IEPM-BW (or PingER on steroids) and the PPDG

Les Cottrell – SLAC
Presented at the PPDG meeting, Toronto, Feb 2002

www.slac.stanford.edu/grp/scs/net/talk/ppdg-feb02.html

Partially funded by DOE/MICS Field Work Proposal on Internet End-to-end Performance Monitoring (IEPM). Supported by IUPAP. PPDG collaborator.
Overview

1. Main issues being addressed by project
2. Other active measurement projects & deployment
3. Deliverables from IEPM-BW
4. Initial results
5. Experiences
6. Forecasting
7. Passive measurements
8. Next steps
9. Scenario
• Provide a simple, robust infrastructure for:
  – Continuous/persistent and one-off measurement of high network AND application performance
  – management infrastructure – flexible remote host configuration

• Optimize impact of measurements
  – Duration, frequency of active measurements, and use passive

• Integrate standard set of measurements including: ping, traceroute, pipechar, iperf, bbcp …

• Allow/encourage adding measure/app tools

• Develop tools to gather, reduce, analyze, and publicly report on the measurements:
  – Web accessible data, tables, time series, scatterplots, histograms, forecasts …

• Compare, evaluate, validate various measurement tools and strategies (minimize impact on others, effects of app self rate limiting, QoS, compression…), find better/simpler tools

• Provide simple forecasting tools to aid applications and to adapt the active measurement frequency

• Provide tool suite for high throughput monitoring and prediction
## Other active measurement projects

<table>
<thead>
<tr>
<th>Surveyor</th>
<th>RIPE</th>
<th>AMP</th>
<th>PingER</th>
<th>NIMI</th>
<th>IEPM-BW</th>
</tr>
</thead>
<tbody>
<tr>
<td>Community</td>
<td>I2</td>
<td>Europe</td>
<td>NSF</td>
<td>HENP/ESnet/ICFA</td>
<td>Research</td>
</tr>
<tr>
<td></td>
<td>ISPs</td>
<td>ISPs</td>
<td></td>
<td></td>
<td>HENP/PPDG/Grid</td>
</tr>
<tr>
<td>Coverage</td>
<td>Mainly US</td>
<td>Mainly</td>
<td>Mainly US</td>
<td>72 Countries</td>
<td>US, CA, JP,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Europe</td>
<td></td>
<td></td>
<td>NL, CH, UK, FR</td>
</tr>
<tr>
<td>Metrics</td>
<td>One way delay, loss</td>
<td>One way delay, loss</td>
<td>RTT, loss</td>
<td>RTT, loss</td>
<td>RTT, loss, net thru, mcast</td>
</tr>
<tr>
<td>Persistent</td>
<td>Yes*</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>On demand</td>
</tr>
<tr>
<td>Topo</td>
<td>Mesh</td>
<td>Mesh</td>
<td>Mesh</td>
<td>Hierarchy</td>
<td>Hierarchy</td>
</tr>
<tr>
<td>Data acc.</td>
<td>Request</td>
<td>Member</td>
<td>Public</td>
<td>Public</td>
<td>Member</td>
</tr>
</tbody>
</table>

**Metrics**
- One way delay, loss
- RTT, loss
- RTT, loss, net thru, mcast
- RTT, loss, net & app thru
<table>
<thead>
<tr>
<th></th>
<th>PingER</th>
<th>AMP</th>
<th>Surveyor</th>
<th>RIPE</th>
<th>NIMI</th>
<th>Trace</th>
<th>IEPM-BW</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLAC</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>LBNL</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>yes</td>
</tr>
<tr>
<td>UWisc</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
<td></td>
<td>Yes</td>
<td>yes</td>
</tr>
<tr>
<td>FNAL</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
<td></td>
<td>Yes</td>
<td>yes</td>
</tr>
<tr>
<td>ANL</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
<td>yes</td>
</tr>
<tr>
<td>BNL</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>JLAB</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
<td>yes</td>
</tr>
<tr>
<td>Caltech</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>yes</td>
</tr>
<tr>
<td>SDSC</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>yes</td>
</tr>
</tbody>
</table>

+CERN, IN2P3, INFN(Milan, Rome, Trieste), KEK, RIKEN, NIKHEF, DL, RAL, TRIUMF
+GSFC, LANL, NERSC, ORNL, Rice, Stanford, SOX, UDelaware, UFla, Umich, UT Dallas
IEPM-BW Deliverables

- Understand and identify resources needed to achieve high throughput performance for Grid and other data intensive applications
- Provide access to archival and near real-time data and results for eyeballs and applications:
  - planning and expectation setting, see effects of upgrades
  - assist in trouble-shooting problems by identifying what is impacted, time and magnitude of changes and anomalies
  - as input for application steering (e.g. data grid bulk data transfer), changing configuration parameters
  - for prediction and further analysis
- Identify critical changes in performance, record and notify administrators and/or users
- Provide a platform for evaluating new SciDAC & base program tools (e.g. pathrate, pathload, GridFTP, INCITE …)
- Provide measurement/analysis/reporting suite for Grid & hi-perf sites
Results so far 1/2

- Reasonable estimates of throughput achievable with 10 sec iperf measurements
- Multiple streams and big windows are critical
  - Improve over default by 5 to 60.
  - There is an optimum windows*streams
- Continuous data at 90 min intervals from SLAC to 33 hosts in 8 countries since Dec ’01
Results so far 2/2

- 1MHz ~ 1Mbps
- Bbcp mem to mem tracks iperf
- BBFTP & bbcp disk to disk tracks iperf until disk performance limits
- High throughput affects RTT for others
  - E.g. to Europe adds ~ 100ms
  - QBSS helps reduce impact
- Archival raw throughput data & graphs already available via http

![Graphs and plots showing results and data analysis]
Forecasting

- Given access to the data one can do real-time forecasting for
  - TCP bandwidth, file transfer/copy throughput
    - E.g. NWS, *Predicting the Performance of Wide Area Data Transfers* by Vazhkudai, Schopf & Foster
- Developing simple prototype using average of previous measurements
  - Validate predictions versus observations
  - Get better estimates to adapt *frequency* of active measurements & reduce impact
    - Also use ping RTTs and route information
  - Look at need for diurnal corrections
  - Use for steering applications
- Working with NWS for more sophisticated forecasting
- Can also use on demand bandwidth estimators (e.g. pipechar, but need to know range of applicability)
**Forecast results**

*Predict* = Moving average of last 5 measurements +/- \( \sigma \)

Iperf TCP throughput SLAC to Wisconsin, Jan ‘02

% average error = \[ \text{average}(\text{abs}(\text{observe}-\text{predict})/\text{observe}) \]

<table>
<thead>
<tr>
<th>33 nodes</th>
<th>Iperf TCP</th>
<th>Bbcp mem</th>
<th>Bbcp disk</th>
<th>bbftp</th>
<th>pipechar</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>13%</td>
<td>23%</td>
<td>15%</td>
<td>14%</td>
<td>13%</td>
</tr>
<tr>
<td>% error</td>
<td>+- 11%</td>
<td>+- 18%</td>
<td>+-13%</td>
<td>+-12%</td>
<td>+-8%</td>
</tr>
</tbody>
</table>
Passive (Netflow) data

- Use Netflow measurements from border router
  - Netflow records time, duration, bytes, packets etc./flow
  - Calculate throughput from Bytes/duration for big flows
  - Validate vs. iperf
Experiences so far (what can go wrong, go wrong, go wrong, go wrong, go wrong, go wrong,...)

- Getting ssh accounts and resources on remote hosts
  - Tremendous variation in account procedures from site to site, takes up to 7 weeks, requires knowing somebody who cares, sites are becoming increasingly circumspect
  - Steep learning curve on ssh, different versions
  - Getting disk space for file copies (100s Mbytes)
- Diversity of OSs, userids, directory structures, where to find perl, iperf ..., contacts
  - Required database to track
    - Also anonymizes hostnames, tracks code versions, whether to execute command (e.g. no ping if site blocks ping) & with what options,
  - Developed tools to download software and to check remote configurations
- Remote server (e.g. iperf) crashes:
  - Start & kill server remotely for each measurement
- Commands lock up or never end:
  - Time out all commands
  - Some commands (e.g. pipechar) take a long time, so run infrequently
- AFS tokens to allow access to .ssh identity timed out, used trscron
- Protocol port blocking
  - Ssh following Xmas attacks; bbftp, iperf ports, big variation between sites
  - Wrote analyses to recognize and worked with site contacts
  - Ongoing issue, especially with increasing need for security, and since we want to measure inside firewalls close to real applications
- Simple tool built for tracking problems
Next steps

- Develop/extend management, analysis, reporting, navigating tools – improve robustness, manageability, optimize measurement frequency
- Understand correlations & validate various tools
- Tie into PingER reporting *(in beta)*
- Improve predictors and quantify how they work, provide tools to access
- Tie in passive Netflow measurements
- Add gridFTP (with Allcock@ANL) & new BW measurers and validate – with Jin@LBNL, Reidi@Rice
- Make data available via http to interested & “friendly” researchers
  - CAIDA for correlation and validation of Pipechar & iperf etc. *(sent documentaion)*
  - NWS for forecasting with UCSB *(sent documentation)*
  - ANL *(done)*
- Make data available by std methods (e.g. MDS, GMA) – with Dantong@BNL
- Make tools portable, set up other monitoring sites, e.g. PPDG sites
- Work with NIMI/GIMI to deploy dedicated engines
  - More uniformity, easier management, greater access granularity & authorization
  - Still need non dedicated:
    - Want measurements from real application hosts, closer to real end user
    - Some apps may not be ported to GIMI OS
    - Not currently funded for GIMI engines
  - Use same analysis, reporting etc.
Scenario

- BaBar user wants to transfer large volume (e.g. TByte) of data from SLAC to IN2P3:
  - Select initial windows and streams from a table of pre-measured optimal values, or use an on demand tool (extended iperf), or reasonable default if none available
  - Application uses data volume to be transferred and simple forecast to estimate how much time is needed
    - Forecasts from active archive, Netflow, on demand use one-end bandwidth estimation tools (e.g. pipechar, NWS TCP throughput estimator)
  - If estimate duration is longer than some threshold, then more careful duration estimate is made using diurnal forecasting
  - Application reports to user who decides whether to proceed
  - Application turns on QBSS and starts transferring

- For long measurements, provide progress feedback, using progress so far, Netflow measurements of this flow for last few half hours, diurnal corrections etc.
  - If falling behind required duration, turn off QBSS, go to best effort
  - If throughput drops off below some threshold, check for other sites
More Information

- IEPM/PingER home site:  
  - www-iepm.slac.stanford.edu/
- IEPM/BW site  
  - www-iepm.slac.stanford.edu/bw
- Bulk throughput site:  
  - www-iepm.slac.stanford.edu/monitoring/bulk/
- SC2001 & high throughput measurements  
  - www-iepm.slac.stanford.edu/monitoring/bulk/sc2001/
- QBSS measurements  
  - www-iepm.slac.stanford.edu/monitoring/qbss/measure.html
- Netflow  