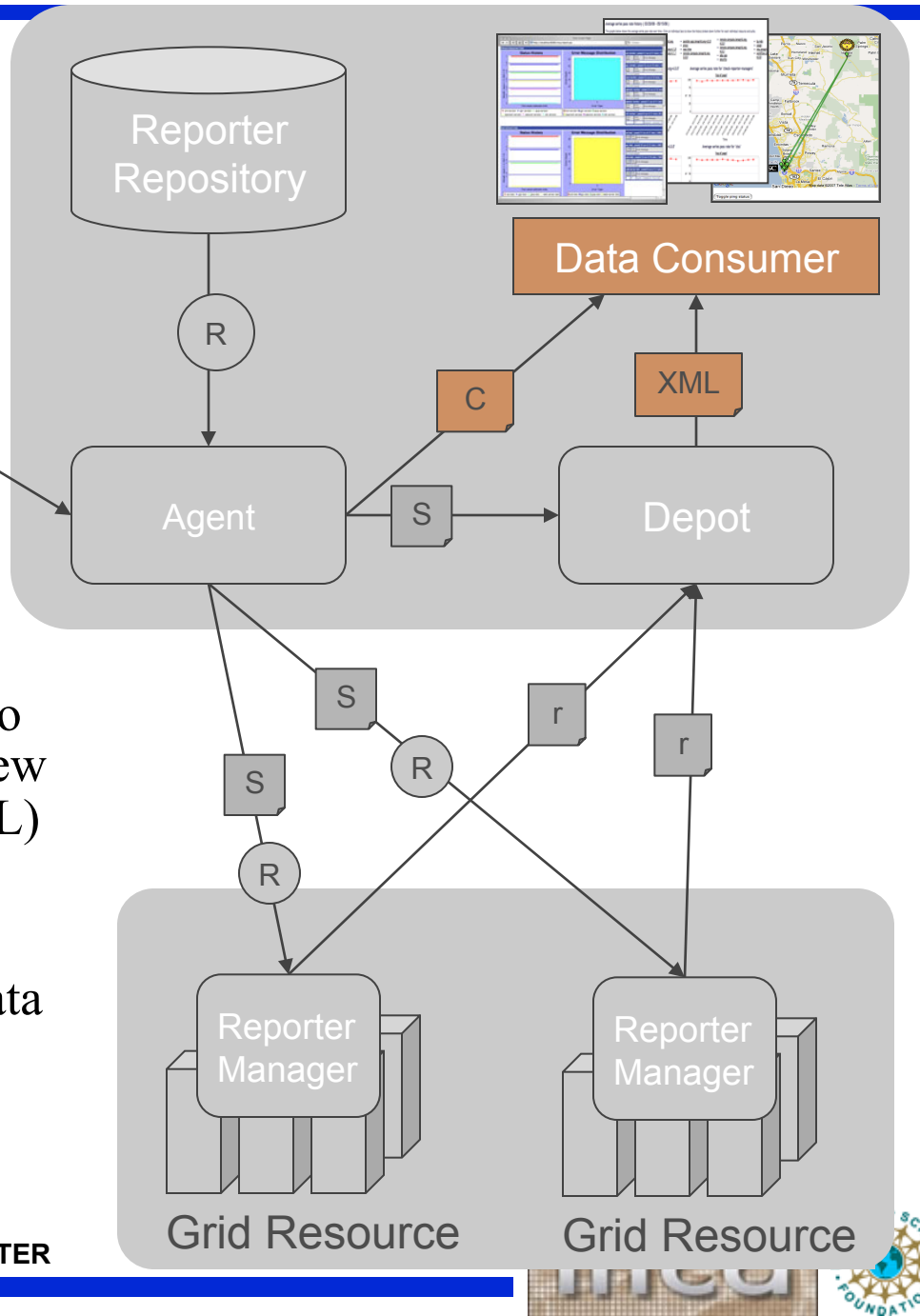
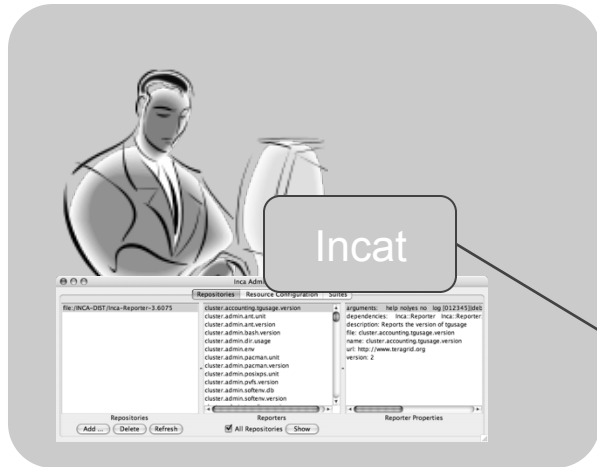


---

# Writing an Inca consumer

Shava Smallen  
**ssmallen@sdsc.edu**

Inca Workshop  
August 27, 2010

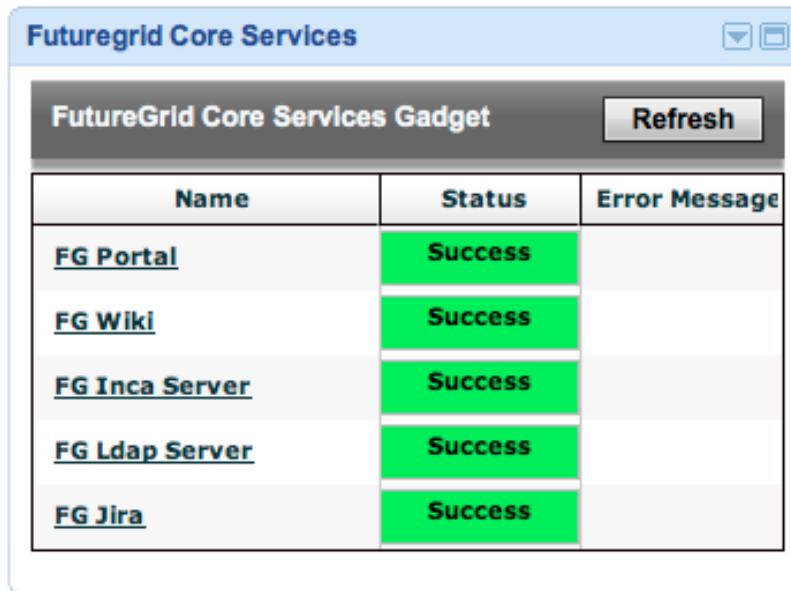


## Writing a data consumer

- Extend the bundled data consumer to format data differently or provide new views (i.e., add new JSP or new XSL)
- Use Inca Client APIs to query data from depot and agent and display data outside of bundled data consumer

# Custom Inca consumer examples

Google  
gadget



Name	Status	Error Message
<a href="#">FG Portal</a>	Success	
<a href="#">FG Wiki</a>	Success	
<a href="#">FG Inca Server</a>	Success	
<a href="#">FG Ldap Server</a>	Success	
<a href="#">FG Jira</a>	Success	

Text History

```
% ./bin/getIncaHistory -a '2010-01-01'  
ranger.tacc.utexas.edu gram-gatekeeper-ping  
18:12:29,157 INFO [main] IncaHistory:118 -  
Attempting to connect to http://  
sapa.sdsc.edu:8001  
2010-01-01 00:00:08.0 true  
2010-01-01 00:00:08.0 true  
...
```

iPhone webapp



sampleSuite

- localResource
  - ant\_helloworld\_compile\_test: pass
  - ant\_version: 1.7.1
  - gcc\_hello\_world: pass
  - gcc\_version: 4.2.1
  - java\_hello\_world: pass
  - java\_version: 1.6.0\_17
  - openssh\_version: 5.2p1
  - openssl\_version: 0.9.8k
  - vtk-nvgl\_version: error
  - wget\_page\_test: error

---

# Inca provides support for custom data consumers

- Inca Client APIs
  - Provide access to report data from the depot and configuration information from the agent (REST, Web Services, Java, Perl)
- Inca XML schemas
  - Provide data for reporting all test details or summaries

---

# Overview of Client Methods

## Client

connect()  
ping()

## DepotClient

insertReport(*resource, report, sysusage, stderr, context*)  
queryDatabase()  
queryGuids()  
queryHql(*hql*)  
queryInstance(*nickname, resource, collected*)  
queryLatest(*expr*)  
queryPeriod(*begin, end, expr*)  
queryStatusHistory(*period, begin, end, select*)  
updateSuite(*xml*)

## AgentClient

getCatalog()  
getConfig  
setConfig(*xml*)

---

# DepotClient queryHql

- Takes an HQL (Hibernate's SQL dialect) query as a parameter
- Returns an array of XML, one element per object selected
- Example:

```
client.queryHql("select ii from Report r, InstanceInfo ii where ii.reportId= r.id  
and r.resource = 'repo'") =>
```

```
<instance>
```

```
<collected>2008-09-03 14:25:03.0</collected>
```

```
<committed>2008-09-03 14:25:04.108</committed>
```

```
<cpuUsageSec>1.462776</cpuUsageSec>
```

```
<memoryUsageMB>25.132812</memoryUsageMB>
```

```
<wallClockTimeSec>2.002927</wallClockTimeSec>
```

```
</instance>
```

```
...
```

---

# DepotClient queryGuids

- Returns an array of suite identifiers
- Example:

`client.queryGuids()` =>

`incas:/sapa.sdsc.edu:6323/app-support.teragrid.org-4.0.3`

`incas:/sapa.sdsc.edu:6323/app-support.teragrid.org-4.2.0`

`incas:/sapa.sdsc.edu:6323/check-reporter-managers`

`incas:/sapa.sdsc.edu:6323/core.teragrid.org-4.0.0`

`incas:/sapa.sdsc.edu:6323/core.teragrid.org-4.0.2`

`incas:/sapa.sdsc.edu:6323/core.teragrid.org-4.1.0`

`incas:/sapa.sdsc.edu:6323/core.teragrid.org-4.2.0`

`incas:/sapa.sdsc.edu:6323/core.teragrid.org-4.2.1`

`incas:/sapa.sdsc.edu:6323/coscheduling.teragrid.org-1.0.0`

`incas:/sapa.sdsc.edu:6323/data-management.teragrid.org-4.0.0`

`incas:/sapa.sdsc.edu:6323/data-movement-clients.teragrid.org-4.2.0`

`incas:/sapa.sdsc.edu:6323/data-movement-servers.teragrid.org-4.2.0`

...

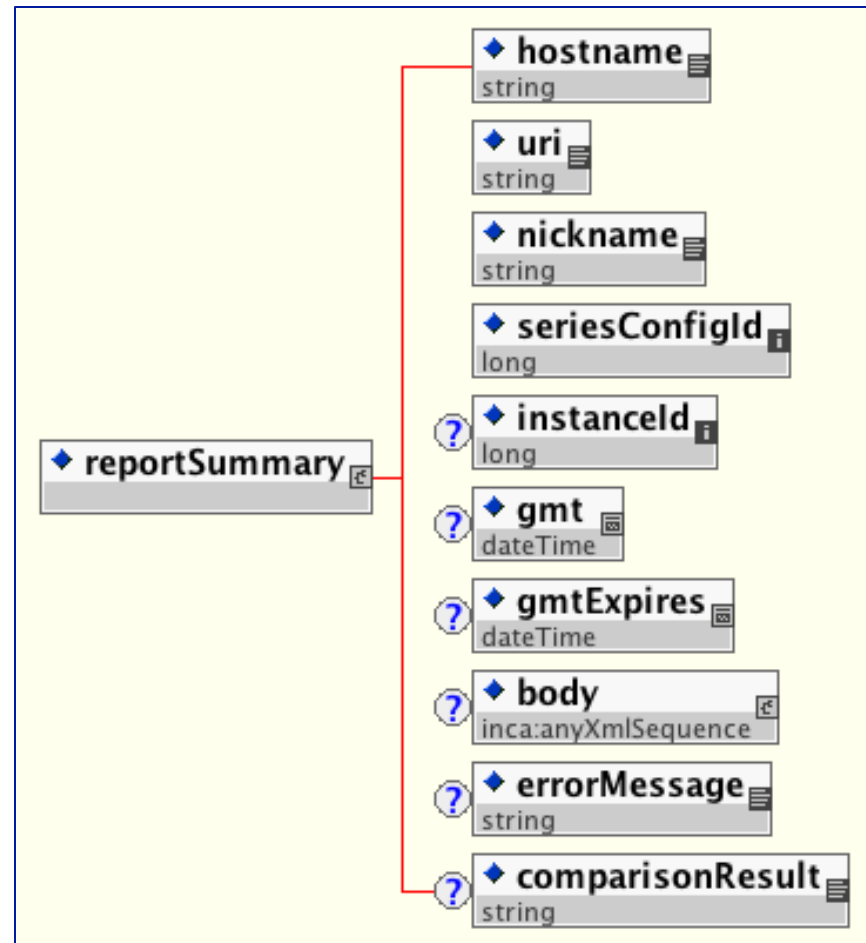
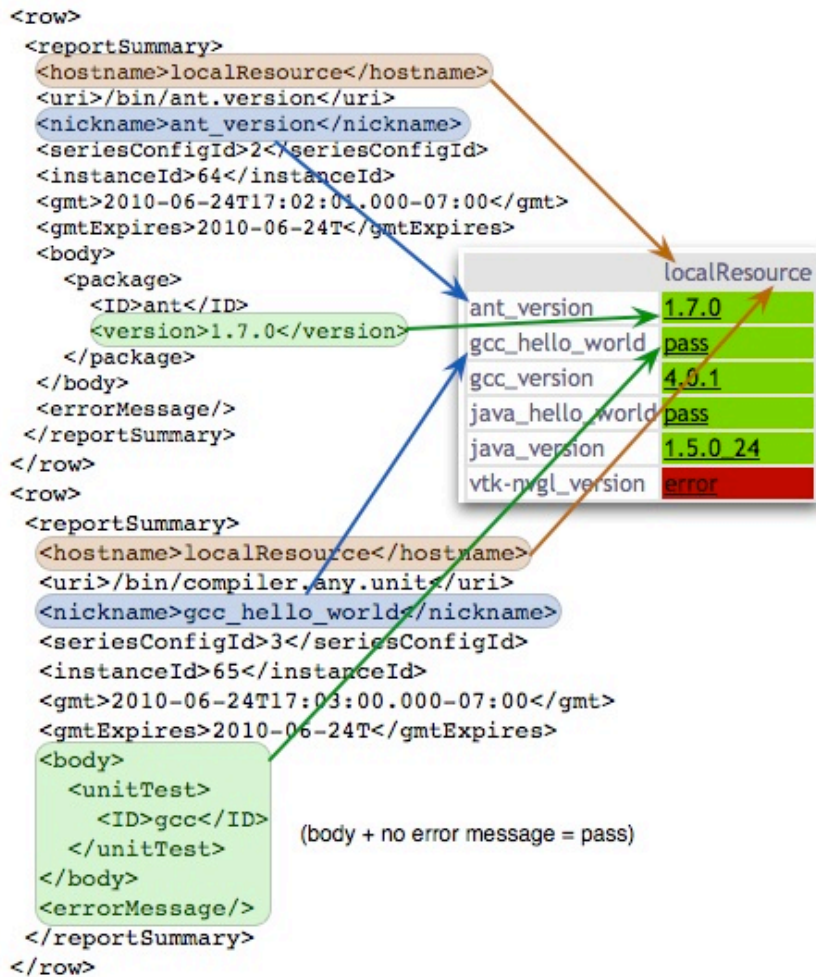
# DepotClient queryLatest

- Takes a select WHERE clause as a parameter
- Returns combined latest report instance info for each series
- Example:

```
client.queryLatest("suite.guid = 'incas://hogg.sdsc.edu:6323/sampleSuite'") =>
<reportSummary xmlns="http://inca.sdsc.edu/queryResult/reportSummary_2.0">
  <hostname xmlns="">localResource</hostname>
  <uri xmlns="">file:///users/u4/train31/IncaInstall/bin/./Inca-Reporter-5.10644/bin/grid.wget.unit</uri>
  <nickname xmlns="">wget_page_test</nickname>
  <seriesConfigId xmlns="">10</seriesConfigId>
  <instanceId xmlns="">169</instanceId>
  <gmt xmlns="">2008-09-03T14:40:01.000-07:00</gmt>
  <gmtExpires xmlns="">2008-09-03T15:00:01.869-07:00</gmtExpires>
  <body xmlns="">
    <unitTest xmlns:rep="http://inca.sdsc.edu/dataModel/report_2.1">
      <ID>wget</ID>
    </unitTest>
  </body>
  <errorMessage xmlns=""></errorMessage>
</reportSummary>
...
```



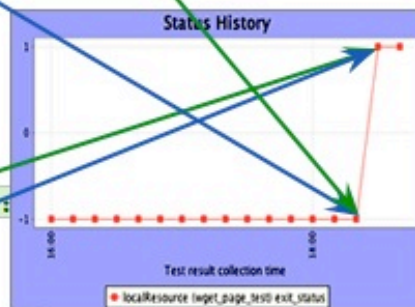
# ReportSummary schema getLatest ()



# Sidebar: interpreting status

```
<row>
  <resource>localResource</resource>
  <nickname>gcc_version</nickname>
  <instanceId>3</instanceId>
  <reportId>3</reportId>
  <configId>10</configId>
  <collected>2010-06-24T18:20:00.000-07:00</collected>
  <exit_status>1</exit_status>
  <exit_message/>
  <body>
    <package>
      <ID>gcc</ID>
      <version>4.1.2</version>
    </package>
  </body>
  <comparisonResult>Failure:gcc</comparisonResult>
</row>
```

```
<row>
  <resource>localResource</resource>
  <nickname>gcc_version</nickname>
  <instanceId>13</instanceId>
  <reportId>2</reportId>
  <configId>11</configId>
  <collected>2010-06-24T18:30:01.000-07:00</collected>
  <exit_status>1</exit_status>
  <exit_message/>
  <body>
    <package>
      <ID>gcc</ID>
      <version>4.1.2</version>
    </package>
  </body>
  <comparisonResult/>
</row>
```



"collected" used for x-axis,  
"comparisonResult" used for y-axis  
when it exists, otherwise  
"exit\_status" used for y-axis

```
if ( exists comparisonResult ) {
  if ( comparisonResult == 'Success' ) {
    println "Pass"
  } else {
    println "Failed: " + comparisonResult;
    if ( exists errorMessage ) {
      println errorMessage;
    }
  }
} else {
  if ( body != '' ) {
    println "Pass";
  } else {
    println errorMessage;
  }
}
```

---

# DepotClient queryPeriod

- Takes a time period (ms) and a select WHERE clause as a parameter
- Returns combined report history instance summary info for each series
- Example:

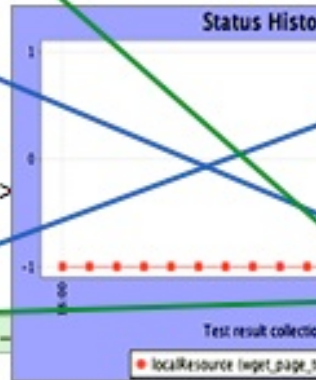
```
client.queryPeriod(1282685125000, 1282771525000, "suite.guid = 'incas://hogg.sdsc.edu:6323/sampleSuite'") =>
<graphInstance>
  <resource>localResource</resource>
  <nickname>java_hello_world</nickname>  <instanceId>2</instanceId>  <reportId>2</reportId>
  <configId>5</configId>
  <collected>2010-08-25T16:45:10.000-07:00</collected>
  <exit_status>1</exit_status>
  <exit_message/>
  <body>
    <unitTest xmlns:rep="http://inca.sdsc.edu/dataModel/report_2.1">
      <ID>javac</ID>
    </unitTest>
  </body>
  <comparisonResult>Success</comparisonResult>
</graphInstance>
...
```

# GraphInstance schema getPeriod

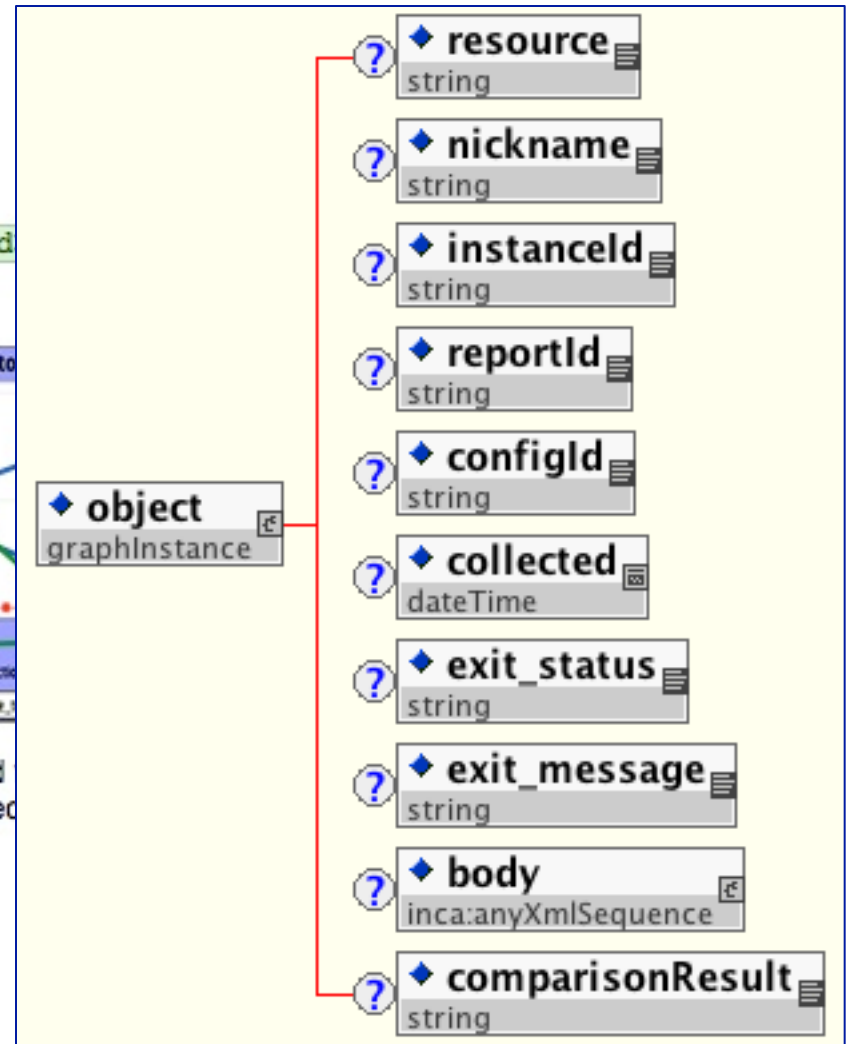
```

<row>
  <resource>localResource</resource>
  <nickname>wget_page_test</nickname>
  <instanceId>142</instanceId>
  <reportId>2</reportId>
  <configId>11</configId>
  <collected>2010-06-24T18:20:00.000-07:00</collected>
  <exit_status>-1</exit_status>
  <exit_message>wget command failed</exit_message>
  <body/>
  <comparisonResult/>
</row>
<row>
  <resource>localResource</resource>
  <nickname>wget_page_test</nickname>
  <instanceId>152</instanceId>
  <reportId>11</reportId>
  <configId>11</configId>
  <collected>2010-06-24T18:30:01.000-07:00</collected>
  <exit_status>1</exit_status>
  <exit_message/>
  <body>
    <unitTest>
      <ID>wget</ID>
    </unitTest>
  </body>
  <comparisonResult/>
</row>

```



"collected" used  
"exit\_status" used



# DepotClient queryStatusHistory

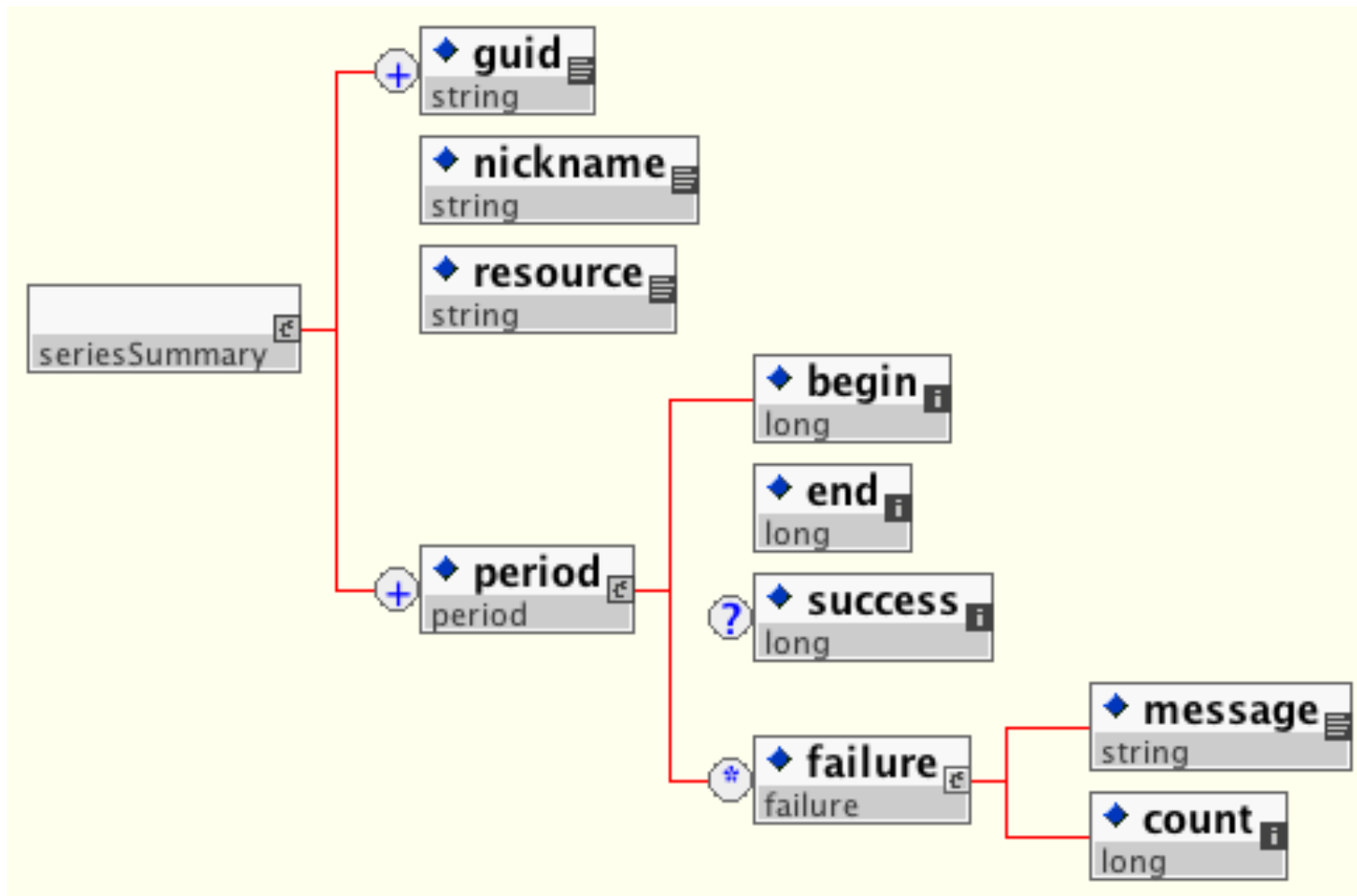
- Takes a time period (ms), begin, end, and a select WHERE clause as parameters
- Returns report history statistics for each series
- Example:

```
client.queryPeriod("WEEK", 1282685125000, 1282771525000, "suite.guid = 'incas://hogg.sdsc.edu:6323/sampleSuite'") =>
```

```
<series>
  <guid>incas://gw41.quarry.iu.teragrid.org:6323/Cluster</guid>
  <nickname>test_batch_job</nickname>
  <resource>iu-xray</resource>
  <period>
    <begin>1280432084455</begin>
    <end>1281036884454</end>
    <success>42</success>
  </period>...
  <period>
    <begin>1282246484455</begin>
    <end>1282851284454</end>
    <success>162</success>
    <failure><message>BATCH: Unable to retrieve reporter output</message><count>2</count></failure>
    <failure><message>BATCH: Unable to submit reporter perl /mnt/lustre_server/u/inca/incaReporterManager/sub32351.pl: System
      call failure. qsub: cannot connect to server (null) (errno=15010) System error:
    </message><count>1</count></failure>
  </period>
</series> ...
```

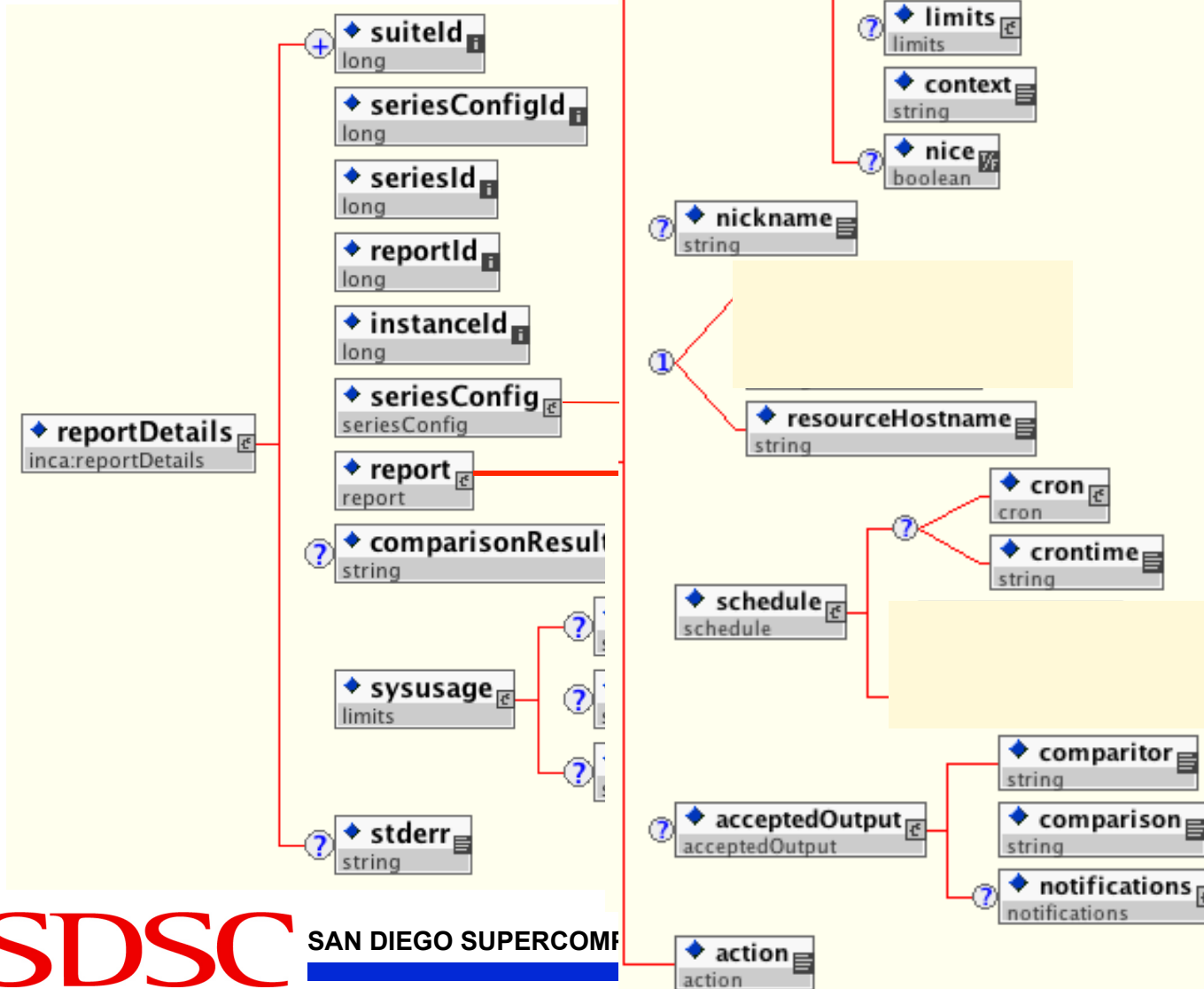


# seriesSummary used to display historical stats queryStatusHistory()



# ReportDetails use get

# instance info



# AgentClient getConfig

- Returns XML for Inca installation configuration
- Example:

`client.getConfig() =>`

```
<inca:inca xmlns:inca="http://inca.sdsc.edu/dataModel/inca_2.0">
  <repositories><repository>file:/users/u4/train31/IncaInstall/bin/./Inca-Reporter-5.10644</repository></repositories>
  <resourceConfig><resources><resource>
    <name>defaultGrid</name>
    <xpath>//resource[matches(name, "localSite")]</xpath>
    <macros> ... </macros>
  </resource></resources></resourceConfig>
  <suites><suite><seriesConfigs><seriesConfig>
    <series>
      <name>cluster.admin.ant.unit</name>
      <args>
        <arg><name>version</name><value>no</value></arg>
      </args>
      <context>@pre@@@@@post@</context>
    </series>
    <nickname>ant_helloworld_compile_test</nickname>
    <resourceSetName>defaultGrid</resourceSetName>
    <schedule><crontime>1-59/10 * * * *</crontime></schedule>
    <action>add</action>
  </seriesConfig></seriesConfigs></suite></suites>
```



---

# AgentClient getCatalog

- Returns Packages.gz contents for a given Reporter Repository
- Example:

```
client.getCatalog('http://inca.sdsc.edu/repository/latest') =>
```

```
version:2
```

```
name:viz.tools.imagemagick.version
```

```
url:http://www.imagemagick.org/
```

```
file:bin/viz.tools.imagemagick.version
```

```
description:Reports the version of the ImageMagick toolkit
```

```
arguments:
```

```
help no|yes no
```

```
log [012345]||debug|error|info|system|warn 0
```

```
verbose [012] 1
```

```
version no|yes no
```

```
dependencies:
```

```
Inca::Reporter
```

```
Inca::Reporter::Version
```

```
version:2
```

```
name:viz.tools.netpbm.version
```

```
url:http://netpbm.sourceforge.net/
```

```
file:bin/viz.tools.netpbm.version
```

```
...
```

---

# Overview of Inca Client APIs

- **REST URLs** - Access to subset of DepotClient functions
- **Web Services API** - Access to subset of DepotClient and AgentClient functions
- **Java API** – Access to full set of DepotClient and AgentClient functions. Requires a valid credential.
- **Perl API** - Access to full set of DepotClient and AgentClient functions. Requires a valid credential.

---

# REST URLs

- Subset of functionality provided by getLatest and getPeriod

[http://hostname:8080/inca/XML/rest/<suiteName>|<queryName>\[/<resourceId>\[/<seriesNickname>\[/week|month|quarter|year\]\]\]](http://hostname:8080/inca/XML/rest/<suiteName>|<queryName>[/<resourceId>[/<seriesNickname>[/week|month|quarter|year]]])

E.g.,

<http://hostname:8080/inca/XML/rest/sampleSuite>

[http://hostname:8080/inca/XML/rest/sampleSuite/localResource/ant\\_version](http://hostname:8080/inca/XML/rest/sampleSuite/localResource/ant_version)

# Web Services API

```
use SOAP::Lite;
my $ws = SOAP::Lite->service
    ("file:/home/inca/2.6/etc/IncaWS.wsdl");
# check agent and depot are available
print $ws->pingAgent('hello agent'), "\n";
print $ws->pingDepot('hello depot'), "\n";
# get the Inca configuration
print $ws->getConfig(), "\n";
my $guid = $ws->queryGuids();
# get the latest instances of a suite
my $results = $ws->querySuite( $guid );
for my $result ( @{$results} ) { print $result;}
```

[http://inca.sdsc.edu/releases/2.6/wsdocs/IncaWS\\_wsdl.html](http://inca.sdsc.edu/releases/2.6/wsdocs/IncaWS_wsdl.html)

# Perl APIs

```
use Inca::AgentClient;

print "Please enter a password: ";
chomp( my $pass = <STDIN> );
my $agentclient = new Inca::AgentClient( host => 'localhost', port => 6323,
    auth => 1, cert => 'etc/clientcert.pem', key => 'etc/clientkey.pem',
    password => $pass, trusted => 'etc/trusted/51b63f6c.0' );
if ( defined $agentclient->getError() ) {
    die "Unable to connect:" . $agentclient->getError();
}
# check agent is available
print $agentclient->ping('hello agent'), "\n";

# get the Inca configuration
print $agentclient->getConfig(), "\n";
```

<http://inca.sdsc.edu/releases/2.6/wsdocs/index.html>

# Java APIs

```
import edu.sdsc.inca.AgentClient;

public class AgentClientTest {
    public static void main(String args[]) {
        AgentClient agentClient = new AgentClient();
        agentClient.setServer("rocks-101.sdsc.edu", 6323);
        agentClient.setCertificatePath("clientcert.pem");
        agentClient.setKeyPath("clientkey.pem");
        agentClient.setTrustedPath("trusted");
        // read password from stdin
        agentClient.setPassword(password);
        try {
            agentClient.connect();
            System.out.println(agentClient.commandPing("hello agent"));
            System.out.println(agentClient.getConfig());
        } catch ( Exception e ) {}
    }
}
```

<http://inca.sdsc.edu/releases/2.6/javawsdocs/overview-summary.html>

---

# Inca provides support for custom data consumers

- Inca Client APIs
  - REST URLs
  - Web Services
  - Java
  - Perl
- Inca XML schemas
  - Report Summaries
  - Graph Instances
  - Series Summary
  - Report Details

---

## Agenda -- Day 2

9:00 - 9:30	Inside the Inca Depot
9:30 - 10:10	Data display (data consumers)
10:20 - 11:00	Writing data consumers
11:00 - 12:00	Hands-on: Data display (data consumers)