GENI Monitoring Collector

*GEC 21*

October 22, 2014
Wednesday, 1.30pm - 3.30pm

History
->
Current State

Central Monitoring/ Status/Utilization/Reporting

- Provide status reporting for all GENI slices and slivers within and between GENI racks.

- Provide "live" measurement information on end-to-end GENI slices via a web/API interface.

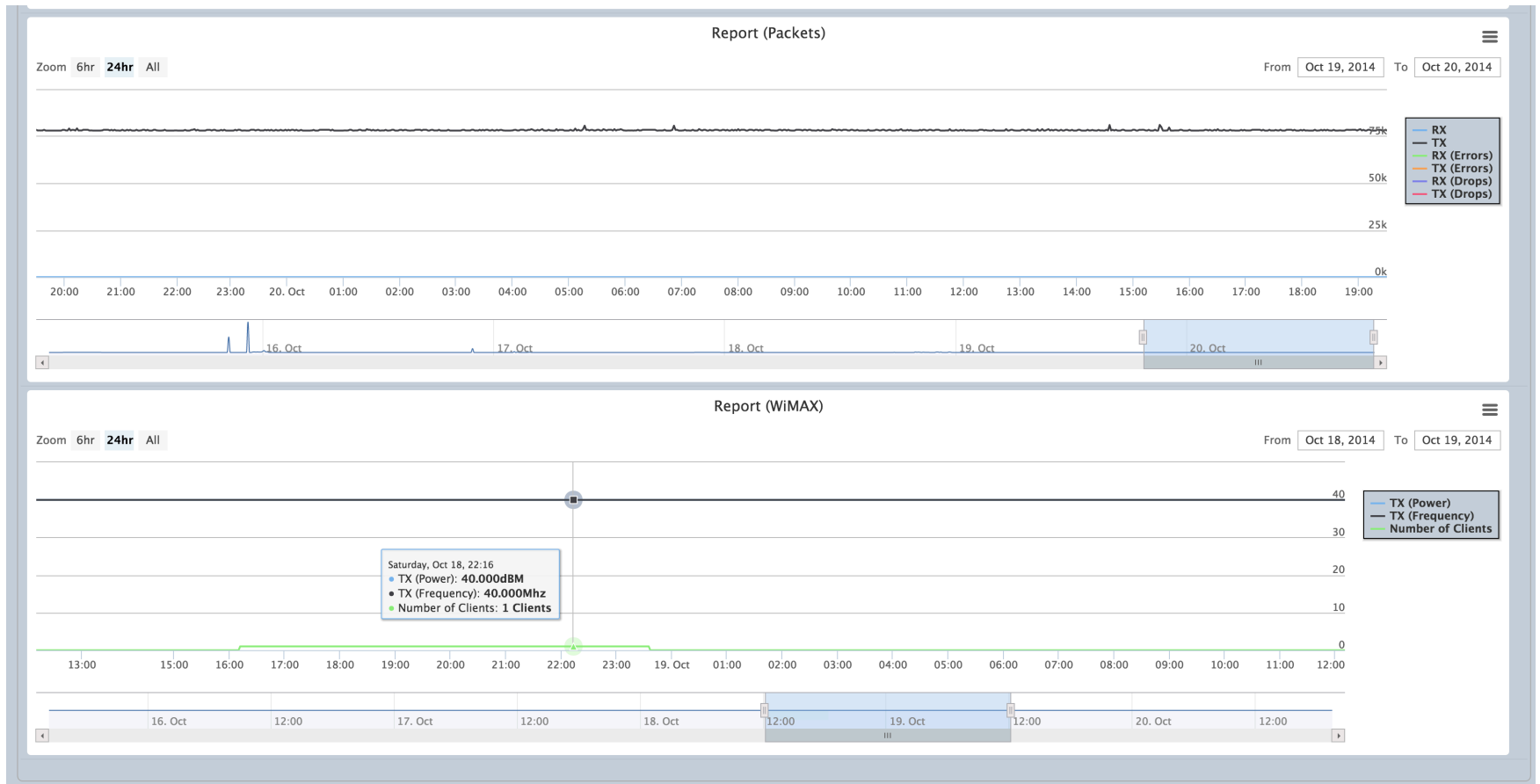- Provide archival storage for GENI rack measurement data.

WEB/API

CENTRAL MONITORING

UTILIZATION STORE

STATUS STORE
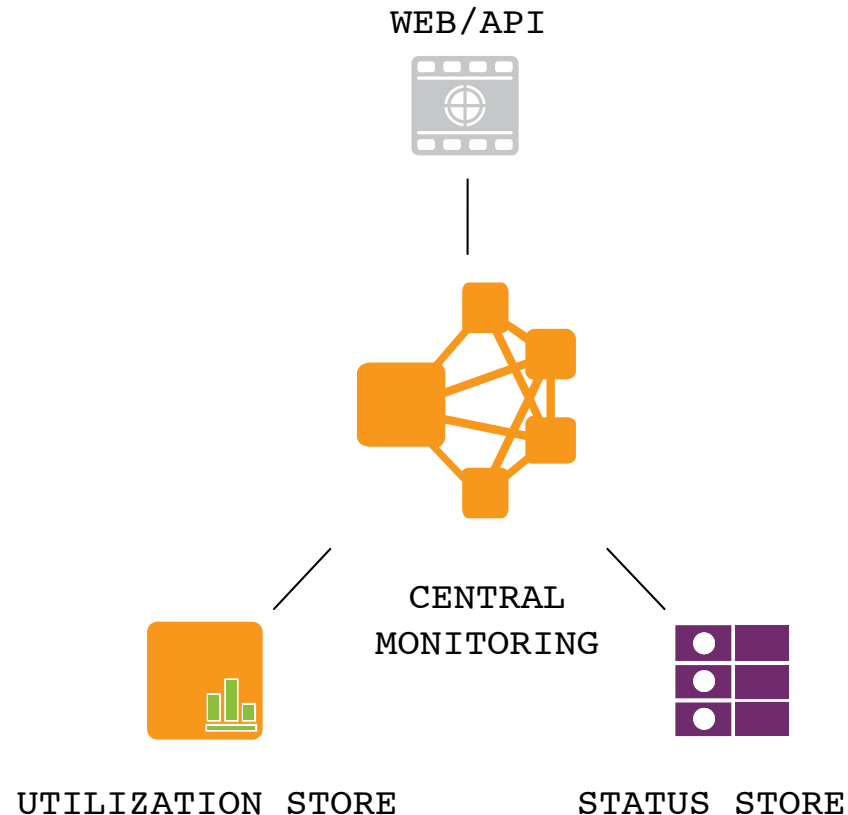
Current Architecture
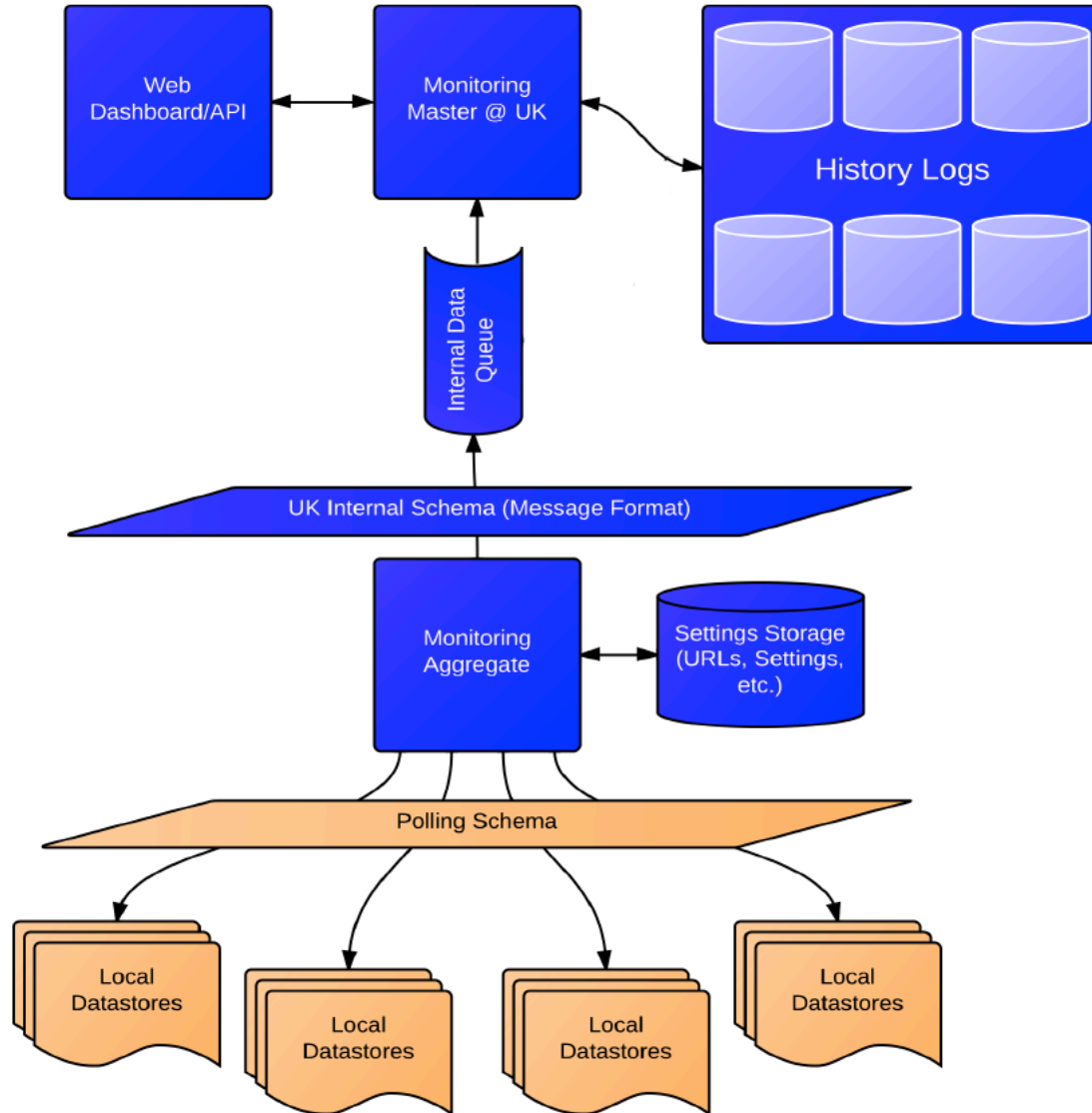
Monitoring Collector Demo

- Updated Collector to handle both v1 and v2 of the Monitoring Schema as well as new WiMAX data.

Slice Topology View Demo

Historic Usage Aggregate Usage Demo
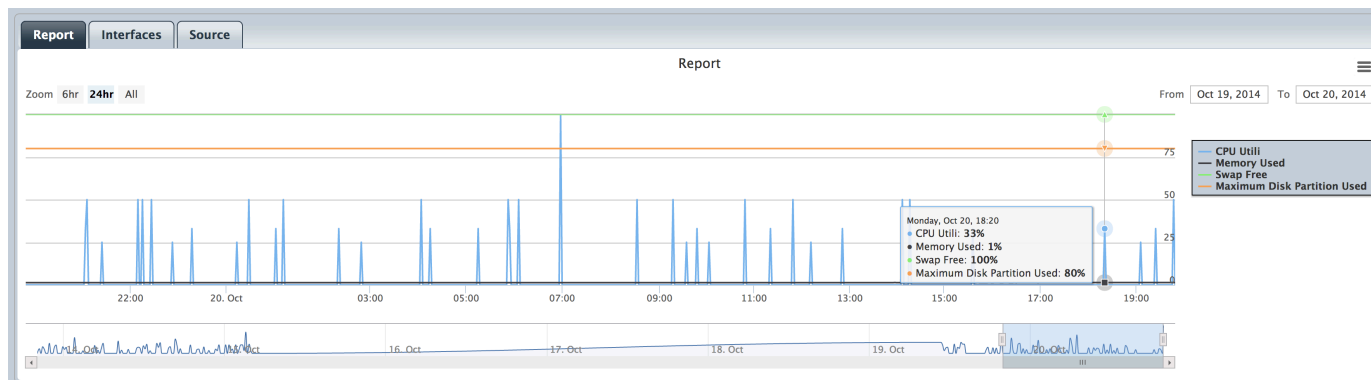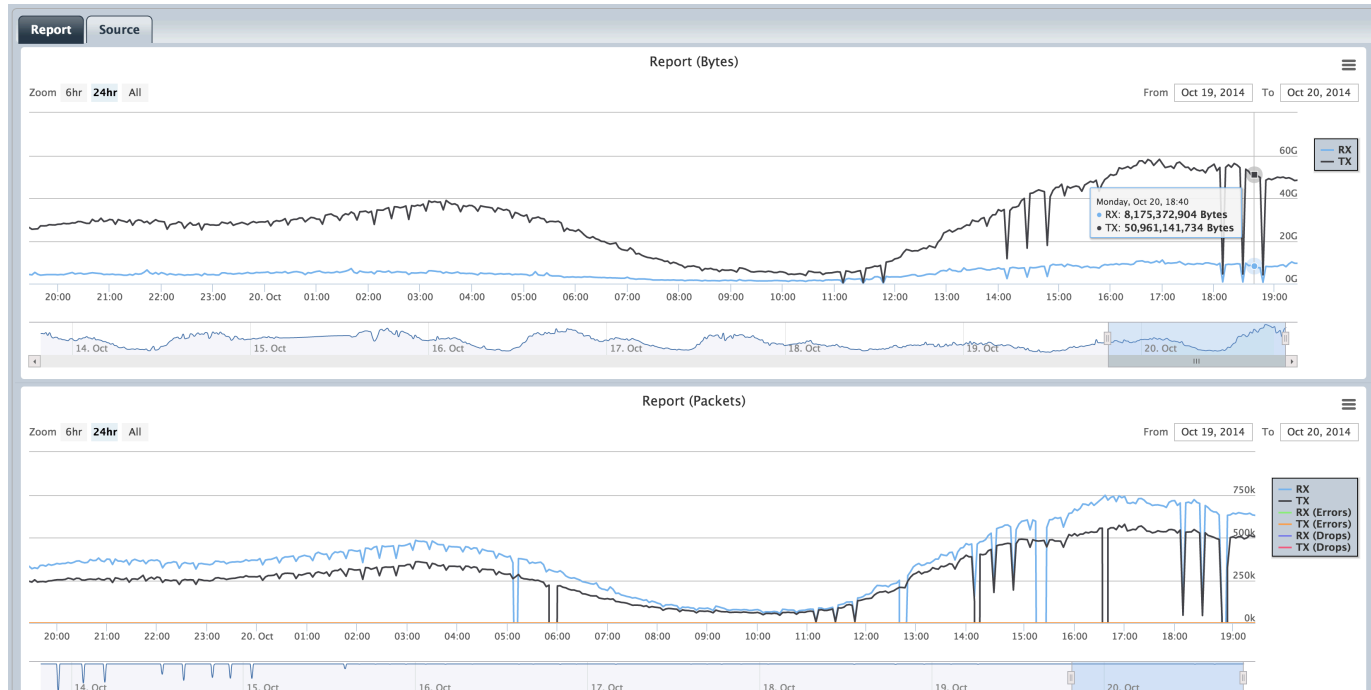
For Next GEC

Next GEC Architecture
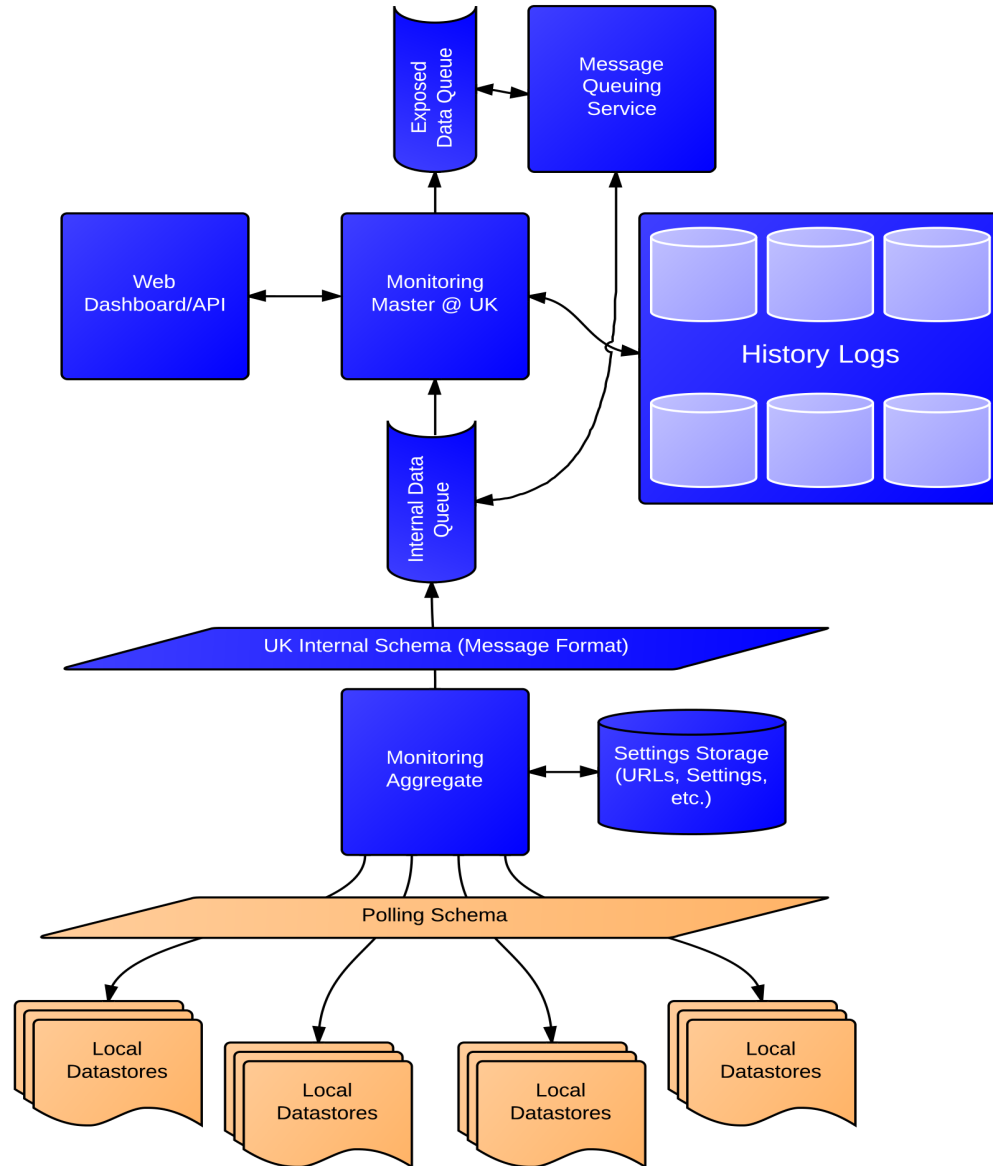
Accessing Historic Data
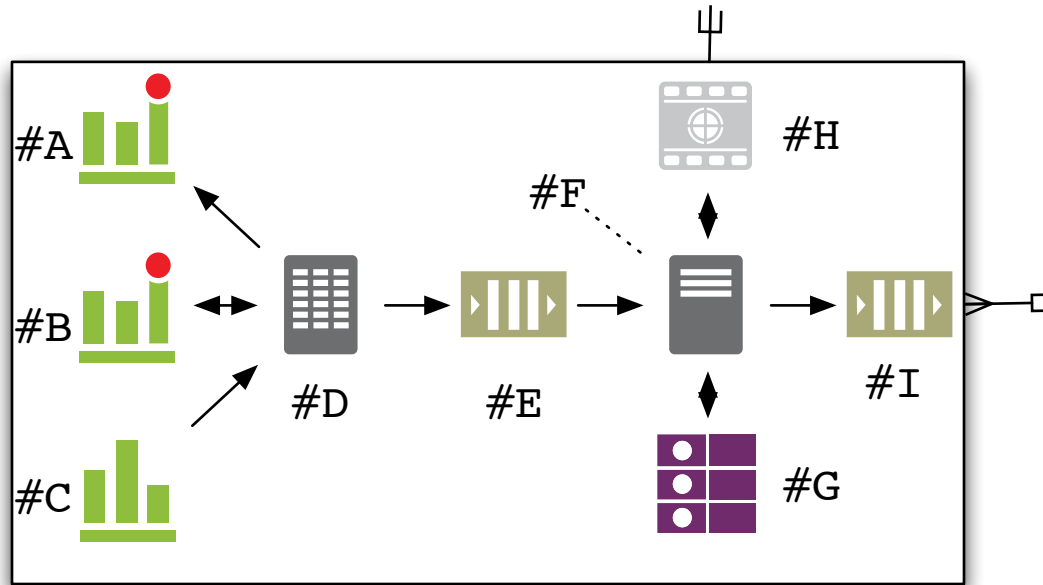
#H: A RESTful API / Web layer will be exposed for accessing data on the *State Controller.* Data exposed will include resource states for resources and causes for the state transitions.

#I: Queuing Publish/Subscribe system that allows for pattern-based (matching) subscriptions. State transition and utilization will be emitted on this queue. *Will be used by central monitoring.

GraphDB

-Keep state information in graph DB.
-Graph operations on monitoring relations.
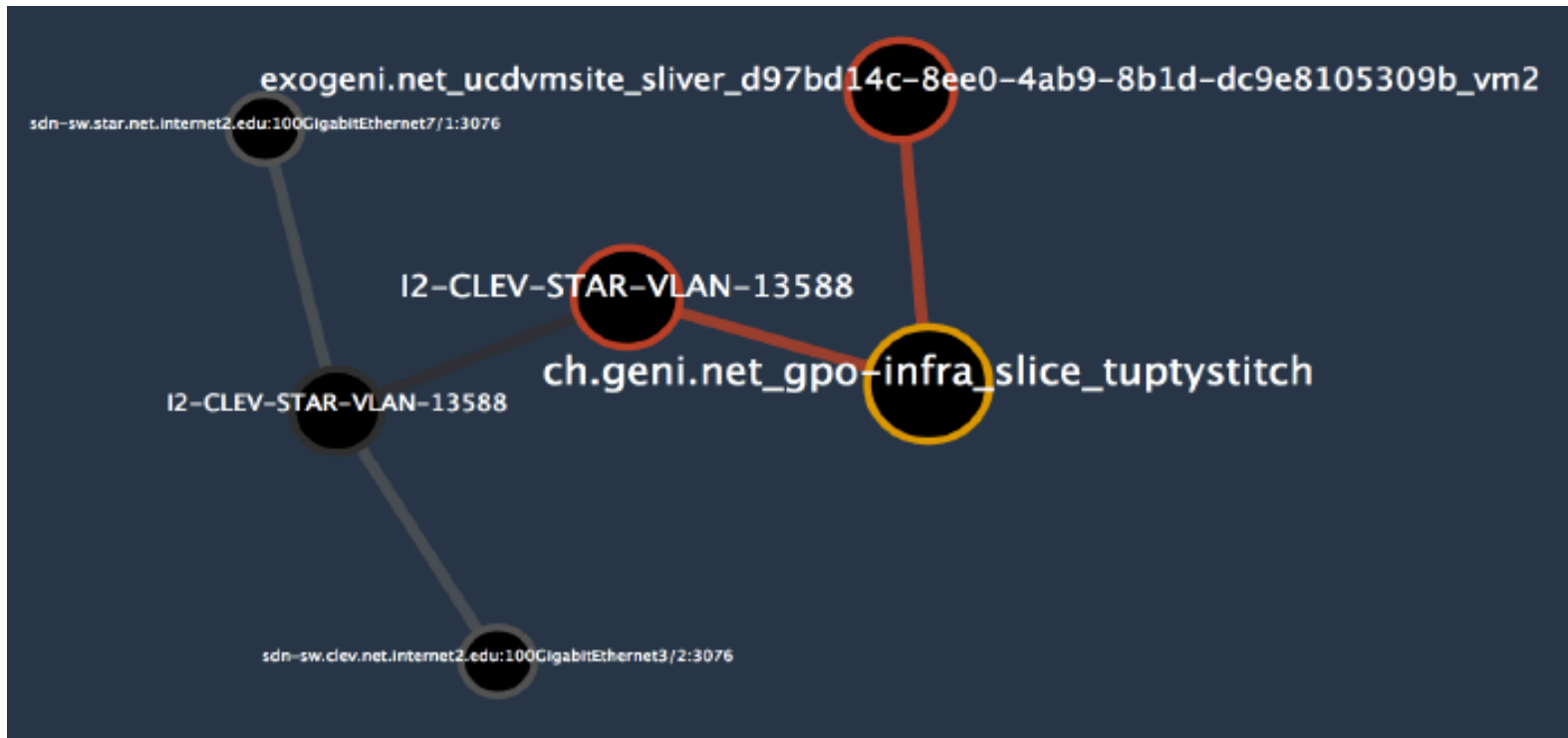
Alerting on Monitoring Stream

-In a typical database we query existing data based on some declarative language.

-We can think of ESPER as like an upside down SQL, where if events occur in the future, results will be emitted.

-Using the ESPER query language, EPL (similar to SQL) complex events can be described.

select count(*) from MyEvent(somefield = 10).win:time(3 min) having count(*) >= 5

-There exist a stream of events named *MyEvent*.
-In the *MyEvent stream* there are events that contain a field named: *somefield*
-In a 3 minute window, if *somefield = 10* five or more times, emit data.
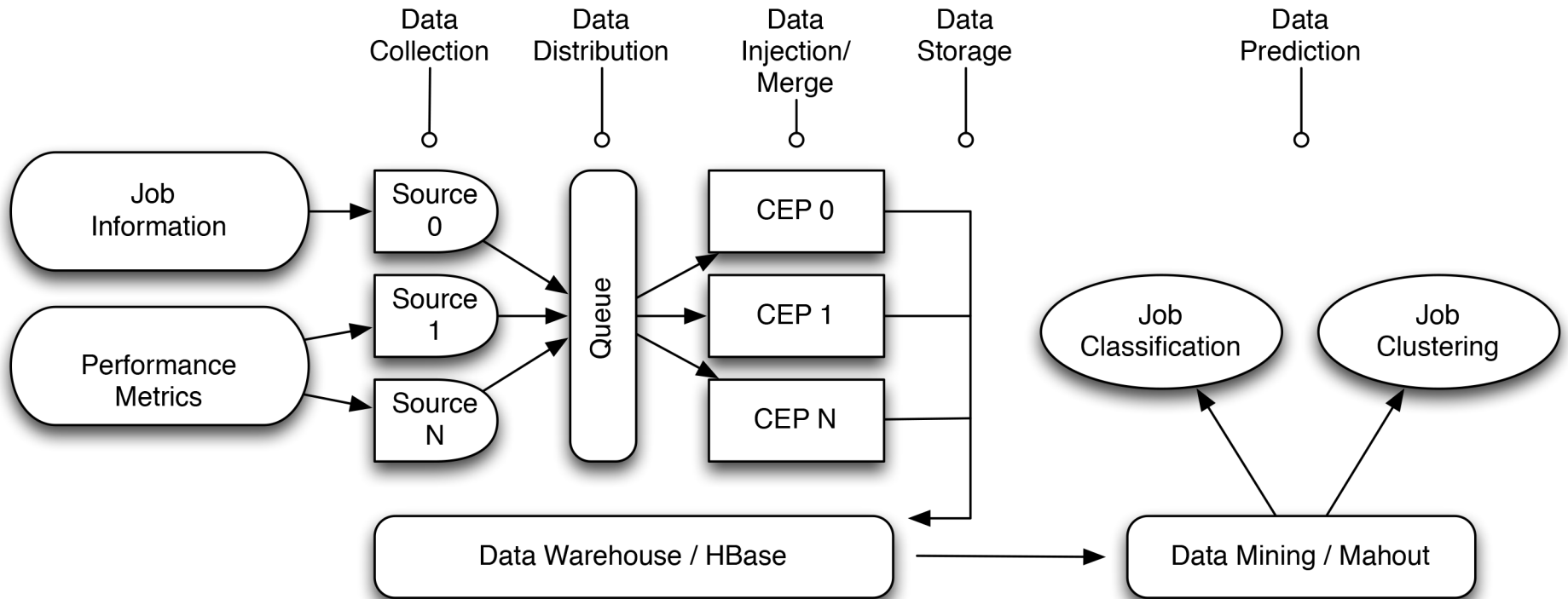
FUTURE STATE

Hybrid Feedback

Monitoring Stream

Knowledge Discovery
–Workload Classification

Resource Profiling

Metric Values — Profile #

Var1: bytes_in, bytes_out, bytes_read, bytes_written, contexts, cpu_system, cpu_user, disk_total, interrupts, mem_buffers, mem_cached, mem_total, page_in, page_out, pkts_in, pkts_out, proc_run, read_time, reads, swap_in, swap_total, write_time, writes

Thanks!

Cody Bumgardner : cody@uky.edu

Caylin Hickey : caylin.hickey@uky.edu

Ray Hyatt : rhyat1@uky.edu