PhEDEx Data Service

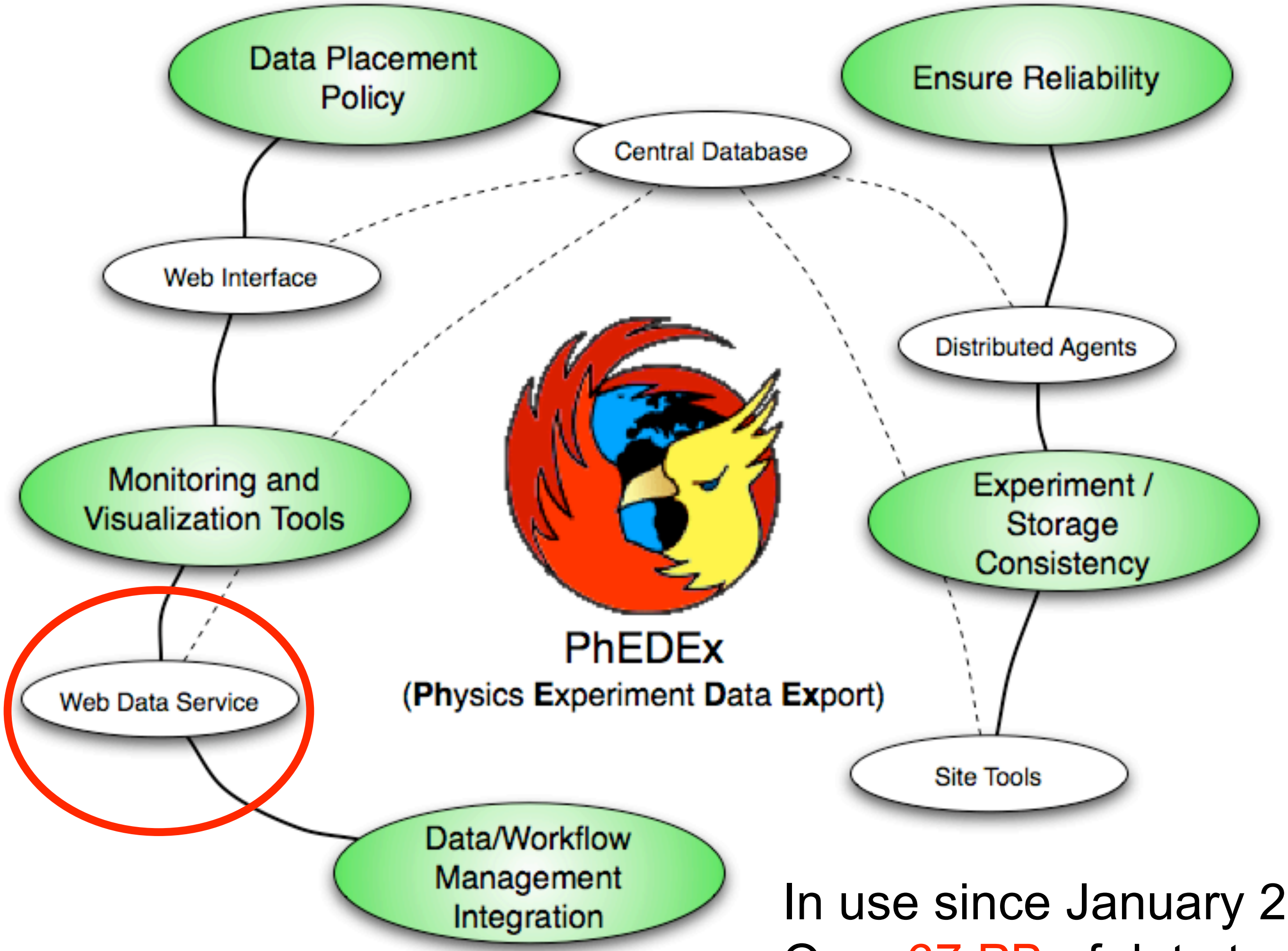
Ricky Egeland, University of Minnesota

on behalf of numerous PhEDEx contributors
and the CMS collaboration

presented at

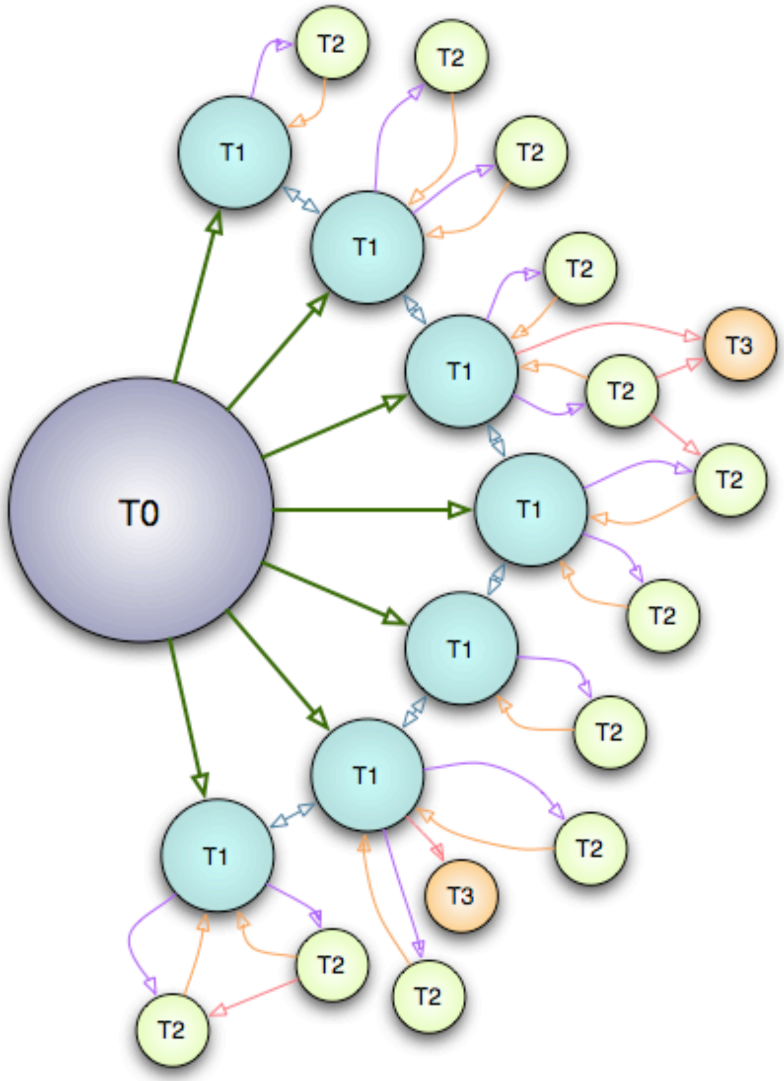
Computing in **H**igh **E**nergy **P**hysics (**CHEP**)
March 26th, 2009

What is PhEDEx?


Data Placement Policy
Ensure Reliability
Central Database
Web Interface
Distributed Agents
Monitoring and Visualization Tools
Experiment / Storage Consistency
Web Data Service
Site Tools
Data/Workflow Management Integration
PhEDEx
(Physics Experiment Data Export)

In use since January 2004
Over 67 PB of data transfers

Data Transfer Requirements



2008 Average Per-Link Requirements

T0 → T1

100 MB/s

T1 ↔ T1

140 MB/s

T1 → T2

25 MB/s

T1 ← T2

6 MB/s

T2 ↔ T2

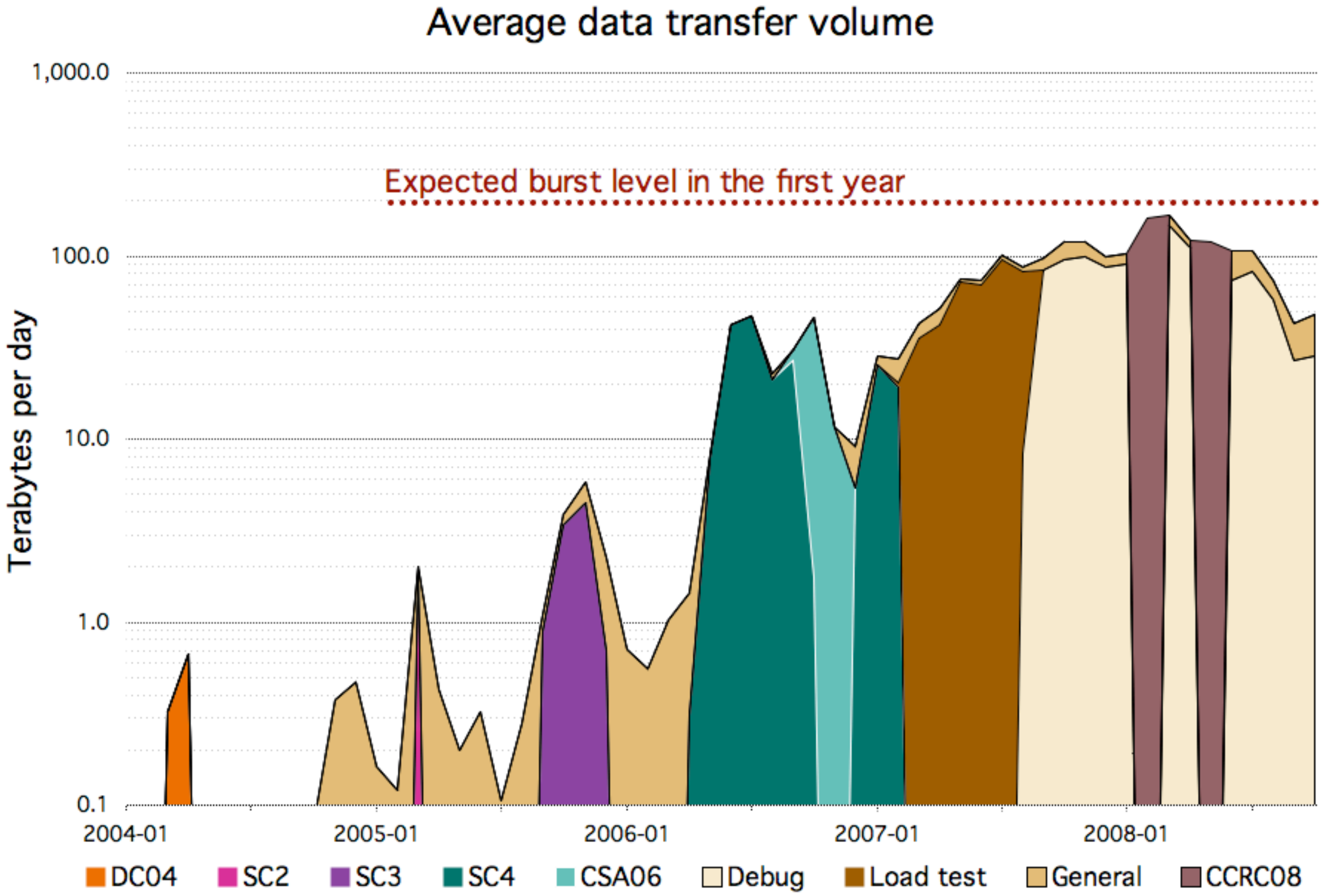
T1 → T3

T2 → T3

undefined, small

Scale of data

production database: 110 nodes, 4.3 M files, 11.7 M replicas



Data available in PhEDEx

- Datasets known to the transfer system
  - datasets / blocks / files
  - filesize, checksum
- Locations of datasets
  - (node, block) => (node, file)
- Requested data transfers
  - who, what, where, when, why
- Current / Historical transfer statistics
  - actively transferring data
  - rate, quality history
- Current / Historical site usage statistics
  - resident data size, requested data size
  - by group or “custodial”
- Monitoring
  - Recent transfer errors
  - Consistency check tests
  - System monitoring (Agent uptime, status)
- System structure
  - Transfer topology
  - Node list

Data Service Motivation

- Integration with other data management components
  - As painless as possible: only requirement is an HTTP client
  - PhEDEx became the "authorative source" for data location
  - Automatic dataset injection, subscription from production components. **The data service is not read-only.**
- Provide monitoring data to custom user scripts
  - As painless as possible: only requirement is an HTTP client
  - No database passwords to distribute
- Integrate transfer data with other monitoring services
- Provide data for next-generation website
- Maximize code re-use within our project

Application Design

https://cmsweb.cern.ch/phedex/datasvc/format/instance/api?options
https://cmsweb.cern.ch/phedex/datasvc/doc/api

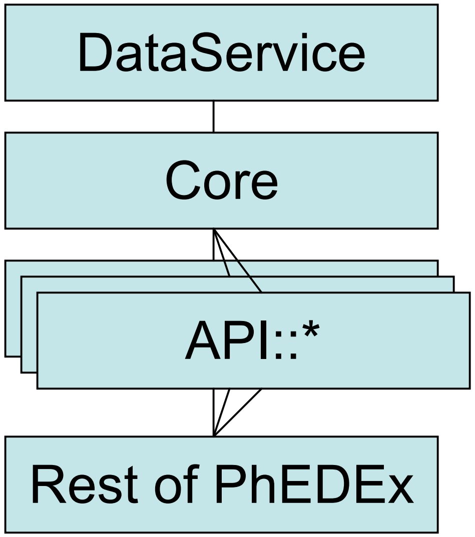
format : output format – xml, json, perl

instance : database instance – prod, debug, test

api : API to use – blockreplicas, subscribe

options : API options (mostly filters) – block=/X/Y/Z#123

doc : output documentation



DataService

Core

API::*

Rest of PhEDEx

configure and initialize application; parse URLs; print headers;

load API and pass options; print data object in requested format; manage caching, authorization;

validate input; call PhEDEx library functions; format and return data object; perldoc API documentation;

PhEDEx business logic and SQL

Design: Conventions

An API call returns only one data structure. An API call does one thing and one thing only. No option shall change the format of the returned data. This is to ensure that clients cannot be surprised by results and know what to expect.

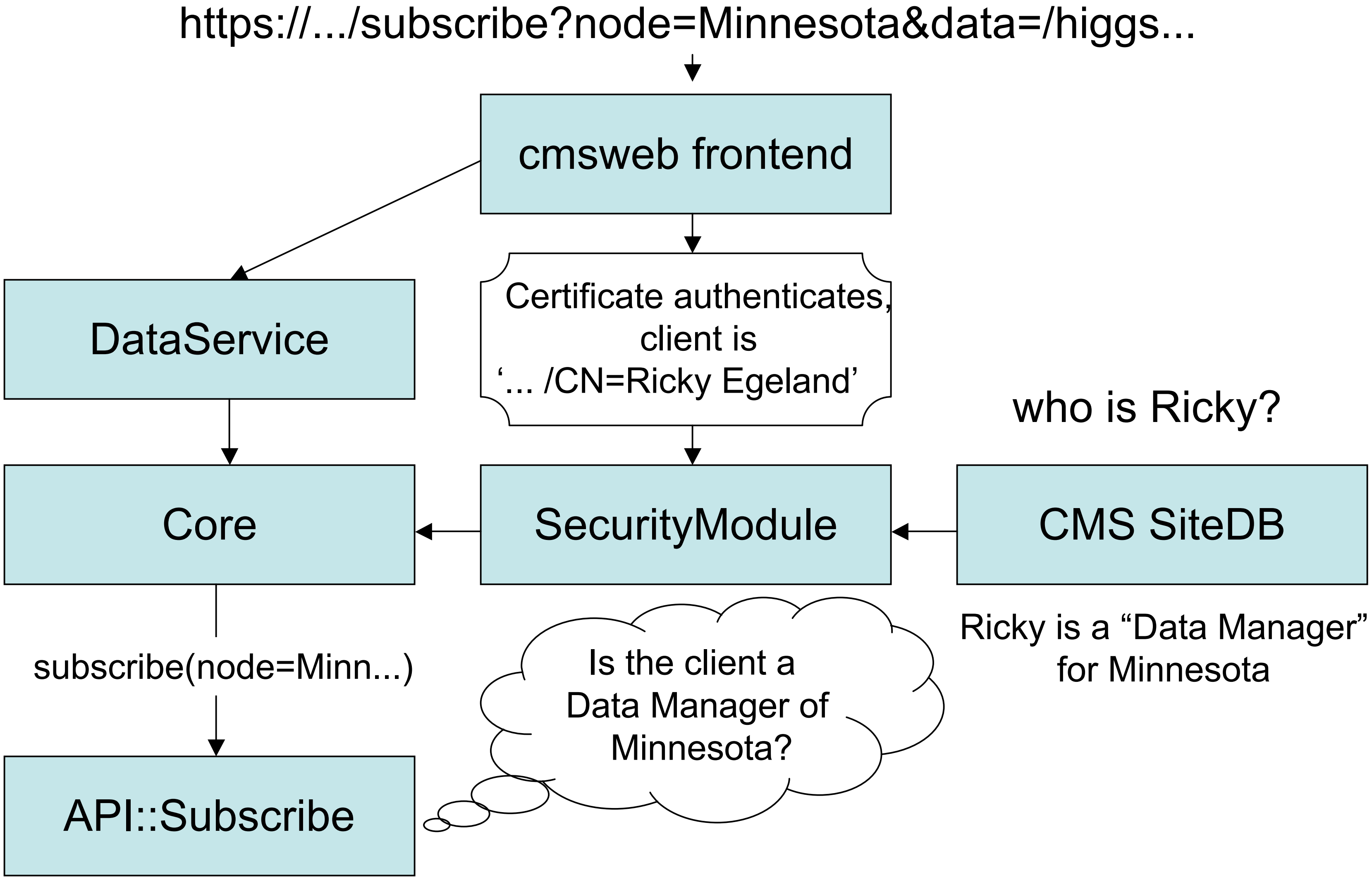
Common entities have required attributes across all API calls. Entities with unique IDs shall always have that ID as an attribute. Basic attributes (e.g. number of files in a block) will appear with that entity no matter what the context is. This is to allow for client-side correlation of results from separate calls.

Utilize hierarchy wherever possible. Do not flatten results, even where it seems convenient. The full context of data entities should be a part of the result, and the client should not have to rely on the options to the call to successfully interpret the results.

Consistent call/response semantics. An attribute which is filterable should have the same name in the response as in the input.

We develop the data service as if we were developing a library for future unknown and inexperienced developers.

Design: SecurityModule



https://.../subscribe?node=Minnesota&data=/higgs...

cmsweb frontend

Certificate authenticates, client is '... /CN=Ricky Egeland'

DataService

who is Ricky?

Core

SecurityModule

CMS SiteDB

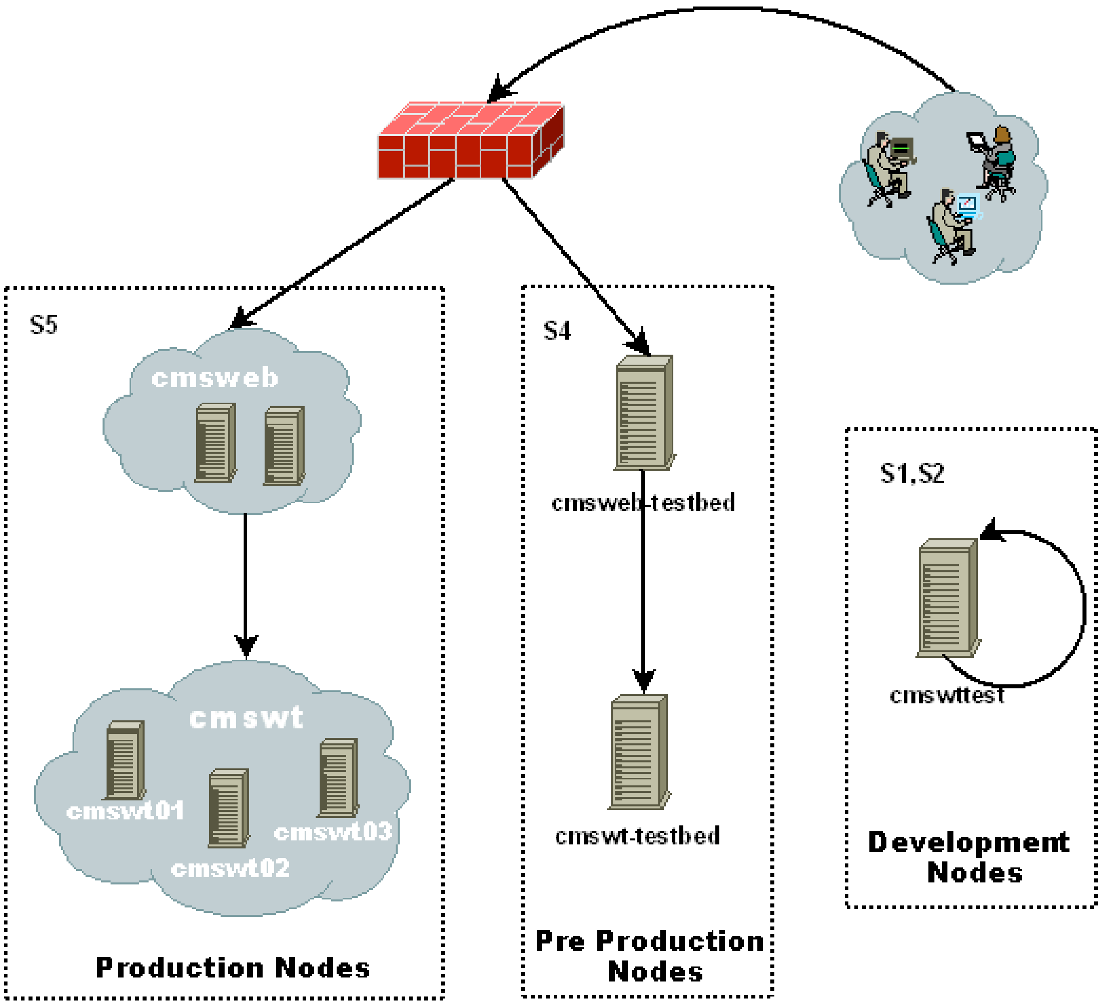
Ricky is a "Data Manager" for Minnesota

subscribe(node=Minn...)

Is the client a Data Manager of Minnesota?

API::Subscribe

Design: Deployment



- Load-balanced frontends providing certificate authentication
- Load-balanced backends hosting the data service
- Backends only accept requests from the frontends

(diagram by Patricia Bittencourt Sampaio, UERJ, Brazil)

Security & Deployment Summary

- cmsweb frontend provides certificate authentication
- SecurityModule+SiteDB provides authorization data
- Data Service implements fine-grained, secure access to APIs
- Only the frontends are accessible to the outside world
- The data service only accepts requests from the frontend
  - enforced both in the host firewall and in the server configuration

Client: Command-line tool

- First officially supported client, 'phedex'
- Uses LWP::UserAgent HTTP library
- Prints data “as-is” from the data service
- “report format” plugins for human-readable output

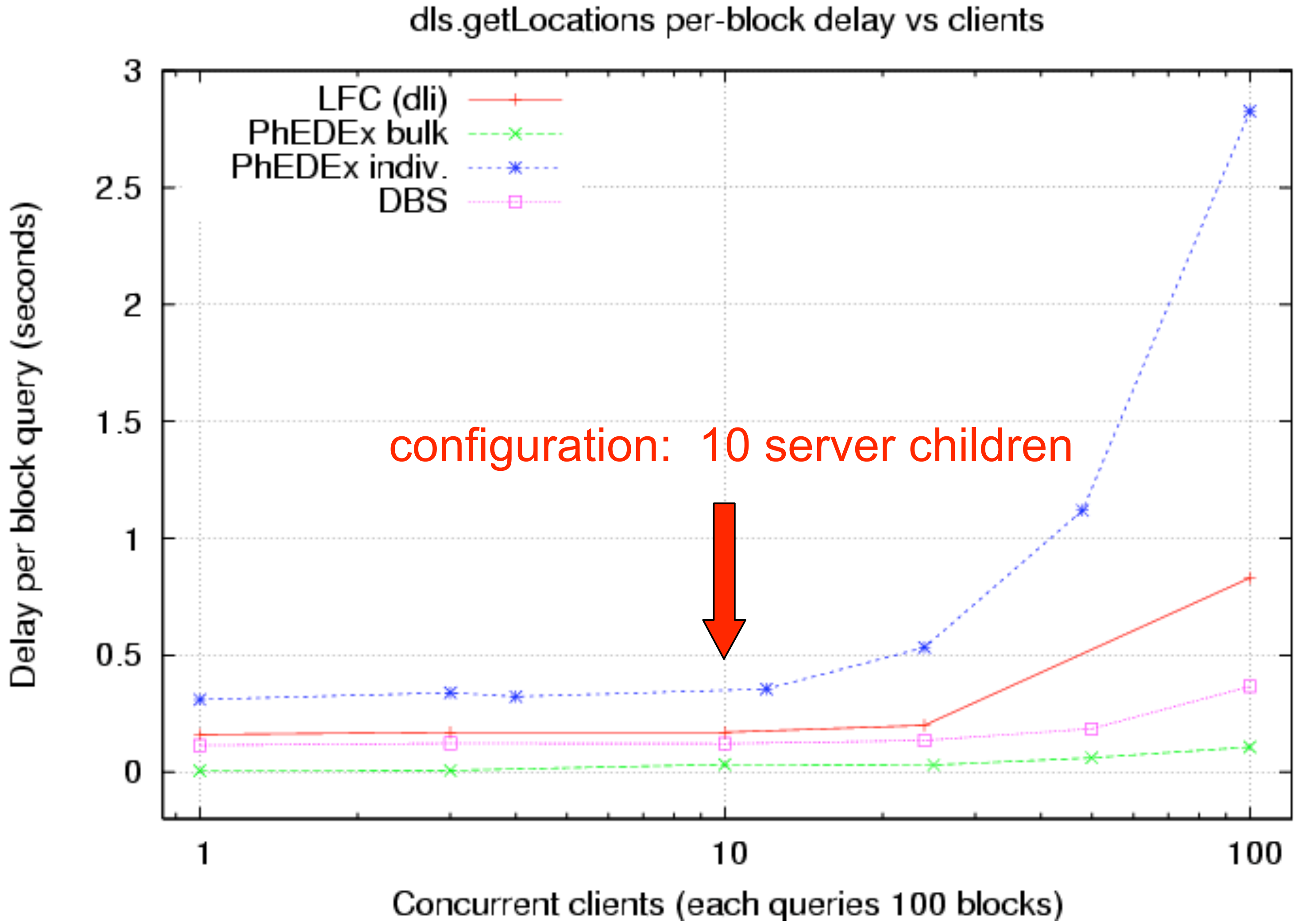
```
> phedex --format xml nodes > nodes.xml

> phedex --format xml blockreplicas --block '...' > myblock.xml

> phedex --format report nodes
NAME, SE, KIND, TECHNOLOGY
T0_CH_CERN_Export, (undef), Buffer, Castor
T0_CH_CERN_MSS, srm-cms.cern.ch, MSS, Castor
...

> phedex --format perl blockreplicas --block '...' | perl -e '...'
```

Design: FakeClient

- For development of APIs without the overhead of running the full apache / mod_perl stack
- A subclass of the 'phedex' command-line tool which fakes the ***server response***
  - faked authentication
  - http server access replaced by direct DataService library invocation
  - Exercises everything in the client/server stack
- Uncaught exceptions are detected immediately and reported at the command prompt
  - no log trawling
- Possible to run in a debugger

Performance: DLS client

DLS = **D**ata **L**ocation **S**ervice

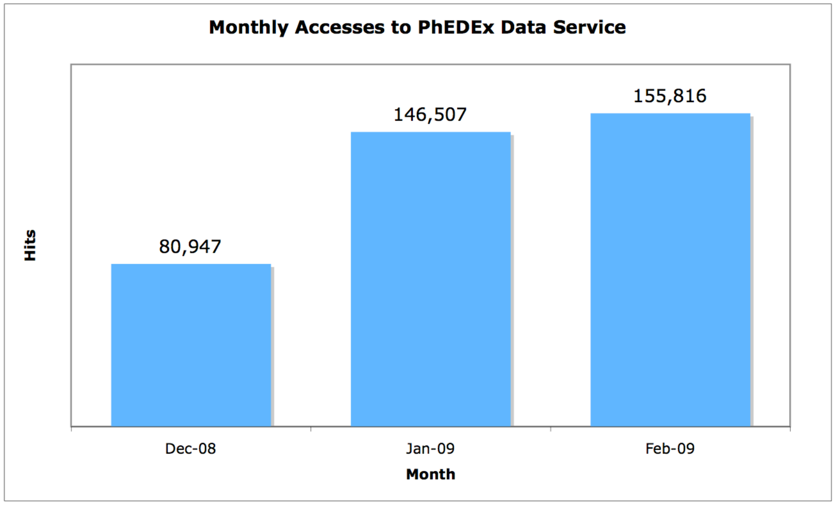


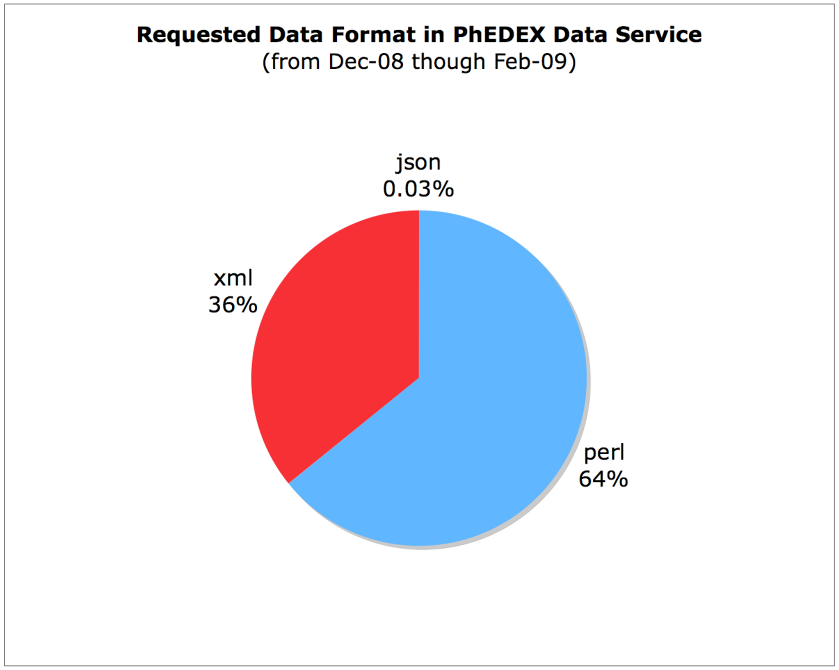
configuration: 10 server children

(plot and study courtesy Antonio Delgado Peris, CIEMAT, Spain)

Usage

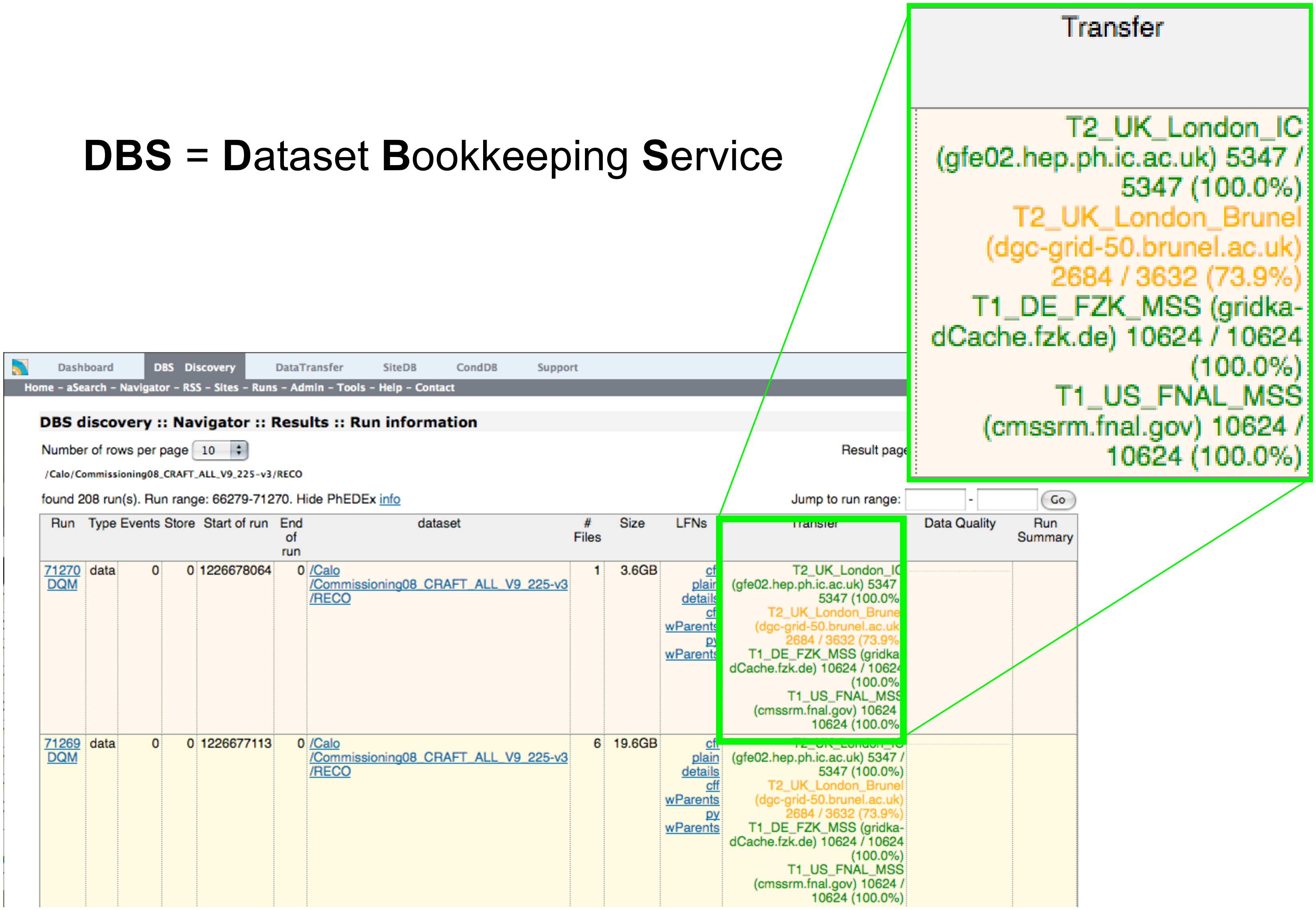
- CMS Computing projects
  - DLS Client / CRAB (Grid Analysis)
  - Data Discovery Page
  - Tier-0
- Other projects
  - Custom site consistency checking tools
  - Netvibes "PhEDEx Download Status" widget
- "Other projects" makeup the majority of hits to the service
  - Great!





Usage : DBS Discovery

DBS = **D**ataset **B**ookkeeping **S**ervice



Dashboard
DBS Discovery
DataTransfer
SiteDB
CondDB
Support
Home – aSearch – Navigator – RSS – Sites – Runs – Admin – Tools – Help – Contact

DBS discovery :: Navigator :: Results :: Run information

Number of rows per page 10

Result page

/Calo/Commissioning08_CRAFT_ALL_V9_225-v3/RECO

found 208 run(s). Run range: 66279-71270. Hide PhEDEx info

Jump to run range: - Go

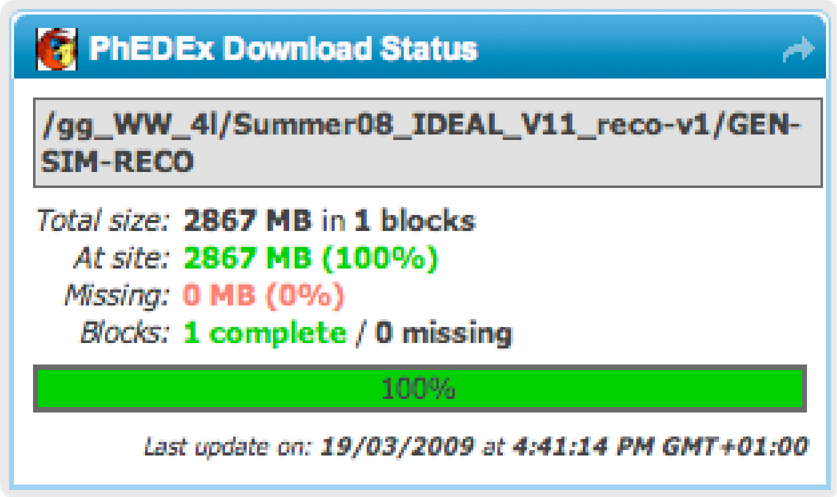
Run	Type	Events	Store	Start of run	End of run	dataset	# Files	Size	LFNs	Transfer	Data Quality	Run Summary
71270 DQM	data	0	0	1226678064	0	/Calo /Commissioning08_CRAFT_ALL_V9_225-v3 /RECO	1	3.6GB	cff plain details cff wParents py wParents	T2_UK_London_IC (gfe02.hep.ph.ic.ac.uk) 5347 / 5347 (100.0%) T2_UK_London_Brunel (dgc-grid-50.brunel.ac.uk) 2684 / 3632 (73.9%) T1_DE_FZK_MSS (gridka-dCache.fzk.de) 10624 / 10624 (100.0%) T1_US_FNAL_MSS (cmssrm.fnal.gov) 10624 / 10624 (100.0%)		
71269 DQM	data	0	0	1226677113	0	/Calo /Commissioning08_CRAFT_ALL_V9_225-v3 /RECO	6	19.6GB	cff plain details cff wParents py wParents	T2_UK_London_IC (gfe02.hep.ph.ic.ac.uk) 5347 / 5347 (100.0%) T2_UK_London_Brunel (dgc-grid-50.brunel.ac.uk) 2684 / 3632 (73.9%) T1_DE_FZK_MSS (gridka-dCache.fzk.de) 10624 / 10624 (100.0%) T1_US_FNAL_MSS (cmssrm.fnal.gov) 10624 / 10624 (100.0%)		

Transfer

T2_UK_London_IC (gfe02.hep.ph.ic.ac.uk) 5347 / 5347 (100.0%)
T2_UK_London_Brunel (dgc-grid-50.brunel.ac.uk) 2684 / 3632 (73.9%)
T1_DE_FZK_MSS (gridka-dCache.fzk.de) 10624 / 10624 (100.0%)
T1_US_FNAL_MSS (cmssrm.fnal.gov) 10624 / 10624 (100.0%)

https://cmsweb.cern.ch/dbs_discovery

Usage : Netvibes Widget



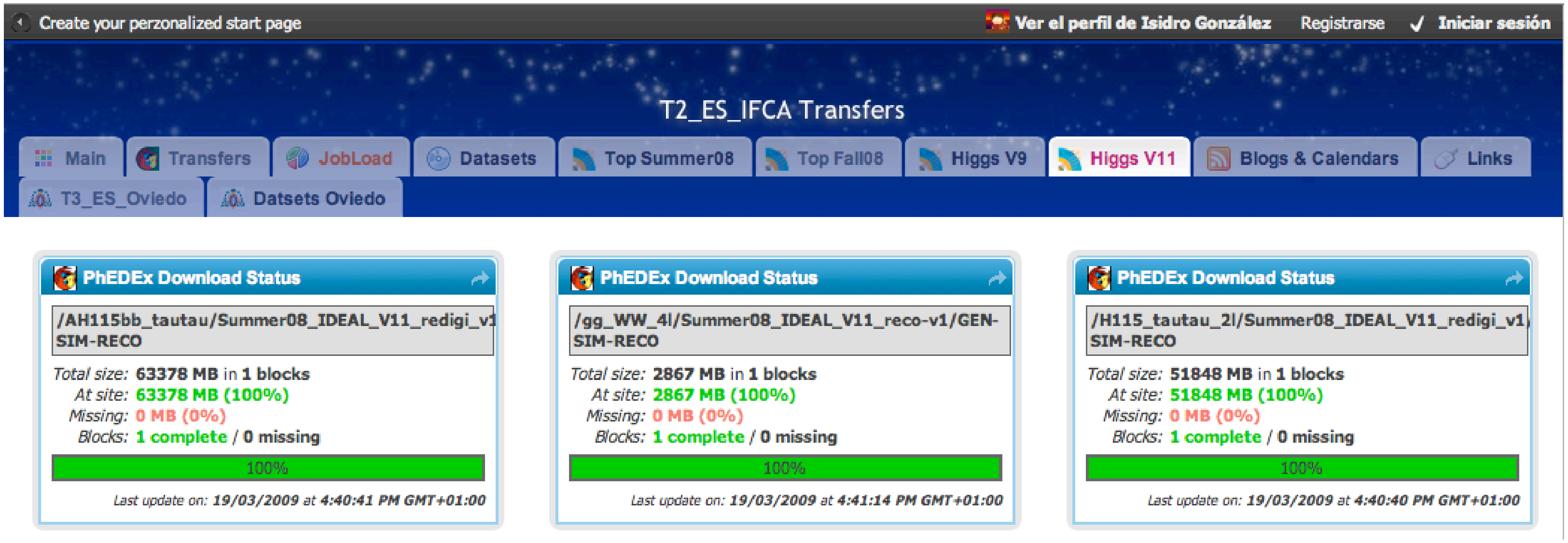
PhEDEx Download Status

/gg_WW_4l/Summer08_IDEAL_V11_reco-v1/GEN-SIM-RECO

Total size: **2867 MB** in **1 blocks**
At site: **2867 MB (100%)**
Missing: **0 MB (0%)**
Blocks: **1 complete** / **0 missing**

100%

Last update on: ***19/03/2009*** *at* ***4:41:14 PM GMT+01:00***



PhEDEx Download Status

/AH115bb_tautau/Summer08_IDEAL_V11_redigi_v1
SIM-RECO

Total size: **63378 MB** in **1 blocks**
At site: **63378 MB (100%)**
Missing: **0 MB (0%)**
Blocks: **1 complete** / **0 missing**

100%

Last update on: ***19/03/2009*** *at* ***4:40:41 PM GMT+01:00***

PhEDEx Download Status

/gg_WW_4l/Summer08_IDEAL_V11_reco-v1/GEN-SIM-RECO

Total size: **2867 MB** in **1 blocks**
At site: **2867 MB (100%)**
Missing: **0 MB (0%)**
Blocks: **1 complete** / **0 missing**

100%

Last update on: ***19/03/2009*** *at* ***4:41:14 PM GMT+01:00***

PhEDEx Download Status

/H115_tautau_2l/Summer08_IDEAL_V11_redigi_v1
SIM-RECO

Total size: **51848 MB** in **1 blocks**
At site: **51848 MB (100%)**
Missing: **0 MB (0%)**
Blocks: **1 complete** / **0 missing**

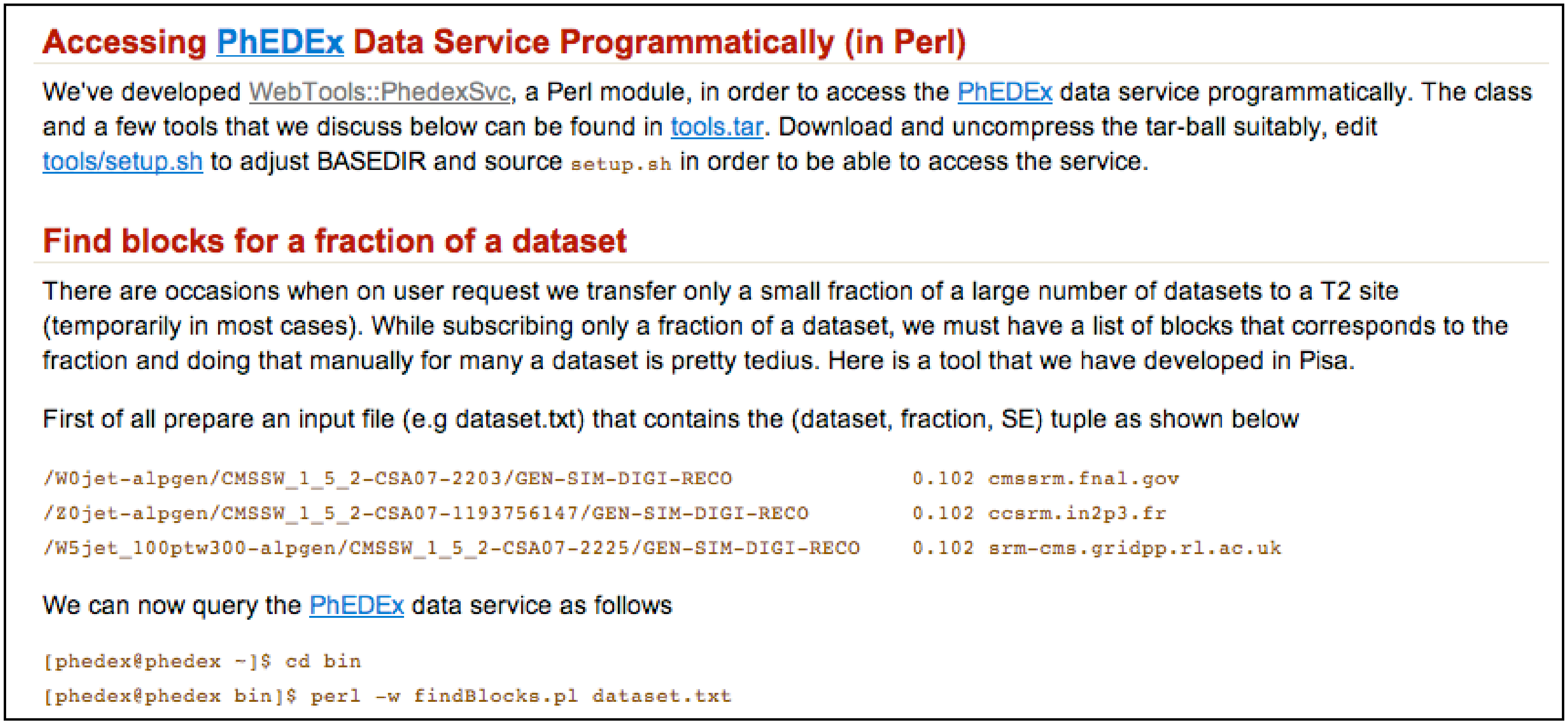
100%

Last update on: ***19/03/2009*** *at* ***4:40:40 PM GMT+01:00***

http://www.netvibes.com/gonis#Higgs_V11

widget by Isidro Gonzàlez – CIEMAT, Spain

Usage : Site Tools



Accessing PhEDEx Data Service Programmatically (in Perl)

We've developed WebTools::PhedexSvc, a Perl module, in order to access the PhEDEx data service programmatically. The class and a few tools that we discuss below can be found in tools.tar. Download and uncompress the tar-ball suitably, edit tools/setup.sh to adjust BASEDIR and source `setup.sh` in order to be able to access the service.

Find blocks for a fraction of a dataset

There are occasions when on user request we transfer only a small fraction of a large number of datasets to a T2 site (temporarily in most cases). While subscribing only a fraction of a dataset, we must have a list of blocks that corresponds to the fraction and doing that manually for many a dataset is pretty tedius. Here is a tool that we have developed in Pisa.

First of all prepare an input file (e.g dataset.txt) that contains the (dataset, fraction, SE) tuple as shown below

```
/W0jet-alpgen/CMSSW_1_5_2-CSA07-2203/GEN-SIM-DIGI-RECO            0.102 cmssrm.fnal.gov
/Z0jet-alpgen/CMSSW_1_5_2-CSA07-1193756147/GEN-SIM-DIGI-RECO      0.102 ccsrm.in2p3.fr
/W5jet_100ptw300-alpgen/CMSSW_1_5_2-CSA07-2225/GEN-SIM-DIGI-RECO  0.102 srm-cms.gridpp.rl.ac.uk
```

We can now query the PhEDEx data service as follows

```
[phedex@phedex ~]$ cd bin
[phedex@phedex bin]$ perl -w findBlocks.pl dataset.txt
```

https://twiki.cern.ch/twiki/bin/view/CMS/ItalianT2ToolsDataTransfers

tools by Subir Sarkar – INFN-Pisa, Italy

Future directions

- 2nd official client: PhEDEx web site

More datasvc APIs for official website → Interesting UIs from outside developers → Integration back into official website

- More RESTful?
  - Current APIs represent a search, not a resource
- Implement caching
  - Caching results of a wildcard search on a dynamic source leads to a low hit ratio
  - Caching results of distinct entity accesses would be more feasible
  - Find a performance balance between bulk accesses and cacheable resources

Conclusion

- PhEDEx data service satisfies integration requirements of CMS computing
- Provides a platform for increased developer involvement
- Allows for increased code re-use within and outside the project
- The data service is planned to become an even more integral component to the PhEDEx project

More info:

PhEDEx Data Service:	http://cmsweb.cern.ch/phedex/datasvc/doc
PhEDEx Web Site:	http://cmsweb.cern.ch/phedex
contact PhEDEx:	cms-phedex-admins@cern.ch