

Lab Metrics

Goals for Spring 2013

- Update aggregator to new windows log format
- Update front end to new windows log format (Do we actually need to do this?)
- Document deploying metrics
- Document metrics implementation
- Determine existing metrics TODO's

Goals for Spring 2012 semester

- Finish aggregation of data
 - Linux side is being worked on by Gary Katsevman
 - Windows side
- Add ability to link to a date/time directly via the new history api with hash fallback using History.js
- Include various other data that would be interesting to graph and could possibly be piped to the Maps project
 - wifi data
- mind meld all that Gary knows to Nick and whomever is willing to listen
- Redesign website to be better and look prettier using twitter's bootstrap

Intro

The Lab Metrics project was started to quantify lab usage. Our primary goals include answering the following questions:

- When should the lab be open?
- How many windows / linux machines do we need?
- Do we need more computers?
- When is the lab used most?

To do this we have researched some existing software packages and come to the conclusion that our best solution would involve building our own service. The closest software package that fits our needs would be LabStats by Computer Lab Solutions. However there are several pitfalls which we want to avoid:

- Required daemon
- Cost
- Lack of Ubuntu and Win7 support

Due to the request to do this sans-daemon and the cost of the package, we believe that we can do better building our own service in house.

Our primary goal is to build a service and web front end that will allow an admin to see useful statistics (time series data) about lab usage. Specifically, our goals will be:

- Web front end
- Data collection service
- Automated time series data backup
- High level of documentation to allow future maintenance

IRC

Have a quick question or comment? Join the Lab Metrics discussion at #metrics on irc.ccs.neu.edu.

Setting up

First download [setupmetrics.sh](#) onto your home directory in a CCIS Linux machine.

This is a script that will attempt to get all sources from crew-git and it will install all dependencies. The sources will go to `./metrics` and the environment (dependencies) will go to `/scratch/$USER/metricsenv`.

Caveat



If you do not want this in your CCIS home, then either edit the script to point `SCRATCH_DIR` to somewhere else or create directory `/scratch/$USER`.

To run the backend or the frontend, you need to have [MongoDB](#) running. There is a copy of MongoDB at `/proj/crew/lee/bin/mongod`. So, you can do

```
mkdir -p /var/tmp/$USER/data
/proj/crew/lee/bin/mongod --dbpath /var/tmp/$USER/data
```

If not on CCIS Linux, download the current 1.6 version from <http://www.mongodb.org/downloads>.

Running Backend

The backend should be installed in your environment, so:

```
cd metrics/backend
source env/bin/activate
python setup.py install
flamongo
```

Running Frontend

Ensure the backend is running.

```
cd metrics/frontend
source env/bin/activate
cd growler
python manage.py runserver 0.0.0.0:8000
```

Current Schedule

- Regroup and identify tasks - 1/31/11
 - Attribute name indirection (Create a model/document that describes the collection of data). - aldwin
 - Frontend. - gkatsev
 - Documentation - lee
 - Howto setup MongoDB with redundancy/failover.
 - The application itself.
 - Finish Linux usage collection - noack
- Stand up a test instance - TBD
- Demo - TBD

Past Schedule

- Project Specification - 10/11/10
- Web API - 10/22/10
- Aggregator - 11/15/10
- Web frontend - 11/20/10
- Backend - 11/10/10
- First demo - ~ 12/1/10

Current Members

- Gary (gkatsev@ccs.neu.edu)
- Nick (aldwin@ccs)
- Geoffrey Wong (gwong@ccs.neu.edu)

old members

Adam Alix - Backend / Aggregator / Frontend / API
Alex Lee - Backend / Aggregator / API
Nick Aldwin - Backend
Sarah Laplante - Aggregator
Chris Jelly - Backend / Aggregator / Frontend
Ali Green - Frontend
Casey Brown - Frontend
Robert Noack - Aggregator
Gary Katsevman - Frontend

The Git repositories

Anonymous access urls:

```
git://crew-git.ccs.neu.edu/metrics/api.git  
git://crew-git.ccs.neu.edu/metrics/backend.git  
git://crew-git.ccs.neu.edu/metrics/frontend.git  
git://crew-git.ccs.neu.edu/metrics/aggregator.git
```

Write access urls:

```
git@crew-git.ccs.neu.edu:metrics/api.git  
git@crew-git.ccs.neu.edu:metrics/backend.git  
git@crew-git.ccs.neu.edu:metrics/frontend.git  
git@crew-git.ccs.neu.edu:metrics/aggregator.git
```

Talk to one of the crew elders for write access.

Links

1. [Time Series Data](#)
2. [RRD](#)
3. [Project Plan](#)
4. [Coding Conventions](#)
5. [Python Development](#) - virtualenv and pip primer.