Inca User-level Grid Monitoring

Shava Smallen
ssmallen@sdsc.edu

SC’08
November 19, 2008
Goal: reliable grid software and services for users

- Over 750 TF
- Over 30 PB of online and archival data storage
- Connected via dedicated multi-Gbps links
- 30-63 software packages and 6-23 services per resource

11 TeraGrid sites, 21 resources
Related Grid monitoring tools

- **BIG BROTHER™**
- **MonALISA**
  - MONitoring Agents using a Large Integrated Services Architecture
- **GridICE**
  - the eyes of the Grid
- **Nagios®**
- **Service Availability Monitoring**

Inca’s primary objective: user-level Grid monitoring
User-level grid monitoring

• Runs from a standard user account
• Executes using a standard GSI credential
• Uses tests that are developed and configured based on user documentation
• Automates periodic execution of tests
• Verifies user-accessible Grid access points
• Centrally manages monitoring configuration
• Easily updates and maintains monitoring deployment
Who benefits from user-level grid monitoring?

- Grid managers
  - Verify requirements are fulfilled by resource providers
  - Identify failure trends

- System administrators
  - Email notification
  - Debugging support

- End users
  - Debug user account/environment issues
  - Advanced users: feedback to Grid/VO
Inca provides user-level grid monitoring

- Stores and archives a wide variety of monitoring results
- Captures context of monitoring result as it is collected
- Eases the writing, deploying, and sharing of new tests or benchmarks
- Flexible and comprehensive web status pages
- Secure
Reporters collect monitoring data

- Executable programs that measure some aspect of the system or installed software
- Supports a set of command-line options and writes XML to stdout
- Schema supports multiple types of data
- Extensive library support for perl and python scripts (most reporters < 30 lines of code)
- Independent of other Inca components
Repositories support sharing

- Collection of reporters available via a URL
- Supports package dependencies
- Packages versioned to allow for automatic updates
- Inca project repository contains 150+ reporters
  - Version, unit test, performance benchmark reporters
  - Grid middleware and tools, compilers, math libraries, data tools, and viz tool
Agent provides centralized configuration and management

- Implements the configuration specified by Inca administrator
- Stages and launches a reporter manager on each resource
- Sends package and configuration updates
- Manages proxy information
- Administration via GUI interface (incat)

Screenshot of Inca GUI tool, incat, showing the reporters that are available from a local repository
Depot stores and publishes data

- Stores configuration information and monitoring results
- Provides full archiving of reports
- Uses relational database backend via Hibernate
- Supports HQL and predefined queries
- Supports plug-in customization (e.g., email notifications, downtimes)
- Web services - Query data from depot and return as XML
Consumer displays data

- Current and historical views
- Web application packaged with Jetty
- JSP 2.0 pages/tags to query data and format using XSLT
- CeWolf/JFreeChart to graph data
Tests Summary

Resource status history

Error history summary

Related test histories

Test status by package and resource

Weekly status report

Cumulative test status by resource

Individual test result details

Individual test history

Historical

Current status
Software status and deployments

Current software version: 2.4
(available from Inca website)

http://inca.sdsc.edu
Inca TeraGrid deployment

- Running since 2003
- Total of 2660 tests running on 20 login nodes, 3 grid nodes, and 3 servers
- Coordinated software and services
- Cross-site tests
- GRAM usage
- CA certificate and CRL checking
- Resource registration in information services

Screenshot of Inca status pages for TeraGrid

http://inca.teragrid.org/
Measuring Performance Variation on the TeraGrid

- **Stage 1:** MPI ping pong
  - Collecting results since Oct 1
  - Runs every 12 hours on 16 processors, <10 minutes
  - Running on NCSA’s Abe, NICS’ kraken

- **Stage 2:** PARATEC
  - Collecting results since November 1
  - Runs every 12 hours on 256 processors, <30 min
  - Running on NCSA’s Abe, NICS’ kraken

```
<table>
<thead>
<tr>
<th></th>
<th>abe</th>
<th>kraken</th>
</tr>
</thead>
<tbody>
<tr>
<td>cluster.compiler.gcc.version</td>
<td>3.4.6</td>
<td>3.3.3</td>
</tr>
<tr>
<td>ofed.version</td>
<td>1.1</td>
<td>n/a</td>
</tr>
<tr>
<td>paratec.performance</td>
<td>1254.436649</td>
<td>920.081393</td>
</tr>
<tr>
<td>pingpong.performance</td>
<td>Min: 871.99</td>
<td>Min: 1411.83</td>
</tr>
<tr>
<td></td>
<td>Mean: 900.19</td>
<td>Mean: 1490.3</td>
</tr>
<tr>
<td></td>
<td>Max: 905.09</td>
<td>Max: 1540.97</td>
</tr>
</tbody>
</table>
```

Latest PingPong and PARATEC results

- Mean MPI ping pong bandwidth history
Inca GLEON deployment

- Sensors in lake: dissolved oxygen level, temperature, velocity (some), etc.

- Monitoring Data Turbine deployments since Oct 2007

- Total of 26 tests running on data server at SDSC and windows box in Northern Temperate Lakes in Wisconsin

http://inca-gleon.sdsc.edu
Inca monitoring benefits end users

“Inca reported errors mirror failures we’ve observed and as they are addressed we’ve noticed an improvement in TeraGrid’s stability.”

-- Suresh Marru (LEAD developer)

- Tests resources and services used by LEAD. E.g.
  - Pings service every 3 mins
  - Verifies batch job submission every hour
- Automatically notifies admins of failures
- Show week of history in custom status pages
Benefits of using Inca

- Detect problems before the users notice them
- Easy to write and share tests and benchmarks
- Easy to deploy and maintain
- Flexible and comprehensive displays
Inca Information

- Announcements: inca-users@sdsc.edu
- Email: inca@sdsc.edu
- Website: http://inca.sdsc.edu

- Supported by: