**Overview of the GIMS (GENI Instrumentation and Measurement Systems) Packet Capture Control and Management System**

Charles Thomas

cthomas@wisc.edu

August 5, 2010

**Design and Implementation**

The GIMS Packet Capture control and management system (i.e. those components of GIMS exclusive of the packet capture system proper) is divided into the following components:

Aggregate Manager:

The GIMS Aggregate Manager (AM) coordinates activities between the GUI, the AJAX server, the GIMS database, and the instrumentation itself. These components allow a user to create, configure, and run the traffic capture facet of a larger ProtoGENI or GENI experiment.

The GIMS Database:

A MySQL database was implemented to keep track of experiment and device information as well as experiment/device pairings. This database replaces the flat-files used to store information in the prototype.

The GIMS DB Admin Tool:

The GIMS Admin tool is a web-based GUI that allows authorized GIMS administrators to manage the GIMS database. Currently-supported functions include:

Add Device: Add a new device to the GIMS database.  Admin can specify a device location, device name, device type, hostname, port, and description.  Admin can also specify which device capabilities are supported by the device in question.  Location name pull-down is pre-populated with a list of existing locations, but user can also specify a new location.  Admin-entered info is sanity-checked via Javascript and errors are pointed out before form submission making it less likely for bogus values to make it into the database.  Upon entry, device capability settings are converted into XML/RPC code for database submission.

Edit Device: Edit an existing device in the GIMS database.  Admin can change any of the afore-mentioned parameters.  All fields are pre-populated with existing device state information, including device capabilities.  As with the "Add Device" functionality, Admin-entered data is sanity-checked via Javascript and errors are pointed out before form submission.

Delete Device: Delete an existing device from the GIMS database.  Admin is prompted to pick from a list of existing devices and once a device is selected the full device listing is displayed in a verification step before device is deleted from the database.  After deletion, a listing is given of the device data that was deleted in case it needs to be recreated later.

**The Life-cycle of an Experiment**

The GIMS control and management system has built-in logic to govern experiment life-cycle (note: “experiment” here refers to a specific instance of traffic capture, not to a larger ProtoGENI or GENI experiment). The packet capture portion of a ProtoGENI experiment can be in one and only one of several states, “Instantiated”, “Configured”, “Running”, “Paused”, “Done”.

Transitions include the following state-changes in response to commands:

 CreateNewExperiment: Instantiated

 ConfigureExperiment: Instantiated->Configured

 StartExperiment: Configured->Running | Paused->Running

 PauseExperiment: Running->Paused

 StopExperiment: Running->Done | Paused->Done

Sanity-checking logic has been implemented at each step of the experiment life-cycle to catch actions that don't make sense given the experiment state.

Better feedback is now provided to user at each step, both for progress and for error conditions. We have also improved logging for troubleshooting and verification.

**Installation**

Download the latest tarball and extract it as usual in a directory in which CGI is enabled and to which a browser can be pointed. The permissions in our directory are set up as follows, but these settings should not be considered authoritative:

drwxrwxr-x CSS

drwxrwxr-x ajax

drwxrwxr-x images

-r-xr-xr-x GIMSAdmin.cgi

-rwxr-xr-x GIMSControl.cgi

-rwxr-xr-x GIMSAuth.pm

-r--r--r-- GIMSDBUtils.pm

-r--r--r-- GIMSJavascript.pm

-r--r--r-- XmlRpcUtils.pm

-rwxrwxrwx gims\_admin.log

-rwxrwxrwx gims\_am.log

-r-xr-xr-x gims\_aggregate\_manager.cgi

-rw-rw-r-- gims\_db.sql

-r-xr-xr-x standin.pl

File explanations:

CSS – Directory contains formatting information for the web pages.

ajax – Directory contains logic for AJAX transactions.

images – Directory of graphics used in the various GUIs.

GIMSAdmin.cgi – GIMS database admin web page.

GIMSControl.cgi – The main GIMS capture web page.

GIMSAuth.pm – Perl module for handling GIMS authentication (incomplete)

GIMSDBUtils.pm – Database utility calls

GIMSJavascript.pm – Javascript logic for the main capture page.

XmlRpcUtils.pm – Utility calls for doing XML/RPC conversion.

gims\_am.log – Logging for GIMS Aggregate Manager goes here.

gims\_admin.log – Logging for the database admin tool goes here.

gims\_aggregate\_manager.cgi – The GIMS Aggregate Manager

gims\_db.sql - A mysqldump file that can be used to create the GIMS database.

standin.pl – A pseudo capture device used for testing when a real capture device isn’t required.

**Use**

Commands and components available from the main GUI (GIMSControl.cgi) are as follows:

**Action Pulldown:**

Create New Experiment: Instantiate a new experiment in the GIMS database. Associate a specific capture device with the experiment. Assign a unique ExperimentID to the experiment.

Set Capture Parameters: Presents the user with a setup UI that is tailored to the capabilities of the device. Allows the user to choose which capture parameters they want to enable and configure. Creates the libpcap string that will be used by the capture device to specify packets of interest to the experiment. Allows the setup of either local or remote data storage, as well as anonymization and/or data aggregation techniques to be applied.

Control Capture Device: Presents the user with a UI that will allow starting, pausing, and stopping of the capture device itself.

Get Experiment Status: Under development

Get Experiment Settings**:** Show the configuration settings for this capture session. Allows for notation or auditing of current configuration.

Capture Device Name Pulldown: This pulldown allows the user to select a device for use in the capture session. Devices can be added to or deleted from this list using the GIMS Database Admin tool.

Valid Experiment IDs Pulldown: Experiments that have been instantiated are available from this pulldown along with the date they were created and the current state of the experiment. This allows the user to select a specific experiment for configuration or control.

**Typical Use Sequence:**

1. User selects “Action To Perform -> Create New Experiment”.
2. User chooses a device from the “Capture Device Name” pulldown, e.g. ‘msn:localhost’.
3. An experiment ID is generated and displayed.
4. User clicks “Submit” button.
5. GUI displays result of action, e.g. “Experiment EXP\_B18E52A8 created. Added device msn:localhost to the experiment.”
6. User selects “Action To Perform -> Set Capture Parameters”.
7. User selects an experiment to configure. The most recently-added experiment and devices should be selected automatically, but double-check that it is correct. Selecting an experiment will automatically select the correct device for that experiment.
8. User clicks “submit”.
9. The experiment setup GUI is displayed.
10. User selects the experiment parameters that are of interest in the capture session.
11. User clicks “submit”.
12. GUI will confirm that configuration happened successfully.
13. User clicks “Return to Start”.
14. User selects “Action To Perform -> Control Capture Devcice”.
15. User selects the ExperimentID to be controlled from the pulldown. Usually this is the most recent one.
16. User clicks “Submit”.
17. Control GUI is displayed. Capture start/pause/stop can be controlled from this GUI. Feedback from the AM and device will be displayed in the window.
18. On clicking “Stop” and verifying, the session has ended. Data will be saved in the directory specified during experiment configuration.