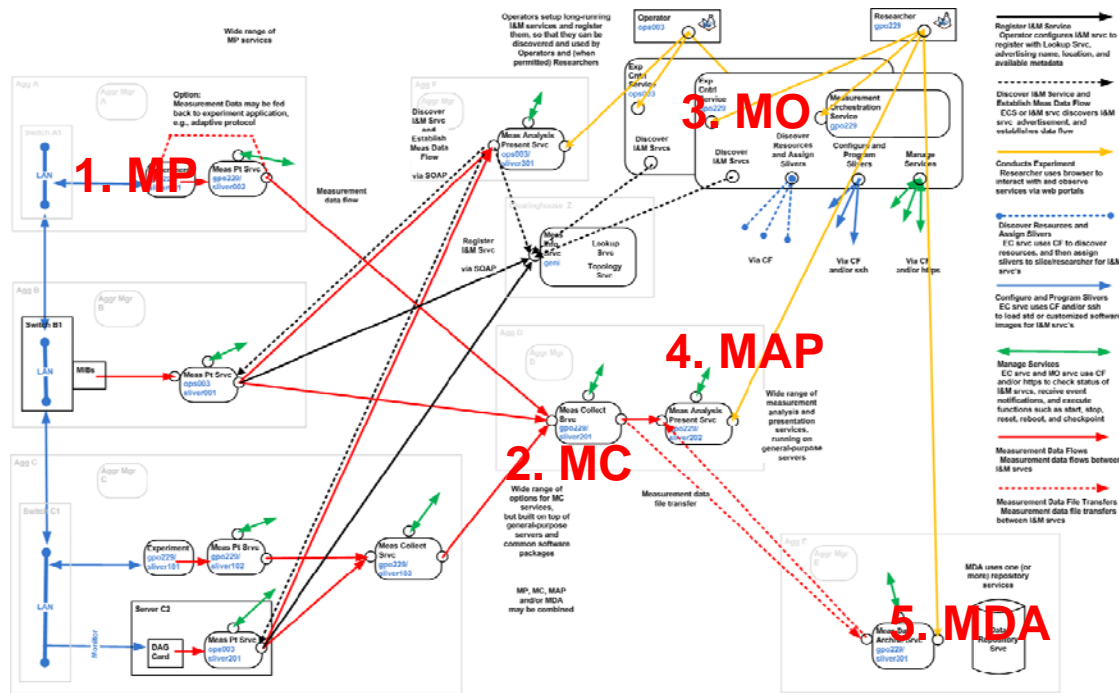
***GEC10 Tutorial***

*March 15<sup>th</sup> 2011*

# Experimenter Use Cases

- Experiment Environment Monitoring
  - Have I got the system and network resources I asked in my slice?
  - Is the experiment environment functioning as expected over my slice lifetime?
  - Can I trace my non-intuitive results in my experiment to a problem in the slice environment?
- Experiment **Instrumentation and Measurements (I&M)** Management
  - I want to collect active and passive measurements at hop, link, path and slice levels for my experiments...
  - I need measurements feedback with anomaly detection and forecasts to control my experiment progress...
  - I would like to have a measurement framework in my experiment to be configured once and be re-usable for subsequent experiment runs...
  - I want an archive of the experiment measurements collected for offline analysis or to share with colleagues...

# DIY: I&M service for Experimenters!



GENI I&M Architecture (Credit: Harry Mussman, GPO)

Sub-services
1. Measurement Point (MP)
2. Measurement Collection (MC)
3. Measurement Orchestrator (MO)
4. Measurement Analysis and Presentation (MAP)
5. Measurement Data Archive (MDA)

**Every experiment requires a custom I&M service; a GENI I&M service should remove most of that burden for an experimenter....**

# Topics of Discussion

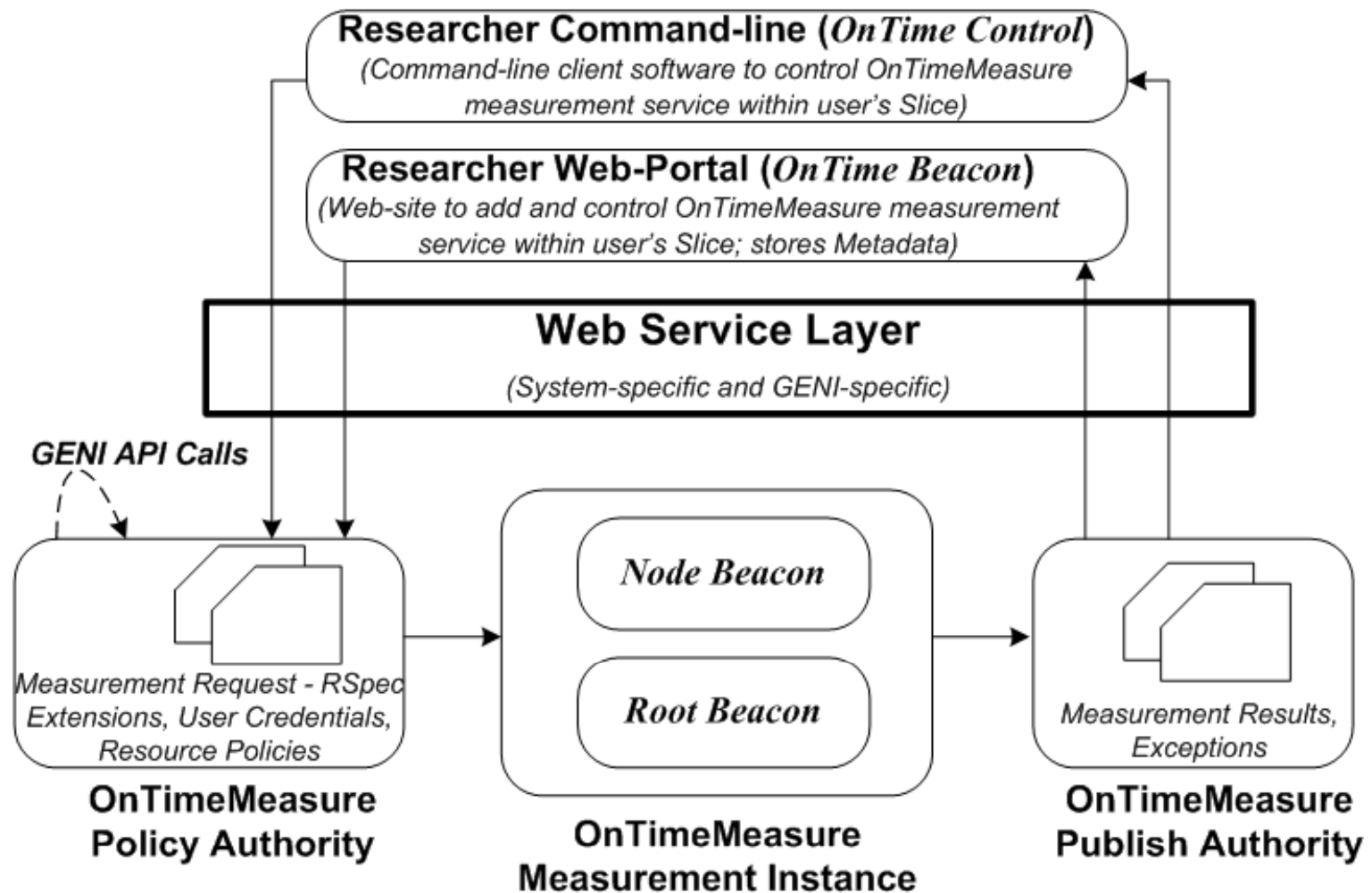
- OnTimeMeasure Project Overview
- OnTimeMeasure Architecture
- Software Components
- User Workflow
- Hands-on: Registration, Installation, Test
- GENI Integration and Interoperability
- Example Experiment Demo

# OnTimeMeasure Project Overview

GENI Project Wiki – <http://groups.geni.net/geni/wiki/OnTimeMeasure>

- Goal: Provide GENI community with capabilities for provisioning on-going and on-demand measurement requests
  - Used in forecasting, anomaly detection, and fault-location diagnosis in GENI experiments and GENI operations
- Outcomes:
  - Software to perform centralized and distributed measurement orchestration and provisioning of measurements
    - *Centralized orchestration* for continuous monitoring, persistent measurements storage and processed network measurement feeds
    - *Distributed orchestration* for on-demand (real-time) measurement requests without need for persistent measurements storage
  - Measurement service that enables users to utilize OnTimeMeasure software in GENI experiments
    - Registers users, slices, maintains meta-data, and allows user control of measurement service functions
      - Researcher Web-portal – <http://ontime.oar.net> for *interactive user control* of measurement service
      - Command-line tools for measurement service *control automation*

# OnTimeMeasure Architecture



# OnTimeMeasure Software Modules

- Customizable software <sup>[\*]</sup> developed at OSC/OARnet
- Two main modules installed **within a GENI experiment slice**
  - **Node Beacon**
    - Installs tools that measure network health metrics such as: route changes, delay, jitter, loss, bandwidth
      - TCP/UDP Iperf, Traceroute, Ping, Pathload, OWAMP, etc.
    - Runs measurements based on a schedule and outputs results
  - **Root Beacon**
    - Installs Apache, MySQL and other packages
    - Creates database tables and configuration files
    - Generates measurement schedules for node beacons
    - Collects data and provides dashboard visualization, statistical analysis (i.e., anomaly detection and weather forecasting) with alarm generation

NOTE: Typically, two or more Node Beacons and one Root Beacon need to be installed on separate dedicated servers in a slice; it is however possible to install Root Beacon on one of the Node Beacon servers

Download Node/Root Beacon Software – [http://ontime.oar.net/download/OnTimeMeasure\\_latest.php](http://ontime.oar.net/download/OnTimeMeasure_latest.php)

[\*] P. Calyam, C.-G.Lee, E. Ekici, M. Haffner, N. Howes, “Orchestrating Network-wide Active Measurements for Supporting Distributed Computing Applications”, *IEEE Transactions on Computers Journal (TC)*, 2006.

# OnTimeMeasure Software Modules (2)

- Two modules that are **external to the GENI experiment slice**

- **OnTime Control**

- Command-line interface client for users to login, manage slice information and control Node/Root Beacons within their slices in an automated manner

NOTE: Typically, users need to install OnTime Control software on their desktops along with other experiment workflow tools such as Gush

Download OnTime Control Software – [http://ontime.oar.net/download/OnTimeControl\\_latest.php](http://ontime.oar.net/download/OnTimeControl_latest.php)

- **OnTime Beacon**

- *User Features:* Web-interface client for users to login, manage slice information and interactively control Node/Root Beacons within their slices
- *Admin Features:* User management (approve new users, maintain user information), Meta-data information repository

NOTE: Typically, users do **NOT** need to install OnTime Beacon software and should use the instance hosted at - <http://ontime.oar.net>; GENI aggregate providers who wish to have a dedicated OnTime Beacon for their users can install their own instance of OnTime Beacon on a dedicated server

Download OnTime Beacon Software – [http://ontime.oar.net/download/OnTimeBeacon\\_latest.php](http://ontime.oar.net/download/OnTimeBeacon_latest.php)

**NOT FOR EXPERIMENTERS!**

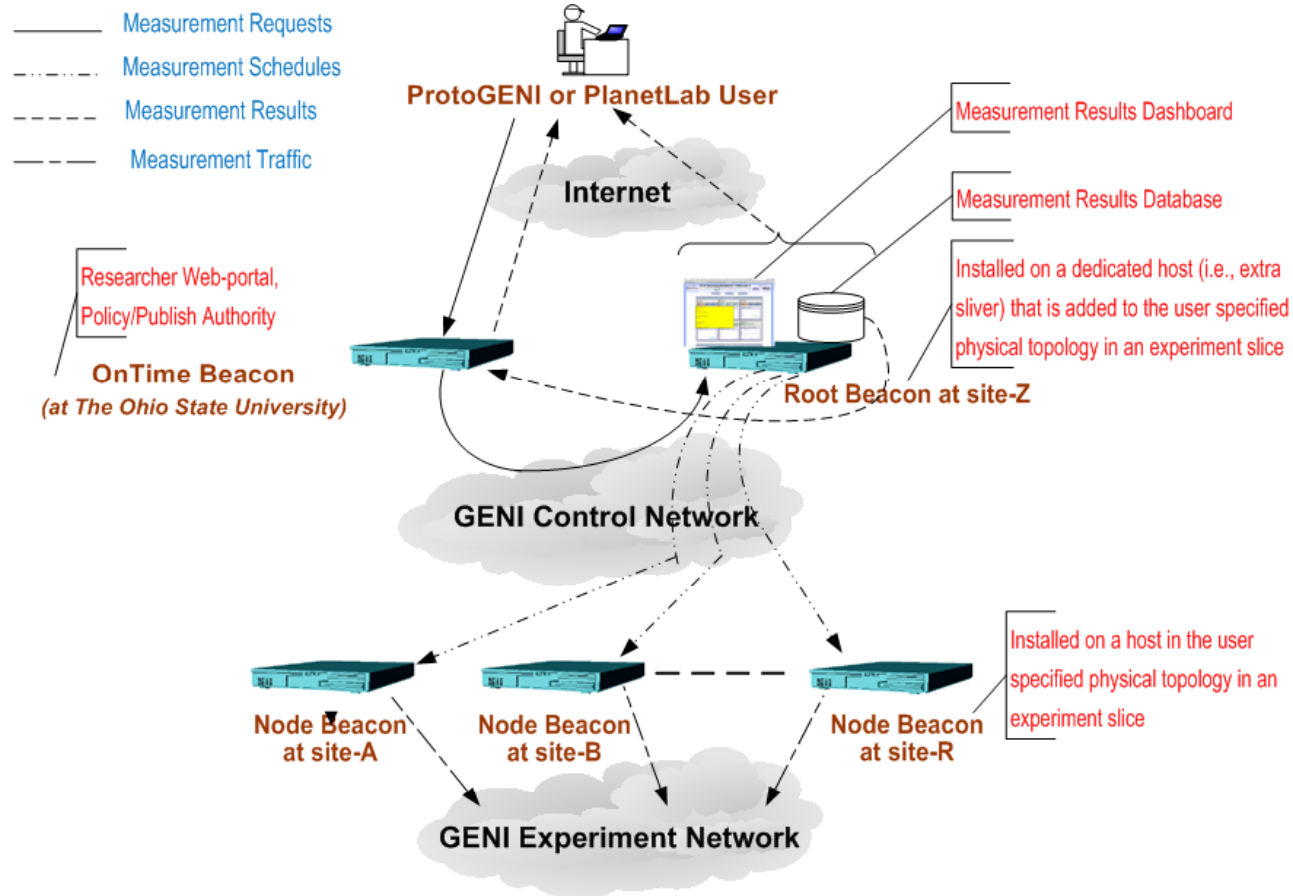


# Mapping to GIMA Services

Service Name	Functions	Module
Measurement Orchestration (MO)	Part of Experiment Control service, uses a language to orchestrate I&M services	Root Beacon
Measurement Point (MP)	Instrumentation that taps into a network and/or systems, links and/or nodes, to capture measurement data and format it using a standardized schema	Node Beacon
Measurement Collection (MC)	Programmable systems that collect, combine, transform and cache measurement data	Root Beacon
Measurement Analysis and Presentation (MAP)	Programmable systems that analyze and then present measurement data	Root Beacon
Measurement Data Archive (MDA)	Measurement data repository, index and portal	Root Beacon

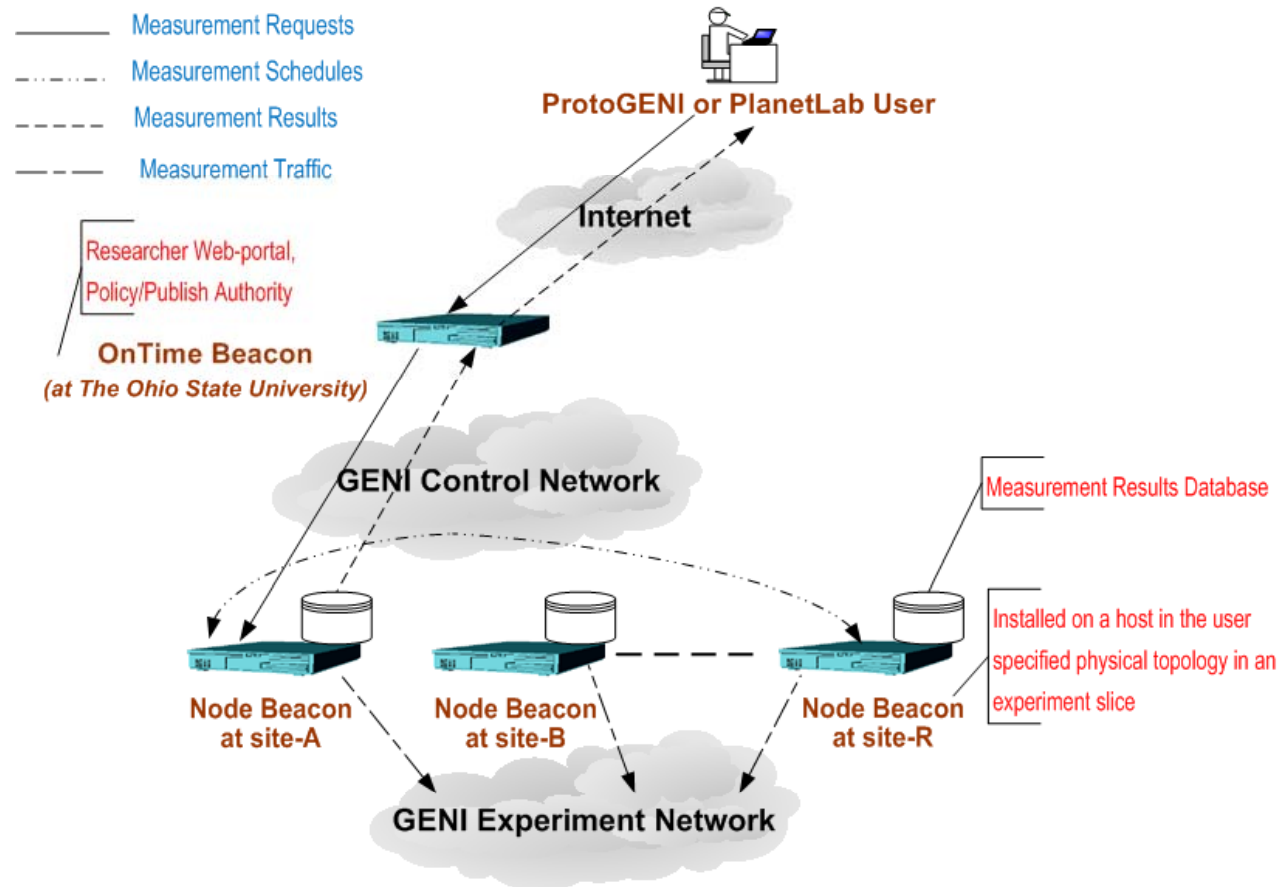
- For details of service requirements, see GENI Instrumentation and Measurement Architecture (GIMA) document - <http://groups.geni.net/geni/wiki/GeniInstMeas>

# Centralized Orchestration



- Centralized scheduling for continuous monitoring, persistent measurements storage and processed network measurement feeds
  - Useful for “network weathermaps” and long-standing experiments with advanced measurement analysis capabilities

# Distributed Orchestration



- Distributed scheduling for on-demand (real-time) measurement requests without need for persistent measurements storage
  - Useful for users or helper apps needing one-off or occasional raw measurement tool outputs

# Measurement Service Capabilities

- Measurement request handling
  - Path-based active measurements (e.g., delay, jitter, loss, throughput, route changes)
    - Measurement topology (e.g., full-mesh, tree, hybrid)
    - Sampling requirements (e.g., periodic, stratified random, random, adaptive)
  - Host-based system performance (e.g., CPU, memory)
- Enforce policies for measurements scheduling
  - Measurement level restrictions for probing tools (e.g., allowable measurement bandwidth and measurement flow duration for different nodes/paths/user-roles)
- Provide raw and processed measurement
  - Measurements provisioning interfaces (e.g., raw data or graph output to human/component-service, processed output of multiple tools to a dashboard)
  - Measurement use context hooks (e.g., verify network path(s) performance in a new slice, network-awareness in an experiment to develop a novel network control scheme)
  - OnTimeMeasure instance metadata (e.g., needed by GMOC, NetKarma)

# User Workflow

## (ProtoGENI or PlanetLab Aggregates)

1. User creates an experiment slice using ProtoGENI or PlanetLab control framework tools
  - For ProtoGENI slice creation, see - <http://groups.geni.net/geni/wiki/OnTime-Install>
  - For PlanetLab slice creation, see - <http://groups.geni.net/geni/wiki/OTM-PlanetLabInstall>
2. User registers at the “Researcher Web-Portal” (<http://ontime.oar.net>)
3. User installs OnTimeMeasure measurement instance in experiment slice
  - Slice RSpec should include reservation of any required measurement resources
  - Node/Root Beacons need to be installed as slivers based on the instructions provided in Step-1 for the specific aggregate
4. Valid login to the “Researcher Web-Portal” allows user to:
  - A. Provide slice RSpec information for the measurement service
  - B. Interact with the measurement service in user’s experiment slice
    - i. Submit measurement requests
    - ii. Control the measurement service
    - iii. Query measurement data

View OnTimeMeasure Demo Videos at – <http://ontime.oar.net/demo>

# Screenshots

## Centralized Service Control

**1 2 3 Measurement Request Submission**

Please review and submit your measurement request to the OnTimeMeasure-GENI measure service.

**STEP 1 Resource Setup**  
Status: Complete ✓ [Modify](#)

Measurement topology architecture selected is: Centralized  
Measurement topology is as follows:  
Slice name: ontime\_m1  
Root Beacon: KANS(64.57.23.133)  
Node Beacon 1: WASH(64.57.23.165)  
Node Beacon 2: SALT(64.57.23.149)

**STEP 2 Request Specification**  
Status: Complete ✓ [Modify](#)

The tasks added to the measurement service are as follows:

Source	Destination	Metric	Pattern
WASH(64.57.23.165)	SALT(64.57.23.149)	Round-trip Delay	Periodic
WASH(64.57.23.165)	SALT(64.57.23.149)	Throughput	Periodic
WASH(64.57.23.165)	SALT(64.57.23.149)	Loss	Periodic
SALT(64.57.23.149)	WASH(64.57.23.165)	Round-trip Delay	Periodic
WASH(64.57.23.165)	SALT(64.57.23.149)	Jitter	Periodic

**STEP 3 Request Submission**  
Submit the request to initialize the measurement service. [Submit Request](#)

## Centralized Measurement Request

**Service Control**

[Start](#) Initiates communications between Root Beacons and/or Node Beacons to start the active measurements data collection

[Stop](#) Terminates communications between Root Beacons and/or Node Beacons to stop the active measurements data collection

Status: ▶ Running: Measurements are being collected in the experiment slice.

[Update](#) Refreshes the service status notification; can be used to verify whether or not any of the service components are functioning as expected

The status of the service components are as follows:

Component	Status
Slice Accessibility	OK ✓
Root Beacon Scheduler	OK ✓
Node and Root Beacon Communications	OK ✓
Measurements Data Collector	OK ✓
Analysis and Publish Authority	OK ✓
Measurement Data Visualization	OK ✓

[Proceed to query measurements data collected: Query Data](#)

**Measurement Query**

Please select from the following query options:

User: Centralized Demo Usr Metric: Round-trip Delay  
Start time: 2010-05-01 23:00:00 Source: WASH(64.57.23.165)  
End time: 2010-05-31 12:00:00 Destination: SALT(64.57.23.149)

Results type:  
 Raw Files  
 Time Series  
 Time Series with Anomalies  
 Time Series with Forecasts

[Submit Query](#)

Status: Measurement query was successful. [View Result](#)

View real-time graphs of measurement data: [View Dashboard](#)

## Centralized Measurement Query

View OnTimeMeasure Demo Videos at – <http://ontime.oar.net/demo>

# Screenshots (2)

Researcher Web-Portal

**Measurement Result**

Measurement completed, please see below. Download raw files: WASH

```
[ 15] local 64.57.23.149 port 5001 connected with 64.57.23.165 port 5001
[ 15] 0.0-10.0 sec 494 MBytes 414 Mbits/sec
[ 15] MSS size 1448 bytes (MTU 1500 bytes, ethernet)
bwctl: stop_exec: 3484246228.794817

RECEIVER END

[WASH]$ bwctl -c 64.57.23.149 -f m -u -b 768k -a 10

RECEIVER START
bwctl: exec_line: iperf -B 64.57.23.149 -s -f m -m -p 5001 -u -t 10
bwctl: start_tool: 3484246254.590082

-----
Server listening on UDP port 5001
Binding to local address 64.57.23.149
Receiving 1470 byte datagrams
UDP buffer size: 0.11 MByte (default)

-----
[ 8] local 64.57.23.149 port 5001 connected with 64.57.23.165 port 5001
[ 8] 0.0-10.0 sec 0.92 MBytes 0.77 Mbits/sec 0.011 ms 0/ 655 (0%)
bwctl: stop_exec: 3484246280.789142

RECEIVER END
```

Distributed Measurement Result

OnTimeMeasure-GENI Measurement Request Creation - Mozilla Firefox

http://ontime.oar.net/MeasurementRequest.php

OnTimeMeasure-GENI | Mea...

CONTACT | LOGIN

OnTimeMeasure 1010001

geni Exploring Networks of the Future

NAVIGATION

MEASUREMENT REQUEST Creation

To submit a measurement request to the OnTimeMeasure-GENI measurement service, please follow a 3-step process:

STEP 1 Resource Setup: You first need to setup a ProtoGENI experiment slice along with Node and Root Beacons

STEP 2 Request Specification: You next need to specify on-going measurement tasks between Node Beacons

STEP 3 Request Submission: You lastly need to review and submit your measurement request to initialize the measurement service

Proceed to Step-1: [Setup Resources](#)

Information

For details of the measurement request creation, please refer to the OnTimeMeasure Design Document.

Ohio | CAAT | S&T

Supported by the National Science Foundation

Mozilla Firefox

http://ontime.oar.net:8080/ov/

OnTimeMeasure-GENI | Mea... | http://192.168.63.8080/ov/

WASH\_AND\_UTAH\_Path\_Over

WASH\_AND\_UTAH\_Path\_Over

WASH\_AND\_UTAH\_Path\_Over

WASH\_AND\_UTAH\_Path\_Throughput

User Customizable Dashboard

## OnTimeMeasure Integration w/ other GENI Projects

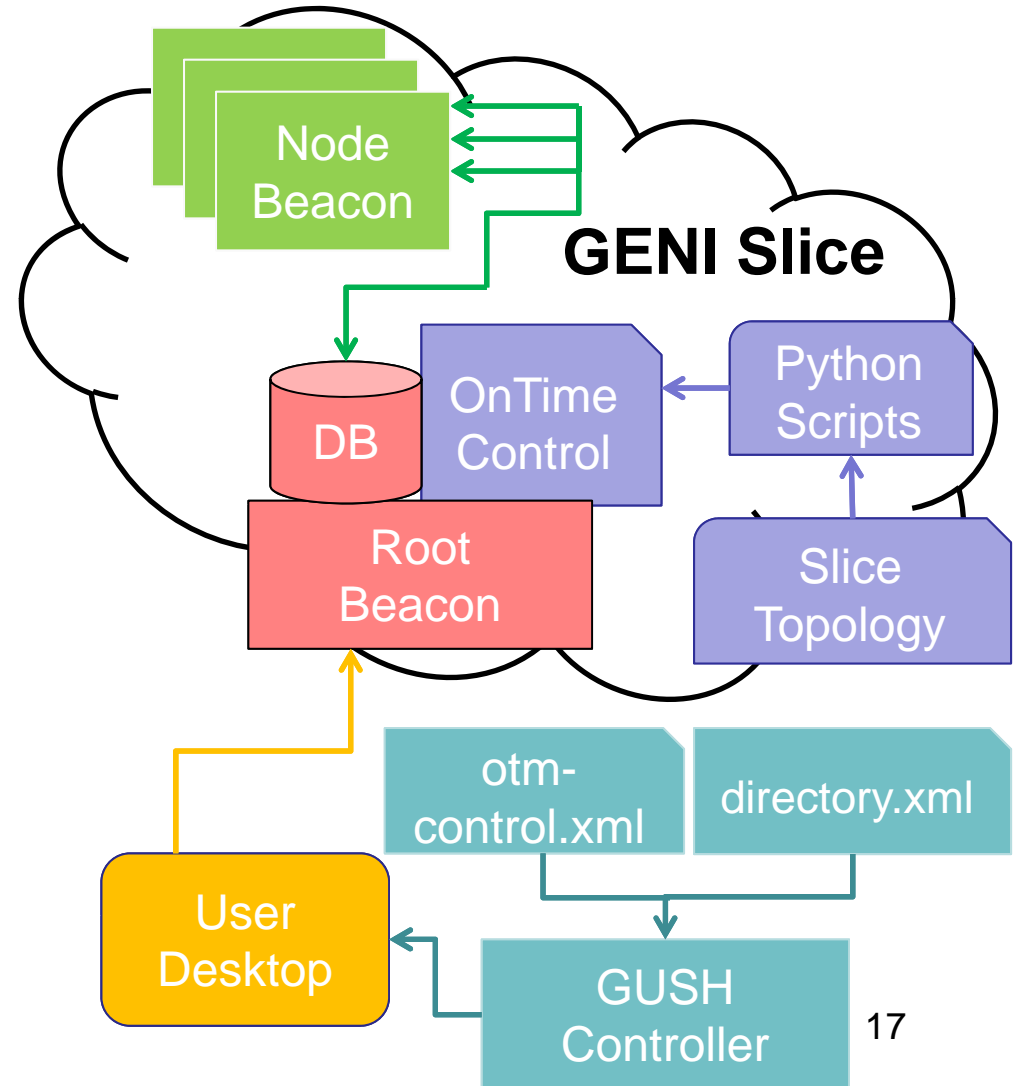
- Not all Experimenter use cases can be supported by OnTimeMeasure alone; integration satisfies more use cases
  - Some use cases will require development based on ***your*** feedback!
- Integrated Projects:
  - OnTimeMeasure-ProtoGENI, OnTimeMeasure-PlanetLab
    - I&M service for GENI aggregate users
  - OnTimeMeasure-Gush
    - I&M service control through Experimenter workflow tool
  - OnTimeMeasure<sup>\*</sup>-INSTOOLS<sup>\*\*</sup>
    - Both active<sup>\*</sup> and passive<sup>\*\*</sup> measurements in experiment slice
  - OnTimeMeasure-CRON
    - I&M service for 10Gbps network path experiments



# Integration with Gush

See details at - <http://groups.geni.net/geni/wiki/OTM-Gush>

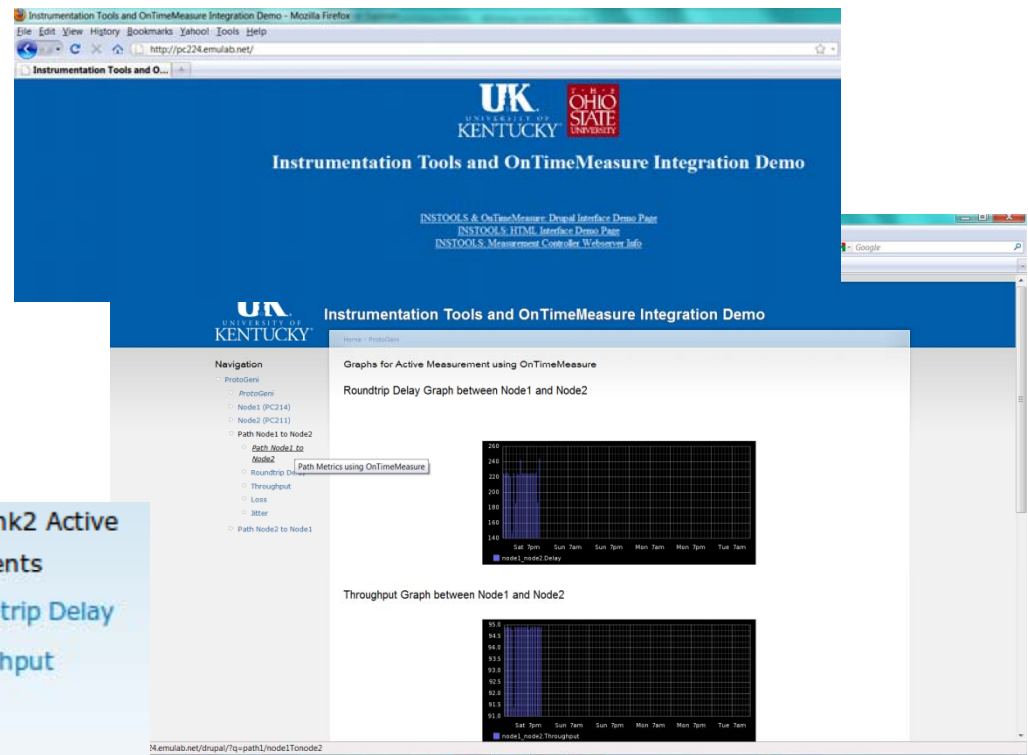
- OnTime Control package resides on Root/Node Beacon; Gush is installed on user desktop
- Set of XML experiment files in the Gush format interact with the OnTime Control Python scripts
- Gush topology description (directory.xml) is passed to the OnTime Control
- Output of OnTime Control is a URL of a text/plot file containing the results of the measurement



# Integration with Instrumentation Tools

See details at - <http://groups.geni.net/geni/wiki/OTM-InsToolsDemo>

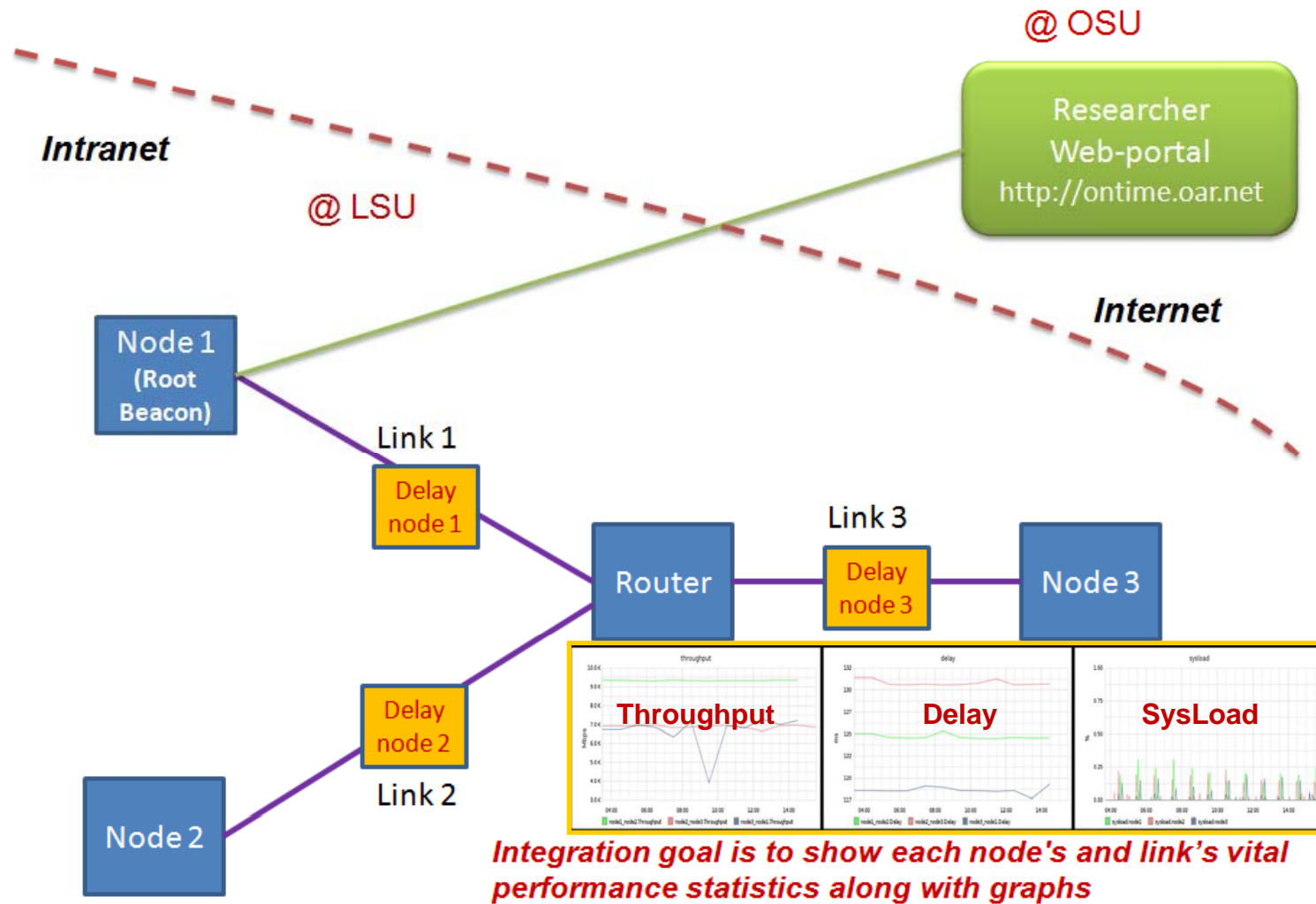
- Integration enables both INSTOOLS and OnTimeMeasure software running within same slice
- Set of INSTOOLS' Drupal configurations have to be modified
- Results and graphs of OnTimeMeasure become accessible through INSTOOLS web-interface



- ▾ Link1 to Link2 Active Measurements
  - Round-trip Delay
  - Throughput
  - Loss
  - Jitter
- Link2 to Link1 Active Measurements

# Integration with CRON 10Gbps Testbed

See details at - <http://groups.geni.net/geni/wiki/OTM-CRONInstall>

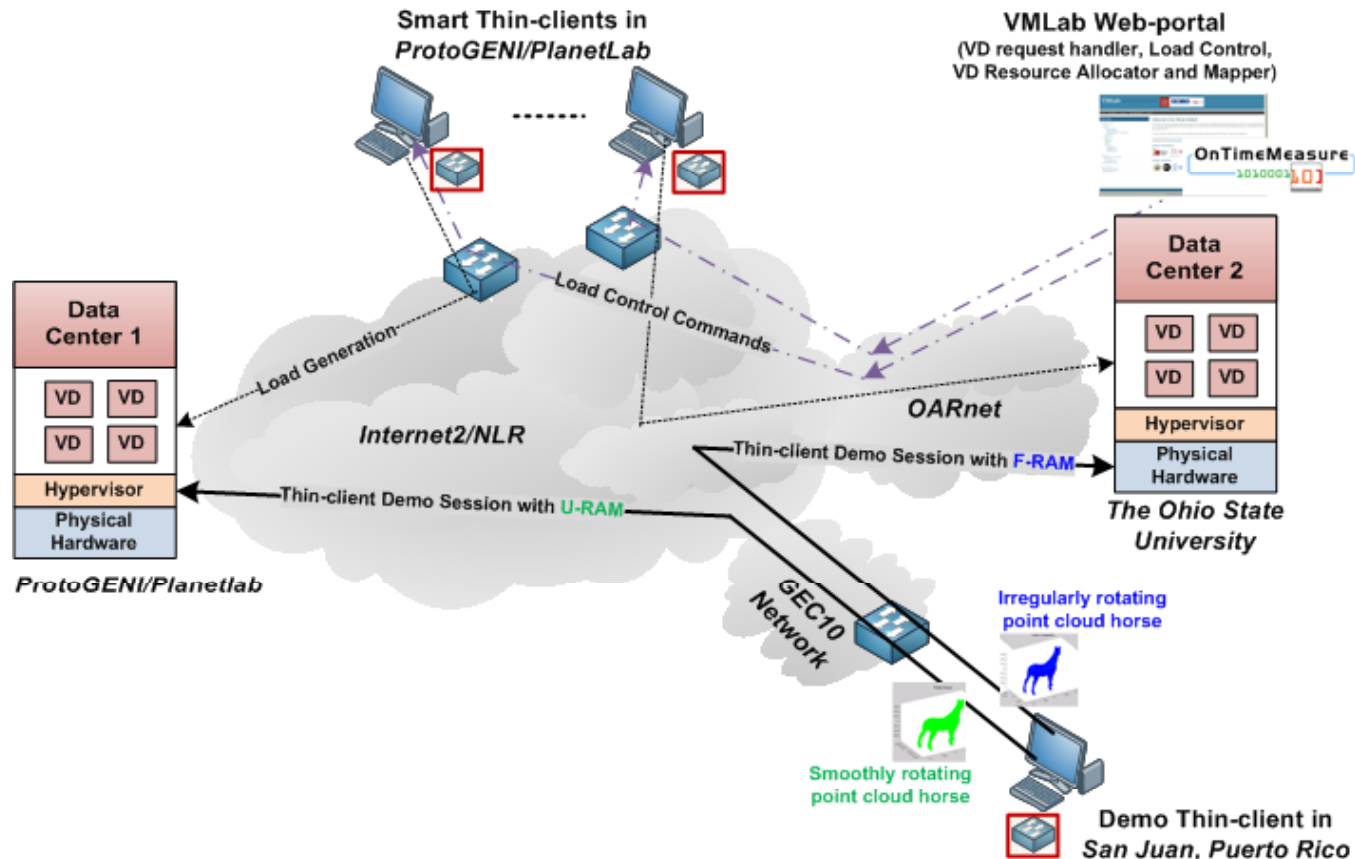


*Integration goal is to show each node's and link's vital performance statistics along with graphs*

## Steps to integrate new metrics custom to your experiment...

- Any active or passive *measurement data source* can be integrated into OnTimeMeasure framework
  - Instantaneous measurement, Measurement time series
  - E.g., OpenFlow, SNMP, *Tools in VMware (today's demo)*, ...
- Integration steps
  - Write a new tool wrapper for Node Beacon (MP Service)
  - Modify measurement collector script and dB schema for Root Beacon (MC Service)
  - Modify scheduler configuration for Root Beacon (MO Service)
  - Request us (Email: [pcalyam@osc.edu](mailto:pcalyam@osc.edu)) to modify analysis and presentation scripts for OnTime Beacon/Control (MAP Service)

# Example Experiment Demo



- Experiment on dynamic resource allocation in virtual desktop clouds
- OnTimeMeasure provides on ProtoGENI and PlanetLab:
  - Active measurements of network resources
  - VMware measurements (via APIs) of system resources

# Conclusion

- OnTimeMeasure measurement service is now available to experimenters
  - Please register at <http://ontime.oar.net>
- Development is on-going, but the core measurement service capabilities are ready for testing and use
- We are seeking ideas to use OnTimeMeasure in GENI experiments
- We are looking to integrate OnTimeMeasure with other GENI software systems

Thank you for your attention ! 😊

*This material is based upon work supported by the National Science Foundation under Grant No. CNS-0940805. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of BBN Technologies, Corp., the GENI Project Office, or the National Science Foundation.*