

IEPM-BW Documentation

Hardware Requirements

The monitoring host should be a machine dedicated to the IEPM-BW software. The host must have enough CPU power to run the probes, drive the MySQL database, and run the analysis. It should also have enough disk space to support the database. The recommended monitoring host hardware configuration is:

- Dual cpu xeon 3 GHz processors with hyperthreading or better
- 4 Gigabytes of memory to allow for plenty of space for MySQL and TCP buffers and workspace as well as a minimum amount of paging
- 1 Gigabit network interface, although a 10 Gigabits nic is also acceptable
- 100 Gigabytes of disk space to allow for the MySQL data base and and all the analysis reports which are generated.
- A separate disk to back up the MySql data base contents

The target host does not need to be a dedicated machine, but it should not be a busy machine.

- A gigabyte of memory is recommended
- If the machine is dedicated to being a target, 1 Gz of CPU is enough.
- Very little disk space is required. The server code is small and can reside in the home directory of the account it runs from.

Software Requirements

Following are the software requirements for the IEPM-BW monitoring node.

- Recommended operating system is the latest stable version of RedHat or any flavour of Fedora, more suitably Fedora Core 5.
- Two accounts should be created on the monitoring machine as these are needed to install and run IEPM-BW.
 - 'iepm' is a user account and is used for running IEPM-BW.
 - Account = iepm, home directory = /home/iepm
 - Second account is an account with root privileges. This is account is used for installation and administration. It's 'cyrus' in my case.
 - Account = cyrus, home directory = /home/cyrus

You can have your own account, but don't forget to replace it at every place with your own account.
- An apache web server has to be available, and the recommended document root is '/var/www'.
- Standard MySQL must be installed on the system. Most recommended is the latest stable release, as this will facilitate assuring the upward compatibility of the code.
- Make sure that the system is configured to autostart MySQL and httpd (apache web server) upon booting. On some systems this can be done with 'chkconfig'. Commands are

```
chkconfig httpd on
chkconfig mysqld on
```

- The following /etc/sysctl.conf parameters are recommended for optimum performance

```
/proc/sys/net/core/wmem_max=20971520
/proc/sys/net/core/rmem_max=20971520
/proc/sys/net/core/rmem_default=65536
/proc/sys/net/core/wmem_default=65536
/proc/sys/net/ipv4/tcp_rmem=4096 87380 20971520
/proc/sys/net/ipv4/tcp_wmem = 4096 65536 20971520
```

The following ports must be open for the probes

Test Name	Port#
iperf	5000
owamp(2.0) link	4824
thrulay	5003
pathchirp	8365 udp
pathload	55001 udp
pathload	55002 tcp

Keep in mind to open port for each and every other service you run. e.g. HTTP, NTP etc

- To allow for running OWAMP, the monitoring and target hosts should be configured with 4 NTP servers. The OWAMP documentation discusses how to setup NTP.

- The latest version of JAVA must be installed on the monitoring host.
- Gnuplot 4.0 is required and must be installed on the monitoring host machine
- Perl should be installed locally on the machine. It can be either in `/usr/bin/perl` or `/usr/local/bin/perl`. There should be a link from where it doesn't exist to where it exists. Suppose if Perl path is `/usr/bin/perl`, its link should be created at `/usr/local/bin/perl`. Perl modules must be installed on the system.

```

Date::Calc
Date::Manip
Date::Parse
Time::localtime
Time::Local
Time::Parse
Sys::Hostname
Proc::ProcessTable
Bundle::DBD::mysql
getopts.pl
getopt.pl
Getopt::Std
strictIO::Handle
File::Temp
SOAP::Lite
CGI
Bit::Vector
GD

```

Don't forget to set the time zone in `Date::Manip` configurations along with other configurations.

Important notes for later reference:

- To start the mysql daemon, use `mysqld_safe`
- To set the root password after installation use: `mysqladmin -u root password "new password"`
- To shut mysql down, use `mysqladmin -u root -p shutdown`

IEPM-BW Software Installation

Obtain software

First you will need the iepm-bw source code. Typically this is obtained from SLAC by logging on (if you have an account) or by an email request to jaraedg@slac.stanford.edu. Either you or he will tar up the entire `v3src` directory, below are the commands to use:

```

>cd $bandsrc
>tar -cvf v3src .tar
>gzip -c v3src.tar /tmp/v3src.tar.gz
>ls -l /tmp/v3src.tar.gz
-rw-r--r-- 1 cottrell sf          125632972 May 17 12:34 /tmp/v3src.tar.gz
>rm v3src.tarcp /tmp/v3
>mv /tmp/v3src.tar.gz ./
>ls -l v3src.tar.gz
-rw-r--r-- 1 cottrell sf          125632972 May 17 12:34 v3src.tar.gz
>cd /afs/slac/public/users/cottrell
>cp $bandsrc/v3src.tar.gz ./

```

This requires a lot of disk space (e.g. `v3src.tar` is ~ 180MBytes). It creates a file `$bandsrc/v3src.tar.gz` which is probably too big to email so it is made available by anonymous ftp. To access it use:

```

>ftp ftp.slac.stanford.edu
Connected to ftp.slac.stanford.edu (134.79.18.30).
220-=====
220-By using this system you expressly consent to the terms and
...220 FTP server ready.
Name (ftp.slac.stanford.edu:cottrell): anonymous331 Anonymous login ok, send your complete email address as
your password.
Password:
230-=====
230-
230-Stanford Linear Accelerator Center Anonymous FTP Server
230-
230-=====...
ftp> binary
200 Type set to I
ftp>cd users/cottrell
ftp> ls
227 Entering Passive Mode (134,79,18,30,188,42).
150 Opening ASCII mode data connection for file list
dr-xr-xr-x  2 ftp      ftp          4096 May 17 19:45 .
dr-xr-xr-x  2 ftp      ftp          4096 May 14 18:21 ..
-r--r--r--  1 ftp      ftp           335 Oct  5  2001 identity.pub
-r--r--r--  1 ftp      ftp           304 Feb  9  1996 pgp.publickey
-r--r--r--  1 ftp      ftp      125632972 May 17 19:45 v3src.tar.gz
226 Transfer complete.ftp> get v3src.tar.gz
local: v3src.tar.gz remote: v3src.tar.gz
227 Entering Passive Mode (134,79,18,30,188,44).
150 Opening BINARY mode data connection for v3src.tar.gz (125632972 bytes)
226 Transfer complete.
125632972 bytes received in 28.4 secs (4.3e+03 Kbytes/sec)
ftp>quit
>

```

You will need to unzip and detar the file using: `tar -xzf v3src.tar.gz`

Install

Currently there is no automated way to install the IEPM-BW monitoring kit on a host. Here are the steps which should be followed, in order, to install the monitoring host software. This assumes that the [requirements](#) for hardware and software configuration have been met. Note that if the default account 'iepm' with home directory '/home/iepm' is used, the source directory '/home/iepm/v3src' is used for the source, '/home/iepm/mysql' is used for the data base, then much of the installation can be done by cutting and pasting from this documentation.

Note that before starting this installation procedure, MySQL must have been successfully installed and all the required perl modules must have been installed. I used to provide MySQL installation instructions, but I have found that the installation procedure depends on the specific system you are installing it on. You must use the information in the MySQL documentation to install it. In addition, the installation of the required perl modules can vary system to system, and so in some cases you can use CPAN directly, but in others you may have to do the installations manually.

The following instructions must be performed as root

1. Decide what accounts are going to be needed. Assume the account 'cal' will have root privileges and be used for operations requiring root. Assume that the monitoring will run under the non-root access account 'iepm'. Both, for purposes of this documentation are assumed to be in group '<account group>', 'slac'. For this example, I will assume that the home directory for the 'cal' account is '/home/cal', and the home directory for the 'iepm' account is '/home/iepm'. For the sake of consistency between monitoring systems, the account 'iepm' with '/home/iepm' is strongly recommended.
2. The monitoring host has a name. This is referred to as '<fully qualified monitoring host name>'.
3. The monitoring host should be given an <aliasname> that starts with 'iepm-bw'. The idea is to hide the real name of the node, but provide it with a name that can recognize its place in the world. This is important. Examples include:
 - iepm-bw.twaren.net
 - iepm-bw.caltech.ul-org
 - iepm-bw.cern.ch
 - iepm-bw.bnl.org
 - iepm-bw.snv.ul-org
 - iepm-bw.fzk.de
4. If the account mysql:mysql has not been allocated, allocate it by:

```

groupadd -g 20000 mysql
useradd -u 20000 -g mysql mysql

```

5. Decide on the directory structure for the mysql data base and the monitoring code. It is best to have the mysql data base on a separate physical disk from the system disk, but if this is not possible, use '/home/iepm/mysql' as the base. Make the mysql directory.

```
mkdir /home/iepm/mysql
chown mysql:mysql /home/iepm/mysql
```

6. Create a configuration file for /etc/my.cnf which MySQL will use when you install the database.

```
[mysqld]
datadir=/home/iepm/mysql
socket=/home/iepm/mysql/mysql.sock
port=3306
[mysqld_safe]
datadir=/home/iepm/mysql
socket=/home/iepm/mysql/mysql.sock
port=3306
[mysql.server]
datadir=/home/iepm/mysql
socket=/home/iepm/mysql/mysql.sock
port=3306
[client]
socket=/home/iepm/mysql/mysql.sock
port=3306
```

7. If mysql is already running, it is running with the 'test' database. Kill the mysql daemons.
8. Restart MySQL with 'mysqld_safe'. This will restart the MySQL server with the configuration file /etc/my.cnf.
9. Create the data base by executing 'mysql_install_db'.
10. Change the ownership of '/home/iepm/mysql' by the command 'chown -R mysql:mysql /home/iepm/mysql'.
11. Set the database root password with 'mysqladmin -u root password "new password" '.
12. You are now ready to create the directories with '/home/iepm/mysql' that must be owned by the iepm account. These directories are used by the iepm monitoring to track the state of the monitoring system. Execute the following commands with 'iepm:slac' replaced with 'slac' replaced by the appropriate group for the 'iepm' account.

```
# the data directory is where the results from the probes are stored.
# These files are picked up by the 'load-datad' daemon and put into the database.
mkdir /home/iepm/mysql/data
chown iepm:slac /home/iepm/mysql/data
# the logs directory contain the log files from each process
mkdir /home/iepm/mysql/logs
chown iepm:slac /home/iepm/mysql/logs
# the pids from the daemons are stored here
mkdir /home/iepm/mysql/pids
chown iepm:slac /home/iepm/mysql/pids
# the files in the keepalives directory are used by the monitoring software to ascertain that
# the daemons are running
mkdir /home/iepm/mysql/keepalives
# prime the pump with:
touch /home/iepm/mysql/keepalives/traced.alive
touch /home/iepm/mysql/keepalives/pingd.alive
touch /home/iepm/mysql/keepalives/bw-synched.alive
touch /home/iepm/mysql/keepalives/pathchirpd.alive
touch /home/iepm/mysql/keepalives/load-datad.alive
touch /home/iepm/mysql/keepalives/load-scheduled.alive
touch /home/iepm/mysql/keepalives/owpingd.alive
# set the permissions so that the iepm account can access them
chown -R iepm:slac /home/iepm/mysql/data
chown -R iepm:slac /home/iepm/mysql/logs
chown -R iepm:slac /home/iepm/mysql/keepalives
chown -R iepm:slac /home/iepm/mysql/pids
```

13. The output from the analysis and reports must go into a directory which is writeable by the 'iepm' account. Generally this is '/home/iepm/public_html'. Make this directory with:

```
mkdir /home/iepm/public_html
mkdir /home/iepm/public_html/<aliasname>
chown -R iepm:<account> /home/iepm/public_html
```

Where <account> is the account your unix id is under. For example, mine is 'cal:sf' (i.e. <account> would be replaced by sf) and our iepm group account was 'iepm:sf'. Depending upon how the /home/iepm/public_html directory was created...for example if I created it via sudo from my cal account, then owner must be set to the 'iepm:<account>' where in SLAC's case <account> is 'sf', but in other cases it might not be.

14. Create a link to this area from /var/www/html

```
ln -s /home/iepm/public_html/<aliasname> /var/www/html/<aliasname>
```

15. IEPM-BW requires a simple configuration file, '/etc/iepm.cnf'. The contents of /etc/iepm.cnf should contain:

```
# required iepm configuration parameters
# /etc/iepm.conf

dataBase = "iepm"
dataBaseHost = "<fully qualified monitoring host name>"
dataBaseDir = "/home/iepm/mysql"
iepmSrcDir = "/home/iepm/v3src"
updateAcct = "iepm"
acctGroup = "<account group>"
readAcct = "readonly"
# This is provided as the monitoring host and database host could be different machines
# It is not advisable for them to be different. In monitoring a network, one should avoid using the
# network for the monitoring control and data logging.
monHost = "<fully qualified monitoring host name>";
aliasName = "<aliasname>";
monHostip = "<monitoring host ip address>";
mysqlport = 3306;

#Connie Logg 5/12/04 cal@slac.stanford.edu <- configurator's name and date please
```

The following installation instructions do not need root privileges can be performed under the iepm account

Installing the iepmtools directory

The 'iepmtools' directory contains the gzipped distribution files for the various tools (probes and otherwise) required for IEPM-BW. The 'install-iepmtools' script can be used to install the tools contained in this directory. There are two associative arrays where the tools, their gzipped files, binaries and required links are specified. This is documented in the code. The 'install-iepmtools' script does the following:

1. 'gunzip's the source distribution
2. Uses tar to extract the source into a directory
3. Runs ./configure in the directory - if there is no ./configure, a message will be printed and the step is ignored
4. Runs make to make the code
5. Locates the needed binaries (as specified in the array %clients_servers)
6. Copies them to the IEPM-BW bin directory
7. Makes any required links.
8. After the binaries are copied to the bin directory, and the links are defined, the installation program pauses to allow the user to look at the output from step 1-7 for each tool that is made.

This is a simple script, and may not be extendable to all packages which someone might want to install with it. ***It should never be run from a cron job.*** only from the command line. It requires that the IEPM-BW installation be in the standard form. The base directory is '/home/iepm/v3src', the binaries are to be loaded into '/home/iepm/v3src/bin', and the iepmtools directory is '/home/iepm/v3src/iepmtools'.

IEPM-BW Database Setup

A MySQL system has its own set of users which are distinct from the unix users, although the names can be the same. The recommended users are:

1. iepm - this is the account the monitoring system runs with
2. readonly - this is a readonly account for general access to the data
3. apache (or nobody) - the account used by httpd to access the database

4. cal (or other user) - the administrator account
5. root - used for MySQL system administration

I remind that these accounts are MySQL user accounts and should not be confused with system user account.

The accounts are defined by issuing grants to the account for accessing the data base.
Create a file named pws in /home/iepm/mysql i.e.

```
touch /home/iepm/mysql/pws
```

edit the file and put the password of your iepm mysql account in it.

Create the IEPM data base. Log into MySQL with the root account and password and issue the following command:

```
create database iepm;
```

The following grant commands can be set up in a text file for 'batch' entry into MySQL, or they can be entered one by one. The root account and password must be known to set up the grants. Make sure, you put the passwords in and don't use the string 'password'. Note the host name component of the grant. All these forms may not be necessary, however I have found that they are for various means of access.

```
#For mysql administrator ('cyrus' in this case) grant all on *.* to cyrus@localhost identified by 'password'
with grant option;
grant all on *.* to cyrus@202.83.167.108 identified by 'password' with grant option;
grant all on *.* to cyrus@iepm-ncp identified by 'password' with grant option;
grant all on *.* to cyrus@iepm-ncp.ncp.edu.pk identified by 'password' with grant option;
grant file on *.* to cyrus;
```

```
# For the iepm account used by iepm to load data into the data base:
grant all on iepm.* to iepm@localhost identified by 'password';
grant all on iepm.* to iepm@202.83.167.108 identified by 'password';
grant all on iepm.* to iepm@iepm-ncp identified by 'password';
grant all on iepm.* to iepm@iepm-ncp.ncp.edu.pk identified by 'password';
grant file on *.* to iepm;
```

```
# For the readonly account (report generation) - no password
grant select on iepm.* to readonly@localhost;
grant select on iepm.* to readonly@'202.83.167.108';
grant select on iepm.* to readonly@iepm-ncp;
grant select on iepm.* to readonly@iepm-ncp.ncp.edu.pk;
grant file on *.* to readonly;
```

```
# For the web server account (may be nobody or apache or other) - no password
grant select on iepm.* to apache@localhost;
grant select on iepm.* to apache@'202.83.167.108';
grant select on iepm.* to apache@iepm-ncp;
grant select on iepm.* to apache@iepm-ncp.ncp.edu.pk;
grant file on *.* to apache;
```

Creating Tables

Create the tables in iepm database. The scripts of these tables can be found here. However you can use the SQL queries in the files in create tables directory. Create tables by giving these files as input to mysql. Please check if you might need to change nodes.txt in case you have different paths for ping, grep, traceroute etc. Use the command

```
mysql -u root -p > file_name Database name
```

For example, for creating monhost table in iepm database

```
mysql -u root -p > monhost.txt iepm
```

It will prompt you for password; enter your mysql root password.

Loading the IEPM Database

Next comes the loading of data in to the database tables. Look at the following links of tables at SLAC IEPM node to have some idea of the data to be inserted in the database tables.

<http://www.slac.stanford.edu/comp/net/iepm-bw.slac.stanford.edu/monitoring-hosts.html>
<http://www.slac.stanford.edu/comp/net/iepm-bw.slac.stanford.edu/NODESPECS.html>
<http://www.slac.stanford.edu/comp/net/iepm-bw.slac.stanford.edu/targets-iepm-bw.slac.stanford.edu.html>
<http://www.slac.stanford.edu/comp/net/iepm-bw.slac.stanford.edu/toolspecs.html>
<http://www.slac.stanford.edu/comp/net/iepm-bw.slac.stanford.edu/plotspecs/pltspecs-entries.html>

The scripts for populating the database tables are located in v3src/mysql. You can load tables with data using CSV files. Other way you can also create your own queries, put them in a file and give as input to mysql using the command

```
mysql -u root -p > file_name databasename
```

At the moment, I recommend creating table by using queries.
For reference, see the following links

[Populating the NODES Table](#)
[Populating the GROUP Tables](#)
[Populating the MONHOST Table](#)
[Populating the TOOLSPECS Table](#)

For more information on the iepm-bw mysql table setup see [IEPM Database Table Summary](#).

Loading the IEPM Database

Currently loading of the monitoring host is done from iepm-bw.slac.stanford.edu.

1. Enter the node into the 'NODES' data base.
2. Enter the node into the 'MONHOST' data base.
3. On iepm-bw.slac.stanford.edu, make a tar file including all the code.

```
cd /afs/slac/package/netmon/bandwidth-tests/v3src
tar -cvf dist.tar -T dist.scripts.list
scp dist.tar iepm@<fully qualified monitoring host name>:/home/iepm/dist.tar
```

4. Log onto the new monitoring host node under the 'iepm' account.
5. Execute the following to install the source code

```
mkdir v3src
cd v3src
tar -xvf ../dist.tar
```

6. Now configure the 'config' directory. The following files may need configuration:
 - The 'config' directory contains a sample .cshrc. Modify this to match the node being installed and copy it to '/home/iepm/.cshrc'. This will defined where java is located, and two environment variables "\$bandsrc" which points to the v3src directory and "\$banddata" which points to the directory where the data and reports are stored. Note that there is a link from '/var/www/html' to this directory.
 - 'cleanup-list' - should be configured correctly. If there are processes which are known to hang, they should go in this file. 'bw-cleanup' uses this file to clean processes known to hang (now or in the past). If the process with the name in this list exists longer than the number of minutes indicated on its line, it will be killed.
An example is:

```

•
# cleanup process list
# Connie Logg 5/18/06 cal@slac.stanford.edu
post-test,120
gnuplot,5
triganal,120
runperiod,120
pathload_rcv,3
pathchirp_rcv,3
/bin/ping,3
/bin/owping,3
/bin/iperf,3
/bin/tlaytcp,3
/bin/bw-iperf-client,3

```

- 'owampd.conf.template', 'owampd.limits', and 'owampd.keys' are configuration files for owamp. 'owampd.limits' and 'owampd.keys' should be ok. Look at owampd.conf.template. 'datadir' and 'vardir' are defined to be '/home/iepm/owamp/data' and '/home/iepm/owamp/run' respectively. If this is ok (recommended default), save 'owampd.conf.template' as 'owampd.conf'. Then create the directories:

```

mkdir /home/iepm/owamp
mkdir /home/iepm/owamp/data
mkdir /home/iepm/owamp/run

```

- 'servers.alive' may need to be configured. This is the file that has the information for starting the servers. It is used by 'keep-servers-alive' to start, and if necessary, restart the servers. For a standard monitoring host, this file should be:

```

bw-iperf-server,bin,-s -p 5000 -w 20M
thrulayd,bin,
pathload_snd,bin, -q -i
map-updated,,
pathchirp_snd,bin,
owampd,bin, -c /home/iepm/v3src/config -Z >& /tmp/owampd.log &

```

The first field is the name of the script. The second field is the directory in \$iepmSrcDir where the script resides. The third is a list of parameters to start it with.

- Make copies of /etc/my.cnf and /etc/iepm.cnf in the config directory. They are used to provide configuration information to the CGI scripts.

```

cp /etc/my.cnf /home/iepm/v3src/config
cp /etc/iepm.cnf /home/iepm/v3src.config/iepm.cnf.<aliasname>

```

7. Install JAVA, if required. If it must be installed, the recommended installation directory is '/home/iepm'. To install it, copy the 'JDK...bin' in 'home/iepm/iepmtools' to '/home/iepm' and execute it.
8. Configure Geoplot.
9. Load the NODES and NODESPECS tables, MONHOST table, TOOLSPECS table, PLOTSPECS table, and ther GROUPS and NODEGROUPS tables if you want to use their functionality.
10. Once the tables are loaded, run the various reports for them. 'report-nodes' should be run first. If MONHOST is not set up correctly, none of the reports will run. You must fix MONHOST first.
11. Other reports to run are:
 - report-monhost
 - report-nodegroups
 - report-nodespecs
 - report-plotspecs
 - report-table-structure
 - report-toolspecs
12. Once all the reports run, you are ready to try the probes. 'schedule-load' will force the probes to be scheduled and run. Look in \$mysqldb/data and check the data files that are output from the probes running. If they look ok, you can execute the various data loading scripts called by 'load-datad'. These include:
 - load-ping-data
 - load-trace-data
 - load-iperf-data
 - load-tlayctp-data
 - load-trace-data
 - load-pathchirp-data
 - load-pathload-data

If all of the above tests work, and the data base is loaded with all probes, nodes, plotspecs, etc. then you can load the crontab. This should start the system up. If there are problems, be sure to check the /home/iepm/mysql/logs directory for the output from the cronjobs.

Understanding TOOLSPECS Information

At the top of the results column on the http://www.slac.stanford.edu/comp/net/iepm-bw.slac.stanford.edu/slac_wan_bw_tests.html there are links to what were the current reference pages on the web for the various tools.

There is a very useful command in \$bandsrc called 'show-probe-commands'. Look at the comment at the top of the code to see how to use it. It reads the toolspec data base and prints the commands that are executed. A useful tutorial can probably be written which shows how the commands are stored in the data base, and how the data base entries are turned into the actual commands executed. Look in the wiki for show-probe-commands. It may or may not be documented.

TOOLSPEC definitions, and plot specifications.

The plot specifications MUST BE ON ONE LINE with NO Carriage returns in them. See the wiki on this.

There is a script 'fetch-toolspecs' that the code calls to fetch the toolspecs from the TOOLSPEC table.

Entries in the TOOLSPEC table look like the following when fetched by 'fetch-toolspecs'. Now the entries in the TOOLSPC table are used to do the individual plots in create-tool-plots and create-history.

```
29cal@iepm-bw: fetch-toolspecs \ | more
\#destnode, toolname, testtype, datatoanalyze, daystoanalyze
iepm-bw.bnl.org,mtlaytcp,background-syn,mtlaytcp:mtlaytcp=w p lt 6 pt 6;mtlaytcp:rttmin=w lt 7 pt 5;
mtlaytcp:rttmax=w lt 2 pt 5;mtlaytcp:rttavg=w lt 3 pt 5,28

iepm-bw.bnl.org,owping,background,owping:tmin=w p lt 3;owping:tmed=w p lt 6;
owping:tmax=w p lt 37;owping:fmin=w p lt 1;owping:fmed=w p lt 7;owping:fmax=w p lt 8,28

iepm-bw.bnl.org,pathchirp,background,pathchirp:pathchirp=w p lt 6 pt 6;
pathchirp:thrumin=w p lt 5 pt 1;pathchirp:thruavg=w p lt 7 pt 1; pathchirp:thrumax=w p lt 3 pt 1,28
iepm-bw.bnl.org,pathload,background-syn,pathload:thrumin=w p lt 6;pathload:thrumax=w p lt 7,28
iepm-bw.bnl.org,ping,background,ping:max=w p lt 7 pt 5;ping:avg=w p lt 2 pt 5;ping:min=w p lt 3 pt 5,28
iepm-bw.bnl.org,tlaytcp,background-syn,tlaytcp:tlaytcp=w p lt 6 pt 6;
tlaytcp:rttmax=axes xly2&nbsp; w p lt 2 pt 5;tlaytcp:rttavg=axes xly2 w p lt 3 pt 5;
tlaytcp:rttmin=axes xly2 w p lt 7 pt 5,28
iepm-bw.bnl.org,traceroute,background,traceroute:numhops=w p lt 6 pt 6,28
iepm-bw.caltech.ul-org,owping,background,owping:tmin=w p lt 3;owping:tmed=w p lt 6;
owping:tmax=w p lt 37;owping:fmin=w p lt 1;owping:fmed=w p lt 7;owping:fmax=w p lt 8,28
```

Note that the filename for each type of data has as a suffix a string to indicate what the data is: i.e.,

mtlaytcp= multi-stream thruway

owping=owamp data

pathchirp=pathchirp data

etc.

Now within each file there is a heading (the first line) which indicates which data is in which column.

#date	time	tmin	tmed	tmax	fmin	fmed	fmax	tsent
tloss	tdup							
terr	tjitter	fsent	floss	fdup	ferr	fjitter	thops	fhops
06/17/2007	10:35:49	39	39.1	58.9	38.9	39.1	39.2	100
?	?							
354	0.2	100	0	0	351	0	12_(consistently)	12_(consistently)
06/17/2007	10:40:38	39.1	39.2	62.2	38.8	38.9	39.1	100
?	?							
?647	0.8	100	0	0	650	0.1	12_(consistently)	12_(consistently)

You may need to refer to the gnuplot guide while reading the following:

Note that on http://www.slac.stanford.edu/comp/net/iepm-bw.slac.stanford.edu/slac_wan_bw_tests.html there is a link called 'gnuplot configuration'. This points to a test that is run nightly to generate the correlations between colors and points in gnuplot. This is run nightly, because if gnuplot is updated, the color correspondence may change.

Let's dissect the toolspec for owping:

```
iepm-bw.bnl.org,owping,background,owping:tmin=w p lt 3;owping:tmed=w p lt 6;owping:tmax=w p lt 37;
owping:fmin=w p lt 1;owping:fmed=w p lt 7;owping:fmax=w p lt 8,28
```

If the toolspec is active (there is a field in the table for deactivating it which is not shown to be in this command),

This spec says for the node iepm-bw.bnl.org, plot the following for the individual and history plots for owping for this node:

1. plot the column 'tmin' 'with points linetype 3' which is encoded as 'w p lt 3', which is a valid short form for gnuplot.
 2. overplot the column 'tmed' 'with points linetype 6' which is encoded as 'w p lt 6'.
- etc.

Thus the data in the columns tmin, tmed, tmax, fmin,fmed,fmax are overplotted on the graph.

I will now go on to explain the 'datatoanalyze' field in the monitoring hosts table (<http://www.slac.stanford.edu/comp/net/iepm-bw.slac.stanford.edu/monitoring-hosts.html>)

Each monitoring host is has a plot specification for the http://www.slac.stanford.edu/comp/net/iepm-bw.slac.stanford.edu/slac_wan_bw_tests.html [graphs.it](http://www.slac.stanford.edu/comp/net/iepm-bw.slac.stanford.edu/slac_wan_bw_tests.html) is applied across all target hosts which it monitors. If there is no data for a particular type of test, it is ignored.

Let us look at one entry. (the one for iepm-bw.slac.stanford.edu):

```
ping:avg=axes xly2 w i lt 1;ping:min=axes xly2 w i lt 9;pathchirp:pathchirp=w p
lt 5 pt 1;iperf:iperf=w p lt 7 pt 18;miperf:miperf=w p lt 4 pt
6;tlaytcp:tlaytcp=w p lt 10;pathload:thrumin=w p lt 2;owping:tmin=axes xly2 w l
lt 10;owping:fmin=axes xly2 w l lt 4;mtlaytcp:mtlaytcp=w p lt 4 pt 6
```

This says, for all the nodes, overplot the following on the graph for that node

1. 'ping' data 'avg' with 'axes xly2 with impulses linetype 1' encoded in the gnuplot short form 'axes xly2 w i lt 1'
 2. 'ping' data min with 'axes xly2 with impulses linetype 0' encodes in the gnuplot short form 'axes xly2 w i lt 9'
 3. 'pathchirp' data 'pathchirp' 'with points linetype 5 pointtype 1' encoded as 'w p lt 5 pt 1'
- etc.

Bear in mind that is a particular data type does not exist for a target node, it is ignored.

There is a table called 'PLOTSPECS' which enables you to specify other types of plots, such as scatterplots, overplot various data.

See :

<http://www.slac.stanford.edu/comp/net/iepm-bw.slac.stanford.edu/html/scatterplots/iepm-bw.bnl.org.html>
for example.

I am not going to go thru this specification now as it follows the previous rules for specifying the data. However if you have questions. please feel free to ask.

Managing IEPM-BW

Log files

Many of the scripts produce logfiles. Basically these are the crontab re-directed STDOUT output from the scripts. At SLAC these are saved on the host iepm-bw.slac.stanford.edu in the directory /u1/myssl/logs/. The crontab appears (May 29th 2007) as:

```

#kill all servers to clean up any hung ones and to rotate the logs
5 0 * * * /afs/slac/package/netmon/bandwidth-tests/v3src/kill-all-servers \
>> /u1/mysql/logs/kill-all-servers 2>&1
# restart the servers
10 0 * * * /afs/slac/package/netmon/bandwidth-tests/v3src/restart-all-servers \
>> /u1/mysql/logs/restart-all-servers.today 2>&1
# copy and date the logs for the day
7 0 * * * /afs/slac/package/netmon/bandwidth-tests/v3src/copylogs /u1/mysql/logs/ \
> /tmp/iepm-bw-logcopy 2>&1
#
# back up the data base
15 0 * * * \
/afs/slac/package/netmon/bandwidth-tests/v3src/backup-iepm-mysql-database \
/nfs/slac/g/net/iepm-bw/iepm-bw.slac.stanford.edu/mysql-backup \
>> /u1/mysql/logs/backup-iepm-mysql-database.today 2>&1
#
# rotate and date the backups
0 3 * * * \
/afs/slac/package/netmon/bandwidth-tests/v3src/copylogs /nfs/slac/g/net/iepm-bw/iepm-bw.slac.stanford.edu
/mysql-backup \
>> /u1/mysql/logs/mysql-backup.today 2>&1
#
# run keepalive check
5,15,25,35,45,55 * * * * /afs/slac/package/netmon/bandwidth-tests/v3src/keep-em-alive \
>> /u1/mysql/logs/keep-em-alive.today 2>&1
#
# run keep server alive check
1,11,21,31,41,51 * * * * /afs/slac/package/netmon/bandwidth-tests/v3src/keep-servers-alive \
>> /u1/mysql/logs/keep-servers-alive.today 2>&1
#
# cleanup hung clients
3,13,23,33,43,53 * * * * /afs/slac/package/netmon/bandwidth-tests/v3src/bw-cleanup \
>> /u1/mysql/logs/bw-cleanup.today 2>&1
#
# run the analyses
#23 1,3,5,7,9,11,13,15,17,19,21,23 * * * \
# /afs/slac/package/netmon/bandwidth-tests/v3src/post-test-processing-script \
# -g 1 \
# >> /u1/mysql/logs/post-test-processing-script.today 2>&1
#23 1,5,9,13,17,21 * * * /afs/slac/package/netmon/bandwidth-tests/v3src/post-test-processing-script \
# -g 1 \
# >> /u1/mysql/logs/post-test-processing-script.today 2>&1
23 1,6,11,16,21 * * * /afs/slac/package/netmon/bandwidth-tests/v3src/post-test-processing-script \
-g 1 \
>> /u1/mysql/logs/post-test-processing-script.today 2>&1
#
# run the overnight analysis
15 3 * * * /afs/slac/package/netmon/bandwidth-tests/v3src/overnight-processing-script \
> /u1/mysql/logs/overnight-processing-script.today 2>&1
#
# run the trace analysis
10 * * * * /afs/slac/package/netmon/bandwidth-tests/v3src/traceanal/traceanal -d today -i 0 \
>> /u1/mysql/logs/traceanal.today 2>&1
#
#Run the new bandwidth change analysis code
5 2,6,10,14,18,22 * * * /afs/slac/package/netmon/bandwidth-tests/v3src/alerts/analyze-for-alerts \
-t "iperf,pathchirp,pathload,miperf,tlaytcp" -p "iperf,pathchirp,thrumin,miperf,tlaytcp" \
>> /u1/mysql/logs/analyze-for-alerts.today 2>&1
#
# run historical alerts web page
15 2,6,10,14,18,22 * * * /afs/slac/package/netmon/bandwidth-tests/v3src/report-alerts \
>> /u1/mysql/logs/report-alerts.today 2>&1

```

Of particular interest (i.e. should be reviewed if any problems are found in the HTML reports) are:

/u1/mysql/logs/post-test-processing-script.today
/u1/mysql/logs/bw-cleanup.today #Tells what processes needed killing

The entries for these logfiles are time stamped, the script originating the log is identified and the message is preceded with a question mark ? for a warning and an exclamation mark (!) for a serious error. The question mark and exclamation mark are to facilitate looking for more important events using grep.

Each night logcopy renames the .today file with .yesterday, the .yesterday with yesterday's date, and removes files older than 7 days. There are typically about 40 .today logfiles.