Big Data - Flexible Data - For HEP

Brian Bockelman
CHEP 2013

Goal for today:

```
cat BigData.txt | grep HEP
```

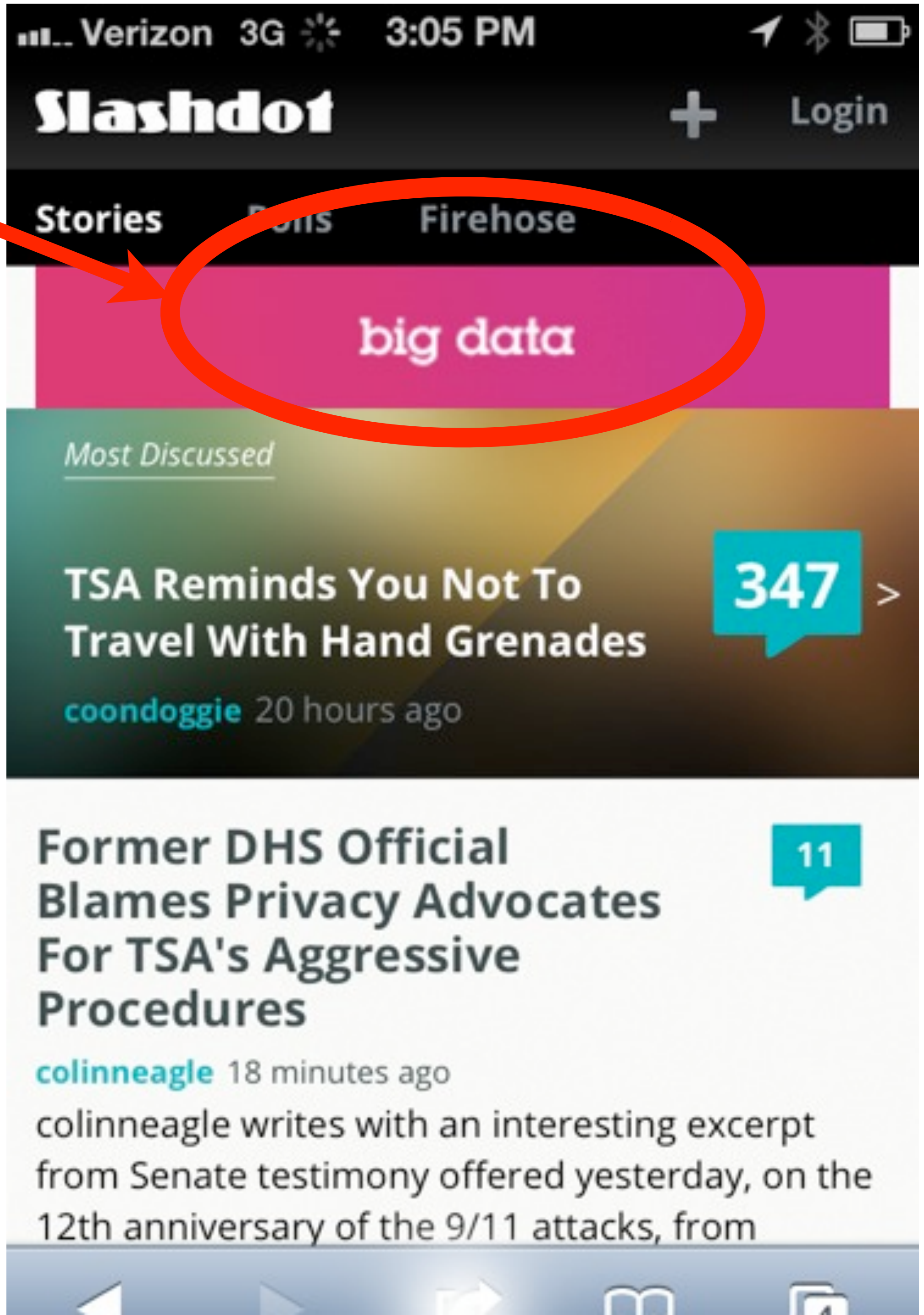
Part I: Big Data

(I think I’ve heard of that!)

Big Data

Big Data ads! Now on your phone!

- I often struggle in dealing with buzzwords.
- There is no doubt that Big Data is a buzzword and we're on the up-swing of the hype cycle!



Verizon 3G 3:05 PM

Slashdot

Login

Stories Polls Firehose

big data

Most Discussed

TSA Reminds You Not To Travel With Hand Grenades

347

coondoggie 20 hours ago

Former DHS Official Blames Privacy Advocates For TSA's Aggressive Procedures

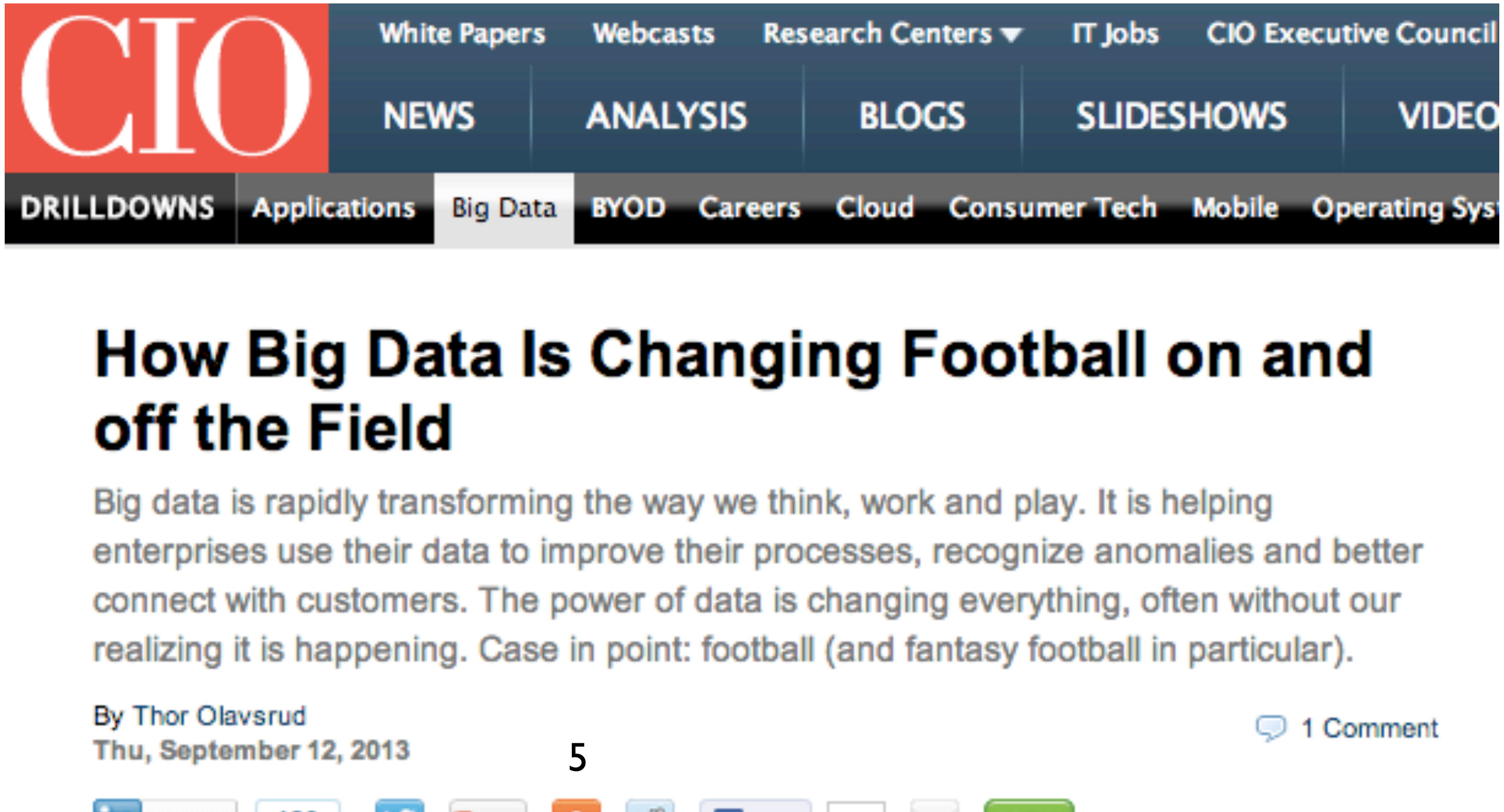
11

colinneagle 18 minutes ago

colinneagle writes with an interesting excerpt from Senate testimony offered yesterday, on the 12th anniversary of the 9/11 attacks, from

Big Data!

- Does anyone else get Facebook ads for Hadoop support?
- Does anyone want to hear football legend Jerry Rice's opinion about Big Data?



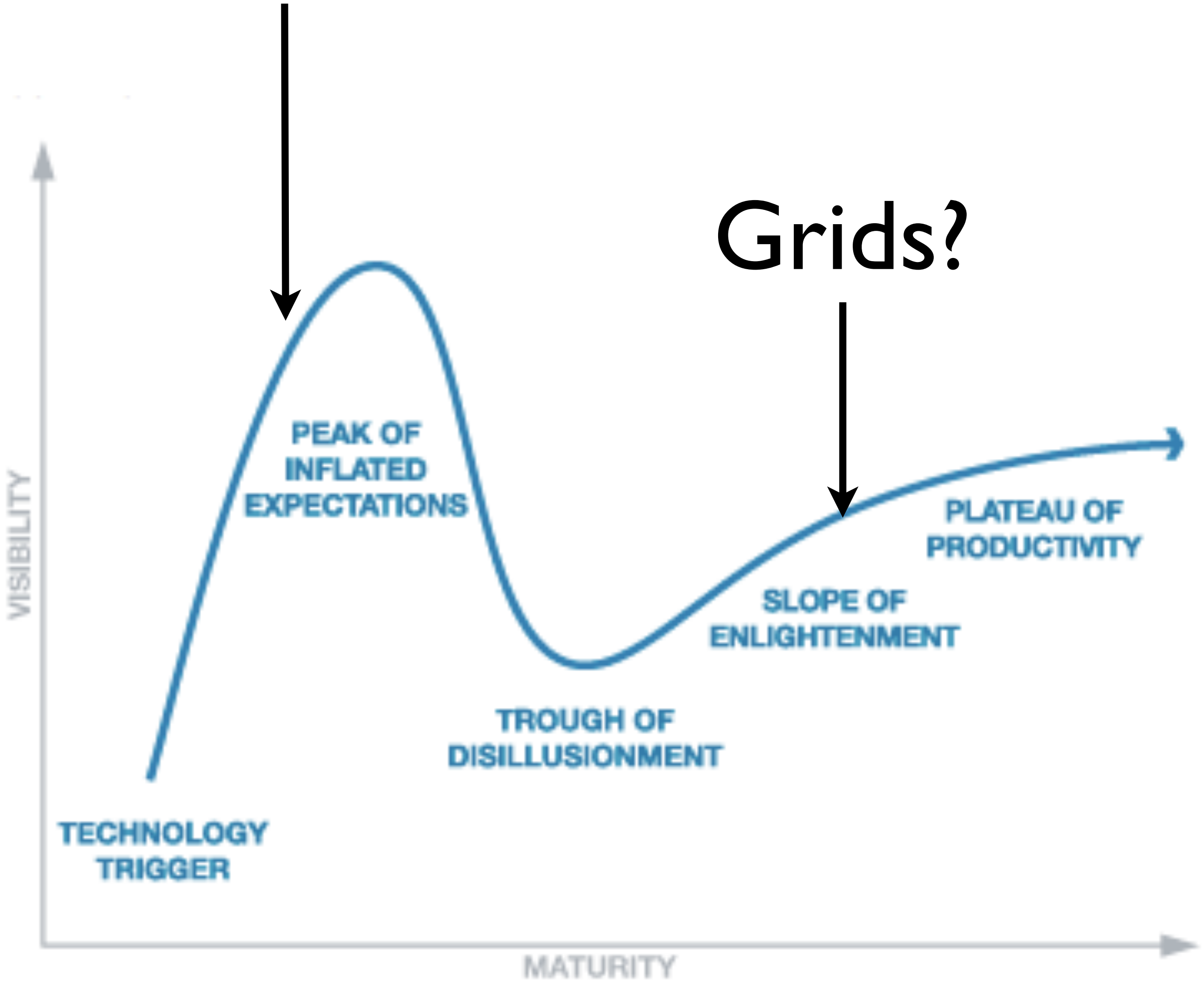
How Big Data Is Changing Football on and off the Field

Big data is rapidly transforming the way we think, work and play. It is helping enterprises use their data to improve their processes, recognize anomalies and better connect with customers. The power of data is changing everything, often without our realizing it is happening. Case in point: football (and fantasy football in particular).

By Thor Olavsrud
Thu, September 12, 2013

1 Comment

Gardner Hype Cycle



Big Data?

Grids?

What is Big Data?
Why would it benefit HEP?

- What is Big Data? I have come across a few formulations:
  - Lots-of-data.
  - “The Three V’s”.
  - A new approach to performing science.
  - A set of frameworks and tools for data analysis.

Big Data = Lots of Data (?)

- One way to evaluate whether you have “Big Data” is to stack your hard drives on a scale and weigh them.
  - If they total more than X kilograms, then you have Big Data.
- This view tends to be a tad self-serving:
  - After all, # of hard drives is a function of budget.
  - And it is called “Big Data”, not “Big Budget”!
- Let’s set aside discussions of petabytes and hard drives for journalists and hype-chasers!

Big Data = “The Three V’s” ?

- Another favorite breakdown of “Big Data” is the three V’s:
  - **Volume**: How many megabytes of data do you have?
  - **Velocity**: How quickly can you analyze your dataset?
  - **Variety**: How many different types of data exist?

HEP versus The Three V's

- Which of the three V's apply to HEP?
  - **Volume**: nearly indisputable. But we agreed to not discuss this!
  - **Velocity**: Like volume, somewhat - but this is a place where HEP can greatly improve.
  - **Variety**: "Variety" in terms of data types; almost everyone has a ROOT IO underpinning.

Big Data =
A new approach for science ?

- Another view of Big Data is applying high-throughput, data-based approaches to fields of science where this previously was not done.
- To first order, this means HEP is *not* Big Data; the field has always been high-throughput and data-based.
  - Let's tweak the definition: Big Data is about applying data analysis in new ways and at higher-throughput.
  - In other words, attempting to increase throughput by an order-of-magnitude. Certainly a relevant topic for HEP!

Big Data == Map Reduce?

- Another way to view **Big Data** is the use of new data processing approaches to existing such as MapReduce.
  - MapReduce provides a simple data processing framework; the premise of this (and other frameworks) is that if you cast your problem to fit a few simple assumptions, they will handle the scaling.
  - These frameworks try to balance the simplicity of use versus number of applicable problems.
- MapReduce - basically, functional(-like) programming - looks surprisingly a lot like HEP workflows.

Big Data

- Which view of **Big Data** is correct? I'm not a futurist and would .
- However, each view has a relevant question for HEP:
  - The three V's: **How do we increase the velocity of our analysis?**
  - New data science: **How do we enable data analysis in new places?**
  - New data platforms: **How do we reuse generic data processing platforms?**
- For my talk, I'd like you to keep these three questions in mind!

Part II:
Flexible Data

How do we increase the velocity of our analysis?

How do we enable data analysis in new places?

Flexible Data

- If Big Data isn't about # of petabytes but enabling new use cases, how do we achieve that?
- I believe the key opportunity of the future is *flexible data*.
  - Flexible data - **the ability to read and process data from anywhere in the world**.

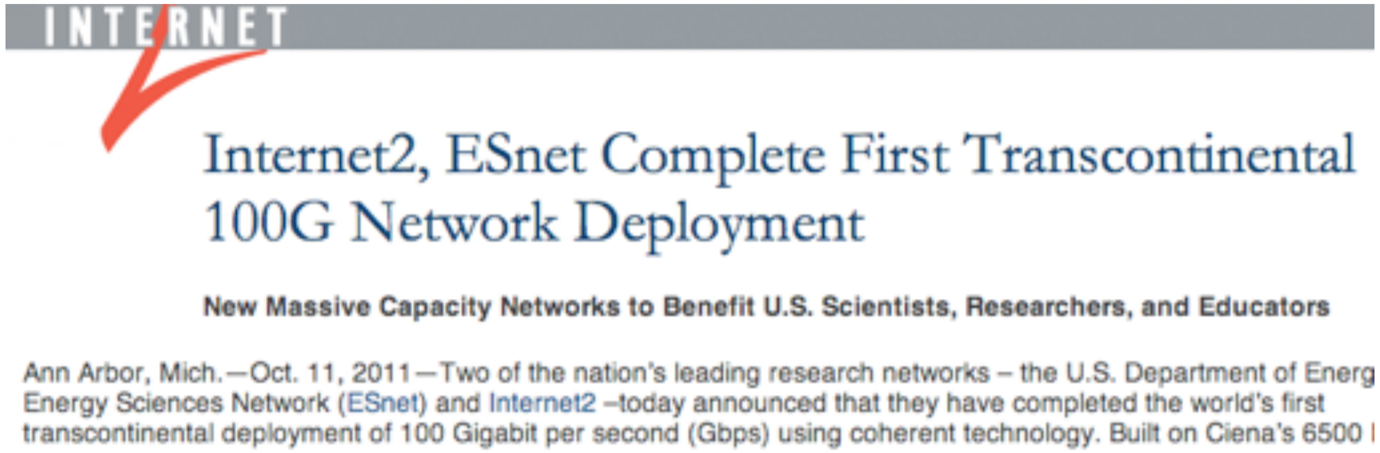
So, how do we get to the future?

Changes happening at sites - Network

2011

↓

2013

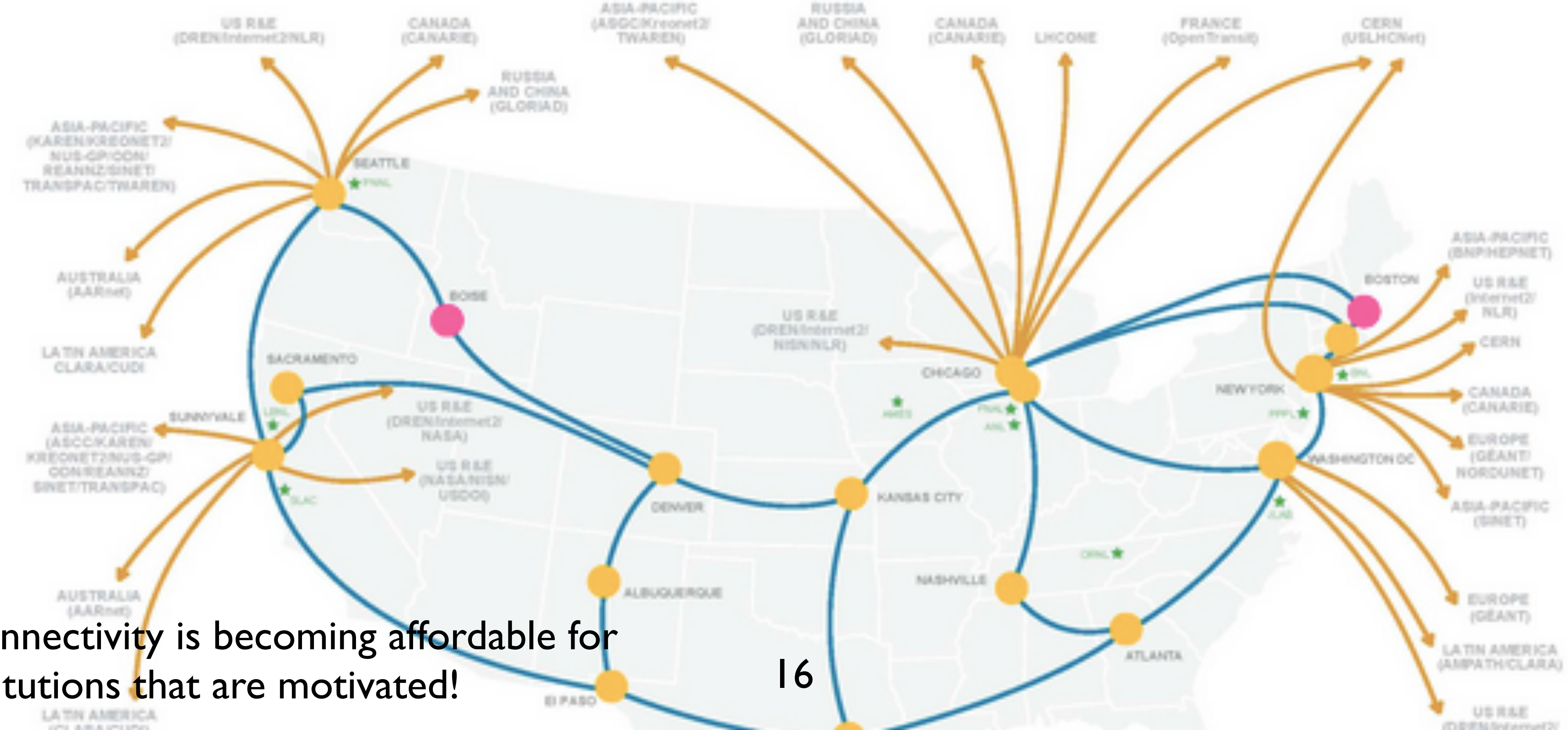


INTERNET

Internet2, ESnet Complete First Transcontinental 100G Network Deployment

New Massive Capacity Networks to Benefit U.S. Scientists, Researchers, and Educators

Ann Arbor, Mich.—Oct. 11, 2011—Two of the nation's leading research networks – the U.S. Department of Energ
Energy Sciences Network (ESnet) and Internet2 –today announced that they have completed the world's first
transcontinental deployment of 100 Gigabit per second (Gbps) using coherent technology. Built on Ciena's 6500 I


US R&E (DREN/Internet2/NLR)
CANADA (CANARIE)
RUSSIA AND CHINA (GLORIAD)
ASIA-PACIFIC (ASGC/Kreonet2/ TWAREN)
RUSSIA AND CHINA (GLORIAD)
CANADA (CANARIE)
LHCONE
FRANCE (OpenTransit)
CERN (USLHCNet)
ASIA-PACIFIC (KAREN/KREONET2/ NUS-GP/ODN/ REANNZ/SINET/ TRANSPAC/TWAREN)
SEATTLE
PNNL
AUSTRALIA (AARnet)
LATIN AMERICA CLARA/CUDI
BOISE
SACRAMENTO
LBNL
SUNNYVALE
ASIA-PACIFIC (ASCC/KAREN/ KREONET2/NUS-GP/ ODN/REANNZ/ SINET/TRANSPAC)
SLAC
US R&E (DREN/Internet2/ NASA)
US R&E (NASA/NISN/ USDOI)
DENVER
ALBUQUERQUE
AUSTRALIA (AARnet)
LATIN AMERICA (CLARA/CUDI)
EL PASO
US R&E (DREN/Internet2/ NISN/NLR)
CHICAGO
AMES
FNAL
ANL
KANSAS CITY
NASHVILLE
ORNL
ATLANTA
BOSTON
NEW YORK
BNL
PPPL
WASHINGTON DC
JLAB
ASIA-PACIFIC (BNP/HEPNET)
US R&E (Internet2/ NLR)
CERN
CANADA (CANARIE)
EUROPE (GÉANT/ NORDUNET)
ASIA-PACIFIC (SINET)
EUROPE (GÉANT)
LATIN AMERICA (AMPATH/CLARA)
US R&E (DREN/Internet2/

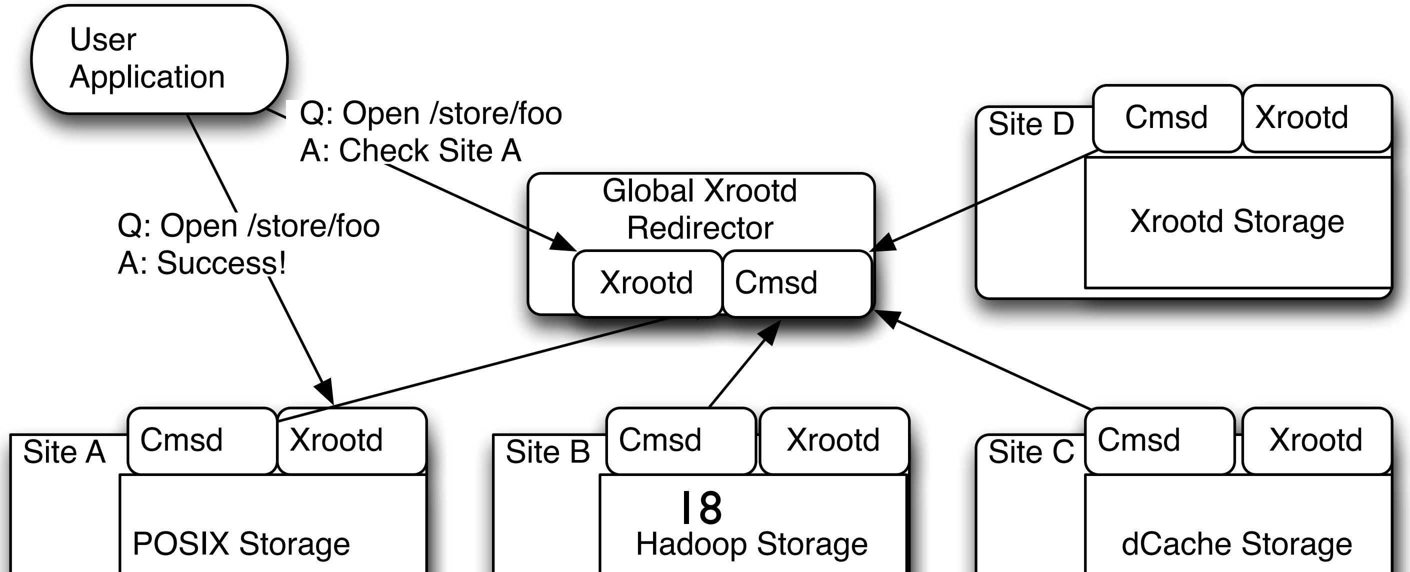
In the US, 100Gbps connectivity is becoming affordable for research institutions that are motivated!

Changes happening at the sites - Remote IO

- Maintaining reliable, scalable site storage is incredibly difficult. Run 1 of the LHC provided many sites & software with “trial by fire”.
    - Sites & software have matured greatly!
- The original WLCG design only allowed application access to storage from within the local site batch system. This allowed sites to tightly control the use of their storage system.
- Sites can now safely and reliably export data to external applications via HTTP(s) or Xrootd.

Data Federations

- In the last few years, we've seen the rise of *data federations*:
  - "A collection of disparate storage resources transparently accessible across a wide area via a common namespace."
- Combines remote I/O from sites with a "redirector". The redirector services client file-open requests; it discovers the locations of a requested file and redirects the client to a particular source.
  - Allows users and applications to *seamlessly* access experiment files without needing to worry about where the file came from.
- Examples I would highlight are the CMS AAA and ATLAS FAX infrastructures.



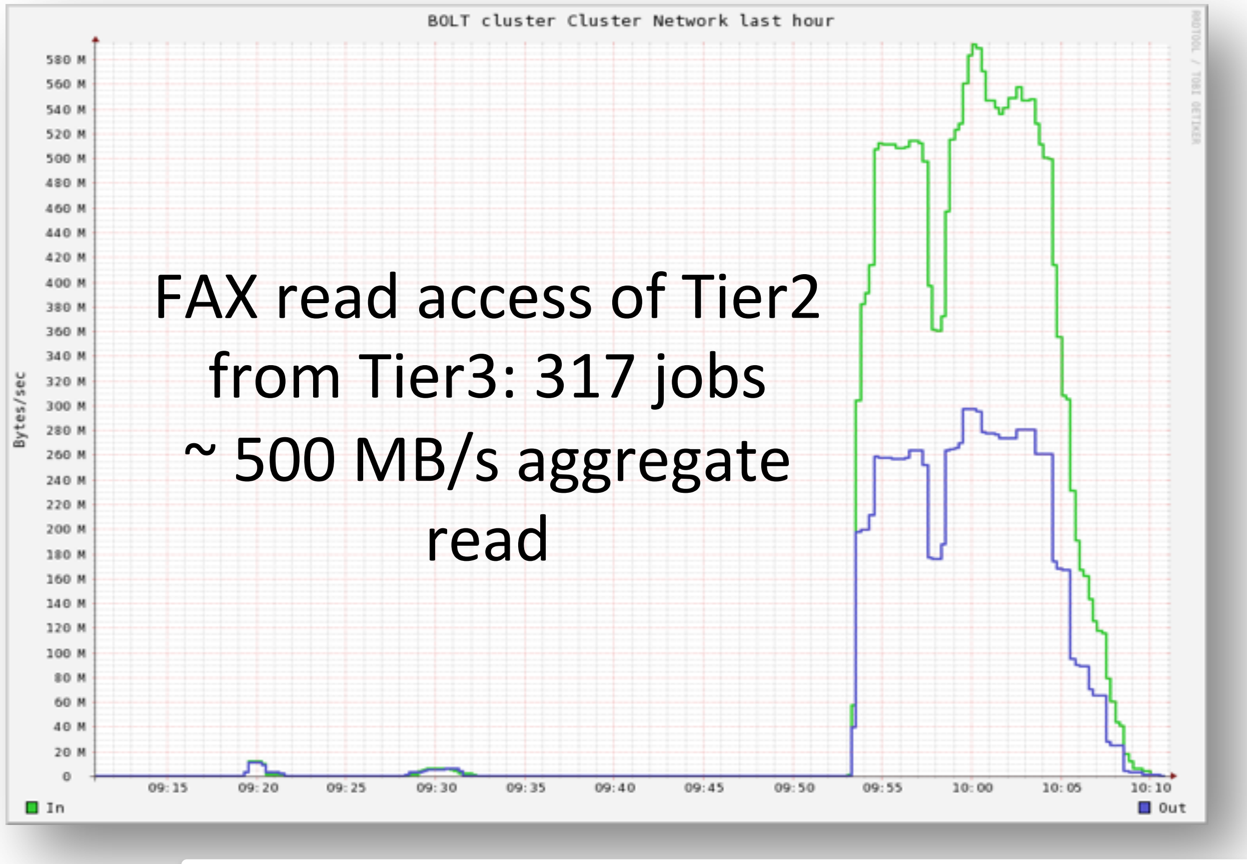
FAX

Slide Courtesy Rob Gardner

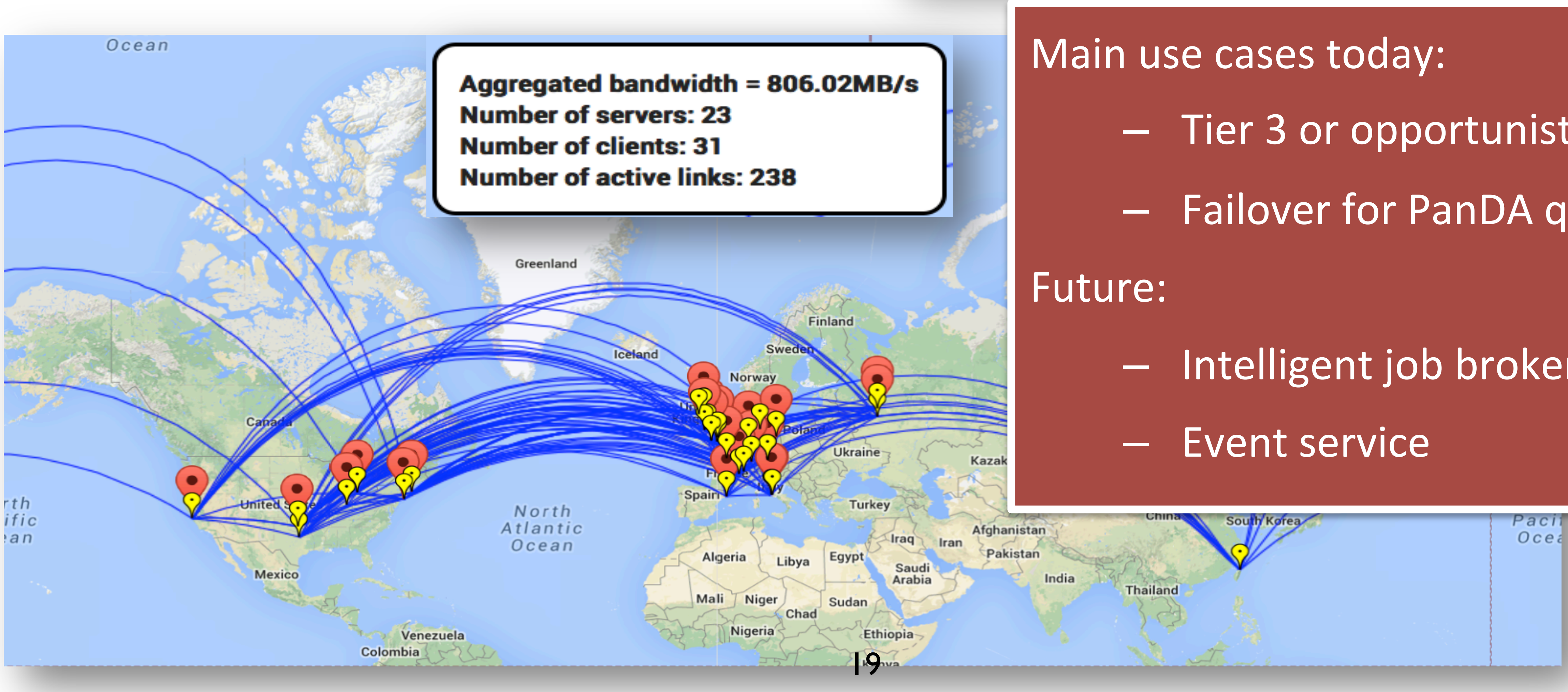
41 storage sites federated with Xrootd

Federating 177 PB of ATLAS data (of total 277 PB total)

Regions: USA, UK, DE, IT, RU, ES, FR + EOS



FAX read access of Tier2 from Tier3: 317 jobs
~ 500 MB/s aggregate read



Aggregated bandwidth = 806.02MB/s
Number of servers: 23
Number of clients: 31
Number of active links: 238

Main use cases today:

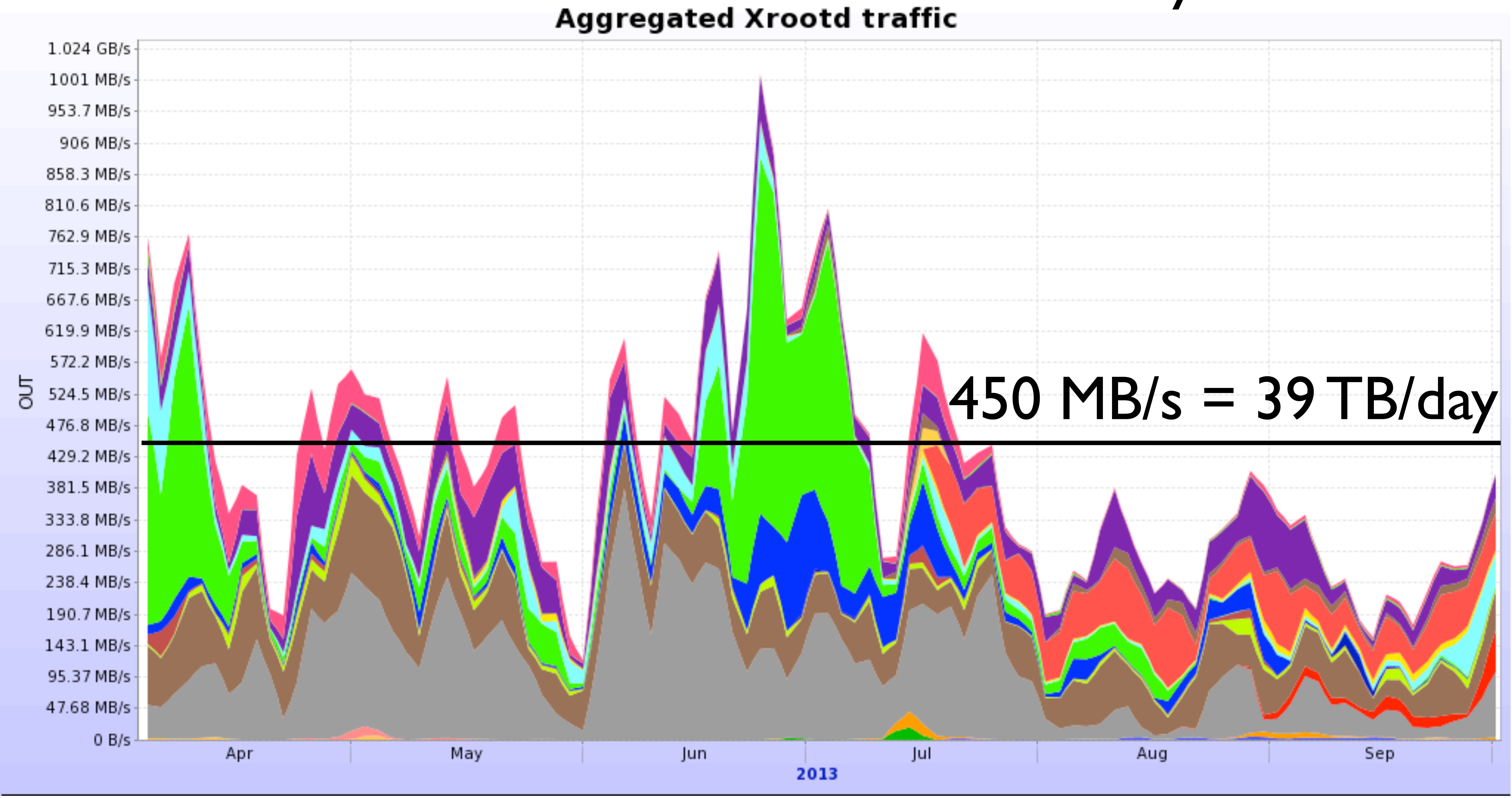
- Tier 3 or opportunistic CPU
- Failover for PanDA queues

Future:

- Intelligent job brokering
- Event service

Global picture

Slide Courtesy Ken Bloom



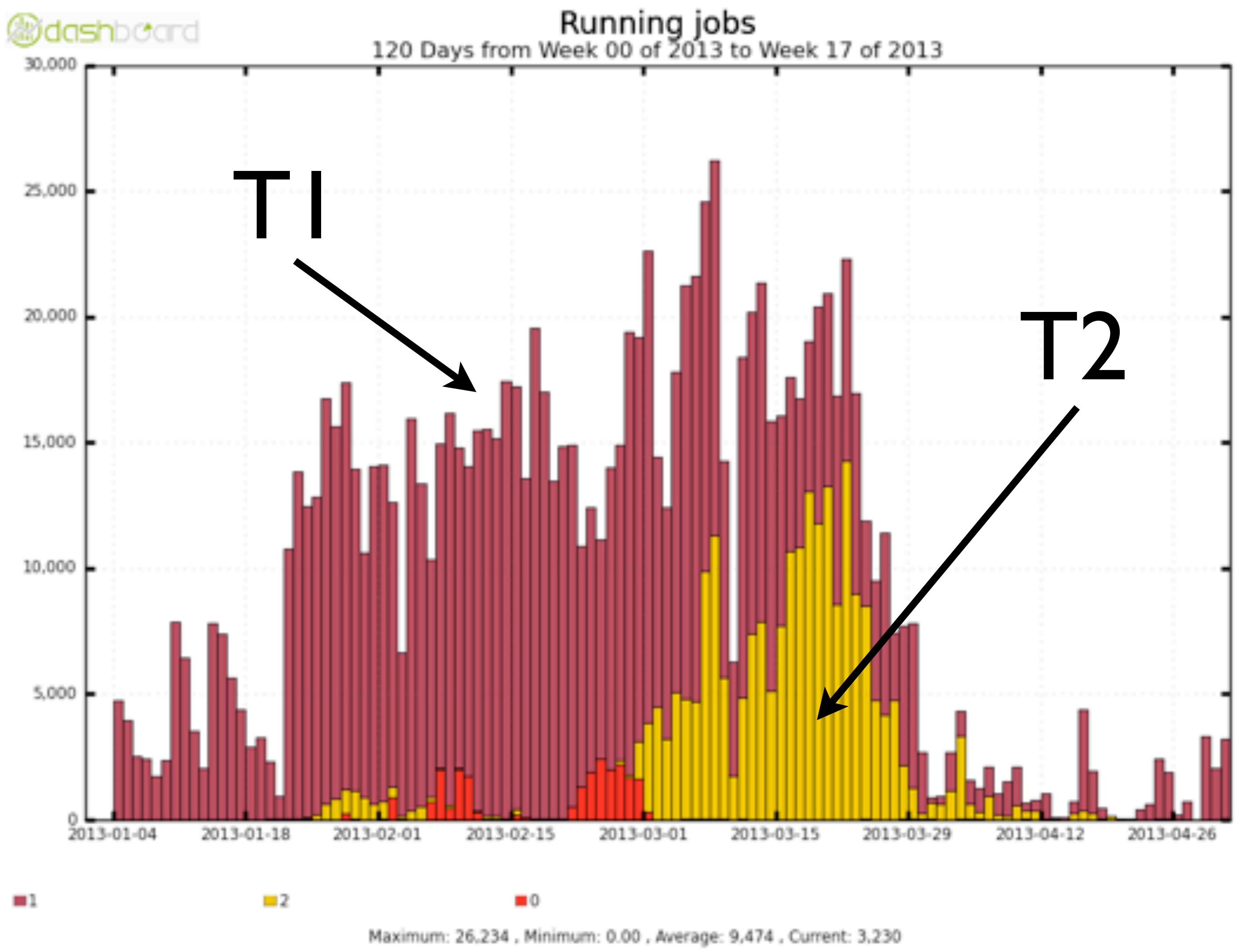
450 MB/s = 39 TB/day

For comparison: transfers via subscriptions = 81 TB/day

Applications: production with remote data

Slide Courtesy Ken Bloom

- “Legacy” reprocessing of 2012 data and associated simulation samples
- Inputs resident at T1 sites
- T1’s ran on data locally
- T2’s ran on simulations read via AAA
- Whole job done faster



Changes in the LHC Computing Models

- Starting in 2012, WLCG Tier-{0,1} sites have been splitting the disk pool and archive, reducing reliance on hierarchical storage management (HSM).
  - The disk pool and archive may still be part of the same system - but staging is not automatic.
  - This allows us to export data on disk to federations without worrying about staging random tapes.
- Remote IO-based workflows are starting to be integrated into more production workflows; this means VOs are depending on their data federations for essential work.

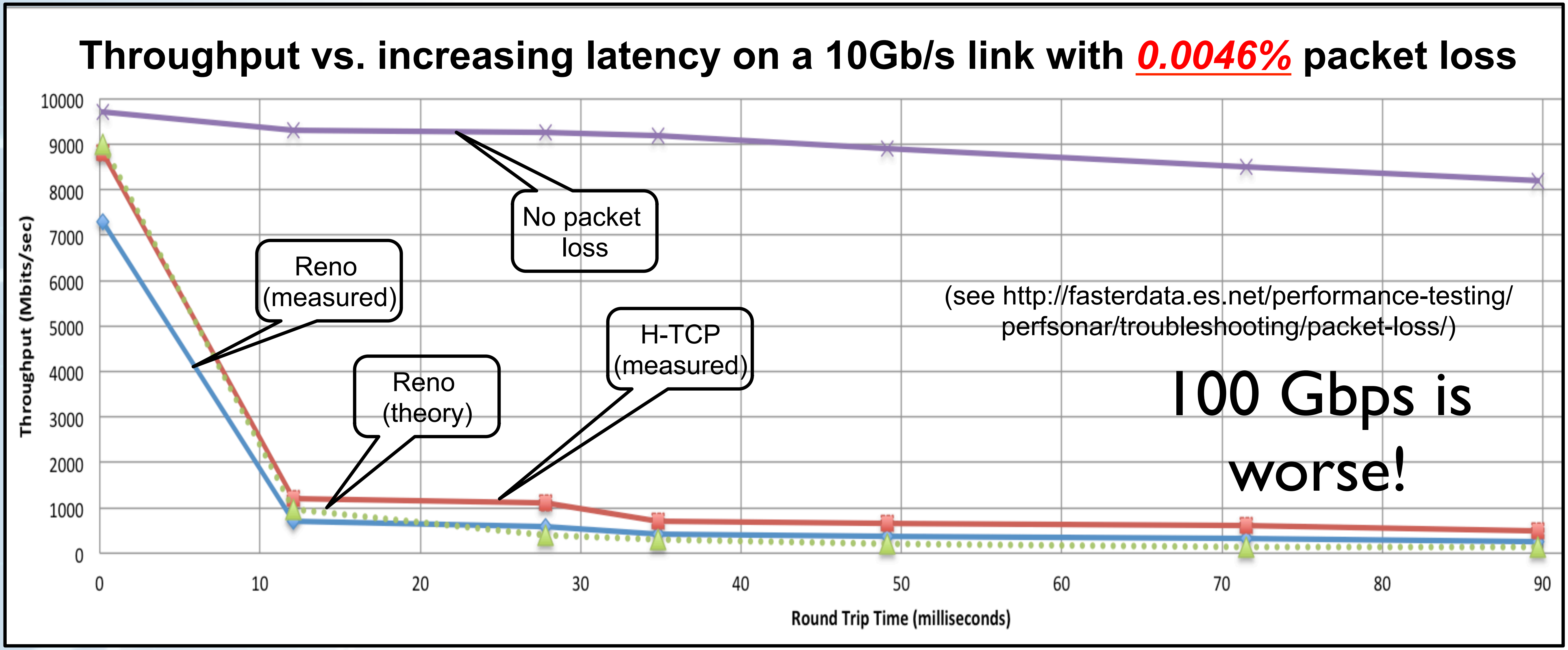
Inflexible Data

- If data flexibility is enabled by reliable networks, remote IO and data federations, what are the barriers to adoption?
  - Authentication and authorization.
  - Applications and frameworks.
  - Networks!

Network Inflexibility?!?

- Networks are the only entity listed as both helping and hindering flexible data. The culprit? Humans and TCP!
  - **TCP is a glass workhorse**. At high-bandwidth and high latency TCP is extraordinarily sensitive to networking problems.
  - Great networks don't live in isolation. For a given flow, one must consider all the pieces involved - endpoint hosts, campus networking, regional networking, and backbone networks. **The humans who run these networks must collaborate closely to fix problems**.
  - To achieve great TCP rates, all must work without a single error or misconfiguration. Error free end-to-end paths are not easily achieved.
    - All our network operators are great, but we place them in an impossible situation.
- Recent trends - such as performance monitoring (perfSonar) and Science DMZs - have made errors easier to spot and less likely to occur.
  - Yet TCP dictates we must have precisely zero errors!

A small amount of packet loss makes a huge difference in TCP performance



100 Gbps is worse!

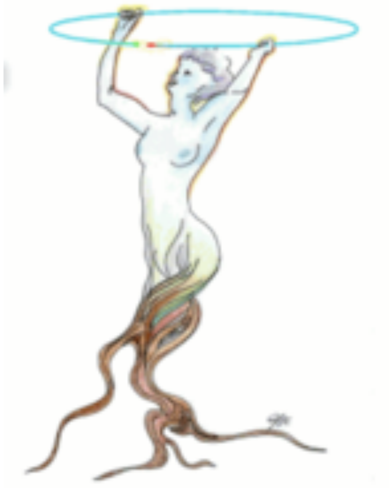
- On a 10 Gb/s LAN path the impact of low packet loss rates is minimal
- On a 10Gb/s WAN path the impact of low packet loss rates is enormous

- ***Implications***: error-free paths are essential for high-volume data transfers

Slide courtesy of Jason Zurawski

Auth{z,n} Limits

- The GSI / VOMS ("grid certificates") architecture provides the WLCG with an unparalleled global SSO infrastructure.
- It is the most successful infrastructure that everyone hates to use / setup.
  - If we want to be serious about remote IO, we must be serious about usability - down to the user's laptop!
  - We have been lucky so far because WLCG users accept grid certificates as a fact of life. Not every experiment has such crazy people!



Applications and Frameworks

- In HEP, we basically standardize on the ROOT IO format. There are many upsides. Let’s discuss the downsides!
- This format allows us to serialize any C++ object we want - a horrible idea!
  - That is, it will dutifully serialize any sloppy user data structure. The user rarely sees the performance penalty.
  - The HEP community forced the ROOT team to do this and they weren’t able to say no. **Big mistake for the community**!
- ROOT allows remote access through a zoo of protocols, but remote access is only a piece of the puzzle.

Flexible ROOT

- CMS has invested effort in analyzing ROOT's IO performance. We have a stress test meant to highlight the worst case (which is, unfortunately, a reasonable analysis pattern); tests are done from Nebraska servers to a CERN client.
    - The ROOT defaults perform poorly over high-latency links; in CMS's measurements, the defaults are **177x** slower than our best optimizations.
    - Even using ROOT's solution - TTreeCache - is about **20x** slower than CMS's optimizations over high-latency links.
    - The full set of CMS optimizations is still slower than just downloading the file and running on it.
- Lesson: **Remote IO != flexible data**. If your experiment doesn't invest time in its applications and framework, *you won't get flexible data*.

Part III:
Small Data

How do we enable data analysis in new places?

How do we reuse generic data processing platforms?

Rough waters ahead!

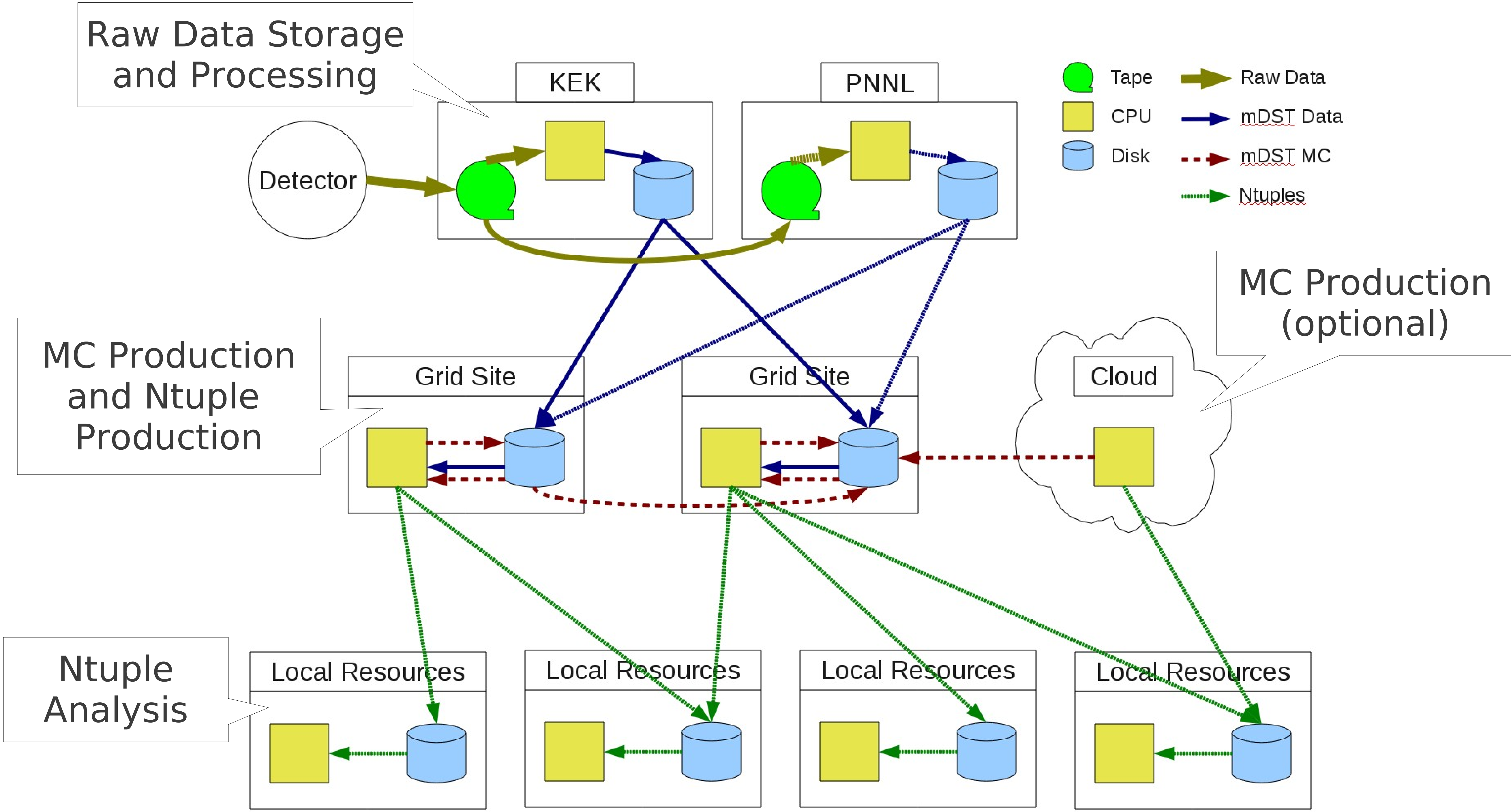
Think Smaller!

- The WLCG experiments historically have had the benefit of being Big Science.
  - **Large** development teams.
  - **Large** operational teams.
  - **Large** number of sites utilized.
- I suspect the *average* HEP experiment has a computing organization that can comfortably fit around a single lunch table.
  - There are few WLCG services that can survive such an environment. *Small* services are much harder than *big* services!
  - There is future in *small data* - data processing enabled by minimal investments in computing infrastructure.

How can we think small?

- Projecting from LHC trends, it is likely future HEP computing sites will utilize these building blocks:
    - A single archive service with excellent network connectivity.
    - Disk service(s) with excellent network connectivity.
    - Analysis & simulation computing services with excellent network connectivity.
    - Simulation computing services with good network connectivity.
- Utilizes a grid-level sources to present the experiment's computing resource as a single batch system (think **PanDA** or **glideinWMS** or DIRAC or Alien2).

Computing Model (Belle II)



Raw Data Storage and Processing

MC Production and Ntuple Production

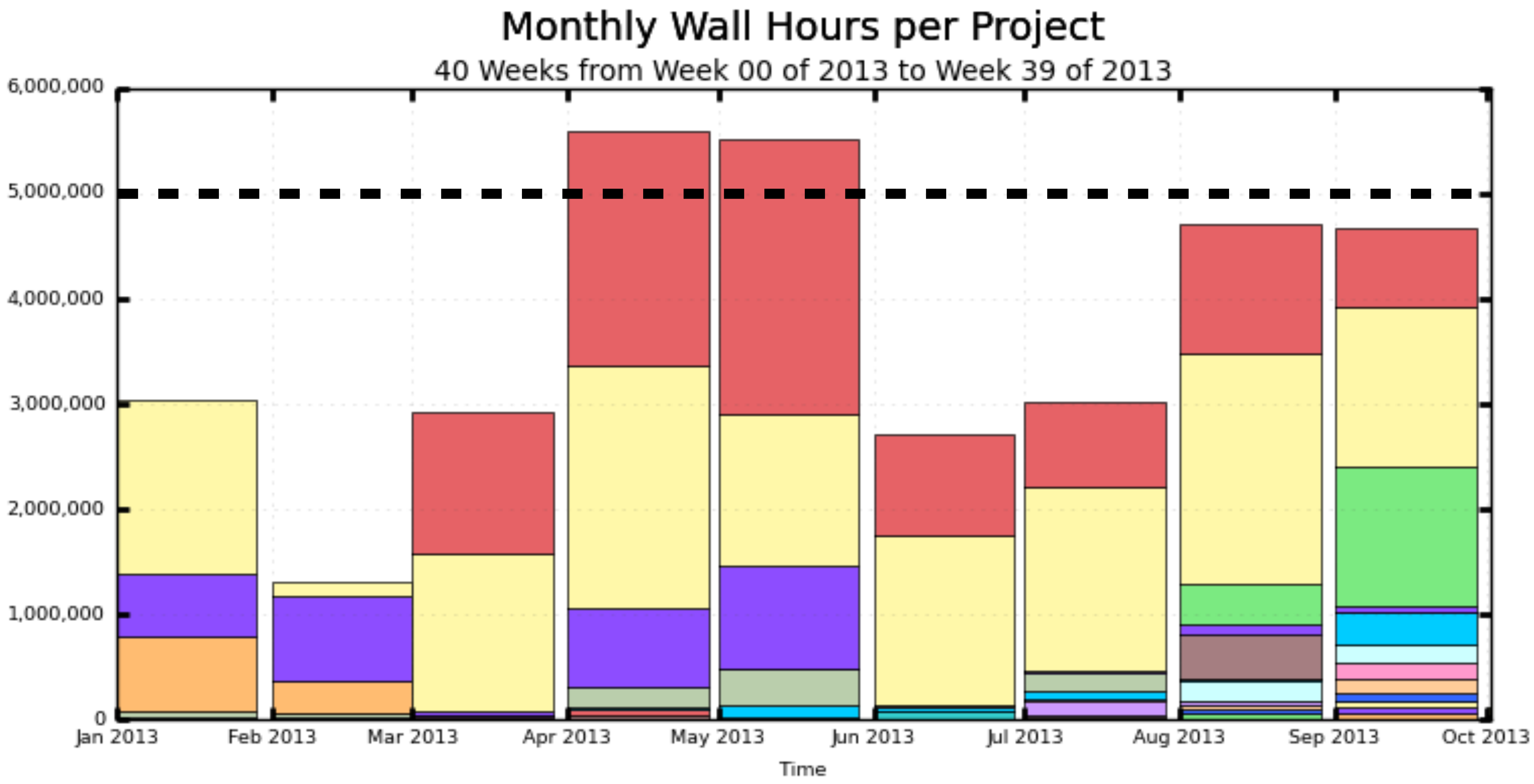
Ntuple Analysis

MC Production (optional)

Platform-as-a-service

- We've historically expected grid infrastructures like OSG to offer services like software packaging, ticketing, registration, and resource discovery.
  - The VO is expected to provide the rest.
- In the last two years, OSG has begun offering VOs and users a service (glideinWMS) which constructs a HTCondor pool on demand.
  - This is a **batch-system-as-a-service**.
  - Resources are provisioned from owned and opportunistic resources; in the near future, we plan to add the ability to use allocation-based and virtualized resources.
- Another recent, less-mature PaaS offering from OSG is a hosted CVMFS service.
  - This allows VOs to harness the power of CVMFS to distribute software without .
- These new services greatly lower the bar of grid adoption.

Small Beans



7,000 cores fully utilized

Small organizations using OSG's HTCondor opportunistic service. Each one has valid science, yet is small compared to a WLCG VO. For comparison, the most active WLCG VO averaged 68M hours / month; the least active averaged 14M hrs / month during 2013.

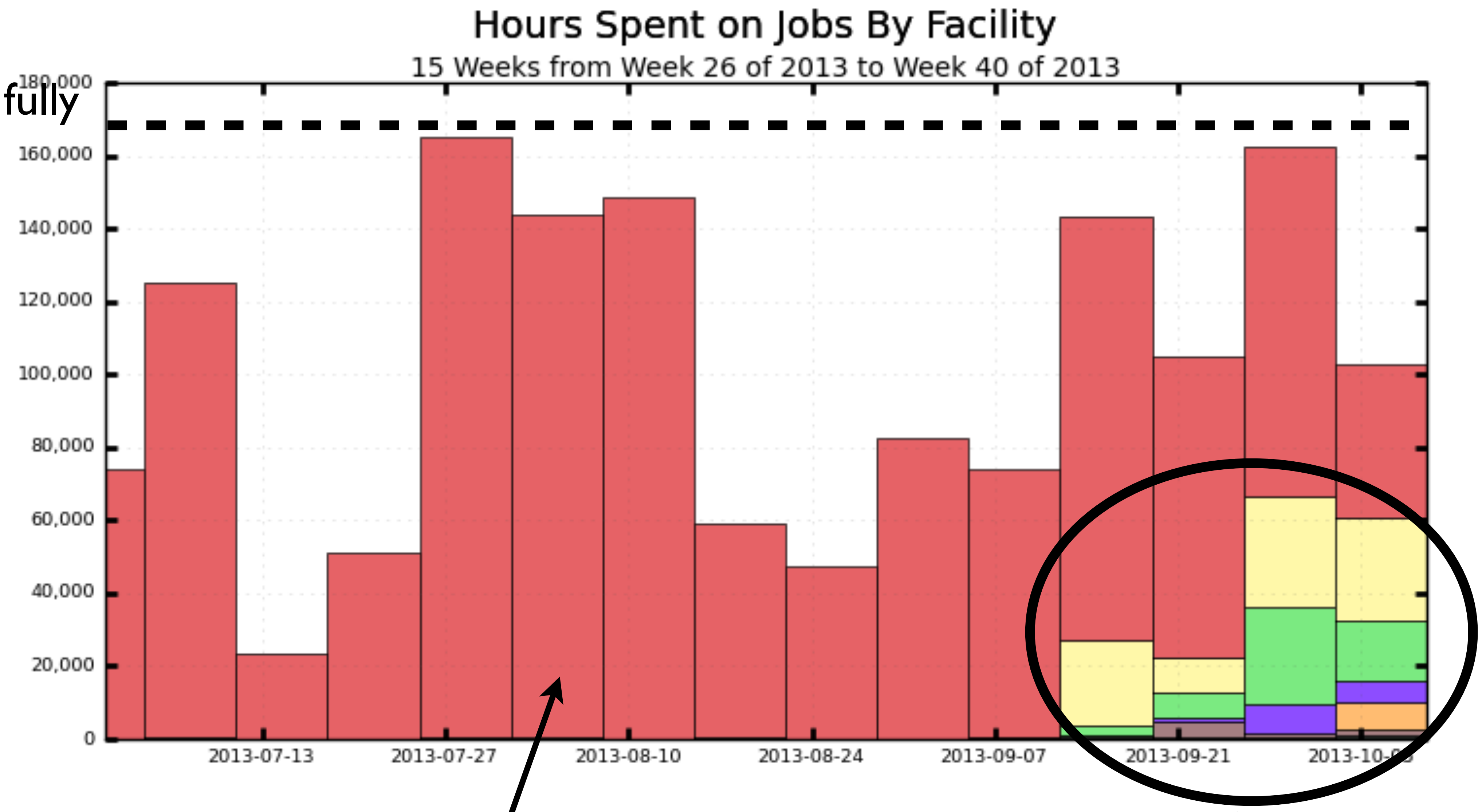
Science Impact does not correlate well with CPU hours!

Thinking Small

- OSG’s CVMFS and HTCondor services provide cleaner “on ramps” for new experiments than we previously could offer.
    - Experiments can start small with HTCondor as the local site batch system, then extend to the grid.
    - Similarly, CVMFS requires little expertise to use.

Recent Example - NoVA

1,000 cores fully utilized



Fermilab

non-Fermilab

Generic Data Processing Platforms

- Why must HEP live on its own “Big Data Island”?
- Why isn’t a “Big Data” generic data platform (Hadoop / NoSQL) utilized more for HEP?
  - On paper, Hadoop MapReduce is better than our custom data processing platforms. Certainly, an all-in-one platform would be amenable to “**small data**”.
  - Extreme-scale databases almost certainly would be faster and better organized than our mishmash of file systems, catalogs, and custom IO layers. Computer Scientists have been telling us this for years!
  - Is HEP simply too lazy to adopt? Has too much baggage? Perhaps...

Generic Data Processing Platforms

- However, I believe the problem is a matter of portability and flexibility. Take Hadoop as an example:
  - **Portability**: For better-or-worse, the "API" exposed by research computing facilities is batch-system & file systems. Running Hadoop MapReduce through a batch system has not yet matured.
  - **Flexibility**: Hadoop has made several design decisions which reduce its performance for running across multiple datacenters.
- The pieces have been demonstrated individually; work has even been done to make ROOT run within MapReduce. However, it seems no one has made it work "end-to-end", in production, and viable for small experiments.
- Portability and flexibility issues tend to be even worse for databases.

Closer than ever before -

but not quite there!

Part IV: Conclusions

What can we learn from Big Data?

Beware Computer Scientists bearing gifts!

Big HEP Data

- HEP, as a field, needs to rise above comparisons of CPU hours, petabytes, and gigabytes per second.
  - These are not Big Data! They are a function of big budget - *easy come, easy go*! - and are intellectually cheap.
- Big Data is about accelerating science:
  - Order-magnitude increases in processing capacity.
  - Order-magnitude decreases in time-to-science.
  - Order-magnitude decreases in dedicated computing personnel.

Absolute size is not as important as growth!

Big HEP Data

- If we want "Big Data" for HEP, we believer there is the most promise in *flexible data* and *small data*.
- The CMS and ATLAS experience with flexible data .
- One potential future area of focus is doing "Big Data" for small HEP experiments.
  - This is not an invitation to start a zoo of "common projects"; rather, an invitation to "creative destruction". How many services can an experiment live without? How many existing common services (**networks**, auth'n, pilot services) can we emphasize?
- The lesson from other fields is about *flexible frameworks* for data analysis.

Looking to the Future

- Where would I invest for HEP's future in Small, Flexible Data?
  - The field doesn't have a sterling history of **sharing** computing software infrastructure (better than most!).
- However, ROOT has become a de-facto standard. I believe the following are achievable:
  - Order-magnitude increase in I/O throughput.
  - Switch ROOT I/O to multithreaded / asynchronous interface.
  - Two-order-magnitude improvement of **default configuration** on high latency links.
- This would unlock flexible data for all. This would be a "*moonshot*" for HEP - an immense impact, but requires far more investment in ROOT IO than currently done.

Questions? Comments? Heckling?