

Web APIs for the Big Science Enterprise

David Skinner

Lawrence Berkeley National Laboratory

April 16, 2013

Our computing & data apparatus is getting bigger.

Increasingly Big Science relies on the machinery of software.

prototypes → reliable infrastructure

GO is getting some serious traction

http://en.wikipedia.org/wiki/Big_Science



How to maintain / sustain?

How can we make reliable plans together?



Evolution of LHC computing

In chronological order:

1. Copy as much data as feasible to analysis centers worldwide, with hierarchical distribution.
2. Relax the hierarchy and rely on caching.
3. Use “federated data stores” to fetch *portions* of relevant data sets from remote storage (anywhere), just before they’re needed.

Increasing faith in global science networks. Enterprise.

Bill and Ted’s Excellent (European) Adventure

	NETFLIX	LHC Computing
Bandwidth per client	1.5Mbit/sec	1MByte/sec
Clients	1M*	100k cores
Serving	1.5Tbits	0.8Tbits
Total Data Distributed	12TB	20PB
Annual Budget	>\$4B	< \$.04B

Similar Problems:
Not all files are equally accessed

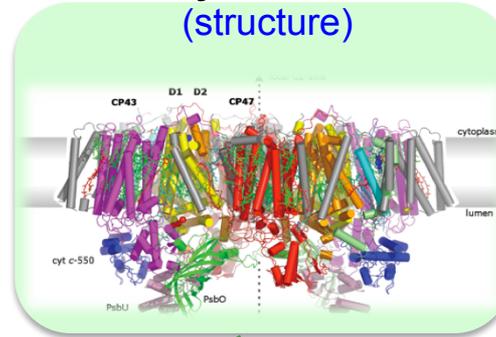


e.g. Forward Physics ;-)

Science Facilities are Now Coupled by Science Networks

- Networks and software are part of the apparatus
- Recent beam time on free-electron laser (LCLS) at SLAC to take 'snapshots' of catalytic reaction in Photosystem II (Nick Sauter et al).
- Data transported to a nearby HPC resource (NERSC) for real-time computational analysis.
- This one experiment *tripled* NERSC's network utilization.

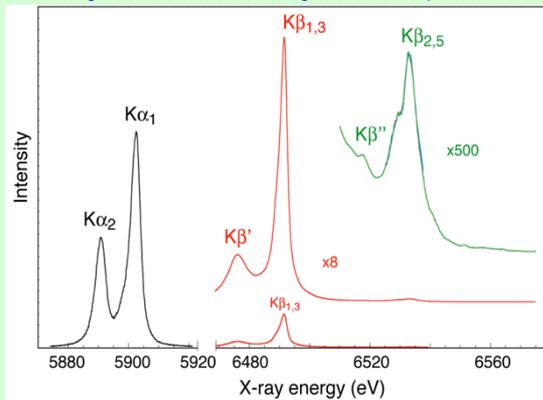
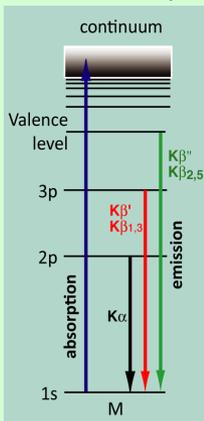
X-ray diffraction (structure)



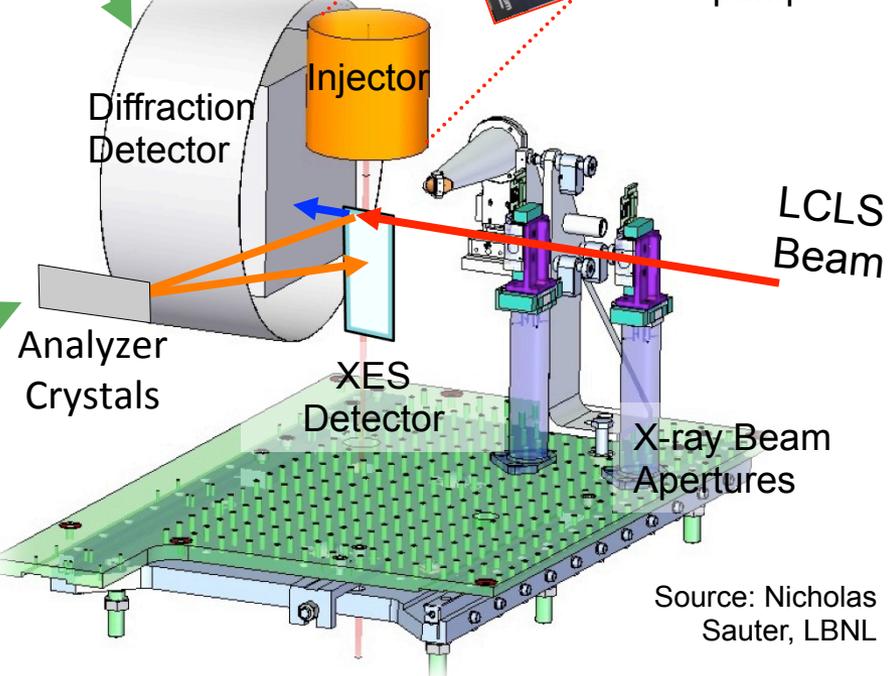
Liquid-jet
Injection of
 μm -size crystals

Optical
pump

X-ray emission spectroscopy (Chemistry at the catalytic site)



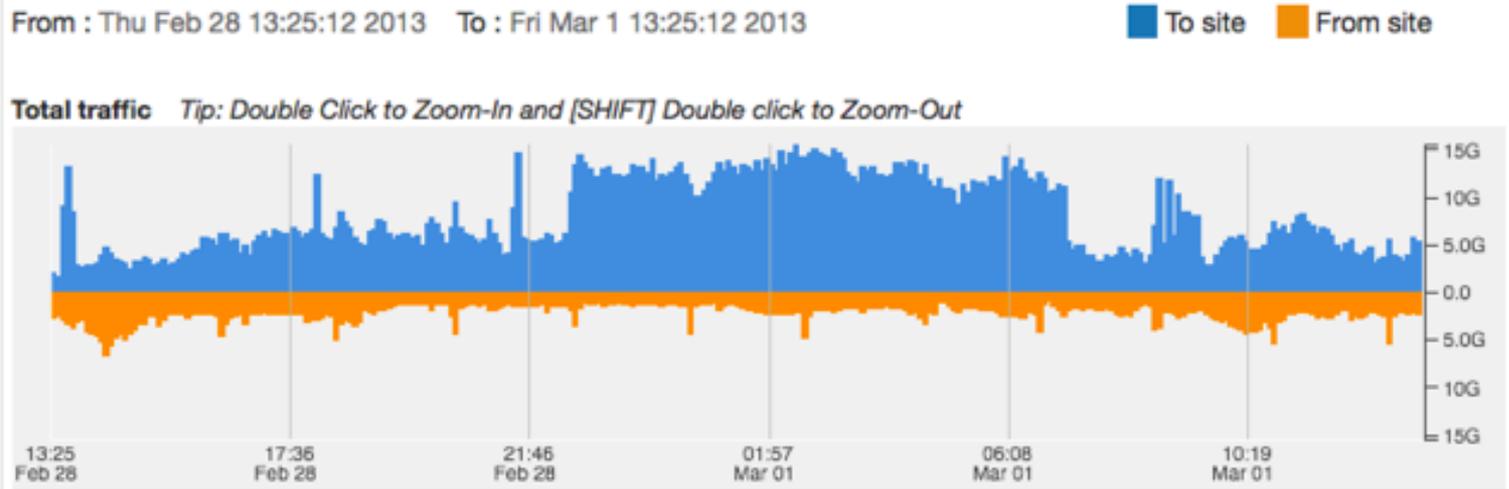
- charge density/spin state
- ligand environment



Source: Nicholas
Sauter, LBNL

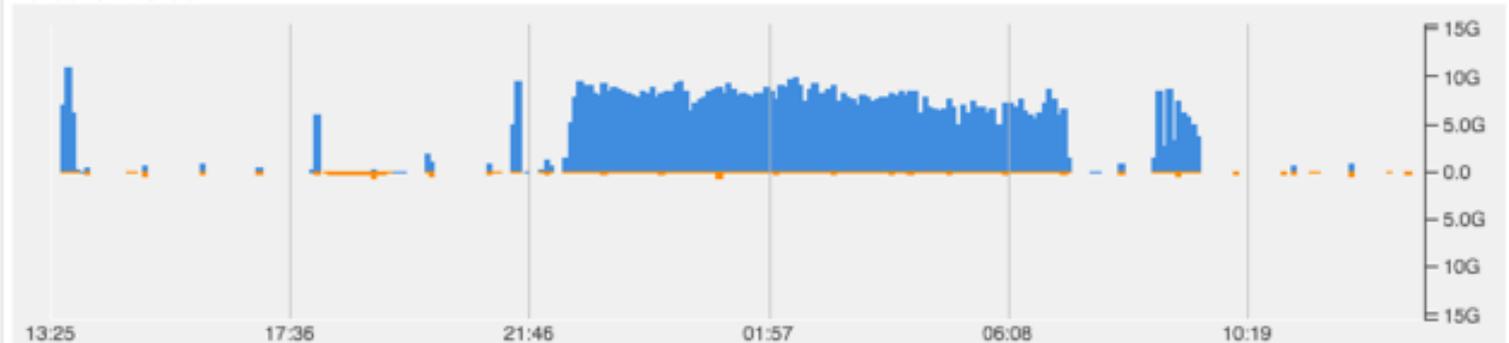
Kern et al (2012) PNAS 109: 9721
Sierra et al (2012) Acta Cryst D68: 1584
Mori et al (2012) PNAS 109: 19103

One Experiment == 150 TB, growing



Traffic split by : 'Autonomous System (origin)'

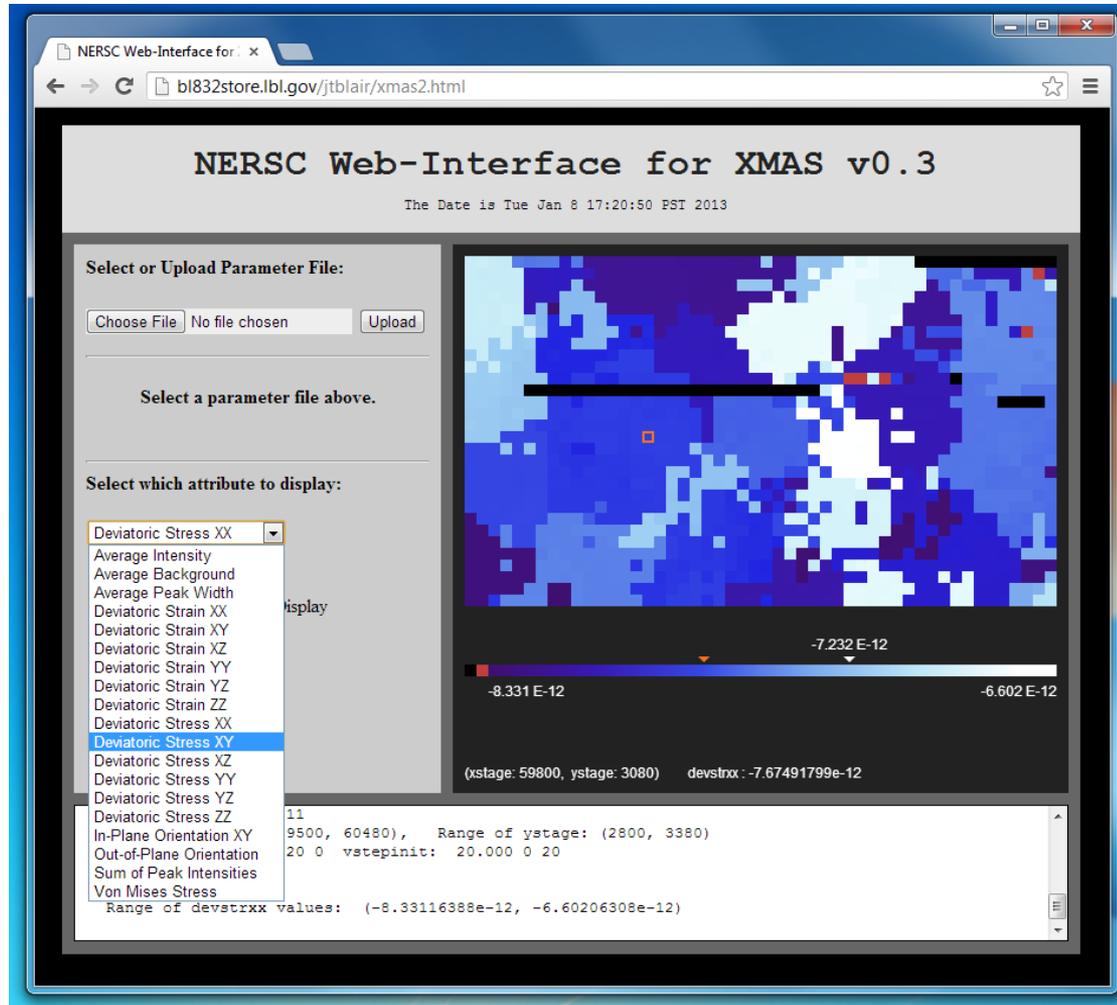
nersc-SLAC:3671



All NERSC
Traffic

PS II Study

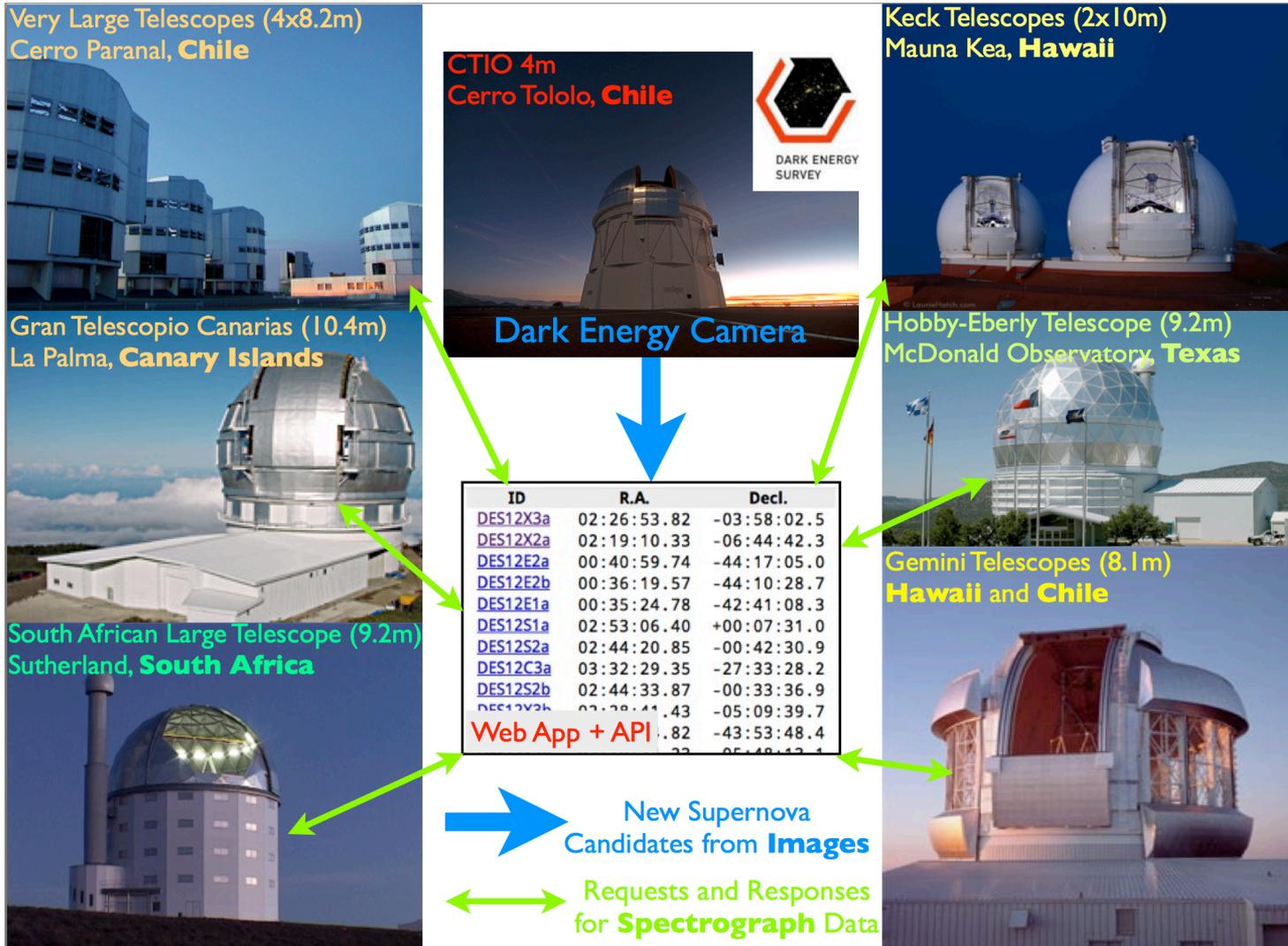
Big Science : Beamline in a Browser



- Establish HPC conduit to NERSC capabilities for ALS users to analyze data in real time
- Develop new visualization tools for displaying multi dimensional datasets
- Introduce reverse modeling tools for understanding plastic deformation at the mesoscale and guiding experiments

Q → Craig Tull (BESCWG)

Big Science : Global Telescope Choreography



Q → Rollin Thomas (LBL)

Let's Talk About Data culture:

Are you a HEP or a BES?



@CERN

Let's talk about a data plan for the next 10 years

Unleash the data team on this problem.

Initiating globally distributed data analysis in 3,2,1.

Let me tell you about our middleware software stack..

@ the Beamline

I need the data tomorrow. Deadline is next week.

Minions? There are 3 people on my team.

How can I trust any computer that's not right in front of me?

Let me tell you about Netflix, Big Data when I want it.

Topics for Globus World

- RESTful Web APIs for Science
- NEWT a web API for HPC
- Examples running at NERSC now
- What's Wrong with APIs

REST: A software architecture (for the web) that we can actually use. Roy did that.

“app enabler” Everyone “knows” how to make a RESTful API.
HATEOAS

Maybe a little too cool. There are now 32 bazillion of them

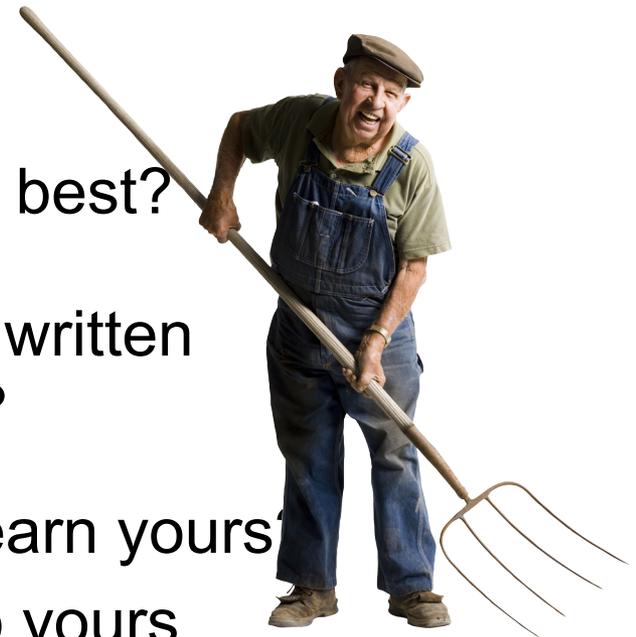
hypermedia

Which one is the best?

Why were they all written
by data providers?

Why do I have to learn yours?

Why didn't you do yours
the “right” way?



REST APIs have brought science *resources* into a new era of accessibility. This magnifies value of those *resources* and enables new science.

Of critical importance going forward into Big Data

Must understand enterprise level value propositions

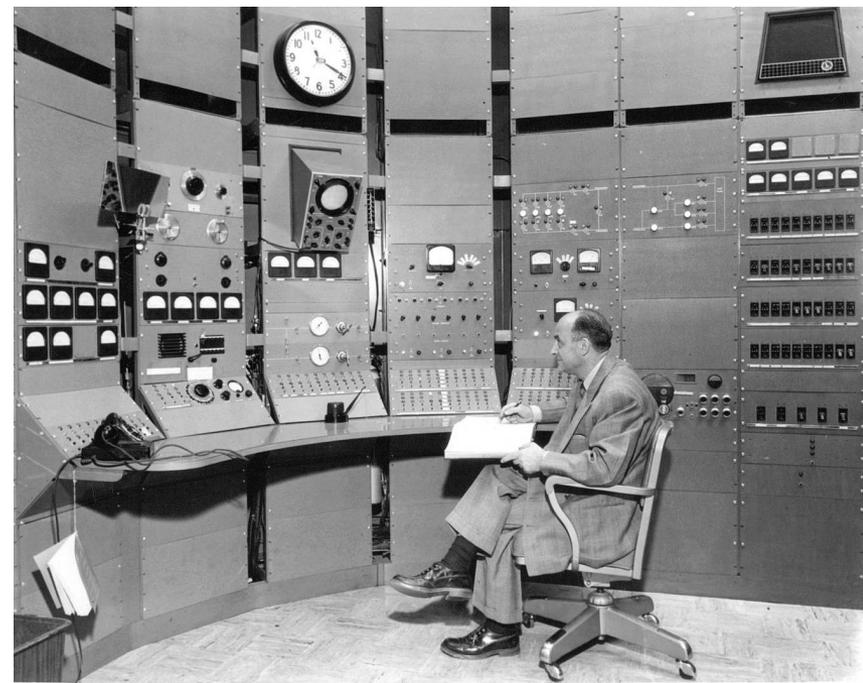
190 REST APIs for Science

RESTful Interface Circa 1955

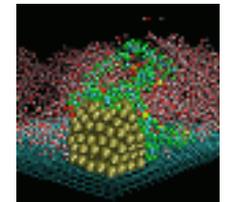
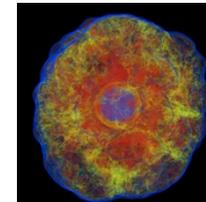
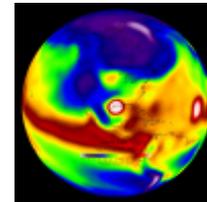
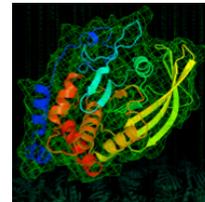
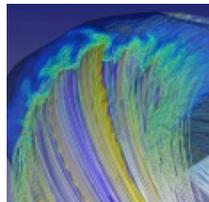
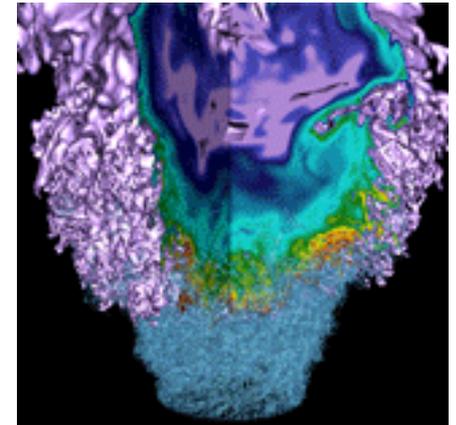


<http://www.w3.org/community/hpcweb/>

W3C Community
and Business
Groups :
HPCWEB



NEWT API

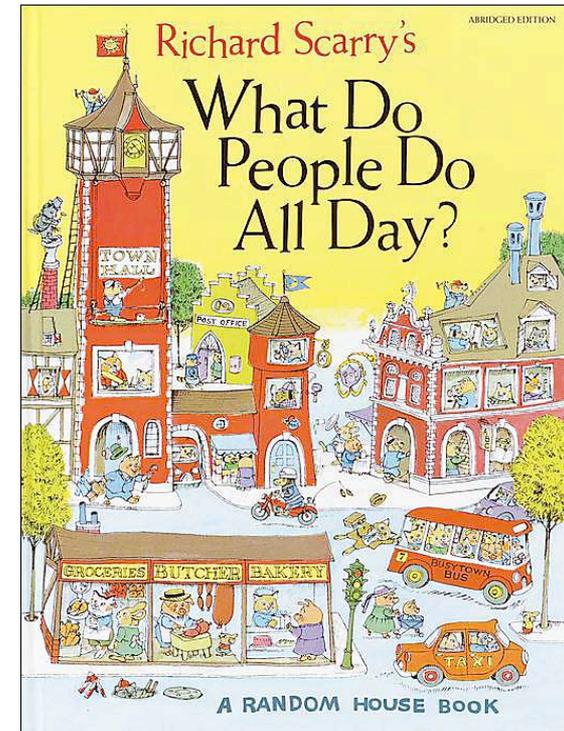


NEWT : An API for HPC Stuff

- Authenticate using NERSC credentials
- Check machine status
- Upload and download files
- Work with Big Data remotely
- Submit a compute job
- Monitor a job, check your workflow
- Get user account information
- Store app data (not scientific data)
- Issue UNIX commands

Q: Do this all in your browser?

A: Yes with a science gateway!

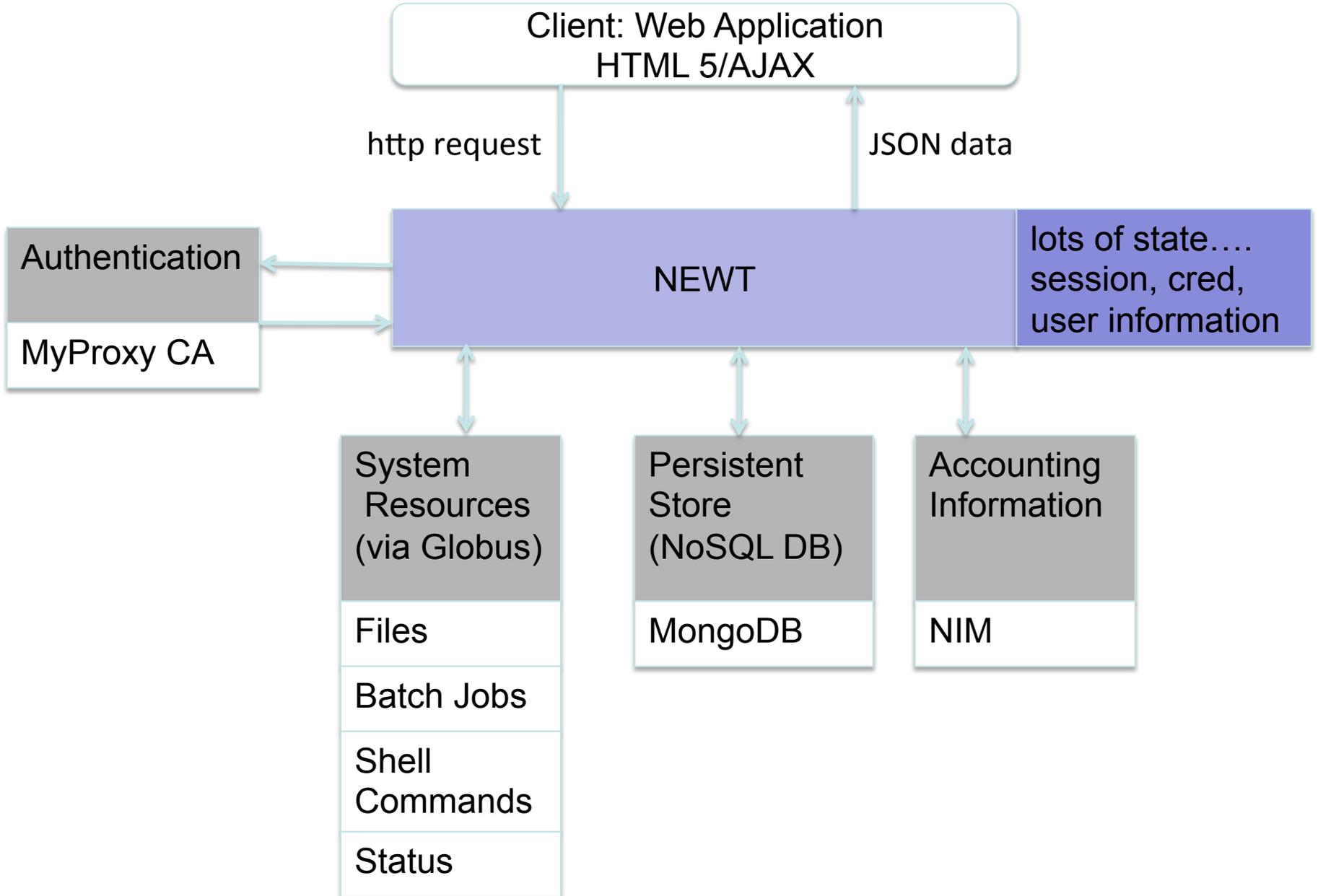


Map resources to URI that use web friendly formats (HTML, JSON, etc.) that deliver or accept information about the state of a resource.

What's a resource? It's whatever you want it to be.

VERB	RESOURCE	DESCRIPTION
POST	/queue/R	Submits POST data to queue on R; returns job id
GET	/file/R/path/	Returns directory listing for /path/ on R
GET	/account/user/U	Returns user account info for U
DEL	/store/DB/DOC	Deletes object DOC in DB

See Roy Fielding's Thesis



Client: Web Application
HTML 5/AJAX

http request

JSON data

Authentication

NEWT

lots of state....
session, cred,
user information

MyProxy CA

System Resources
(via Globus)

Persistent Store
(NoSQL DB)

Accounting Information

Files

MongoDB

NIM

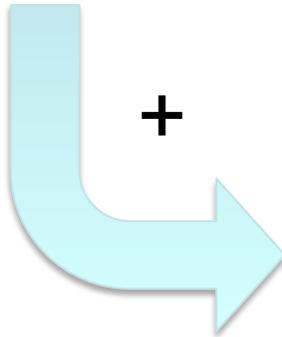
Batch Jobs

Shell Commands

Status

Job id	Name	User	Time Use S Queue
997298.sdb	zigzagBwH+600	huangj3	0 H reg_small
997312.sdb	zigzagBwH+100	huangj3	0 H reg_small
997314.sdb	zigzagNwH+600	huangj3	0 H reg_small
997319.sdb	zigzagNwH+100	huangj3	0 H reg_small
1274346.sdb	QD09a	zhaohui	0 H reg_small
1300198.sdb	QD65	zhaohui	0 H reg_small
1340353.sdb	...ne1KL12h.R2.1	lanxin	0 H reg_small
1399578.sdb	T833N010ppm	helasrag	0 H reg_small
1399579.sdb	T833N010ppm	helasrag	0 H reg_small
1399582.sdb	T833N010ppm	helasrag	0 H reg_small
1399583.sdb	T833N010ppm	helasrag	0 H reg_small

dskinner@hopper02:~>



Command line when you need it, web when you don't

Use HTML+APIs to make science gateways (web apps)

QSUB Demo

Choose one of the following:

Enter a job file path on franklin:

Submit the batch script below to Franklin:

ID:

Status:

Data Headaches Circa 2013

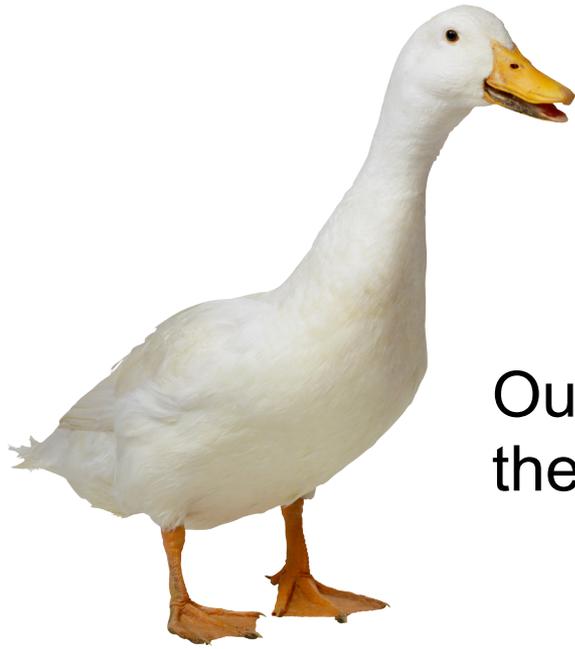
Overheard between Dr. Computing & Dr. Science

“I can’t collaborate with you because my data is too big for you kiddo. You could get pieces, but your connection is so weak that you’d never be able to really work with the data. Best if we just say no.”

“I can’t show you my data analysis workflow because the moving parts are too big. There’s no small parts you can interact with.”

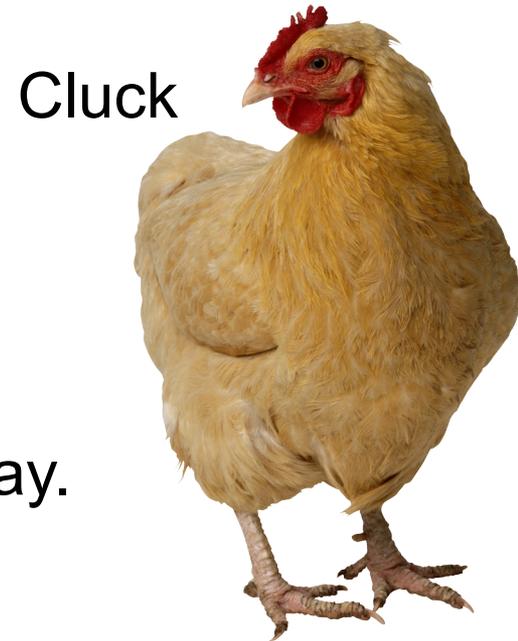
“I’d write an API but I am not sure what people want. Could be a huge waste for my team, and a little embarrassing, if nobody came knocking.”

“How can I provide data access in graduated ways? I want to provide more data to teams who prove they are doing world-class stuff with my world-class data, not waste time on tourists.”



Quack

Our progress is measured by the extent to which these go away.



Cluck

Data Headaches Circa 2013

Overheard between Dr. Science vs. Dr. Computing:

“I can’t collaborate with you on that Big Data proposal because I am a simple spectroscopist. You’d probably ask me to learn a bunch a CS rigamarole before I could even ‘ls’ your data. Phooey.”

“I know I could demonstrate something very cool if I had access to that data set, even a little slice. How can I persuade Dr. Computing of this without asking for the whole thing? I don’t even know which part to ask for? How can I stay away from the parts I don’t need?”

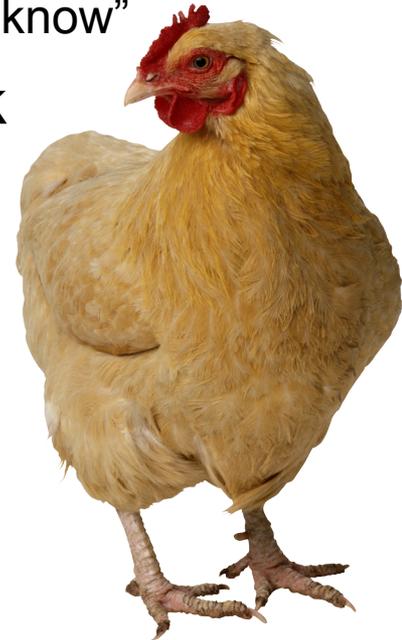
“I can’t collaborate with you because your API doesn’t do what I want. Asking you to change it would probably be imposing. You wrote the API so you must know”



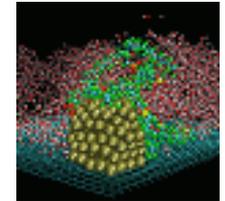
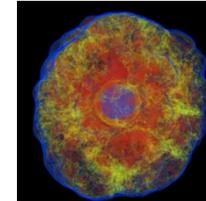
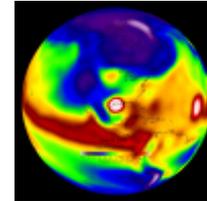
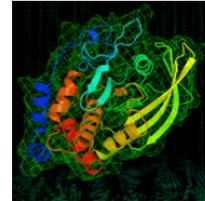
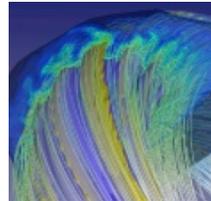
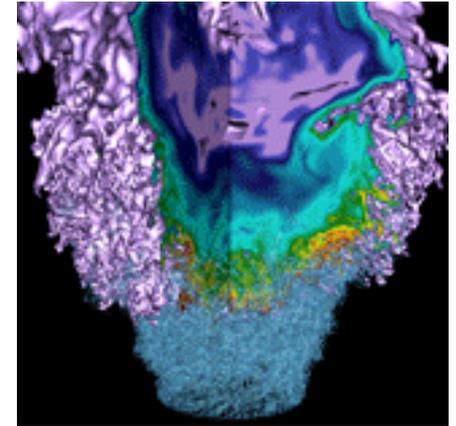
Quack

Our progress is measured by
the extent to which these go away.

Cluck



NEWT Futures



NEWT 1.0 :

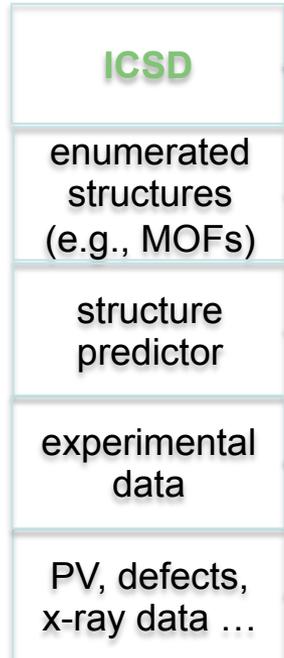
- NERSC Web Toolkit REST API that covers things people do at HPC centers
- Batch and interactive oriented execution
- HTTP based file/data movement

NEWT 2.0:

- Refactoring around flatter URLs
- Batch, interactive, and pipeline oriented execution
- High Throughput Computing (HTC)
- New approaches for wildcarding
- Data analysis provenance through tagging
- Extensibility: requests to extend/modify API through POSTS
- Take discussion to W3C

A Design Target for NEWT : Enable Materials Genomics

computational survey drivers



simulation engines



Hands-off Expertise Automation (A)
High Throughput Computing (A)
Big Data Materials Ontologies (B)
Probabilistic Data Management (B)
Anomaly Detection (C)
Consistency / V&V (C)
Query Language / Web Gateway (D)
Experimental validation is key(!)

operational
In 2012

functional
electronic
materials

phosphors
&
scintillator

carbon
storage

critical
materials...

materials genome

Community
Web
Gateway

Big Data Enterprise means acknowledging true costs

Big Data is not free (as in beer)



Real infrastructure,
Real costs
Real money

Nights and weekends are easier for my team.

Awesome.
I've got \$3.50.
Make mine a
double. I'll
take a small
schema too.
To go.



I'd buy four if you charged \$1.25.

Sold.

Data Discovery

Whatcha looking for? Whatcha got?

I need it in another format. No problem?

What does this data even look like? Give me just a little bit and I'll get back to you.
"Insight before a terabyte"

People who liked data XYZ also like YZXV.

Data is the easiest product in the universe to test-drive.
Why aren't we doing that?



Reflections on NEWT 1.0, thoughts on 2.0

A web API tailored for HPC purposes

Very Accessible

Took 2 people 7 years to write, we needed a REST

Toolkit approach (ugh)

small data

Popular w/ NERSC users
(always a good sign)

“stiff”

no HTC

limited vocabulary



We can rebuild him better. We have the technology.
(remember that one time when scientists invented the world wide web?)

The Tyranny of the API

How many APIs do you want to learn?

Written in a one-sided way

Speculation by publisher as to user

No way to vote on what you like, suggest improvements

My team wants that data/resource our way.

Sometimes need to mask multiple APIs
under a single unified view



All those errors

Why are so many APIs written by the data publisher? I could do better.

Closing Thoughts

Imagine that the API spec was a google doc someplace.

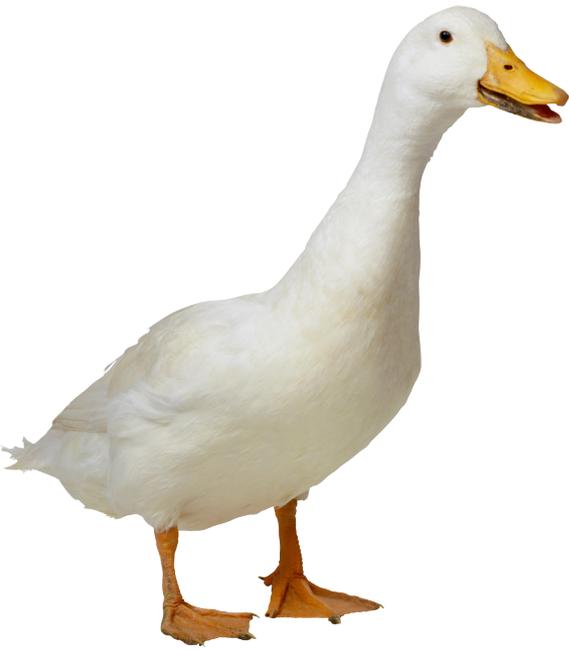
What if you let some people comment/suggest changes or alternatives?

What if you let people you trust make actual changes to the spec?

What if every version of the spec and every transaction against every version was logged?

What if you could copy one spec to another? Imagine you had fine grained controls.

What if you could do all this through a RESTful interface?



Quack?presto!..... Ok. Quack.

Thanks!



Big Data. Big Science. Let's Build Something Great.

