GEMINI: A GENI Instrumentation and Measurement System

Overview and Status

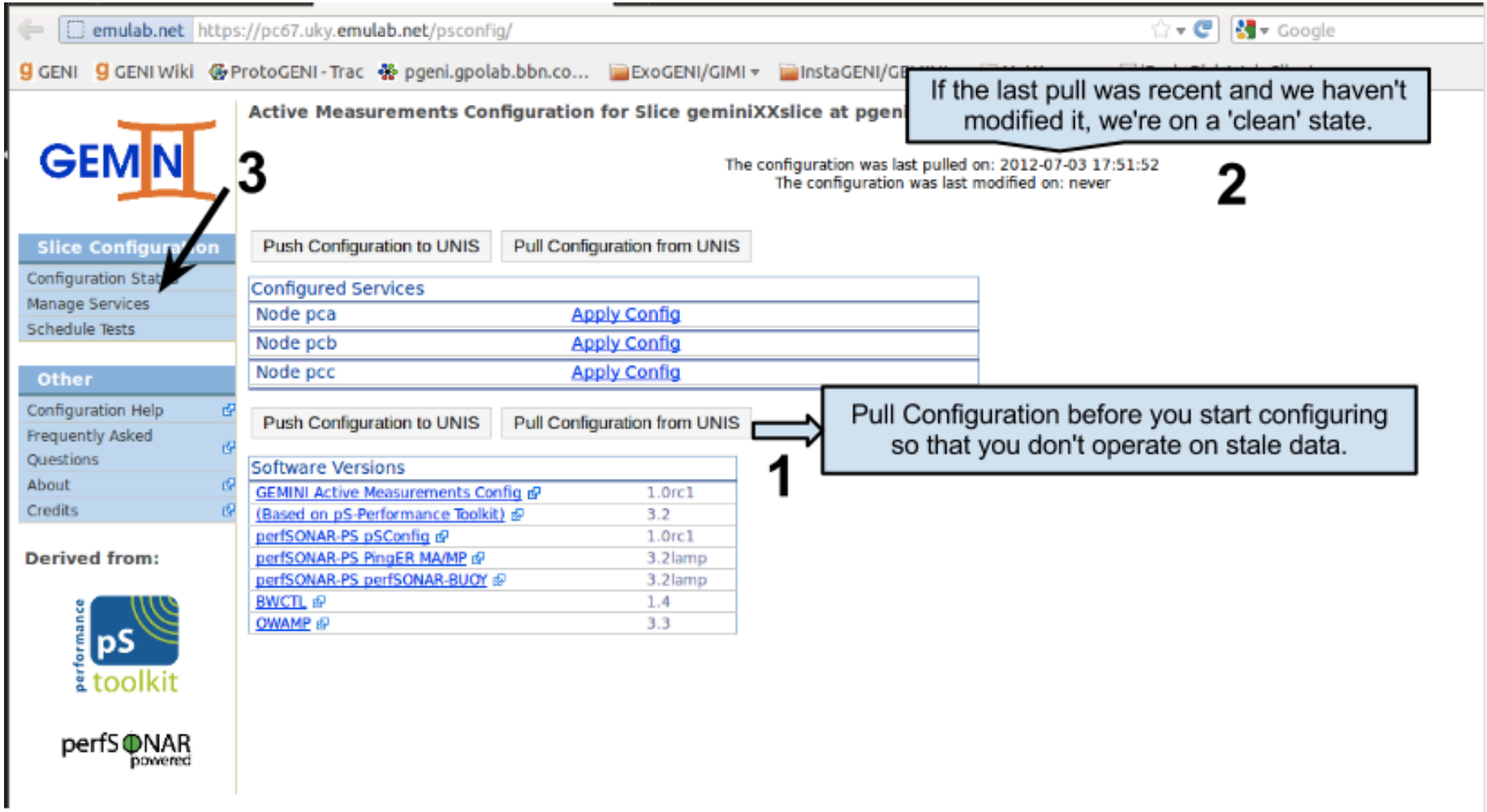
Current Status
(Tutorial today 1:00-5:00)

To have an instrumented slice:

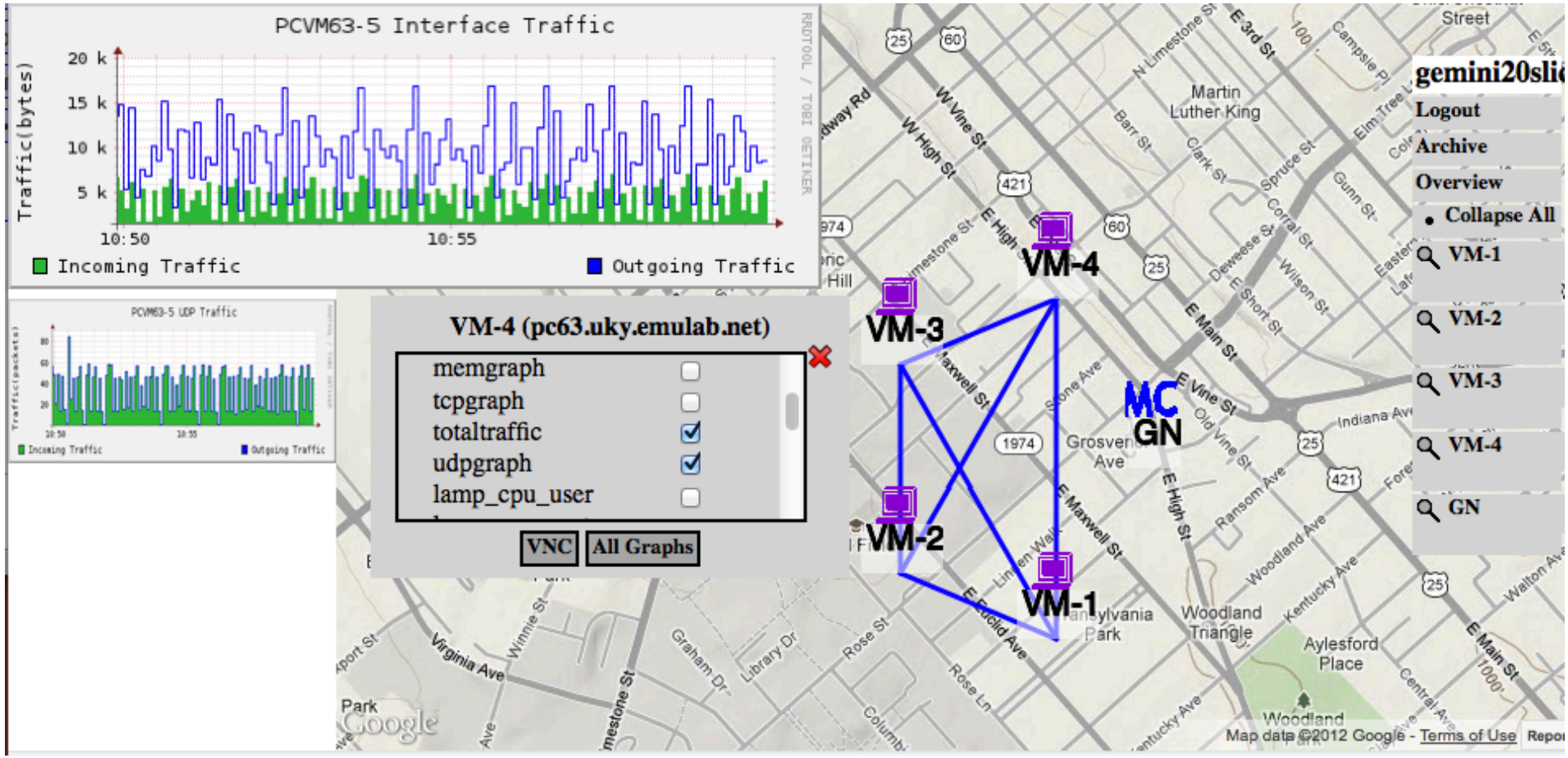
1. Run the instrumentize.py script with appropriate credentials and slice name

2. Configure activities through the portal and configuration interfaces

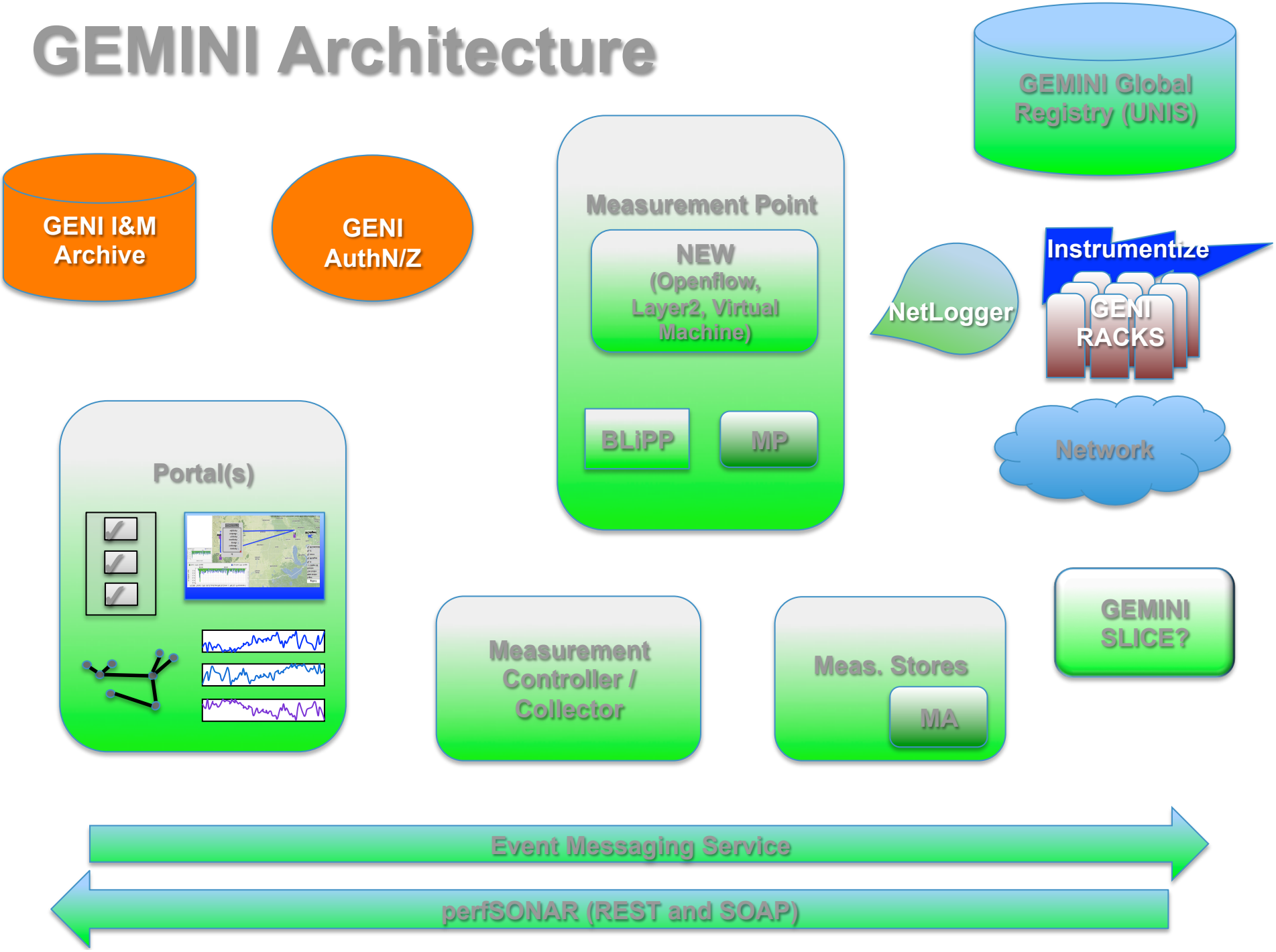
Configuration Interface (via UNIS)



Portal for visualizing resources / data



GEMINI Architecture



What you get:

The merging of INSTOOLS and LAMP (perfSONAR for GENI) together

Passive Metrics (INSTOOLS+LAMP)

Active Measurements (LAMP)

Archiving (INSTOOLS)

User workspace for ease of setup and use (New)

Unified Network Information Service (UNIS) running as a service with other uses (stitching talk at 1:00)

Current Capabilities

Active Network Metrics

Bandwidth (bwctl/iperf)

One-way latency (OWAMP)

Latency (Pinger)

Host Passive Metrics

CPU and Memory Utilization

TCP, UDP, ICMP Traffic counters

IP Traffic

Host Network Info (Routing tables etc.)

Host Process Info

Any performance metric from /proc

Passive Network Metrics

In/Out Octets

In/Out Errors

Basic OpenFlow monitoring

Archiving

IRODS

CNRI

GMOC

Issues with old version

- Multiple interfaces
- Configuration confusion
- perfSONAR services are specialized and hard to extend, many services for different metrics.
- Different stores/databases for different metrics.
- Old/Clunky SOAP and XML
- Inefficient use of SNMP on hosts

New Architecture Features

- New Measurement Store (MS)
  - NoSQL data store
  - Stores all metrics in a common (JSON) format
  - REST API
- New UNIS
  - Also NoSQL/JSON
  - REST API
- BLiPP (A General Measurement Point)
  - Collects many host metrics and sends to MS
  - Registers metadata and config/status with UNIS

Application level Monitoring

- BliPP and NetLogger (from LBL)
- NetLogger
  - Provides an API in a variety of languages
  - Emit perfSONAR-compatible events
  - Dynamic instrumentation of system calls with LD_PRELOAD
  - Details available after GEC15

Campus Infrastructure Monitoring

- OpenFlow monitoring
- Using OF as a monitoring enabler
- Monitoring of the OF infrastructure itself
- Entities that aren't GENI resources, but which are used by GENI in some way
- Active probes for dynamic resources
  - Circuit "acceptance testing"

Next Steps

- Integration of ABAC support (prototyped)
- GENI OneStop interface
  - Experiment and measurement orchestration and management
- Application metrics with NetLogger
- Configurable, in-line, on-line summarization and data reduction
- Cooperation with CFs/rack operators/GMOC
  - Common metrics across infrastructures
  - Reduction of duplicate measurements
  - Common collection tools

Goals:

- High rate, low impact monitoring
  - We want the ability to have very fine grained data.
- Single consistent interface
  - We aim to simplify configuration while still leaving advanced options open
- Fewer moving parts
  - Consolidated MS
  - UNIS unifies topology, configuration, and metadata
- Easy processing, retrieval, and visualization of data

Discussion Topics

- What do users and operators want?
- AuthN/Z
- Measurement should be a first class entity
  - Standard images should have embedded monitoring!
- Ongoing measurements need to be addressed
  - Access to hypervisor, or provide code
  - AuthN/Z to initiate new measurement activities
- Global naming, global information service

Questions?

Thanks for your attention!