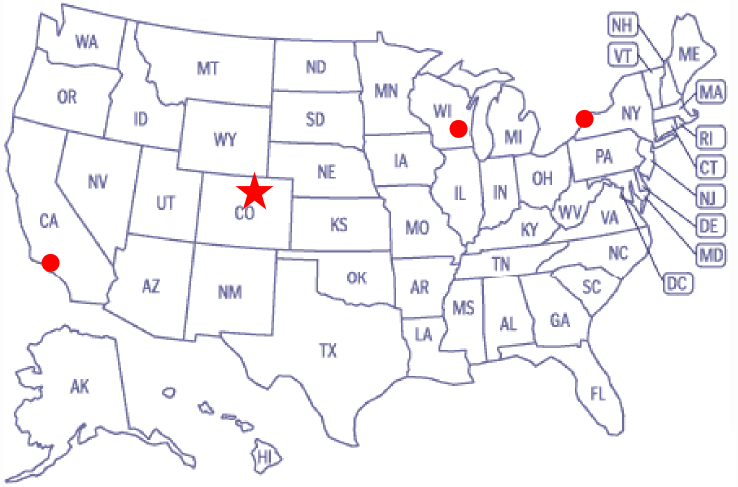
CWS4DB: A Customizable Web Service for Efficient Access to Distributed Nuclear Physics Relational Databases

FY 2008 SBIR Phase II Proposal Award Number: DE-FG02-07ER84757

Dr. Mark L. Green, PI

Tech-X Corporation, Buffalo Office
Systems Integration Group

Headquarters Boulder, CO



Buffalo, NY

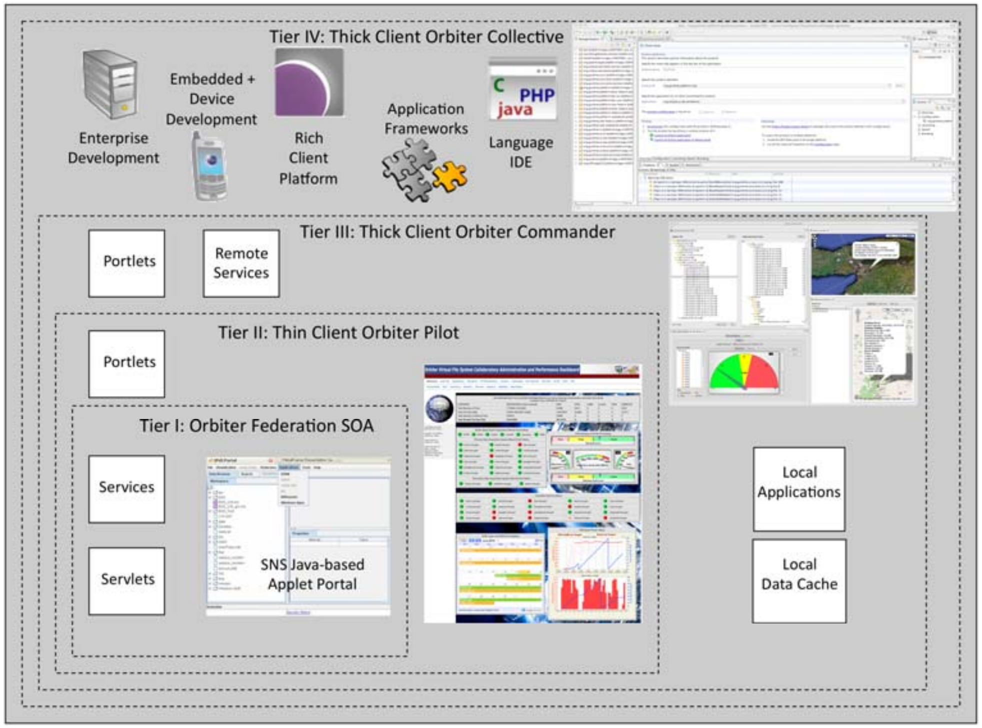
Tech-X Orbiter Project

- **Orbiter is an end-to-end framework** delivering fast and secure solutions through both thin-client web access and thick-client desktop application suites and modules. These applications leverage the information-sharing capabilities of Orbiter in providing powerful and personalized web-accessible components.
- **Service Oriented Architectures (SOAs)** have been proven to be a popular design for building reliable and scalable large-scale software systems, borrowing from earlier Object Oriented Programming (OOP) techniques of encapsulation, cohesion, and the use of abstraction layers behind well-defined public APIs. Orbiter Federation services, built upon industry standards, offer fast and secure access to a wide range of capabilities.

https://orbiter.txcorp.com

Orbiter Multitier Portal Architecture (MPA)

- Through the Multitier Portal Architecture (MPA) Orbiter Federation services are delivered directly to end-users via a variety of rich interactive interfaces. The MPA allows increasingly sophisticated capabilities to be rapidly developed to suit a wide range of user requirements, and the foundation provided by Orbiter Federation enables these capabilities to be delivered swiftly and securely to end-users.
- Framework for delivering capabilities to thin- and thick-clients using the Orbiter RESTful SOA
- Flexible and re-usable architecture for developing capabilities for thin web clients and thick local clients
- Comprised of four tiers:
    - Tier I: Orbiter Federation SOA
        - Low-level RESTful services
    - Tier II: Thin-Client Orbiter Pilot
        - Light weight client access
    - Tier III: Thick-Client Orbiter Commander
        - Fully capable installed application
    - Tier IV: Thick-Client Orbiter Collective
        - IDE for Orbiter development


Tier IV: Thick Client Orbiter Collective
Enterprise Development
Embedded + Device Development
Rich Client Platform
Application Frameworks
C
PHP
java
Language IDE
Tier III: Thick Client Orbiter Commander
Portlets
Remote Services
Tier II: Thin Client Orbiter Pilot
Portlets
Tier I: Orbiter Federation SOA
Services
Servlets
SNS Java-based Applet Portal
Local Applications
Local Data Cache

https://orbiter.txcorp.com

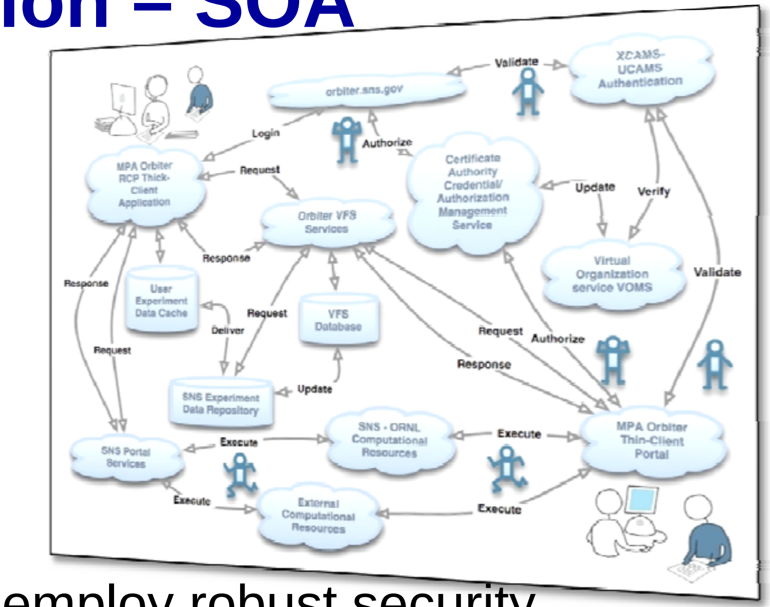
TECH-X CORPORATION

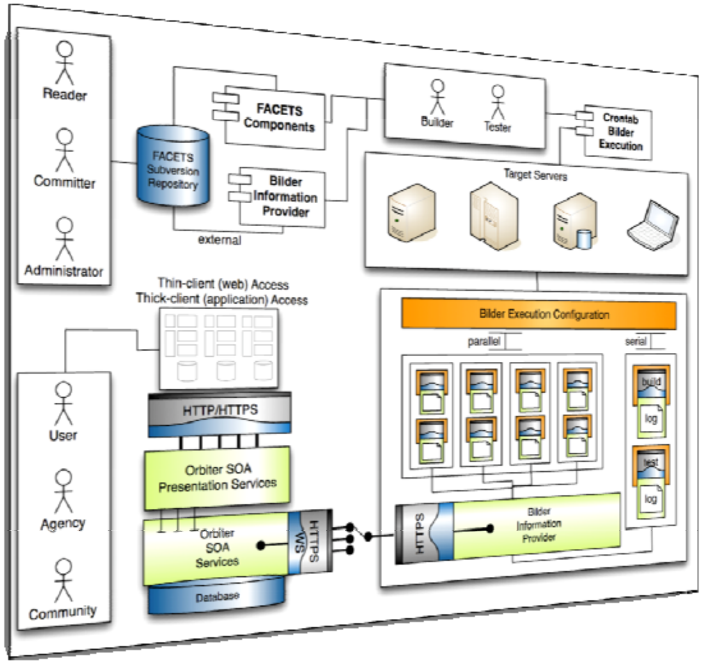
Tech-X Orbiter Project

- **Federation** provides a Service Oriented Architecture (SOA) of web services, delivering powerful, lightweight, secure, and scalable capabilities.
- **Pilot** delivers Federation web services through web-accessible thin-clients. These gateway and portal clients deliver Orbiter capabilities through easy-to-use web interfaces.
- **Commander** is a rich cross-platform desktop application that provides access to Federation services while allowing Orbiter systems to interact directly with local compute resources.
- **Collective** opens the door to advanced collaboration across a wide range of associations, facilities, and institutions. Orbiter meets the needs of these organizations through the development of integrated cross-platform applications that enable the full value of third party products and services.

Orbiter Federation – SOA

- Tier I of the Orbiter Multitier Portal Architecture
- Orbiter services are implemented as Representational State Transfer (RESTful) web services that deliver functionality through a well-defined API.


orbiter.sns.gov
Validate
XCAMS-UCAMS Authentication
Login
Authorize
MPA Orbiter RCP Thick-Client Application
Request
Certificate Authority Credential/ Authorization Management Service
Update
Verify
Orbiter VFS Services
Response
Response
User Experiment Data Cache
Virtual Organization service VOMS
Validate
Request
Deliver
VFS Database
Request
Authorize
Request
Response
SNS Experiment Data Repository
Update
SNS Portal Services
Execute
SNS - ORNL Computational Resources
Execute
MPA Orbiter Thin-Client Portal
Execute
External Computational Resources
Execute



- These services employ robust security standards including SSL and signed requests that ensure client identities, the integrity of their RESTful service calls, and the privacy of their transmissions.
- Orbiter Web Services use SSL encryption, access key identifiers, timestamps, and private key signatures, ensuring the privacy, authorization, and request integrity of all interactions.

Tech-X Orbiter Project

- Orbiter Pilot, Orbiter Commander, and the Orbiter Collective demonstrate how access to Orbiter Federation resources and services can be provided through **reliable, scalable, and interactive scientific gateways**.
- Its **flexible cross-platform desktop solutions** present the user with a rich and customizable interface to data, information, computational resources, and enterprise application bases.
- Orbiter **solutions are inherently scalable**, where Federation, Pilot, Commander, and Collective each build modular capabilities that are focused on particular needs.
- Orbiter solutions have been routinely used for the **management and retrieval of large amounts of data** and information.

CWS4DB Project

A customizable Web Service for Efficient Access to Distributed Nuclear Physics Relational Databases

DOE NP Phase I and II – Manouchehr Farkhondeh

Tech-X: Mark L. Green (PI), Catherine L. Ruby, Sean Burley, Krishna Kantam, Srilakshmi Ramireddy

Need: As the size of NP data grows and the collaborative nature of HENP experiments increases, the ability to access differently organized relational databases remotely, efficiently, and yet in a user-friendly and interoperable manner is becoming very important.

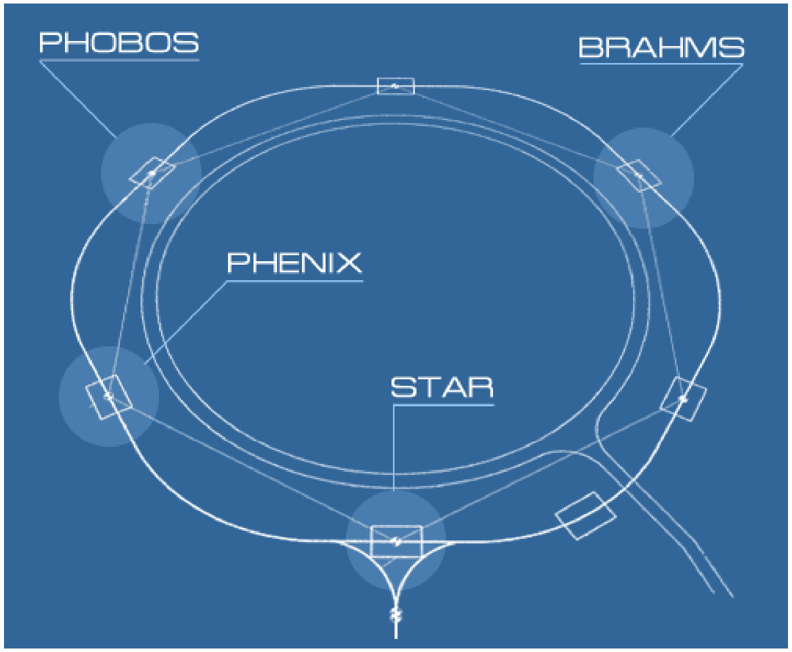
Partners: Jerome Lauret, Dmitry Arkhipkin (STAR project at BNL), Kate Keahey (Nimbus project at ANL), Doug Olson (Open Science Grid), Alexandre Vaniachine (ATLAS project ANL/CERN)

DOE Beneficiaries: Nuclear and high energy physics communities, national laboratories, and collaborative projects

Commercial Beneficiaries: Companies requiring efficient web service access to distributed relational databases with high-level database and user APIs

Problem Identification

- The importance of this project comes from the fact that a large fraction of the ever-growing data generated by Nuclear Physics (NP) experiments is stored in relational databases. For example:
  - The BNL Relativistic Heavy Ion Collider (RHIC) supports STAR (Solenoidal Tracker at the at the RHIC) which composed of 52 institutions from 12 countries, with a total of 529 collaborators;
  - relational databases (such as Condition databases, Calibration databases, and Geometry databases) are heavily used in the STAR experiment;
  - while accessing data in such databases is convenient and available for local users who are familiar with a particular database, the situation becomes more complicated when the databases are distributed and heterogeneous.

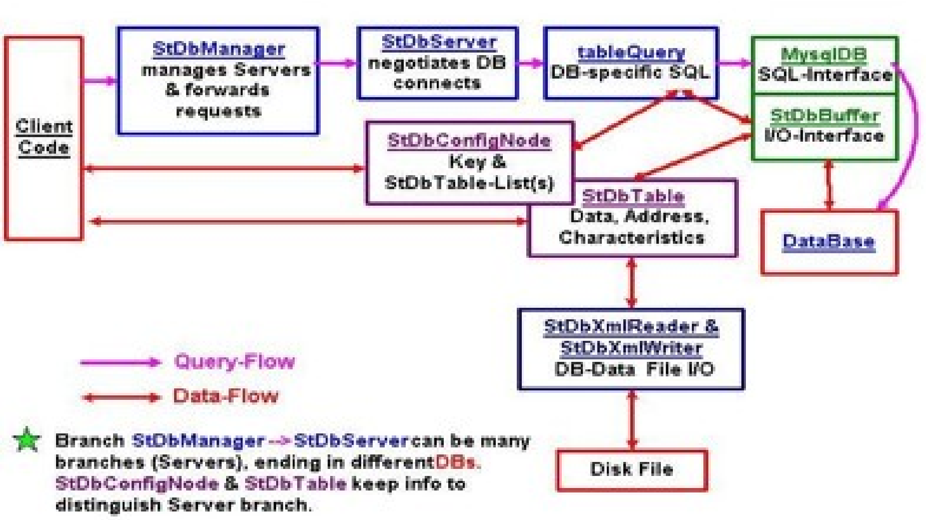


- Tech-X therefore proposed a system to overcome the outlined challenges by bridging relational databases with high-level APIs through Web services.
  - In particular, the distributed and heterogeneous nature of the databases will be addressed by creating Web services in the Orbiter Federation Service Oriented Architecture (SOA), which provides mechanisms coordinating access to diversified data resources through REST (Representational State Transfer) services, caching, authentication, and authorization.

CWS4DB Technical Objectives

- Tech-X proposes to develop a customizable Web service for efficient access to distributed NP databases. The proposed system will consist of:
  - a generic Web service for accessing arbitrary distributed relational databases,
  - a reference client implemented at the Relativistic Heavy Ion Collider (RHIC) at Brookhaven National Laboratory (BNL), for the Solenoidal Tracker at the at the RHIC (STAR) experiment, and
  - a tool for creation of the high-level and domain-specific clients required by particular applications.
- The Phase II objectives include:
  - Take into account what was learned from the research in Phase I and extend the CWS4DB prototype into a production-quality, load-balanced, auto-caching, grid-enabled, fault-tolerant, and on-demand system.
  - Use a flexible work plan involving a separate piece of technical functionality that can be implemented in a way that can be exercised in the STAR computing environment, yet developed in a general way for application’s from other NP projects.
  - ***The ultimate goal is to produce a set of software tools and services that can be easily adapted by the NP application developer.***

DataBase API Class Interactions



★ Branch StDbManager --> StDbServer can be many branches (Servers), ending in different DBs. StDbConfigNode & StDbTable keep info to distinguish Server branch.

TECH-X CORPORATION

CWS4DB Tasks

- Task 1: Determine CWS4DB System and Load Balancing Additional Requirements and Properties (Tech-X & BNL)
  - Extend the Phase I developed requirements and properties and continue prototype work with our partners.
- Task 2: Design and Implement Tiered Deployment Capabilities (Tech-X)
  - Develop a tiered deployment based protocol for the CWS4DB system.
- Task 3: Design and Implement Auto-Caching Infrastructure (Tech-X & BNL)
  - Provide a sophisticated auto-caching mechanism in order to increase the effective system performance based on work with our partners.
- Task 4: Enable Multi-Virtual Organization Role-Based Capabilities (Tech-X)
  - Develop the CWS4DB infrastructure required for user-friendly management and caching capabilities.
- Task 5: Develop Dynamic On-Demand Data Resource Access (Tech-X)
  - This on-demand service provides a STAR MySQL database instance using the Amazon EC2 deployments.

CWS4DB Tasks Continued

- Task 6: Develop Fault Resilient Data Resource Pathways (Tech-X)
  - Eliminated a single point of failure for the STAR C++ API bound codes database query requests.
- Task 7: Develop a Prototype On-Demand Data Resource Node (Tech-X & BNL)
  - Prototyped the deployment of a on-demand data resource node to meet the dynamic data demands of the STAR collaboration.
- Task 8: Prototype Pre-Cache Capabilities for Production Job Workflow (Tech-X & BNL)
  - Pathway for an authenticated and authorized user upon configuration of the CWS4DB system to execute the customizable site specific test suite for pre-caching production job queries is complete.
- Task 9: Develop a Customizable Site Specific Test Suite (Tech-X)
  - In order to deliver a high quality of service infrastructure a customizable and site specific test suite is required to validate and verify the performance and data delivery capabilities of the CWS4DB system.

CWS4DB Tasks

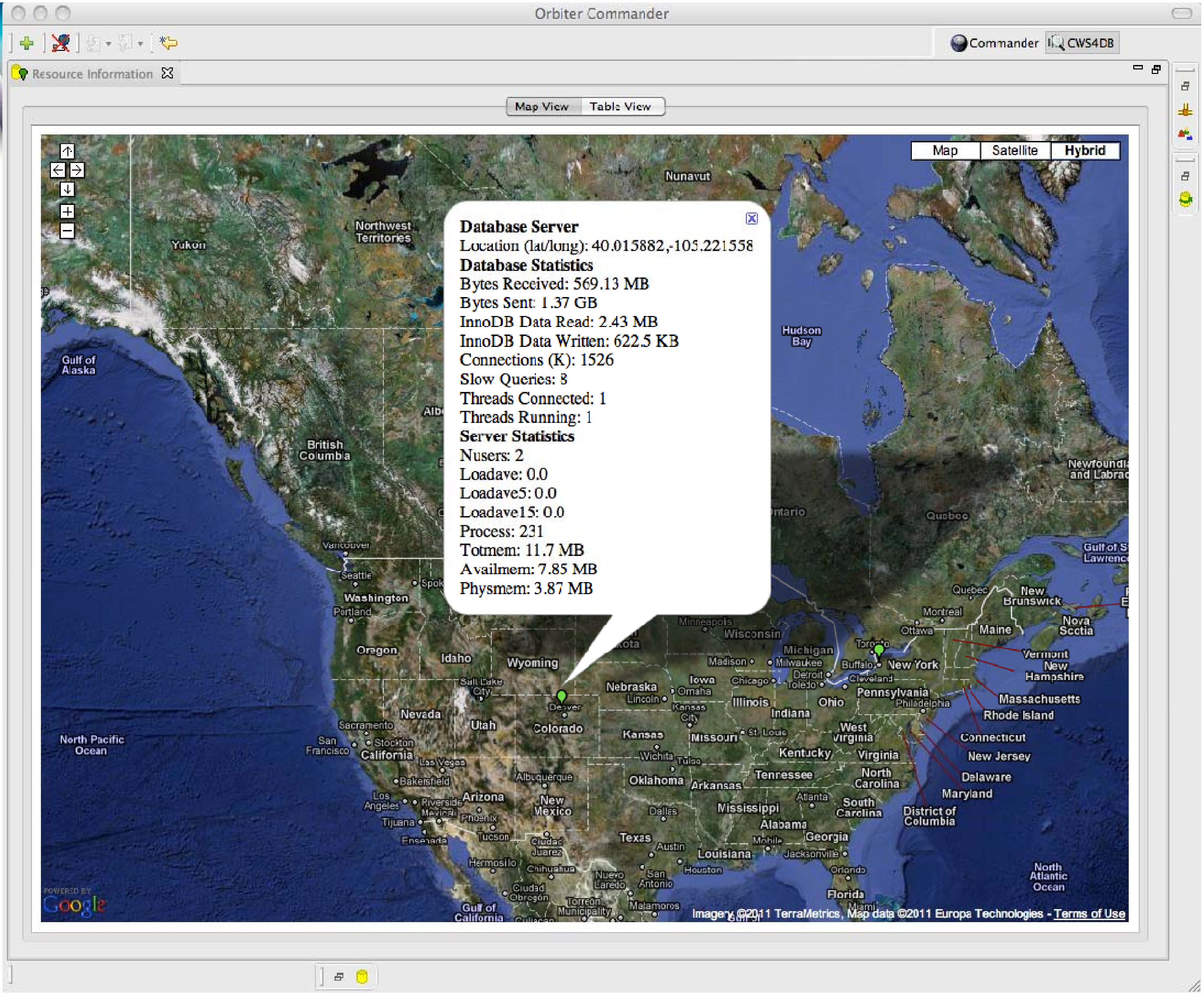
- Task 1: Determine CWS4DB System and Load Balancing Additional Requirements and Properties (Tech-X & BNL)
  - Extend the Phase I developed requirements and properties and continue prototype work with our partners.
- Task 2: Design and Implement Tiered Deployment Capabilities (Tech-X)
  - Develop a tiered deployment based protocol for the CWS4DB system.
- Task 3: Design and Implement Auto-Caching Infrastructure (Tech-X & BNL)
  - Provide a sophisticated auto-caching mechanism in order to increase the effective system performance based on work with our partners.
- Task 4: Enable Multi-Virtual Organization Role-Based Capabilities (Tech-X)
  - Develop the CWS4DB infrastructure required for user-friendly management and caching capabilities.
- Task 5: Develop Dynamic On-Demand Data Resource Access (Tech-X)
  - This on-demand service provides a STAR MySQL database instance using the Amazon EC2 deployments.

Orbiter Federation – SOA

https://{ServiceProvider}/{ResourceAddress}/{Attributes}/{ID}/{ExpirationTime}/{Signature}

- Orbiter Access Key *{ID}* declares user identity
- *{Expiration Time}* ensures request lifetime/validity
- RSA Private Key *{Signature}* ensures data integrity

Similar to the Amazon AWS Security Model



Database Server

Location (lat/long): 40.015882,-105.221558

Database Statistics

Bytes Received: 569.13 MB

Bytes Sent: 1.37 GB

InnoDB Data Read: 2.43 MB

InnoDB Data Written: 622.5 KB

Connections (K): 1526

Slow Queries: 8

Threads Connected: 1

Threads Running: 1

Server Statistics

Nusers: 2

Loadave: 0.0

Loadave5: 0.0

Loadave15: 0.0

Process: 231

Totmem: 11.7 MB

Availmem: 7.85 MB

Physmem: 3.87 MB

CWS4DB Tasks Continued

- Task 6: Develop Fault Resilient Data Resource Pathways (Tech-X)
  - Eliminated a single point of failure for the STAR C++ API bound codes database query requests.
- Task 7: Develop a Prototype On-Demand Data Resource Node (Tech-X & BNL)
  - Prototyped the deployment of a on-demand data resource node to meet the dynamic data demands of the STAR collaboration.
- Task 8: Prototype Pre-Cache Capabilities for Production Job Workflow (Tech-X & BNL)
  - Pathway for an authenticated and authorized user upon configuration of the CWS4DB system to execute the customizable site specific test suite for pre-caching production job queries is complete.
- Task 9: Develop a Customizable Site Specific Test Suite (Tech-X)
  - In order to deliver a high quality of service infrastructure a customizable and site specific test suite is required to validate and verify the performance and data delivery capabilities of the CWS4DB system.

CWS4DB Summary

File Name : txc02.ccr.buffalo.edu.config.inc.php

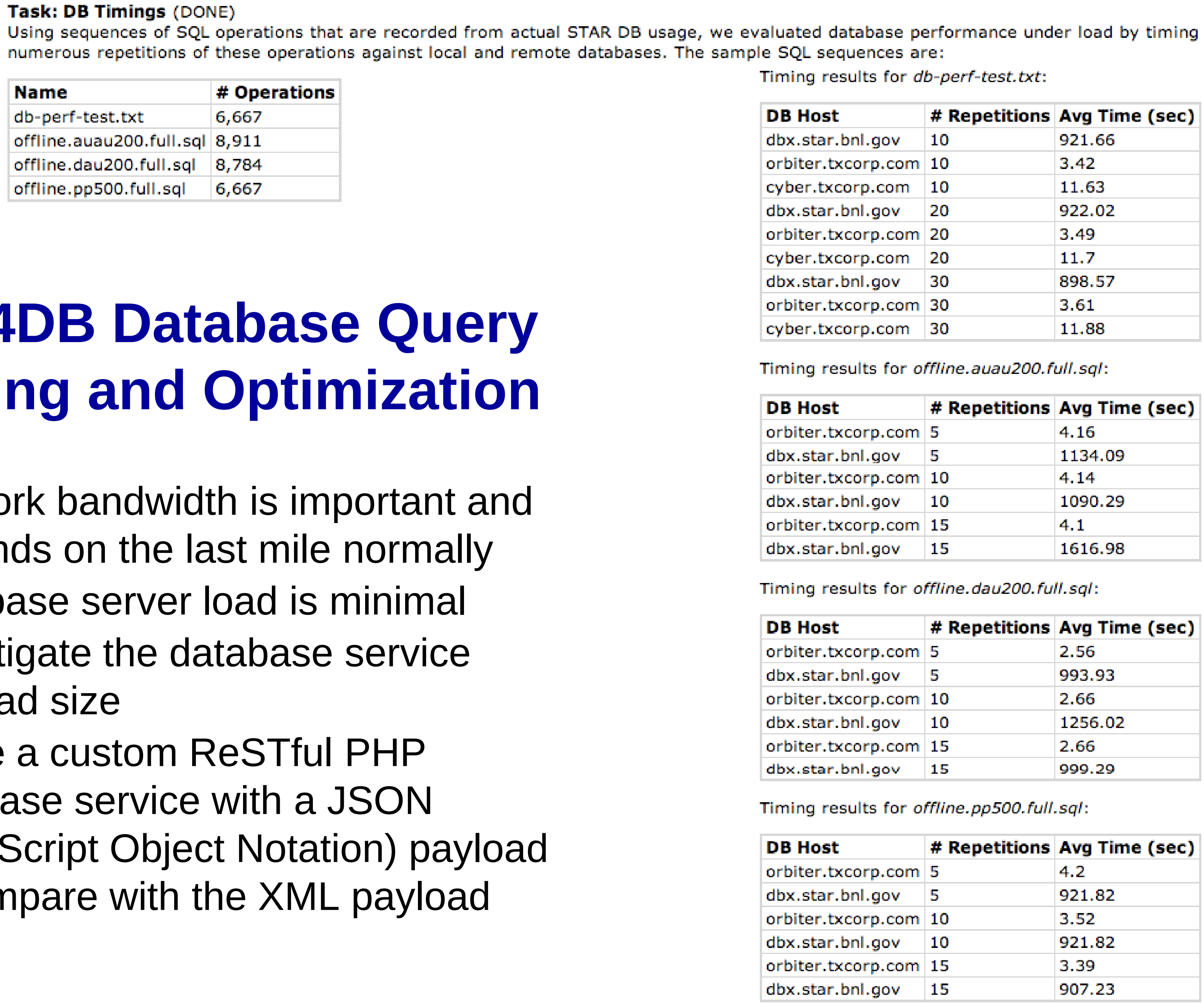
```
// ******************************* STAR specific config ***************************************
/**
 * @var STRING ORBITERCACHEFILELOCATION Cache file location.
 */
define('ORBITERCACHEFILELOCATION', '/tmp/cache');
/**
 * @var STRING ORBITERHASHTYPE Orbiter Hash type
 */
define('ORBITERHASHTYPE', 'sha1');
/**
 * @var integer ORBITERQUERYCONNECTIONSTRINGS Number of Orbiter Query Connection strings.
 */
define('ORBITERQUERYCONNECTIONSTRINGS', 2);
/**
 * @var boolean ORBITERUSEQUERYDB Defines whether to use Orbiter Query DB.
 */
define('ORBITERUSEQUERYDB', true);
/**
 * @var integer ORBITERQUERYDBSERVICEADDRESS Orbiter Query DB Load balancer service address.
 */
define('ORBITERQUERYDBSERVICEADDRESS', 'http://txc02.ccr.buffalo.edu/orbiter/'.ORBITERVERSION.'/service/webservice');
/**
 * @var string ORBITERSQLFILELOCATION Orbiter Sql file location to run the pre-cache for new resource.
 */
define('ORBITERSQLFILELOCATION', '/tmp/sqlfiles/auau200_log.txt');
```

CWS4DB Tasks

- Task 3: Design and Implement Auto-Caching Infrastructure (Tech-X & BNL)
  - Provide a sophisticated auto-caching mechanism in order to increase the effective system performance based on work with our partners.
- Task 4: Enable Multi-Virtual Organization Role-Based Capabilities (Tech-X)
  - Develop the CWS4DB infrastructure required for user-friendly management and caching capabilities.
- Task 7: Develop a Prototype On-Demand Data Resource Node (Tech-X & BNL)
  - Prototyped the deployment of a on-demand data resource node to meet the dynamic data demands of the STAR collaboration.
- Task 8: Prototype Pre-Cache Capabilities for Production Job Workflow (Tech-X & BNL)
  - Pathway for an authenticated and authorized user upon configuration of the CWS4DB system to execute the customizable site specific test suite for pre-caching production job queries is complete.

CWS4DB Database Query Caching and Optimization

- Network bandwidth is important and depends on the last mile normally
- Database server load is minimal
- Investigate the database service payload size
- Wrote a custom ReSTful PHP database service with a JSON (JavaScript Object Notation) payload to compare with the XML payload



Task: DB Timings (DONE)
Using sequences of SQL operations that are recorded from actual STAR DB usage, we evaluated database performance under load by timing numerous repetitions of these operations against local and remote databases. The sample SQL sequences are:

Name	# Operations
db-perf-test.txt	6,667
offline.auau200.full.sql	8,911
offline.dau200.full.sql	8,784
offline.pp500.full.sql	6,667

Timing results for *db-perf-test.txt*:

DB Host	# Repetitions	Avg Time (sec)
dbx.star.bnl.gov	10	921.66
orbiter.txcorp.com	10	3.42
cyber.txcorp.com	10	11.63
dbx.star.bnl.gov	20	922.02
orbiter.txcorp.com	20	3.49
cyber.txcorp.com	20	11.7
dbx.star.bnl.gov	30	898.57
orbiter.txcorp.com	30	3.61
cyber.txcorp.com	30	11.88

Timing results for *offline.auau200.full.sql*:

DB Host	# Repetitions	Avg Time (sec)
orbiter.txcorp.com	5	4.16
dbx.star.bnl.gov	5	1134.09
orbiter.txcorp.com	10	4.14
dbx.star.bnl.gov	10	1090.29
orbiter.txcorp.com	15	4.1
dbx.star.bnl.gov	15	1616.98

Timing results for *offline.dau200.full.sql*:

DB Host	# Repetitions	Avg Time (sec)
orbiter.txcorp.com	5	2.56
dbx.star.bnl.gov	5	993.93
orbiter.txcorp.com	10	2.66
dbx.star.bnl.gov	10	1256.02
orbiter.txcorp.com	15	2.66
dbx.star.bnl.gov	15	999.29

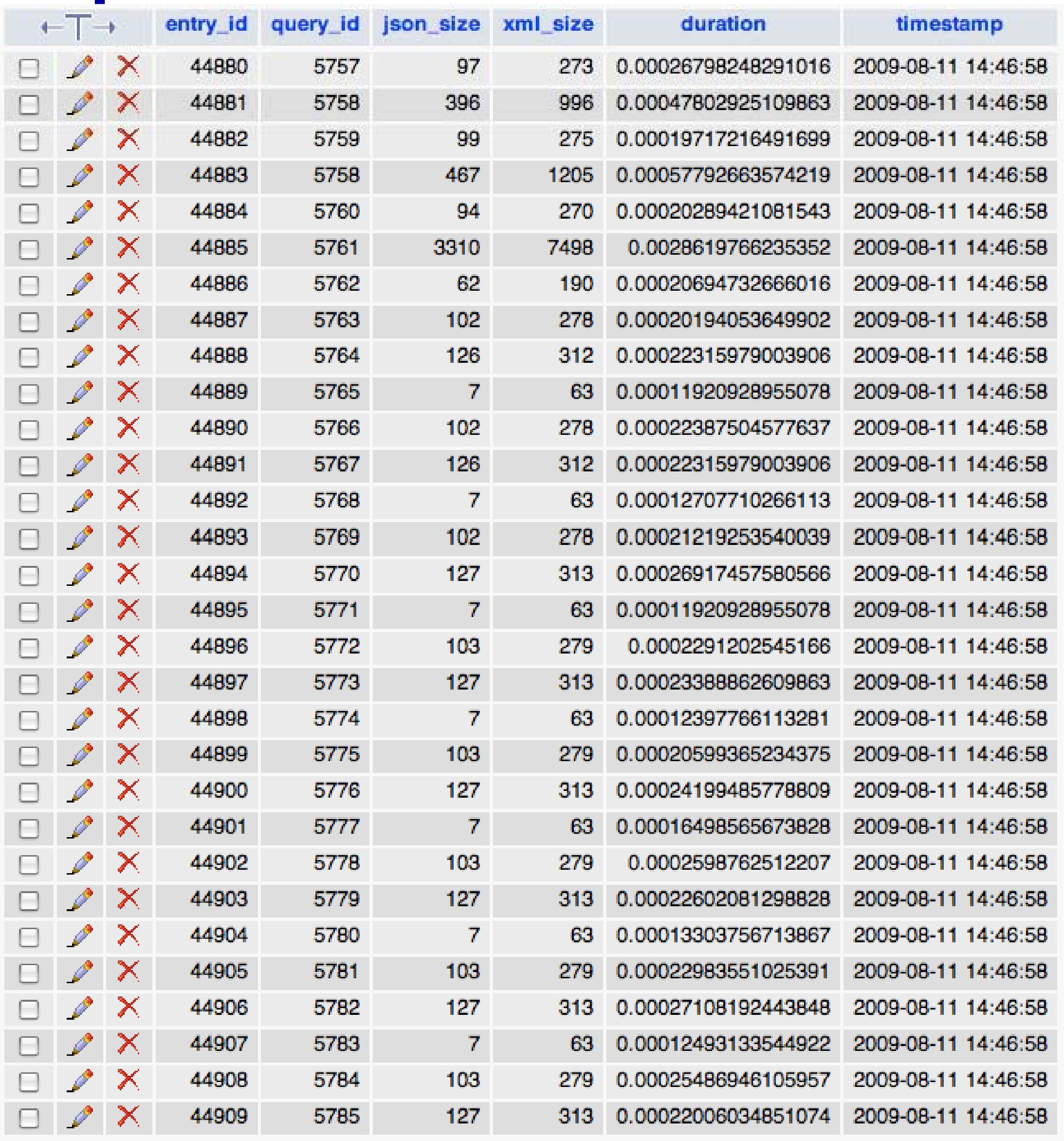
Timing results for *offline.pp500.full.sql*:

DB Host	# Repetitions	Avg Time (sec)
orbiter.txcorp.com	5	4.2
dbx.star.bnl.gov	5	921.82
orbiter.txcorp.com	10	3.52
dbx.star.bnl.gov	10	921.82
orbiter.txcorp.com	15	3.39
dbx.star.bnl.gov	15	907.23

CWS4DB Database Query Caching and Optimization

- Log performance data for each SQL operation
- Calculate and log JSON and XML payload size

- On average over a dataset the equivalent JSON payload is 8.8 – 10.1 times smaller
- In general an order of magnitude lower bandwidth loading is required with the JSON PHP service



←T→	entry_id	query_id	json_size	xml_size	duration	timestamp
□ ✎ ✕	44880	5757	97	273	0.00026798248291016	2009-08-11 14:46:58
□ ✎ ✕	44881	5758	396	996	0.00047802925109863	2009-08-11 14:46:58
□ ✎ ✕	44882	5759	99	275	0.00019717216491699	2009-08-11 14:46:58
□ ✎ ✕	44883	5758	467	1205	0.00057792663574219	2009-08-11 14:46:58
□ ✎ ✕	44884	5760	94	270	0.00020289421081543	2009-08-11 14:46:58
□ ✎ ✕	44885	5761	3310	7498	0.0028619766235352	2009-08-11 14:46:58
□ ✎ ✕	44886	5762	62	190	0.00020694732666016	2009-08-11 14:46:58
□ ✎ ✕	44887	5763	102	278	0.00020194053649902	2009-08-11 14:46:58
□ ✎ ✕	44888	5764	126	312	0.00022315979003906	2009-08-11 14:46:58
□ ✎ ✕	44889	5765	7	63	0.00011920928955078	2009-08-11 14:46:58
□ ✎ ✕	44890	5766	102	278	0.00022387504577637	2009-08-11 14:46:58
□ ✎ ✕	44891	5767	126	312	0.00022315979003906	2009-08-11 14:46:58
□ ✎ ✕	44892	5768	7	63	0.00012707710266113	2009-08-11 14:46:58
□ ✎ ✕	44893	5769	102	278	0.00021219253540039	2009-08-11 14:46:58
□ ✎ ✕	44894	5770	127	313	0.00026917457580566	2009-08-11 14:46:58
□ ✎ ✕	44895	5771	7	63	0.00011920928955078	2009-08-11 14:46:58
□ ✎ ✕	44896	5772	103	279	0.0002291202545166	2009-08-11 14:46:58
□ ✎ ✕	44897	5773	127	313	0.00023388862609863	2009-08-11 14:46:58
□ ✎ ✕	44898	5774	7	63	0.00012397766113281	2009-08-11 14:46:58
□ ✎ ✕	44899	5775	103	279	0.00020599365234375	2009-08-11 14:46:58
□ ✎ ✕	44900	5776	127	313	0.00024199485778809	2009-08-11 14:46:58
□ ✎ ✕	44901	5777	7	63	0.00016498565673828	2009-08-11 14:46:58
□ ✎ ✕	44902	5778	103	279	0.0002598762512207	2009-08-11 14:46:58
□ ✎ ✕	44903	5779	127	313	0.00022602081298828	2009-08-11 14:46:58
□ ✎ ✕	44904	5780	7	63	0.00013303756713867	2009-08-11 14:46:58
□ ✎ ✕	44905	5781	103	279	0.00022983551025391	2009-08-11 14:46:58
□ ✎ ✕	44906	5782	127	313	0.00027108192443848	2009-08-11 14:46:58
□ ✎ ✕	44907	5783	7	63	0.00012493133544922	2009-08-11 14:46:58
□ ✎ ✕	44908	5784	103	279	0.00025486946105957	2009-08-11 14:46:58
□ ✎ ✕	44909	5785	127	313	0.00022006034851074	2009-08-11 14:46:58

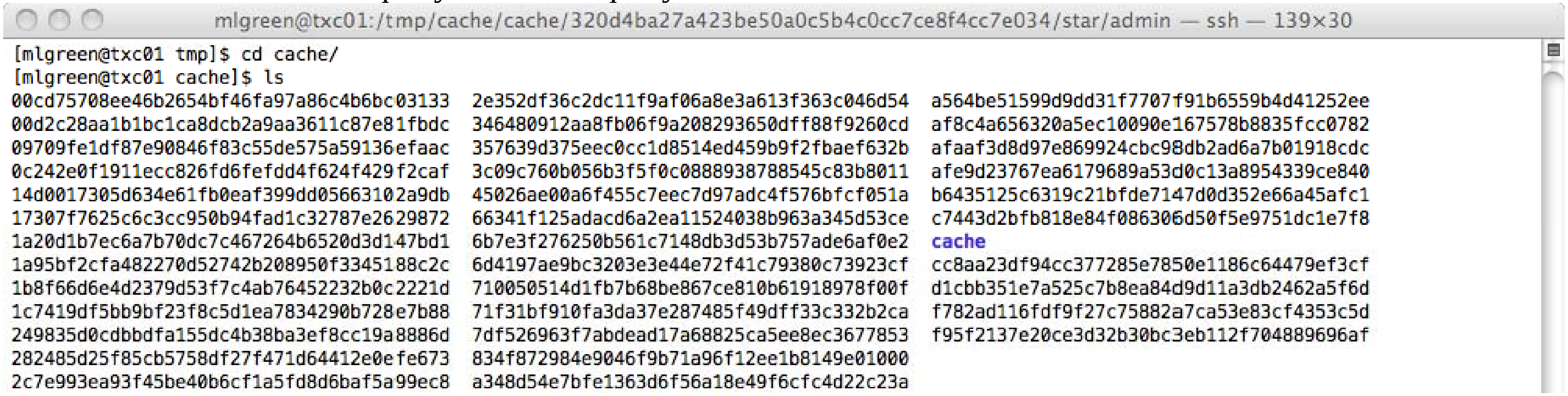
CWS4DB Summary

File Name : star.pp500.full.sql

https://cyber.txcorp.com/orbiter/service/star/OrbiterStarSimulatorService.php
/cache/off/
/format/XML/
/host/local/
/file//tmp/testfiles/star.pp500.full.sql/
/address/http://64.240.154.24/orbiter/service/star/

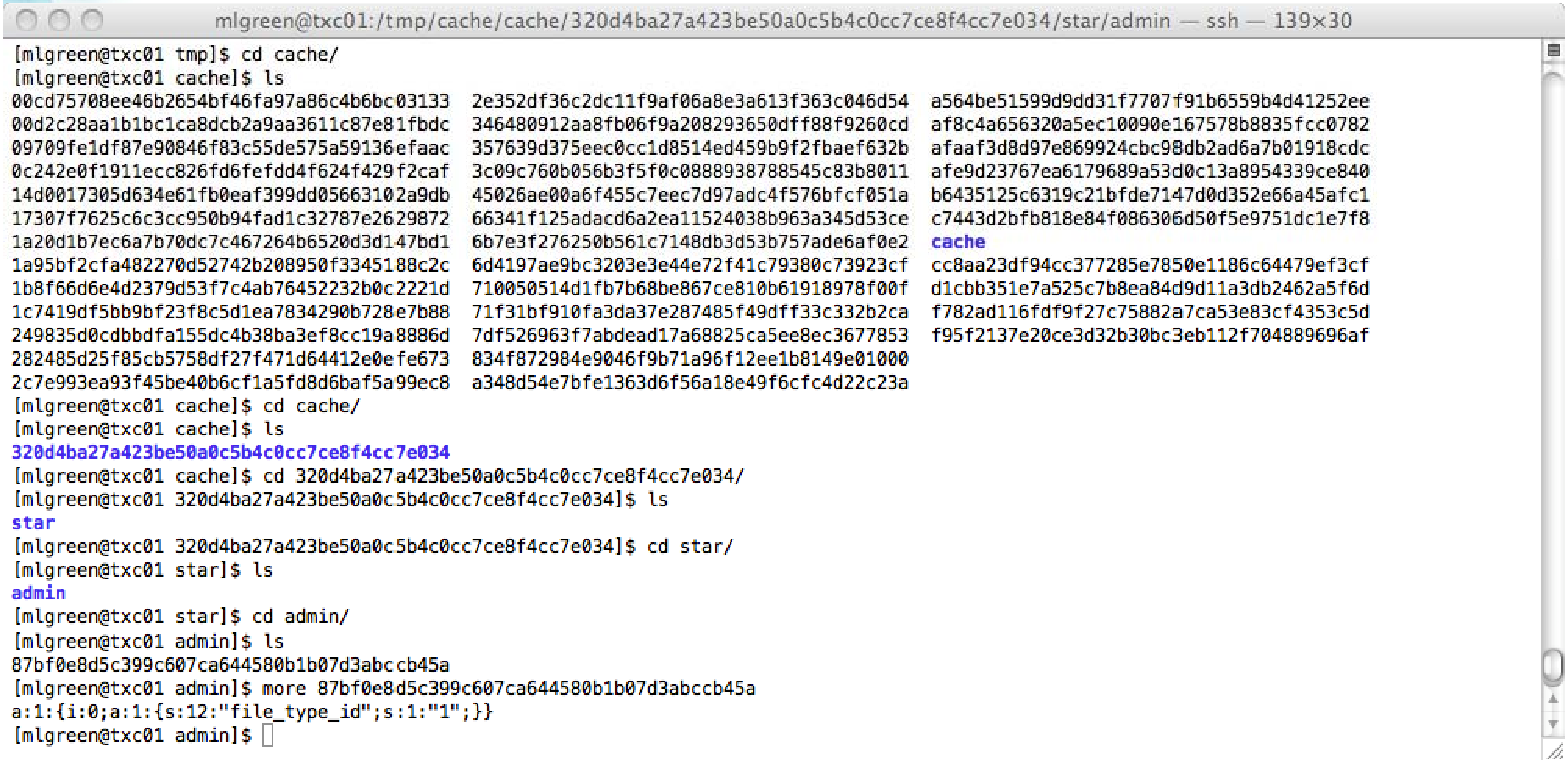
Result:

Number of trials averaged: 1
Total number of queries: 6549
Total size of queries: 38,926,201 bytes
Total query time: 76.9 seconds
Total query rate: 85.1 query/second.


mlgreen@txc01:/tmp/cache/cache/320d4ba27a423be50a0c5b4c0cc7ce8f4cc7e034/star/admin — ssh — 139×30

```
[mlgreen@txc01 tmp]$ cd cache/
[mlgreen@txc01 cache]$ ls
00cd75708ee46b2654bf46fa97a86c4b6bc03133  2e352df36c2dc11f9af06a8e3a613f363c046d54  a564be51599d9dd31f7707f91b6559b4d41252ee
00d2c28aa1b1bc1ca8dcb2a9aa3611c87e81fbdc  346480912aa8fb06f9a208293650dff88f9260cd  af8c4a656320a5ec10090e167578b8835fcc0782
09709fe1df87e90846f83c55de575a59136efaac  357639d375eec0cc1d8514ed459b9f2fbaef632b  afaaf3d8d97e869924cbc98db2ad6a7b01918cdc
0c242e0f1911ecc826fd6fefdd4f624f429f2caf  3c09c760b056b3f5f0c0888938788545c83b8011  afe9d23767ea6179689a53d0c13a8954339ce840
14d0017305d634e61fb0eaf399dd05663102a9db  45026ae00a6f455c7eec7d97adc4f576bfcf051a  b6435125c6319c21bfde7147d0d352e66a45afc1
17307f7625c6c3cc950b94fad1c32787e2629872  66341f125adacd6a2ea11524038b963a345d53ce  c7443d2bfb818e84f086306d50f5e9751dc1e7f8
1a20d1b7ec6a7b70dc7c467264b6520d3d147bd1  6b7e3f276250b561c7148db3d53b757ade6af0e2  cache
1a95bf2cfa482270d52742b208950f3345188c2c  6d4197ae9bc3203e3e44e72f41c79380c73923cf  cc8aa23df94cc377285e7850e1186c64479ef3cf
1b8f66d6e4d2379d53f7c4ab76452232b0c2221d  710050514d1fb7b68be867ce810b61918978f00f  d1cbb351e7a525c7b8ea84d9d11a3db2462a5f6d
1c7419df5bb9bf23f8c5d1ea7834290b728e7b88  71f31bf910fa3da37e287485f49dff33c332b2ca  f782ad116fdf9f27c75882a7ca53e83cf4353c5d
249835d0cdbbdfa155dc4b38ba3ef8cc19a8886d  7df526963f7abdead17a68825ca5ee8ec3677853  f95f2137e20ce3d32b30bc3eb112f704889696af
282485d25f85cb5758df27f471d64412e0efe673  834f872984e9046f9b71a96f12ee1b8149e01000
2c7e993ea93f45be40b6cf1a5fd8d6baf5a99ec8  a348d54e7bfe1363d6f56a18e49f6cfc4d22c23a
```

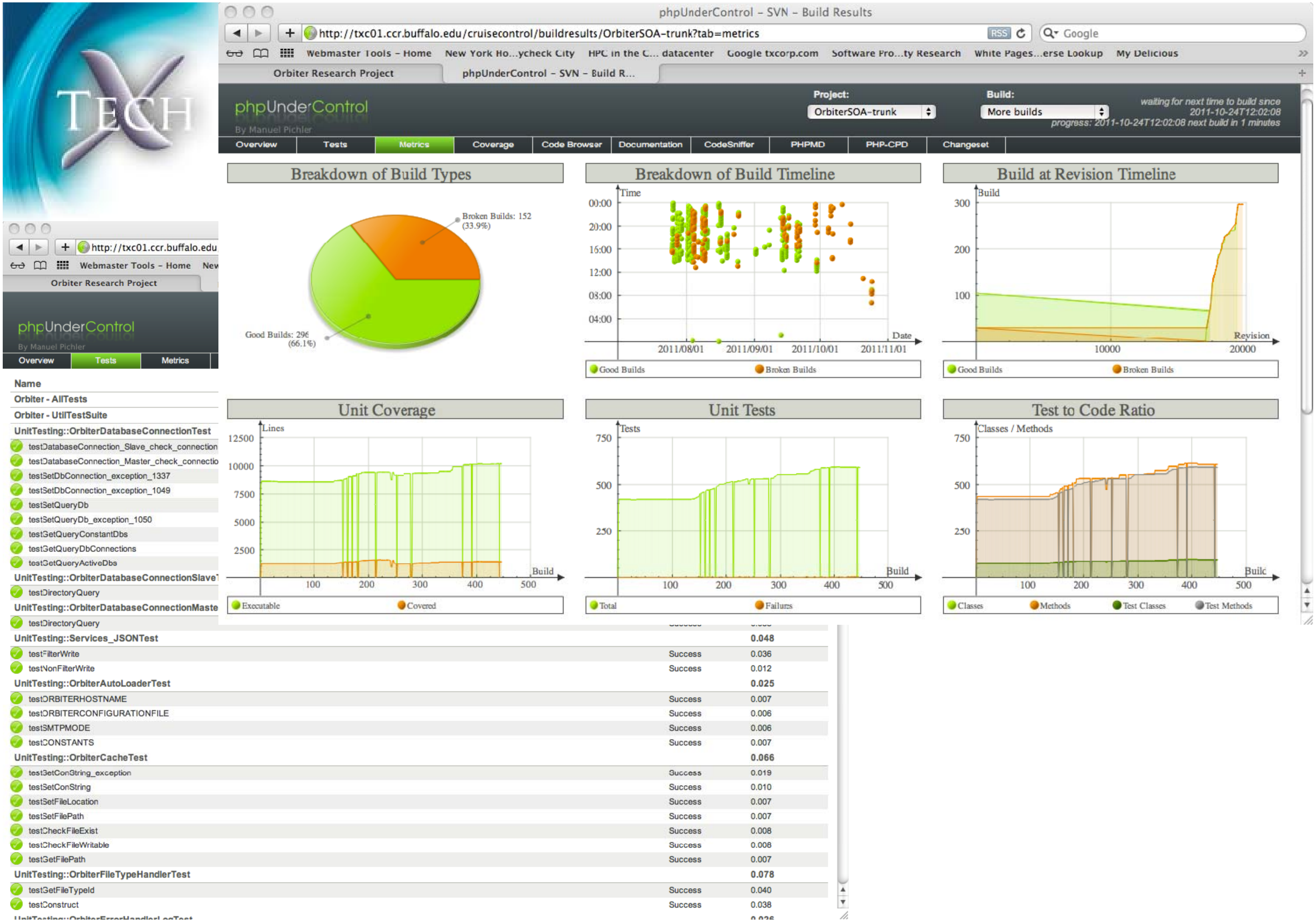
CWS4DB Summary


mlgreen@txc01:/tmp/cache/cache/320d4ba27a423be50a0c5b4c0cc7ce8f4cc7e034/star/admin — ssh — 139×30

```
[mlgreen@txc01 tmp]$ cd cache/
[mlgreen@txc01 cache]$ ls
00cd75708ee46b2654bf46fa97a86c4b6bc03133  2e352df36c2dc11f9af06a8e3a613f363c046d54  a564be51599d9dd31f7707f91b6559b4d41252ee
00d2c28aa1b1bc1ca8dcb2a9aa3611c87e81fbdc  346480912aa8fb06f9a208293650dff88f9260cd  af8c4a656320a5ec10090e167578b8835fcc0782
09709fe1df87e90846f83c55de575a59136efaac  357639d375eec0cc1d8514ed459b9f2fbaef632b  afaaf3d8d97e869924cbc98db2ad6a7b01918cdc
0c242e0f1911ecc826fd6fefdd4f624f429f2caf  3c09c760b056b3f5f0c0888938788545c83b8011  afe9d23767ea6179689a53d0c13a8954339ce840
14d0017305d634e61fb0eaf399dd05663102a9db  45026ae00a6f455c7eec7d97adc4f576bfcf051a  b6435125c6319c21bfde7147d0d352e66a45afc1
17307f7625c6c3cc950b94fad1c32787e2629872  66341f125adacd6a2ea11524038b963a345d53ce  c7443d2bfb818e84f086306d50f5e9751dc1e7f8
1a20d1b7ec6a7b70dc7c467264b6520d3d147bd1  6b7e3f276250b561c7148db3d53b757ade6af0e2  cache
1a95bf2cfa482270d52742b208950f3345188c2c  6d4197ae9bc3203e3e44e72f41c79380c73923cf  cc8aa23df94cc377285e7850e1186c64479ef3cf
1b8f66d6e4d2379d53f7c4ab76452232b0c2221d  710050514d1fb7b68be867ce810b61918978f00f  d1cbb351e7a525c7b8ea84d9d11a3db2462a5f6d
1c7419df5bb9bf23f8c5d1ea7834290b728e7b88  71f31bf910fa3da37e287485f49dff33c332b2ca  f782ad116fdf9f27c75882a7ca53e83cf4353c5d
249835d0cdbbdfa155dc4b38ba3ef8cc19a8886d  7df526963f7abdead17a68825ca5ee8ec3677853  f95f2137e20ce3d32b30bc3eb112f704889696af
282485d25f85cb5758df27f471d64412e0efe673  834f872984e9046f9b71a96f12ee1b8149e01000
2c7e993ea93f45be40b6cf1a5fd8d6baf5a99ec8  a348d54e7bfe1363d6f56a18e49f6cfc4d22c23a
[mlgreen@txc01 cache]$ cd cache/
[mlgreen@txc01 cache]$ ls
320d4ba27a423be50a0c5b4c0cc7ce8f4cc7e034
[mlgreen@txc01 cache]$ cd 320d4ba27a423be50a0c5b4c0cc7ce8f4cc7e034/
[mlgreen@txc01 320d4ba27a423be50a0c5b4c0cc7ce8f4cc7e034]$ ls
star
[mlgreen@txc01 320d4ba27a423be50a0c5b4c0cc7ce8f4cc7e034]$ cd star/
[mlgreen@txc01 star]$ ls
admin
[mlgreen@txc01 star]$ cd admin/
[mlgreen@txc01 admin]$ ls
87bf0e8d5c399c607ca644580b1b07d3abccb45a
[mlgreen@txc01 admin]$ more 87bf0e8d5c399c607ca644580b1b07d3abccb45a
a:1:{i:0;a:1:{s:12:"file_type_id";s:1:"1";}}
[mlgreen@txc01 admin]$ 
```

CWS4DB Tasks Continued

- Task 6: Develop Fault Resilient Data Resource Pathways (Tech-X)
  - Eliminated a single point of failure for the STAR C++ API bound codes database query requests.
- Task 7: Develop a Prototype On-Demand Data Resource Node (Tech-X & BNL)
  - Prototyped the deployment of a on-demand data resource node to meet the dynamic data demands of the STAR collaboration.
- Task 8: Prototype Pre-Cache Capabilities for Production Job Workflow (Tech-X & BNL)
  - Pathway for an authenticated and authorized user upon configuration of the CWS4DB system to execute the customizable site specific test suite for pre-caching production job queries is complete.
- Task 9: Develop a Customizable Site Specific Test Suite (Tech-X)
  - In order to deliver a high quality of service infrastructure a customizable and site specific test suite is required to validate and verify the performance and data delivery capabilities of the CWS4DB system.



phpUnderControl – SVN – Build Results

http://txc01.ccr.buffalo.edu/cruisecontrol/buildresults/OrbiterSOA-trunk?tab=metrics

Project: OrbiterSOA-trunk

Build: More builds

waiting for next time to build since 2011-10-24T12:02:08
progress: 2011-10-24T12:02:08 next build in 1 minutes

Name		
Orbiter - AllTests		
Orbiter - UtilTestSuite		
UnitTesting::OrbiterDatabaseConnectionTest		
testDatabaseConnection_Slave_check_connection		
testDatabaseConnection_Master_check_connectio		
testSetDbConnection_exception_1337		
testSetDbConnection_exception_1049		
testSetQueryDb		
testSetQueryDb_exception_1050		
testGetQueryConstantDbs		
testGetQueryDbConnections		
testGetQueryActiveDbs		
UnitTesting::OrbiterDatabaseConnectionSlave		
testDirectoryQuery		
UnitTesting::OrbiterDatabaseConnectionMaste		
testDirectoryQuery		
UnitTesting::Services_JSONTest		0.048
testFilterWrite	Success	0.036
testNonFilterWrite	Success	0.012
UnitTesting::OrbiterAutoLoaderTest		0.025
testORBITERHOSTNAME	Success	0.007
testORBITERCONFIGURATIONFILE	Success	0.006
testSMTPMODE	Success	0.006
testCONSTANTS	Success	0.007
UnitTesting::OrbiterCacheTest		0.066
testGetConString_exception	Success	0.019
testSetConString	Success	0.010
testSetFileLocation	Success	0.007
testSetFilePath	Success	0.007
testCheckFileExist	Success	0.008
testCheckFileWritable	Success	0.008
testGetFilePath	Success	0.007
UnitTesting::OrbiterFileTypeHandlerTest		0.078
testGetFileTypeId	Success	0.040
testConstruct	Success	0.038

TECH-X CORPORATION

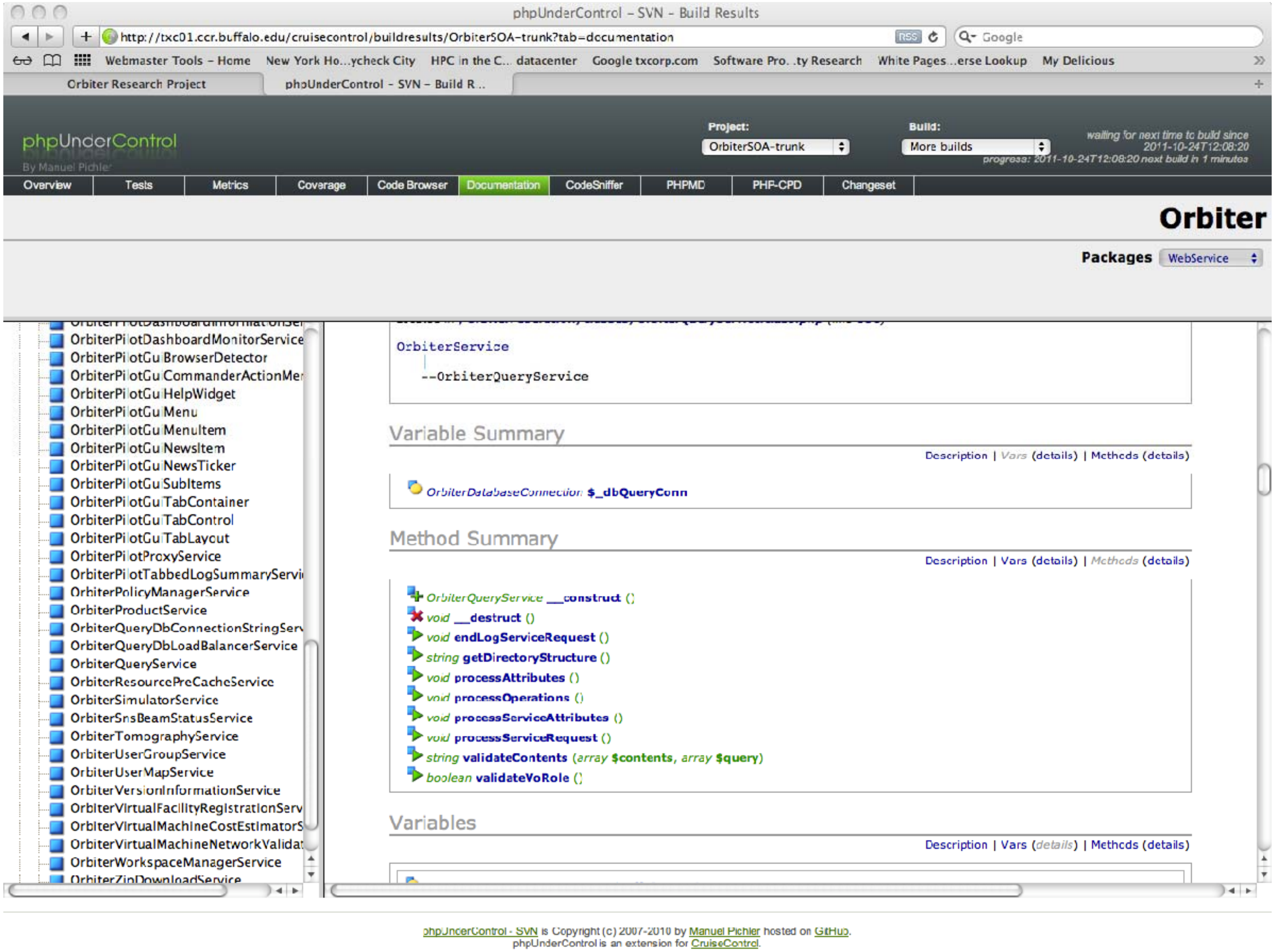
Project Summary

PHP Code:

- 92 classes
- 642 functions/methods
- 10200 lines

API:

- Includes source code links
- Usage
- Dynamically updated



waiting for next time to build since 2011-10-24T12:08:20
progress: 2011-10-24T12:08:20 next build in 1 minutes

Orbiter

- OrbiterPilotDashboardMonitorService
- OrbiterPilotGuiBrowserDetector
- OrbiterPilotGuiCommandActionMe
- OrbiterPilotGuiHelpWidget
- OrbiterPilotGuiMenu
- OrbiterPilotGuiMenuItem
- OrbiterPilotGuiNewsItem
- OrbiterPilotGuiNewsTicker
- OrbiterPilotGuiSubItems
- OrbiterPilotGuiTabContainer
- OrbiterPilotGuiTabControl
- OrbiterPilotGuiTabLayout
- OrbiterPilotProxyService
- OrbiterPilotTabbedLogSummaryServi
- OrbiterPolicyManagerService
- OrbiterProductService
- OrbiterQueryDbConnectionStringSer
- OrbiterQueryDbLoadBalancerService
- OrbiterQueryService
- OrbiterResourcePreCacheService
- OrbiterSimulatorService
- OrbiterSnsBeamStatusService
- OrbiterTomographyService
- OrbiterUserGroupService
- OrbiterUserMapService
- OrbiterVersionInformationService
- OrbiterVirtualFacilityRegistrationServ
- OrbiterVirtualMachineCostEstimatorS
- OrbiterVirtualMachineNetworkValidat
- OrbiterWorkspaceManagerService
- OrbiterZipDownloadService

```
OrbiterService
   |
   --OrbiterQueryService
```

Variable Summary

Description | Vars (details) | Methods (details)

OrbiterDatabaseConnection **$_dbQueryConn**

Method Summary

Description | Vars (details) | Methods (details)

- *OrbiterQueryService* **__construct** ()
- *void* **__destruct** ()
- *void* **endLogServiceRequest** ()
- *string* **getDirectoryStructure** ()
- *void* **processAttributes** ()
- *void* **processOperations** ()
- *void* **processServiceAttributes** ()
- *void* **processServiceRequest** ()
- *string* **validateContents** (*array* **$contents**, *array* **$query**)
- *boolean* **validateVoRole** ()

Variables

Description | Vars (details) | Methods (details)

phpUnderControl - SVN is Copyright (c) 2007-2010 by Manuel Pichler hosted on GitHub.
phpUnderControl is an extension for CruiseControl.

Orbiter Federation SOA:
Python Client Service Access Example

```
#!/usr/bin/python
import os, sys, base64, hmac, commands, time
from hashlib import sha1 as sha
from urllib import urlencode
from urllib import urlopen
from urllib import quote_plus

myhome = os.environ.get('HOME')
os.environ['TZ']='GMT'
time.tzset()

idfile = open(myhome + "/.orbiter/my.id")
ACCESS_KEY = idfile.read().strip()
idfile.close()
keyfile = open(myhome + "/.orbiter/user.key")
PRIVATE_KEY = keyfile.read()
keyfile.close()

URI = sys.argv[1]
EXPIRES = str(int(time.mktime(time.localtime(time.time()+60))))
str = URI + '/OrbiterAccessKeyId/' + ACCESS_KEY + '/Expires/' + EXPIRES
SIGNATURE = base64.b64encode(hmac.new(PRIVATE_KEY, str, sha).digest()).strip()
print urlopen(str + '/Signature/' + SIGNATURE, params).read()
```

Scripts and libraries
are also available for:
C/C++, CURL, Java,
Python, PHP that can
access the Orbiter Federation.

Orbiter Federation – SOA

- Orbiter Federation addresses security threats facing web services as identified by Web Services Interoperability Organization which include:
  - **Message alteration:** Attackers cannot alter an Orbiter request without breaking the RSA SHA hash signature. Orbiter will reject a request that does not match canonical string signed resource identifier for the specified Orbiter access key ID.
  - **Loss of confidentiality:** The SSL protocol ensures that Orbiter service transactions are handled privately and provides transport-level encryption.
  - **Falsified messages:** Secure Orbiter services cannot be reached without a signed canonical string resource identifier that matches the signature for the specified Orbiter Federation SOA resource address.
  - **Man in the middle:** The SSL protocol prevents an attacker from reviewing requests and responses send securely between the Orbiter Federation SOA web services and their clients.
  - **Principal spoofing:** The Orbiter infrastructure is the only provider of valid Orbiter access key identifiers and RSA private keys that are authorized to use Orbiter Federation SOA secure web services.

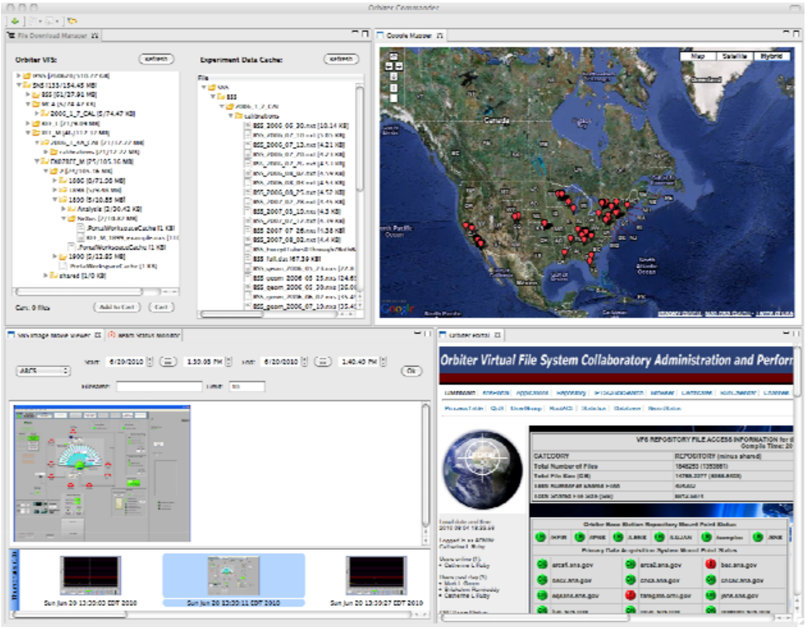
Orbiter Federation – SOA

- Orbiter Federation addresses security threats continued:

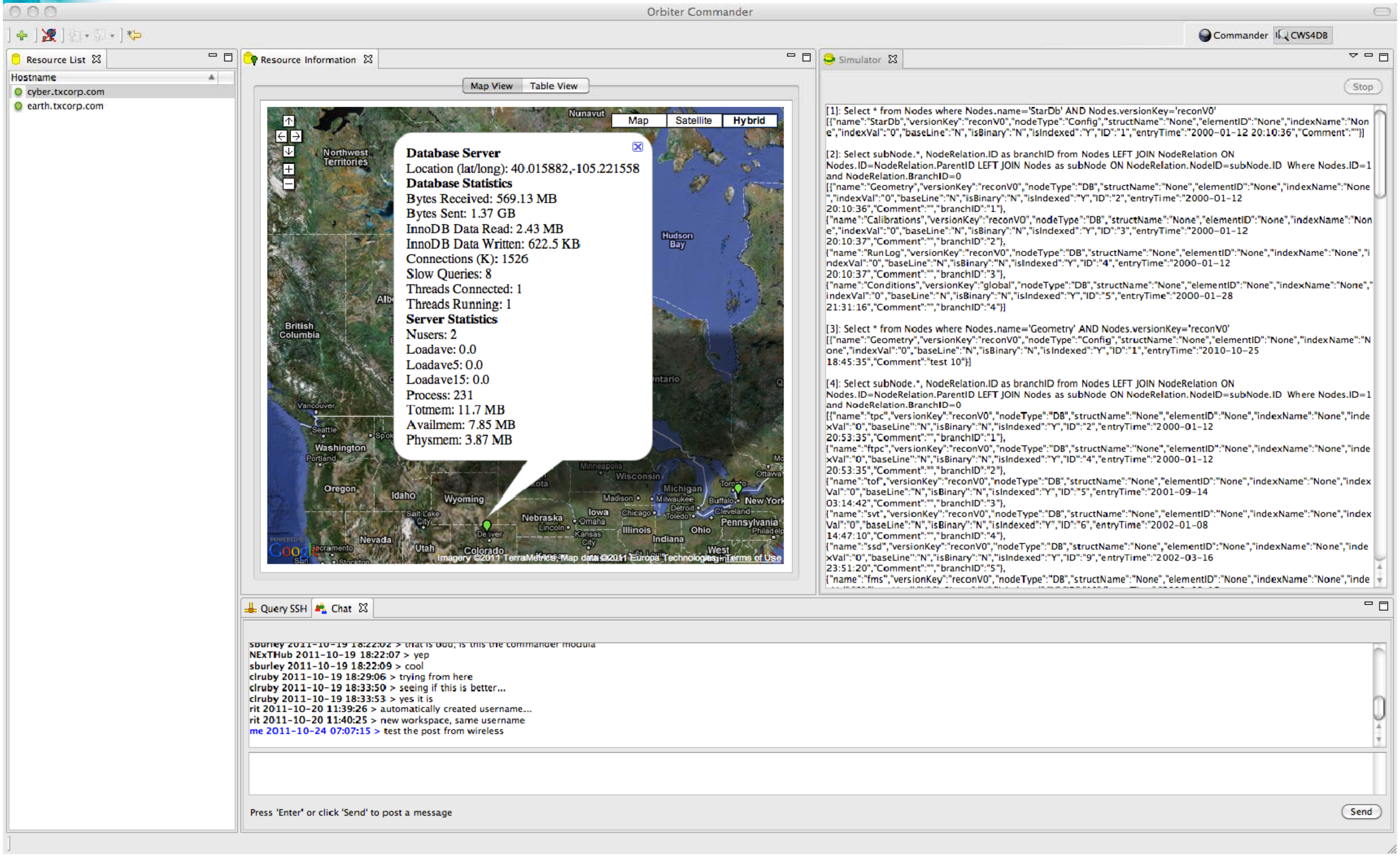
  - **Forging claims:** Attackers cannot create valid Orbiter Federation SOA service requests without obtaining an Orbiter access key identifier and valid RSA private key from the Orbiter Federation SOA authentication/authorization infrastructure.
  - **Replay of message:** Attackers cannot repeat a RESTful request to secure Orbiter Federation services, as subsequent identical requests will be rejected. Attackers cannot alter the user-provided expiration time without breaking the RSA signature.
  - **Replay of message parts:** An Orbiter RESTful service request is not complete without a valid signature that is applied to all other message parts. Attackers cannot construct a new request from any part of a previous request without altering the service request canonical string resource identifier and generating a valid signature.
  - **Denial of service*:** The denial of service propensity is greatly reduced by the listed security measures in place at the current time within the Orbiter Federation SOA. Furthermore, more specific measures are planned which will ban specific offending IP addresses to further reduce the threat. *Distributed Denial of Service (DDOS) attacks, however, are extremely difficult to defend against utilizing known security measures.

Orbiter Commander – Thick Client

- Tier III of the Orbiter Multitier Portal Architecture
- Orbiter Commander is a Application Framework where a:
  - **Application Framework** defines how to solve common problems, not solutions themselves
    - Using well placed abstraction layers
    - Defining points for extending functionality
- Utilizes Eclipse Rich Client Platform for generating multi-platform applications
  - Mature Integrated Development Environment
  - Simplified Standard Widget Toolkit
  - Automatic handling of core GUI implementation
- Built on Orbiter Federation Services
- Modular design for plug-and-play capabilities
- Well-defined extension points for code-reuse and future development
- Highly customizable interface



STAR Commander Implementation



Database Server
Location (lat/long): 40.015882,-105.221558
Database Statistics
Bytes Received: 569.13 MB
Bytes Sent: 1.37 GB
InnoDB Data Read: 2.43 MB
InnoDB Data Written: 622.5 KB
Connections (K): 1526
Slow Queries: 8
Threads Connected: 1
Threads Running: 1
Server Statistics
Nusers: 2
Loadave: 0.0
Loadave5: 0.0
Loadave15: 0.0
Process: 231
Totmem: 11.7 MB
Availmem: 7.85 MB
Physmem: 3.87 MB

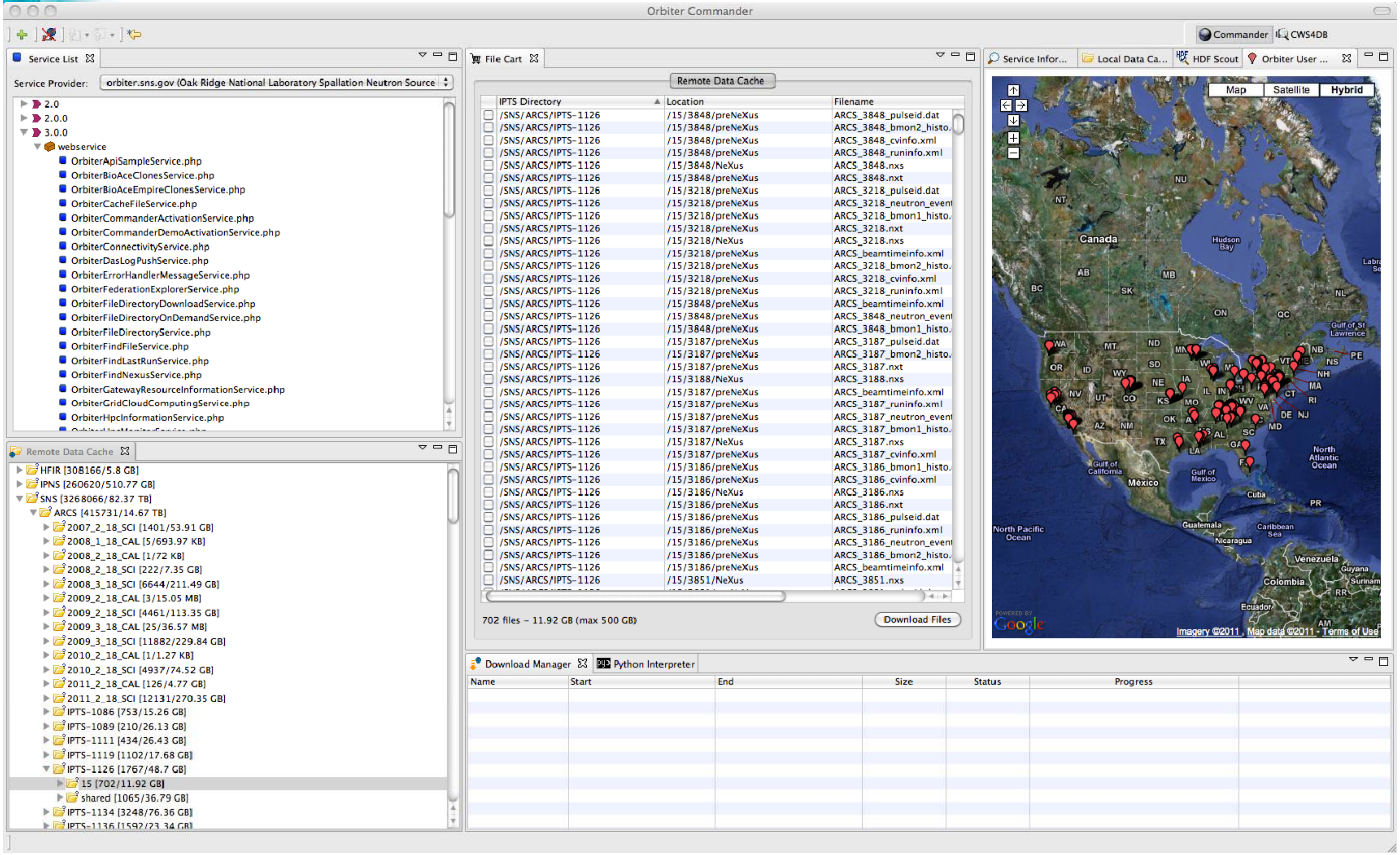
Imagery ©2011 TerraMetrics, Map data ©2011 Europa Technologies - Terms of Use

```
[1]: Select * from Nodes where Nodes.name='StarDb' AND Nodes.versionKey='reconV0'
[{"name":"StarDb","versionKey":"reconV0","nodeType":"Config","structName":"None","elementID":"None","indexName":"None","indexVal":"0","baseLine":"N","isBinary":"N","isIndexed":"Y","ID":"1","entryTime":"2000-01-12 20:10:36","Comment":""}]

[2]: Select subNode.*, NodeRelation.ID as branchID from Nodes LEFT JOIN NodeRelation ON Nodes.ID=NodeRelation.ParentID LEFT JOIN Nodes as subNode ON NodeRelation.NodeID=subNode.ID  Where Nodes.ID=1 and NodeRelation.BranchID=0
[{"name":"Geometry","versionKey":"reconV0","nodeType":"DB","structName":"None","elementID":"None","indexName":"None","indexVal":"0","baseLine":"N","isBinary":"N","isIndexed":"Y","ID":"2","entryTime":"2000-01-12 20:10:36","Comment":"","branchID":"1"},
{"name":"Calibrations","versionKey":"reconV0","nodeType":"DB","structName":"None","elementID":"None","indexName":"None","indexVal":"0","baseLine":"N","isBinary":"N","isIndexed":"Y","ID":"3","entryTime":"2000-01-12 20:10:37","Comment":"","branchID":"2"},
{"name":"RunLog","versionKey":"reconV0","nodeType":"DB","structName":"None","elementID":"None","indexName":"None","indexVal":"0","baseLine":"N","isBinary":"N","isIndexed":"Y","ID":"4","entryTime":"2000-01-12 20:10:37","Comment":"","branchID":"3"},
{"name":"Conditions","versionKey":"global","nodeType":"DB","structName":"None","elementID":"None","indexName":"None","indexVal":"0","baseLine":"N","isBinary":"N","isIndexed":"Y","ID":"5","entryTime":"2000-01-28 21:31:16","Comment":"","branchID":"4"}]

[3]: Select * from Nodes where Nodes.name='Geometry' AND Nodes.versionKey='reconV0'
[{"name":"Geometry","versionKey":"reconV0","nodeType":"Config","structName":"None","elementID":"None","indexName":"None","indexVal":"0","baseLine":"N","isBinary":"N","isIndexed":"Y","ID":"1","entryTime":"2010-10-25 18:45:35","Comment":"test 10"}]

[4]: Select subNode.*, NodeRelation.ID as branchID from Nodes LEFT JOIN NodeRelation ON Nodes.ID=NodeRelation.ParentID LEFT JOIN Nodes as subNode ON NodeRelation.NodeID=subNode.ID  Where Nodes.ID=1 and NodeRelation.BranchID=0
[{"name":"tpc","versionKey":"reconV0","nodeType":"DB","structName":"None","elementID":"None","indexName":"None","indexVal":"0","baseLine":"N","isBinary":"N","isIndexed":"Y","ID":"2","entryTime":"2000-01-12 20:53:35","Comment":"","branchID":"1"},
{"name":"ftpc","versionKey":"reconV0","nodeType":"DB","structName":"None","elementID":"None","indexName":"None","indexVal":"0","baseLine":"N","isBinary":"N","isIndexed":"Y","ID":"4","entryTime":"2000-01-12 20:53:35","Comment":"","branchID":"2"},
{"name":"tof","versionKey":"reconV0","nodeType":"DB","structName":"None","elementID":"None","indexName":"None","indexVal":"0","baseLine":"N","isBinary":"N","isIndexed":"Y","ID":"5","entryTime":"2001-09-14 03:14:42","Comment":"","branchID":"3"},
{"name":"svt","versionKey":"reconV0","nodeType":"DB","structName":"None","elementID":"None","indexName":"None","indexVal":"0","baseLine":"N","isBinary":"N","isIndexed":"Y","ID":"6","entryTime":"2002-01-08 14:47:10","Comment":"","branchID":"4"},
{"name":"ssd","versionKey":"reconV0","nodeType":"DB","structName":"None","elementID":"None","indexName":"None","indexVal":"0","baseLine":"N","isBinary":"N","isIndexed":"Y","ID":"9","entryTime":"2002-03-16 23:51:20","Comment":"","branchID":"5"},
{"name":"fms","versionKey":"reconV0","nodeType":"DB","structName":"None","elementID":"None","indexName":"None","inde
```

```
sburley 2011-10-19 18:22:02 > that is odd; is this the commander modula
NExTHub 2011-10-19 18:22:07 > yep
sburley 2011-10-19 18:22:09 > cool
clruby 2011-10-19 18:29:06 > trying from here
clruby 2011-10-19 18:33:50 > seeing if this is better...
clruby 2011-10-19 18:33:53 > yes it is
rit 2011-10-20 11:39:26 > automatically created username...
rit 2011-10-20 11:40:25 > new workspace, same username
me 2011-10-24 07:07:15 > test the post from wireless
```

Press 'Enter' or click 'Send' to post a message

Commander Explorer Implementation



Service Provider: orbiter.sns.gov (Oak Ridge National Laboratory Spallation Neutron Source

- 2.0
- 2.0.0
- 3.0.0
  - webservice
    - OrbiterApiSampleService.php
    - OrbiterBioAceClonesService.php
    - OrbiterBioAceEmpireClonesService.php
    - OrbiterCacheFileService.php
    - OrbiterCommanderActivationService.php
    - OrbiterCommanderDemoActivationService.php
    - OrbiterConnectivityService.php
    - OrbiterDasLogPushService.php
    - OrbiterErrorHandlerMessageService.php
    - OrbiterFederationExplorerService.php
    - OrbiterFileDirectoryDownloadService.php
    - OrbiterFileDirectoryOnDemandService.php
    - OrbiterFileDirectoryService.php
    - OrbiterFindFileService.php
    - OrbiterFindLastRunService.php
    - OrbiterFindNexusService.php
    - OrbiterGatewayResourceInformationService.php
    - OrbiterGridCloudComputingService.php
    - OrbiterHpcInformationService.php

Remote Data Cache

- HFIR [308166/5.8 GB]
- IPNS [260620/510.77 GB]
- SNS [3268066/82.37 TB]
  - ARCS [415731/14.67 TB]
    - 2007_2_18_SCI [1401/53.91 GB]
    - 2008_1_18_CAL [5/693.97 KB]
    - 2008_2_18_CAL [1/72 KB]
    - 2008_2_18_SCI [222/7.35 GB]
    - 2008_3_18_SCI [6644/211.49 GB]
    - 2009_2_18_CAL [3/15.05 MB]
    - 2009_2_18_SCI [4461/113.35 GB]
    - 2009_3_18_CAL [25/36.57 MB]
    - 2009_3_18_SCI [11882/229.84 GB]
    - 2010_2_18_CAL [1/1.27 KB]
    - 2010_2_18_SCI [4937/74.52 GB]
    - 2011_2_18_CAL [126/4.77 GB]
    - 2011_2_18_SCI [12131/270.35 GB]
    - IPTS-1086 [753/15.26 GB]
    - IPTS-1089 [210/26.13 GB]
    - IPTS-1111 [434/26.43 GB]
    - IPTS-1119 [1102/17.68 GB]
    - IPTS-1126 [1767/48.7 GB]
      - 15 [702/11.92 GB]
      - shared [1065/36.79 GB]
    - IPTS-1134 [3248/76.36 GB]
    - IPTS-1136 [1592/23.34 GB]

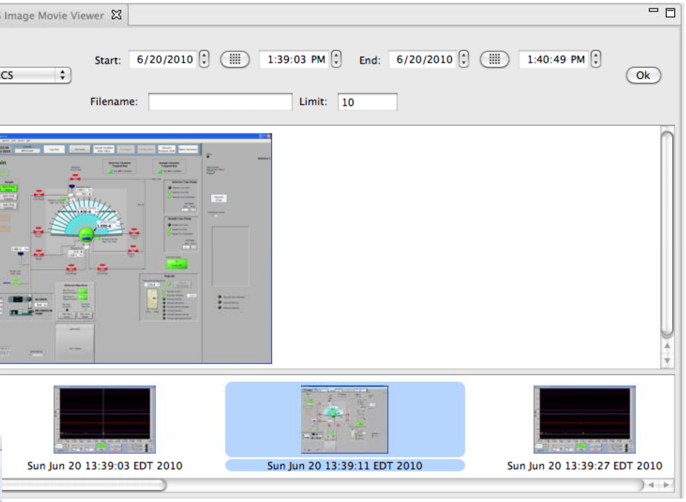
	IPTS Directory	Location	Filename
□	/SNS/ARCS/IPTS-1126	/15/3848/preNeXus	ARCS_3848_pulseid.dat
□	/SNS/ARCS/IPTS-1126	/15/3848/preNeXus	ARCS_3848_bmon2_histo.
□	/SNS/ARCS/IPTS-1126	/15/3848/preNeXus	ARCS_3848_cvinfo.xml
□	/SNS/ARCS/IPTS-1126	/15/3848/preNeXus	ARCS_3848_runinfo.xml
□	/SNS/ARCS/IPTS-1126	/15/3848/NeXus	ARCS_3848.nxs
□	/SNS/ARCS/IPTS-1126	/15/3848/preNeXus	ARCS_3848.nxt
□	/SNS/ARCS/IPTS-1126	/15/3218/preNeXus	ARCS_3218_pulseid.dat
□	/SNS/ARCS/IPTS-1126	/15/3218/preNeXus	ARCS_3218_neutron_event
□	/SNS/ARCS/IPTS-1126	/15/3218/preNeXus	ARCS_3218_bmon1_histo.
□	/SNS/ARCS/IPTS-1126	/15/3218/preNeXus	ARCS_3218.nxt
□	/SNS/ARCS/IPTS-1126	/15/3218/NeXus	ARCS_3218.nxs
□	/SNS/ARCS/IPTS-1126	/15/3218/preNeXus	ARCS_beamtimeinfo.xml
□	/SNS/ARCS/IPTS-1126	/15/3218/preNeXus	ARCS_3218_bmon2_histo.
□	/SNS/ARCS/IPTS-1126	/15/3218/preNeXus	ARCS_3218_cvinfo.xml
□	/SNS/ARCS/IPTS-1126	/15/3218/preNeXus	ARCS_3218_runinfo.xml
□	/SNS/ARCS/IPTS-1126	/15/3848/preNeXus	ARCS_beamtimeinfo.xml
□	/SNS/ARCS/IPTS-1126	/15/3848/preNeXus	ARCS_3848_neutron_event
□	/SNS/ARCS/IPTS-1126	/15/3848/preNeXus	ARCS_3848_bmon1_histo.
□	/SNS/ARCS/IPTS-1126	/15/3187/preNeXus	ARCS_3187_pulseid.dat
□	/SNS/ARCS/IPTS-1126	/15/3187/preNeXus	ARCS_3187_bmon2_histo.
□	/SNS/ARCS/IPTS-1126	/15/3187/preNeXus	ARCS_3187.nxt
□	/SNS/ARCS/IPTS-1126	/15/3188/NeXus	ARCS_3188.nxs
□	/SNS/ARCS/IPTS-1126	/15/3187/preNeXus	ARCS_beamtimeinfo.xml
□	/SNS/ARCS/IPTS-1126	/15/3187/preNeXus	ARCS_3187_runinfo.xml
□	/SNS/ARCS/IPTS-1126	/15/3187/preNeXus	ARCS_3187_neutron_event
□	/SNS/ARCS/IPTS-1126	/15/3187/preNeXus	ARCS_3187_bmon1_histo.
□	/SNS/ARCS/IPTS-1126	/15/3187/NeXus	ARCS_3187.nxs
□	/SNS/ARCS/IPTS-1126	/15/3187/preNeXus	ARCS_3187_cvinfo.xml
□	/SNS/ARCS/IPTS-1126	/15/3186/preNeXus	ARCS_3186_bmon1_histo.
□	/SNS/ARCS/IPTS-1126	/15/3186/preNeXus	ARCS_3186_cvinfo.xml
□	/SNS/ARCS/IPTS-1126	/15/3186/NeXus	ARCS_3186.nxs
□	/SNS/ARCS/IPTS-1126	/15/3186/preNeXus	ARCS_3186.nxt
□	/SNS/ARCS/IPTS-1126	/15/3186/preNeXus	ARCS_3186_pulseid.dat
□	/SNS/ARCS/IPTS-1126	/15/3186/preNeXus	ARCS_3186_runinfo.xml
□	/SNS/ARCS/IPTS-1126	/15/3186/preNeXus	ARCS_3186_neutron_event
□	/SNS/ARCS/IPTS-1126	/15/3186/preNeXus	ARCS_3186_bmon2_histo.
□	/SNS/ARCS/IPTS-1126	/15/3186/preNeXus	ARCS_beamtimeinfo.xml
□	/SNS/ARCS/IPTS-1126	/15/3851/NeXus	ARCS_3851.nxs

702 files – 11.92 GB (max 500 GB)

Name	Start	End	Size	Status	Progress

Orbiter Commander – Thick Client (continued)

- Atomic capabilities are provided as *modules* that can be installed as needed from a central module repository
- The Orbiter Federation RESTful SOA provides robust access to diverse capabilities, such as:
    - Multi-threaded streaming downloads of repository files
    - Live status monitoring of the beam
    - Slideshows of instrument application screenshots
    - Organization of modules into "*Suites*"



ARCS

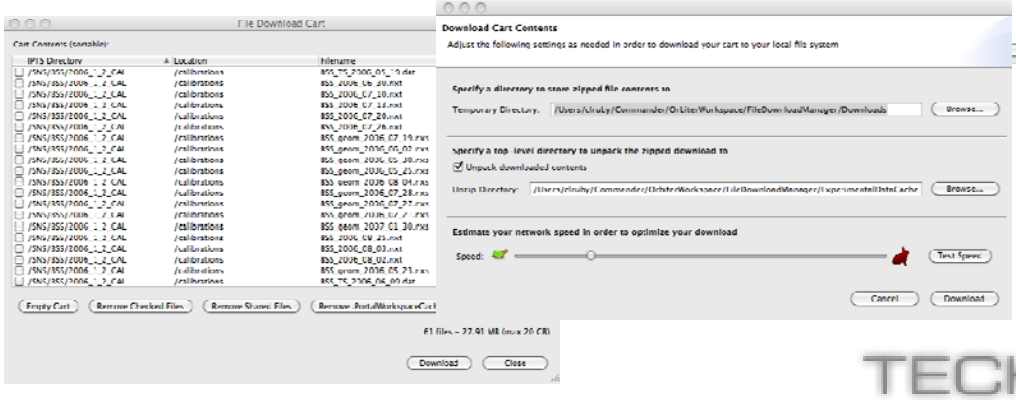
Start: 6/20/2010 1:39:03 PM End: 6/20/2010 1:40:49 PM

Filename: Limit: 10

Sun Jun 20 13:39:03 EDT 2010

Sun Jun 20 13:39:11 EDT 2010

Sun Jun 20 13:39:27 EDT 2010



Cart Contents (sortable):

	IPTS Directory	Location	Filename
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_TS_2006_05_19.dat
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_2006_06_30.nxt
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_2006_07_10.nxt
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_2006_07_13.nxt
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_2006_07_20.nxt
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_2006_07_26.nxt
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_geom_2006_07_19.nxs
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_geom_2006_06_02.nxs
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_geom_2006_05_30.nxs
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_geom_2006_05_25.nxs
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_geom_2006_08_04.nxs
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_geom_2006_07_28.nxs
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_geom_2006_07_27.nxs
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_geom_2006_07_21.nxs
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_geom_2007_01_30.nxs
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_2006_08_25.nxt
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_2006_08_03.nxt
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_2006_08_02.nxt
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_geom_2006_05_23.nxs
☐	/SNS/BSS/2006_1_2_CAL	/calibrations	BSS_TS_2006_06_09.dat

61 files – 27.91 MB (max 20 GB)

Download Cart Contents

Adjust the following settings as needed in order to download your cart to your local file system

Specify a directory to store zipped file contents to

Temporary Directory: /Users/druby/Commander/OrbiterWorkspace/FileDownloadManager/Downloads

Specify a top-level directory to unpack the zipped download to

☑ Unpack downloaded contents

Unzip Directory: /Users/druby/Commander/OrbiterWorkspace/FileDownloadManager/ExperimentalDataCache

Estimate your network speed in order to optimize your download

Speed:

For More Information

Contact:

Mark L. Green, Vice President of Systems Integration

716-204-8690

mlgreen@txcorp.com

https://orbiter.txcorp.com

TECH-X CORPORATION