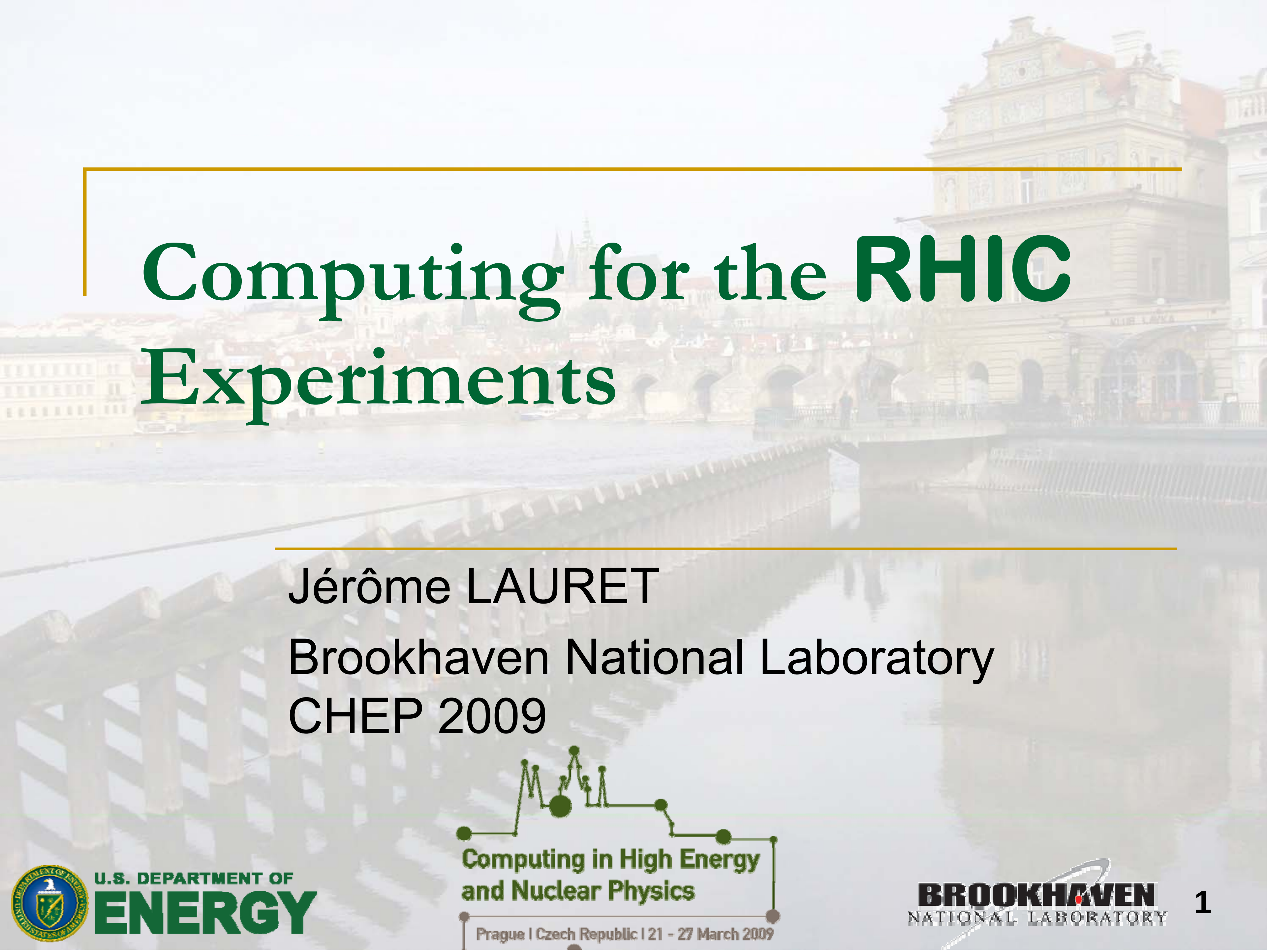


Computing for the **RHIC** Experiments

Jérôme LAURET

Brookhaven National Laboratory

CHEP 2009

Computing in High Energy and Nuclear Physics

Prague | Czech Republic | 21 - 27 March 2009

U.S. DEPARTMENT OF ENERGY

BROOKHAVEN NATIONAL LABORATORY

Outline

sw evolution, upgrades, lessons learned

Carla Vale, Flemming Videbaek, Peter Steinberg, Martin Purschke, Chris Pinkenburg, Thomas Ullrich, Micheal Ernst, …

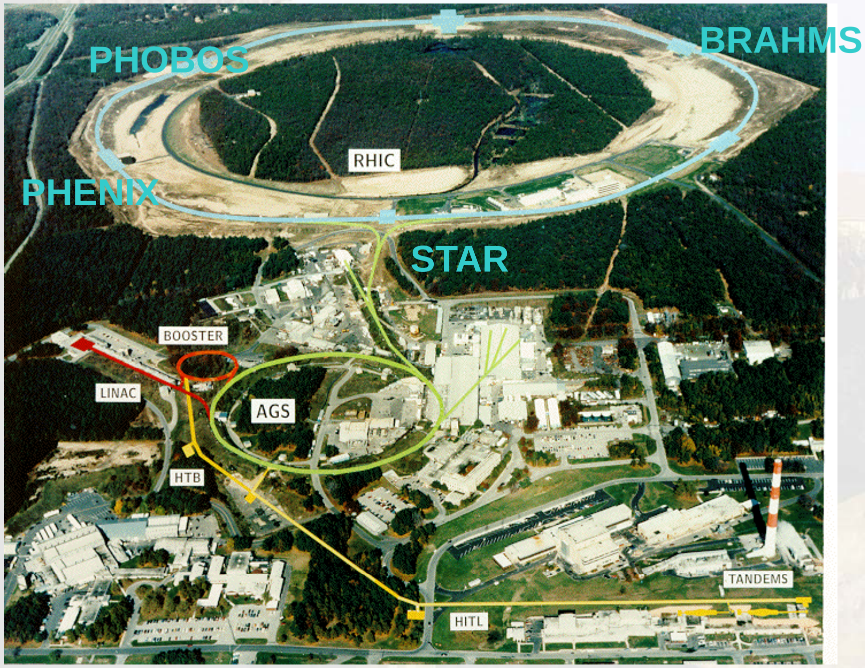
upgrades, concept and approach evolution & lessons learned (what worked)

- Introduction to RHIC & experiments
  - DAQ rates, data growth and CPU analysis
- How did we make it?
  - Practical choices (frameworks, output formats evolution)
  - Resource saving “toolkit”
- Distributed computing and Grids
- Conclusions

The Relativistic Heavy Ion Collider (RHIC) complex & program

Scientific program in Heavy-Ion and Spin program

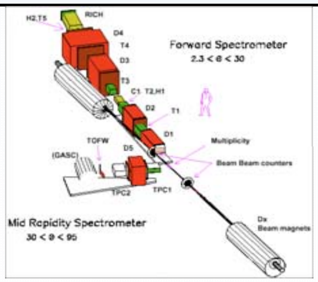
- Heavy Ion: QGP
  - provide unique insight into how quark and gluons behaved collectively at the very first moment our universe was born.
  - Critical temperature Tc $\approx 2.10^{12}$ K
    - The sun core is ~ 10^{7} K
    - Tc ⇔ 170 MeV
- Spin program
  - understanding how mass and spin combine into building blocks of nature
- **Versatile machine- Flexibility is key to understanding complicated problems**
  - *Polarized* protons sqrt(s_{NN}) = 50-500 GeV
  - Nuclei from d to Au (U), sqrt(s_{NN}) = 20-200 GeV



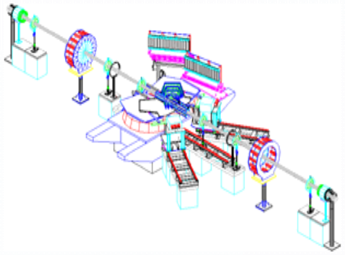
The experiments

- RHIC = 4 experiments
- Two small phased-out in 2006: **BRAHMS** (particle production over large rapidity range) & **PHOBOS** (4π multiplicity and correlations)
  - ~ 50 participants
  - very un-distributed geographically
- Two large experiments: **PHENIX** (tracking, electromagnetic probes near mid-rapidity) & **STAR** (precision global tracking and calorimetry over large acceptance)
  - participants 500+
  - Distribution: 12 countries, 50+ institutions

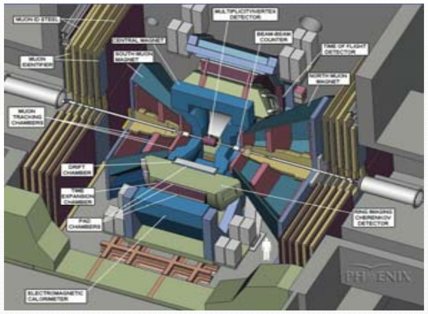
BRAHMS



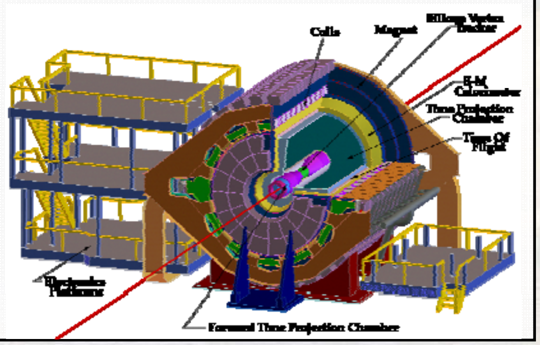
PHOBOS



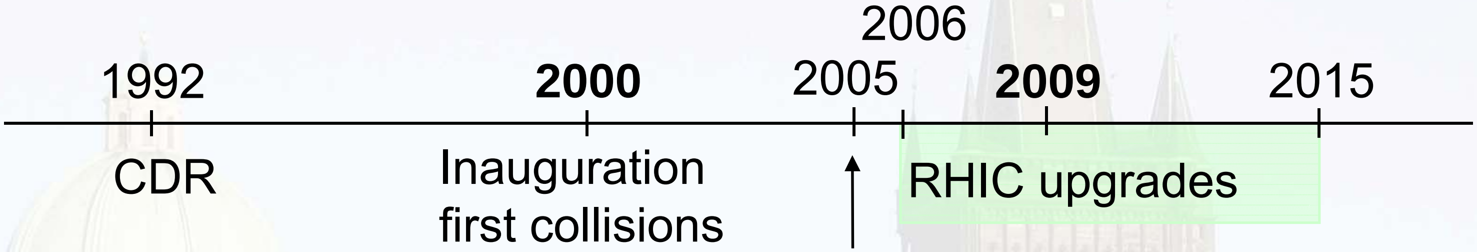
PHENIX



STAR



Status & upgrades


1992

CDR

2000

Inauguration
first collisions

2005

2006

2009

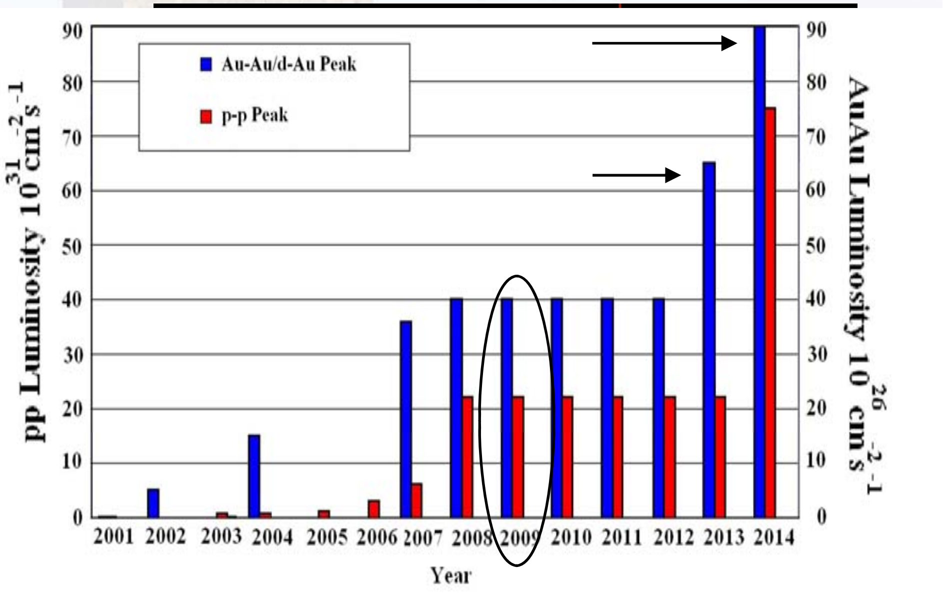
2015

RHIC upgrades

Announce of the Perfect Liquid (DOE, BNL). Full story video available.

PHENIX and STAR

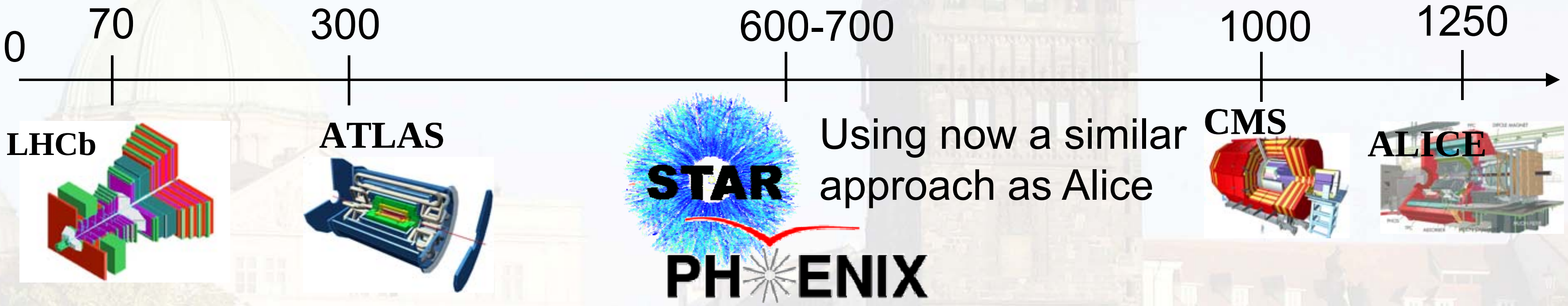
- Detector upgrades to address the more complex Physics
  - Heavy Flavor, Silicon Vertex, Reaction plane, forward Physics
- Machine / luminosity upgrade
  - First installment in 2007
- DAQ upgrades:
  - Early high rate for PHENIX
  - Staged DAQ upgrade for STAR (x100 in 2004, x1000 in 2008)



DAQ rates – the perspective and the consequence

Niko Neufeld, CHEP 2009
Martin Purschke, CHEP 2004

Number in MB/sec



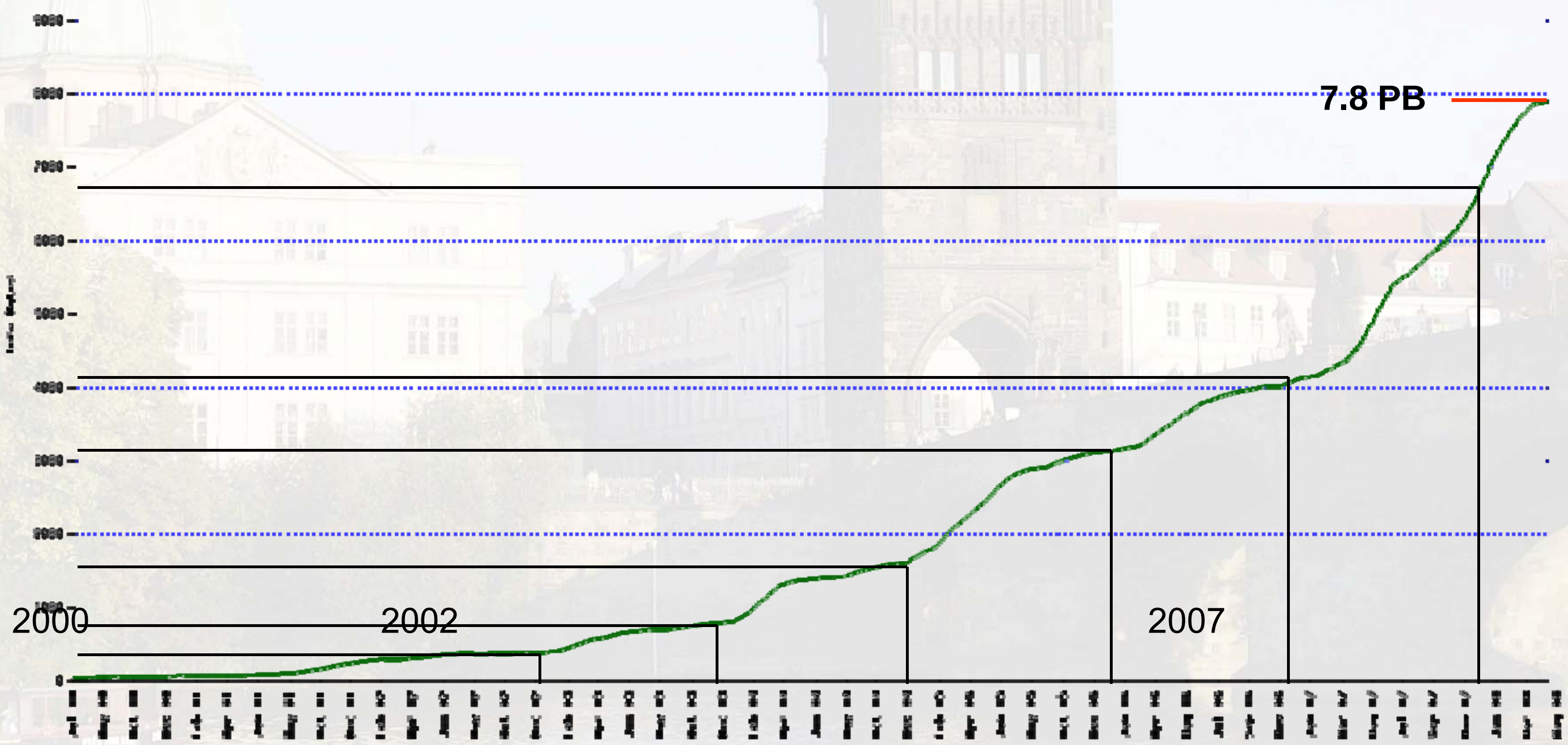
Using now a similar approach as Alice

- Data size
  - RHIC (from p+p to Au+Au) within LHC’s range (p+p or Pb+Pb)
  - PHENIX – Largest datasets
  - STAR sustained 400-600 MB/sec
  - STAR current program requires 250-300 MB/sec (22 weeks) in 2012

PB/year raw data sample recorded on tape
Will double (luminosity) in the coming years

Data so far, …

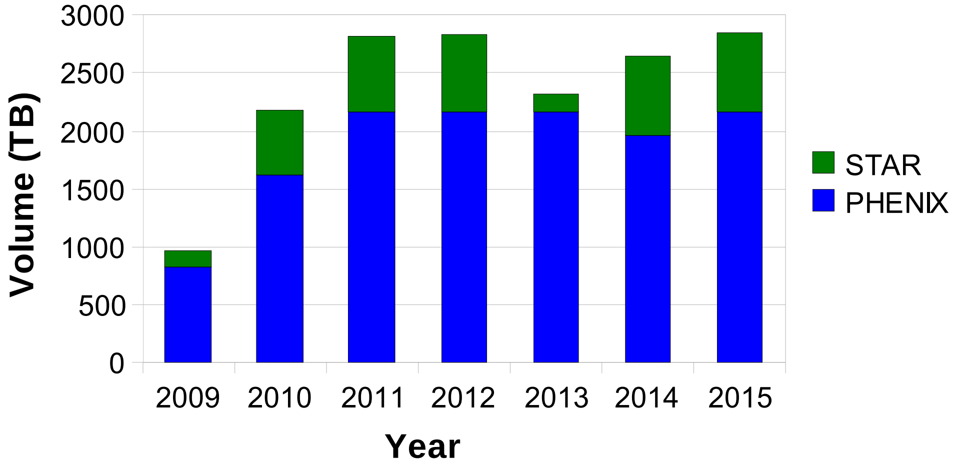
Data Volume archived at the RACF (managed by HPSS)



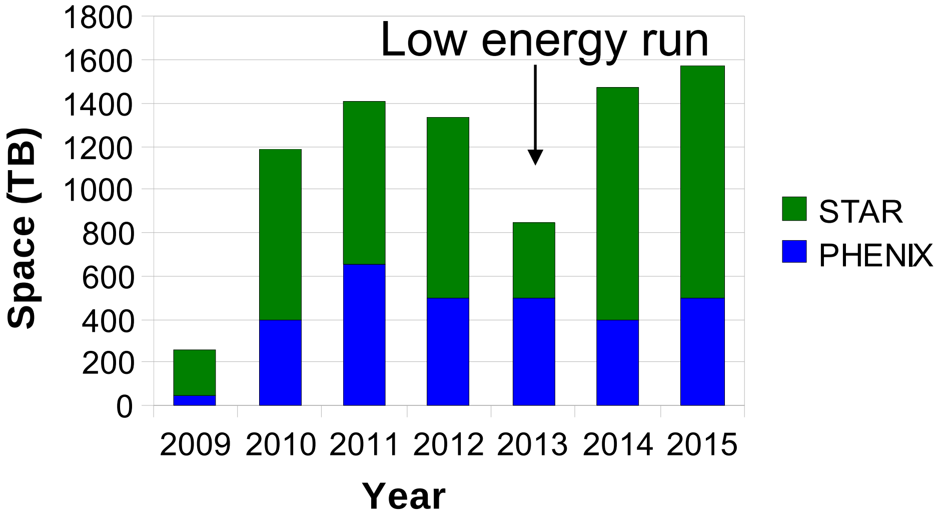
Data growth outlook

- Initial model:
  - fraction of data from previous years on disk and/or analyzed?
  - **WRONG!**
- RHIC Experience:
  - **nearly all data from all years are being constantly analyzed, cross-compared, merged (analysis)**
- ~ ½ of the cost in storage (tracked 2005-2008)

Estimated raw data volume (TB) for PHENIX and STAR exp.



Derived data volume (TB)



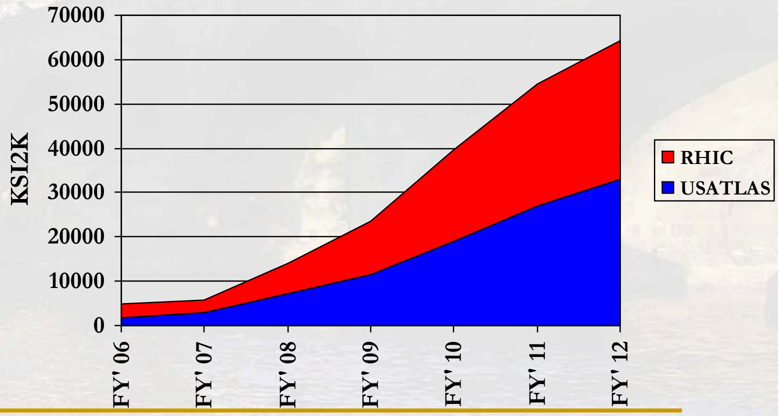
Facility and relation to experiment

- **BNL/Tier0 Facility – RACF**
  - Mission: *Online Recording of Raw Data, Production reconstruction of Raw Data, Primary Facility for Data Selection and Analysis, Long time Archiving and Serving of all Data*
  - Share, leverage, consolidate, focus on robust solutions
  - Maximize CPU cycles – Shared (queues) if not used (cross experiments, EOL)

- Procurement cycles
  - Base funding for equipment shared by the experiments and the facility
  - Cycle: 5 years plans, long term projections
  - Common pool for facility + experiments

- Issue
  - **Facility + experiment shared pool of money Zero sum principle ⇔ balance**
  - Storage: tape is a fixed cost
  - CPU needed for processing
    - **# of passes @ RHIC have been low (<< 3)**
    - **Implied Out-sourcing from the start**

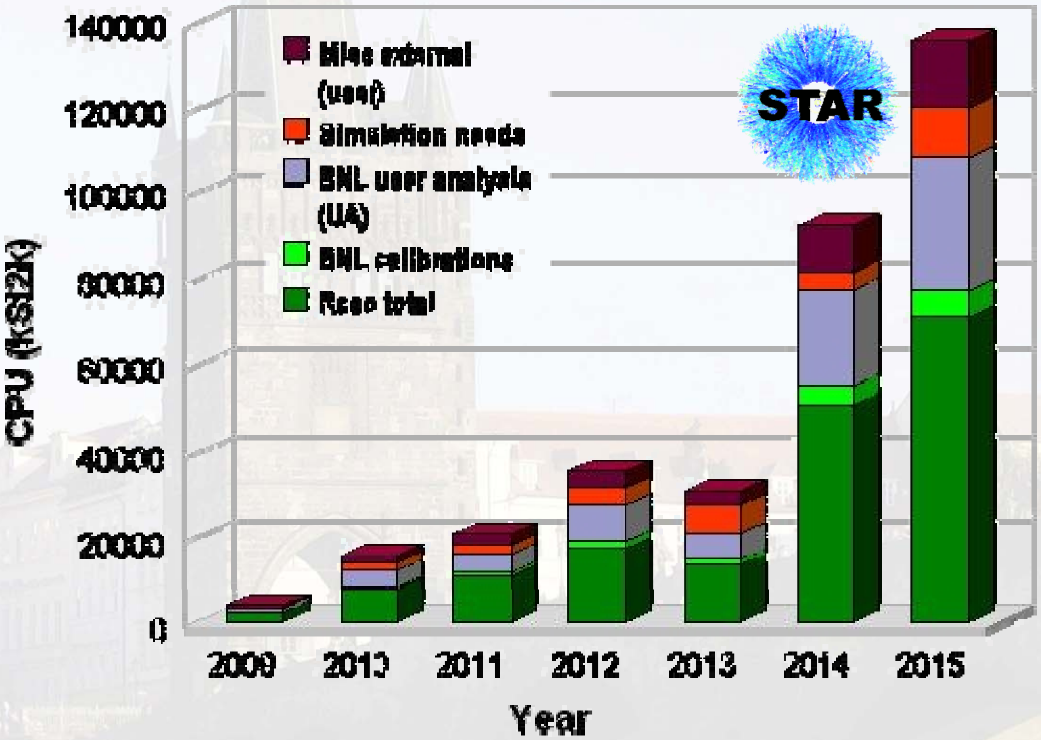
Processing Power - 65 MSI2k in 2012. Numbers may change with RHIC revised plan



CPU overview

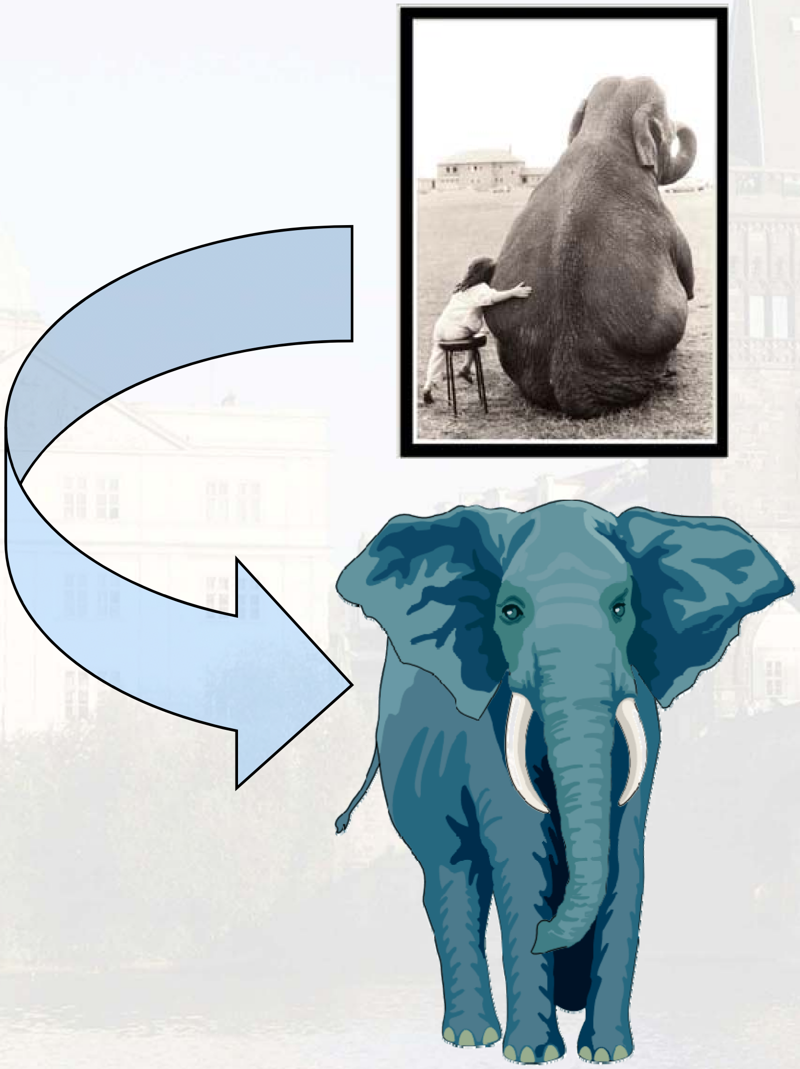
- Comparing to one pass data production
  - Event generator / simulation ~ +10%
  - Embedding ~ +15%
  - One pass analysis = one reconstruction
  - Calibration:
    - At least +20%
    - New detector +30%
- Shortfalls
  - Tier0 – no simulation support
  - User analysis estimated +150-200%
  - Additional production passes
  - One pass done in “humanly” acceptable

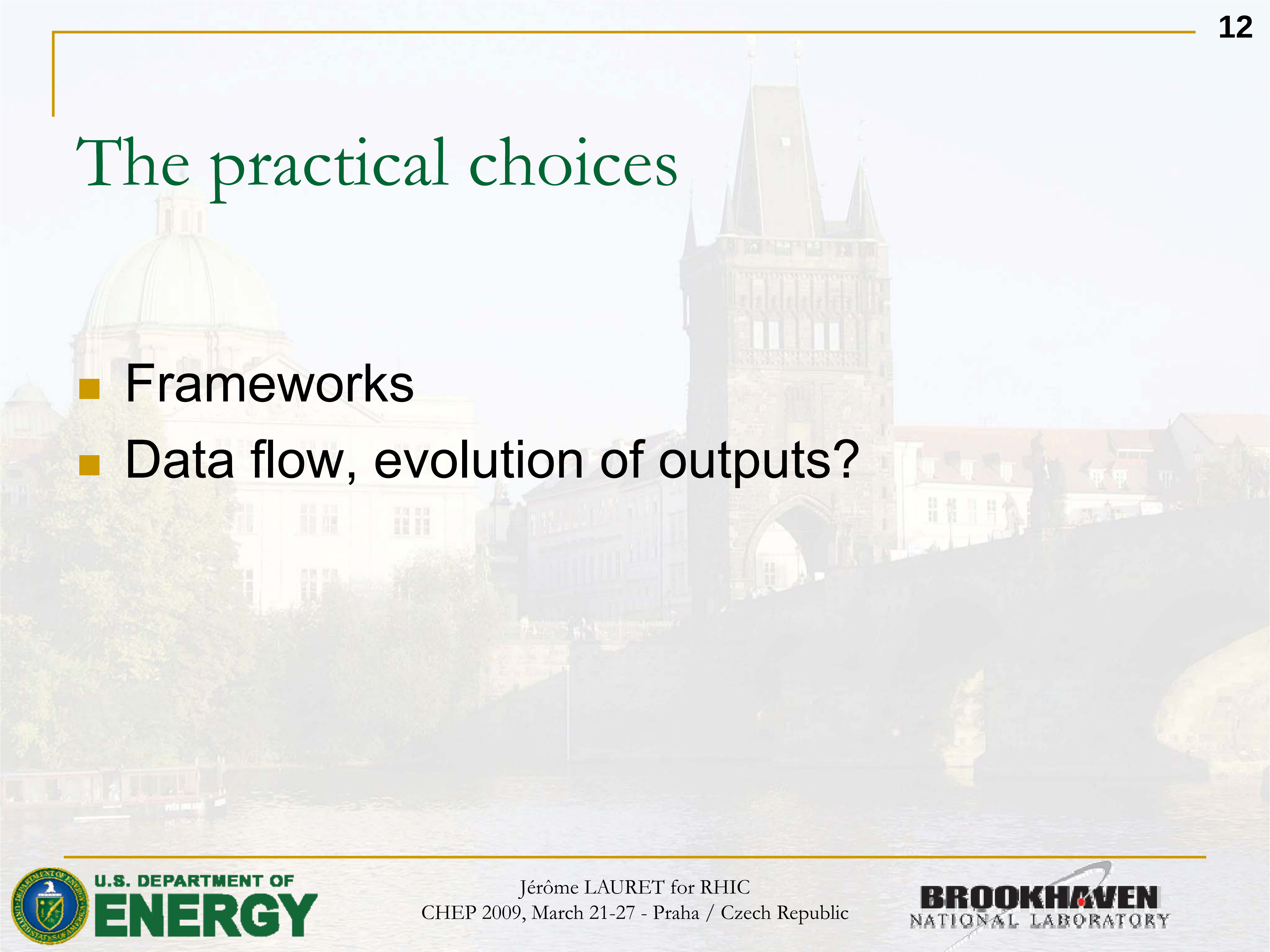
Based on current funding guidance



2.2 passes target

All red from off-Tier0 resources





The practical choices

- Frameworks
- Data flow, evolution of outputs?

Frameworks

- Rapid switch to ROOT framework
  - Provided all the basic we needed: histogram, NTuples, IO, version evolution, framework, visualization … **Good mileage**
  - Experiments built on top – **OO models all the way**
    - `PhAT` (Phobos Analysis Toolkit), `PHool` (PHenix OO Library), `BRAT` (BRahmsAnalysis Tool), `root4star`
  - Do-it-all frameworks: saves time and development cycles [Analysis, calibration, data production] merged
    - Often include simulations as well; online computing later (from 4 to 6 years after the start)

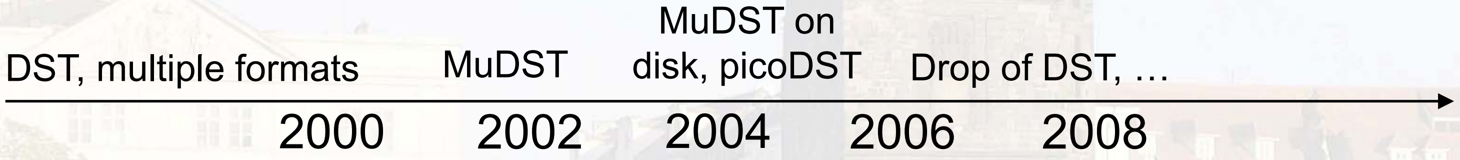
- Use **Freeware & Open source packages**
  - Initial disaster – the Objectivity DB lesson
  - Since, reliable technologies wherever possible: Use of MySQL, Postgres / Oracle (proven tech. across), …

Almost no change in frameworks since (IO, calibration, DB, framework)
Phased-in other paradigm through generic interfaces / implementations

Data, evolution of output “formats”

DST = Data Summary Tables

- Typical “data flow”: **DAQ** $\xrightarrow[2:1]{1:1}$ **DST** $\xrightarrow[10:1]{5:1}$ **MuDST** … {pico|nano}DST,



Object level access
Did not take-off – no real need? Too early?

- **New:** DST dropped after content matured (4-5 years)
  - Only a fraction kept for calibration checks purposes
  - Embedding simulation process raw signal merging

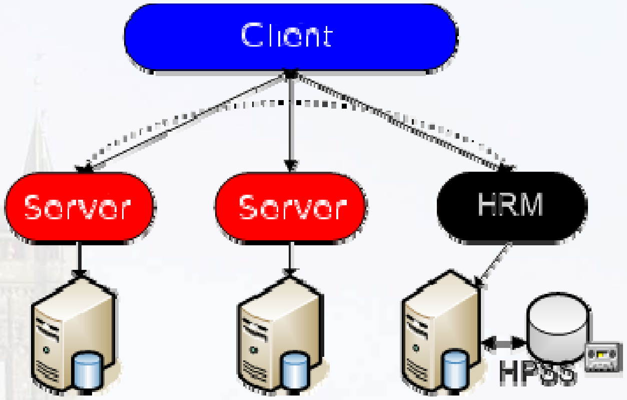
The "what worked" list
The resource saving toolkit …

- Resource saving
  - Use of distributed disks
  - Analysis Trains
  - Out-sourcing

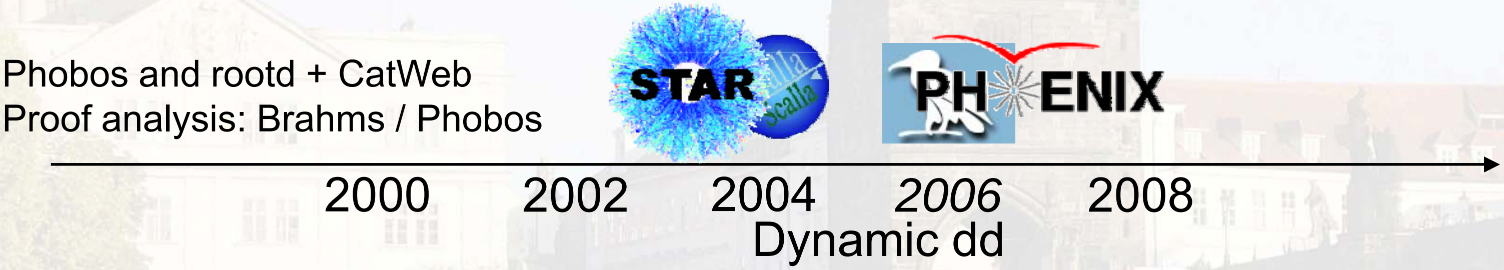
The "what worked" list
The resource saving toolkit …

- Resource saving
  - Use of distributed disks
  - Analysis Trains
  - Out-sourcing

Use of distributed disk



- The promises (or hopes)
  - NFS solutions are costly (order of magnitude vs local storage).
  - Unless high-end solutions, scalability is doubtful
  - Lots of data – dynamic restore from MSS
  - Access from remote



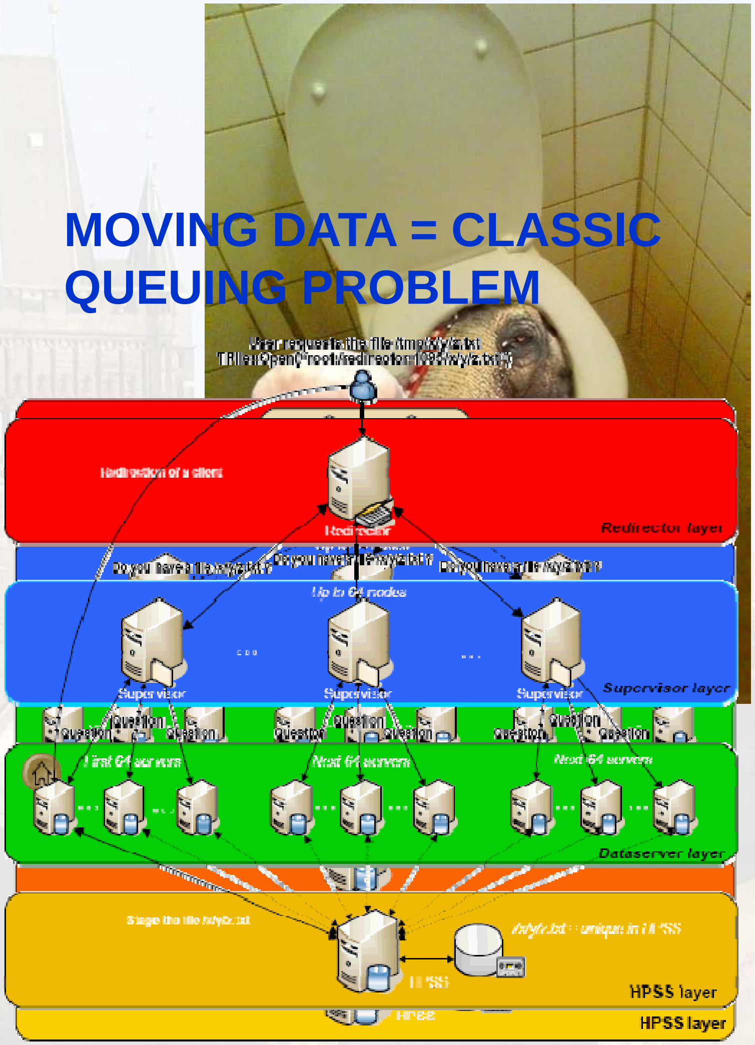
- Proof: Initially not practical as needed 100% availability, good mileage nonetheless
  - LAN was sufficient to NOT use proof at all
  - Would multi-core change this?
- Overall
  - Delivered on “cheap”
  - Would have benefited from systems such as Xrootd/dCache

- Dynamic recovery from HPSS/MSS in 2006/2007. *Disaster* with un-coordinated IO
- Disabled dynamic disk population, staged separately from single account

- Initial *disasters* with too many requests to HPSS & frequent *failure* of dCache doors
- Access restricted to single account

The problem

- Classic problem?
  - Large amount of data, not enough cache + small amount of tape drive → Throw more resource at it, restrict usage
- Not so classic solution … (CHEP 09 contribID 431)
  - Requests from Many sources (in the 1000+)
  - Tape systems: minimal file size for maximal performance: LTO3 size > 4 GB, LTO4 size > 5 GB
  - Tape degradation – increase with reads and mounts
  - Mixed environment
    - Performance matters (possible resource starvation)
    - User's perception matters
- The solution(s)?
  - Stage once or pre-stage (bring file to cache before you even need it) if possible
  - Larger file in MSS
  - Coordinate IO – DataCarousel+HPSS batch
    - Queuing system for restore / avoid costly tape mount and dismount
    - Apply policies and priority strategies, faire-shareness



MOVING DATA = CLASSIC QUEUING PROBLEM

Are we forgetting basic principles?
Will the problem of moving data on WAN from Tier sites be similar?

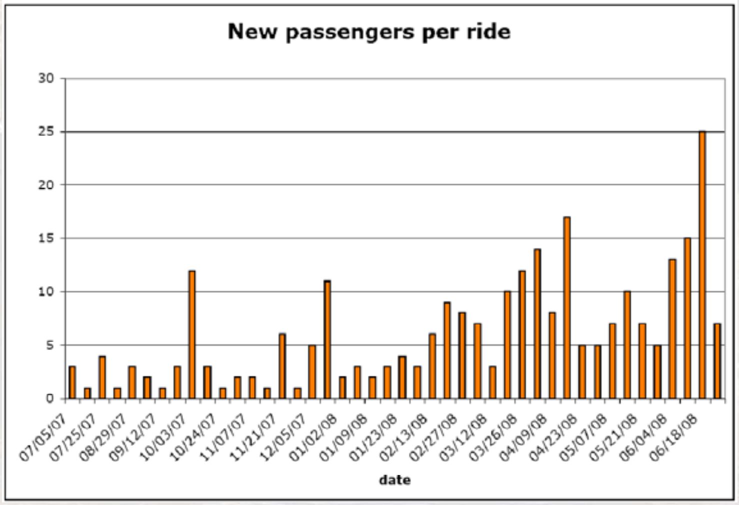
Poster #65, Thursday

The "what worked" list
The resource saving toolkit …

- Resource saving
  - ❑ Use of distributed disks
  - ❑ Analysis Trains
  - ❑ Out-sourcing

Analysis Chains, Analysis Trains, Taxis

- General idea: group analysis together
  - Run once over the data, multiple analysis done in one pass
  - N users, N reads => N users, 1 read
- Initial starts
  - Early start in STAR (day 1 design) – un-maintainable after 2 years
  - PHOBOS model based on a few users & Proof data access – some success
- PHENIX: Analysis trains, best success after initial tuning
  - Spin over pre-staged and pinned partial data, replace data, go-to next sample, re-launch the train. Datasets of interest covered in ~ 3 weeks
  - Larger cache added in 2006/2007 allows for a 1 week turn around



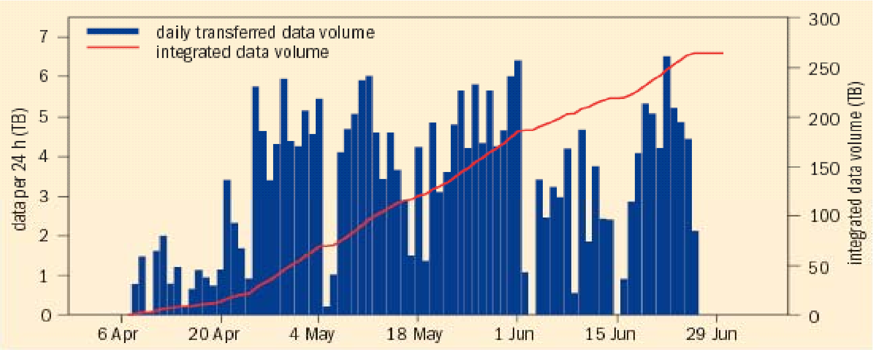
Analysis trains have matured and so has the community. This mode of operation will be on the increase with resource demands.

The "what worked" list
The resource saving toolkit …

- Resource saving
  - ❑ Use of distributed disks
  - ❑ Analysis Trains
  - ❑ Out-sourcing

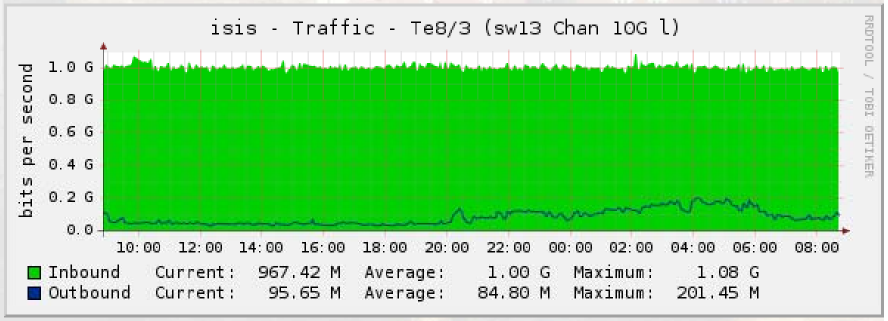
Out-sourcing?

- Main focus @ RHIC
  - move data to other sites, recover or offload cycles
  - ***Network data transfers ARE undeniable science enablers***
- Data transfers in STAR
  - Bulk transfers using BeStMan/SRM (NERSC/PDSF) since 2002
  - Transfer to China in 2004 (picoDST)
  - Routine transfer to Prague in 2008
  - “Raw” Grid ftp (KISTI/Korea, …)
- Data transfers PHENIX
  - Grid ftp for Phenix (to/from CC-J Japan)
  - Transfers to/from CCF (srm-cp)



Daily rates of data transferred from the PHENIX experiment to the CCJ computing centre in Japan (blue), and the integrated data volume (red). Overall, 270 TB of data were transferred.

Phenix data transfer to CC-J/Japan in near real-time, Story here.



STAR Data transfer from BNL to KISTI/Daejeon Korea sustained at WAN speed 1 Gb/sec
Story here

Poster 092, Indico contribID=432,
Setting up Tier2 site at Golias/ Prague farm

Distributed computing

Grid-ing or not Griding?

- What are the RHIC experiments doing Grid-wise (data movement apart)?
  - STAR: only active experiment to routinely run jobs on Grids (+dev)
  - **So, what is/are the problems if any?**

- Are Grids usable?
  - Outstanding efficiencies – efficiency > 97%
    - Operation support from Grid projects (OpenScience-Grid)
  - Justified to move all STAR Monte-Carlo productions on Grid (2006)
  - ✓ **USABLE**

Open Science Grid

- Where are the problems for production environments?
  - Grids are complex and too dynamic for production environment
  - Troubleshooting is simply inadequate (globus error # anyone?)
  - VO mainly using dedicated sites with pre-installed software stack
    - **Little to no opportunistic use**

Or is it Clouding or …?

ContribID # 516

ContribID # 475

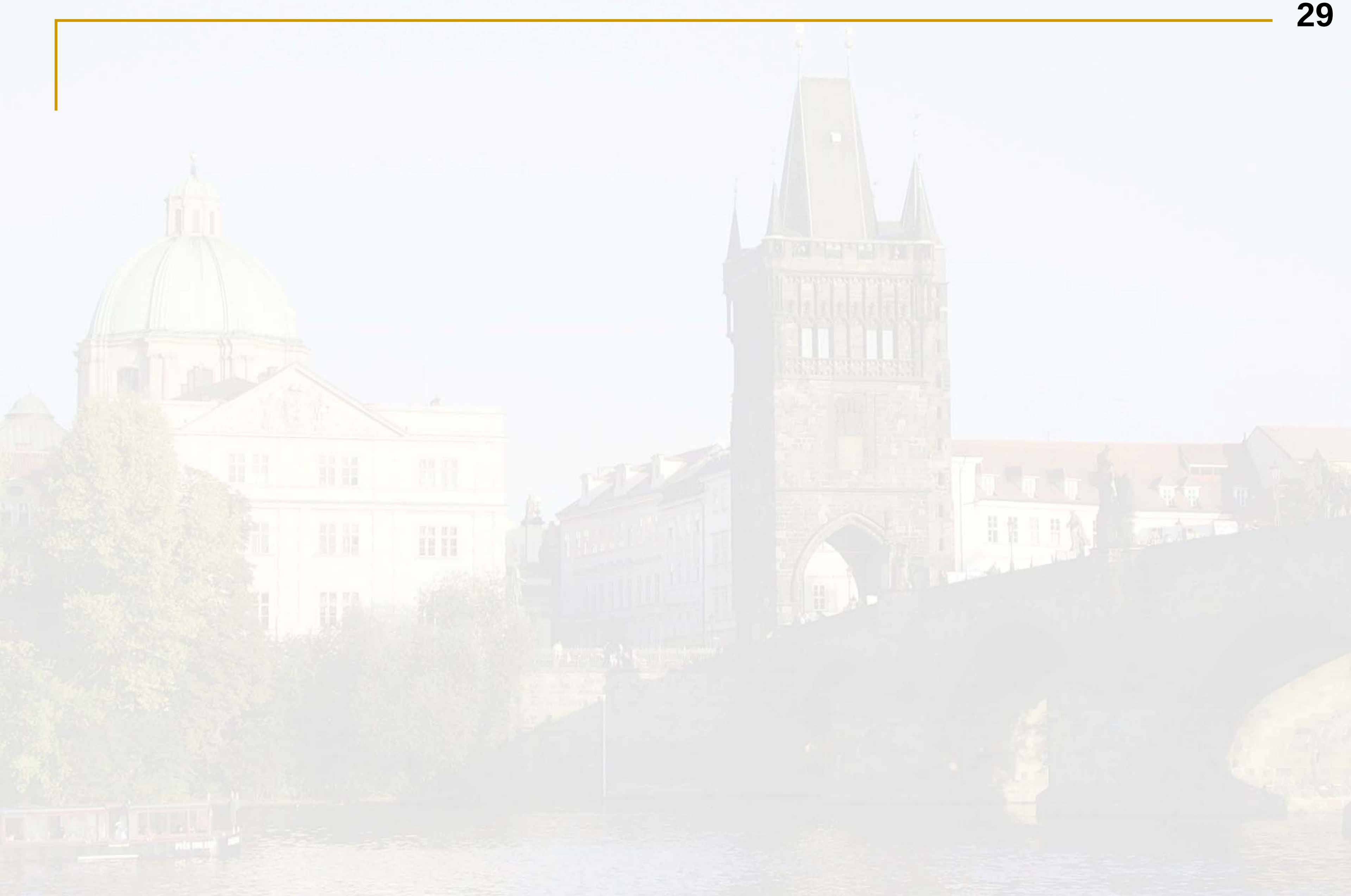
- **Are Cloud usable?**
  - STAR Use *Amazon/EC2* / Elastic Cloud Computing (Nimbus / Test in 2007/2008)
  - Scale & Performance: ~ 300 jobs at all times, weeks long
    - Similar efficiencies than normal Grids measured so far
    - 5 MB/sec data transfer / WN – for simulation, enough
    - **NOT A SILVER BULLET** (under the hood, still the grid stack)
  - ✓ **USABLE**
  - Status: STAR run on EC2 to handle MC production (event generator + response simulator + full reconstruction) – Emergency request
    - **Results have been used for analysis to be presented for Quark Matter 2009 [real practical use of Clouds helping science deliverables]**
- Economics of Clouds remain puzzling (within range of facility costs to first order)
  - Cons: MSS unlikely on Clouds, Network performance low
  - Pros : Truly opportunistic used at reach, software provisioning is immediate to any site
  - **IMMEDIATE benefits, LEAST efforts, MAXIMAL confidence**
- Prospects? Technology rapidly changing …
  - **Grid and clouds are NOT orthogonal – VM provide on the fly resources**
  - **Integrating technology on OSG, enhance/complement grids**
  - **Truly opportunistic implies network dynamic circuit provisioning?**
  - **…**
- **Are we ready?**

Conclusions

- RHIC experiments
  - Large data samples, LHC comparable according to current numbers
- Using pragmatic approaches and principles
  - Survived two order of magnitude more data
  - With overall little changes in initial schemes and designs
  - ROOT works: let’s use it; OpenSource all the way
- Maturing
  - Better handle on how to allocate resources
  - Matured data format allows reduction of x5 to x10
  - AnaTrain, distributed disks, data transfers are routine
    - Avoid access to tape at all cost, optimize otherwise
- Grids
  - Beyond data transfers, only 1 out of 4 using Grid job production
  - Perhaps Clouds (VM) will change the game – truly opportunistic use of resources / last moment requests possible, easy provisioning (advantages go beyond grid)

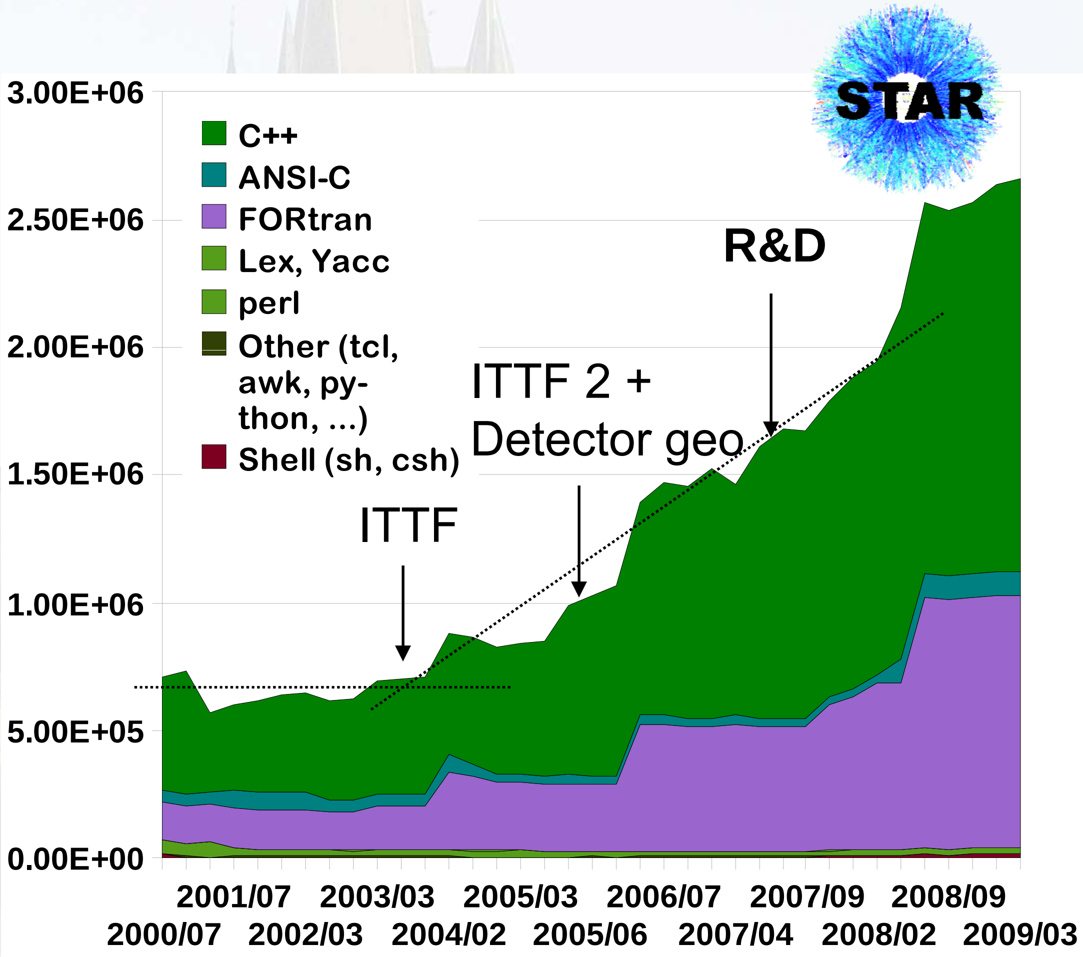
Did great, accumulated experiences and ready for the challenges ahead …

Backup slides



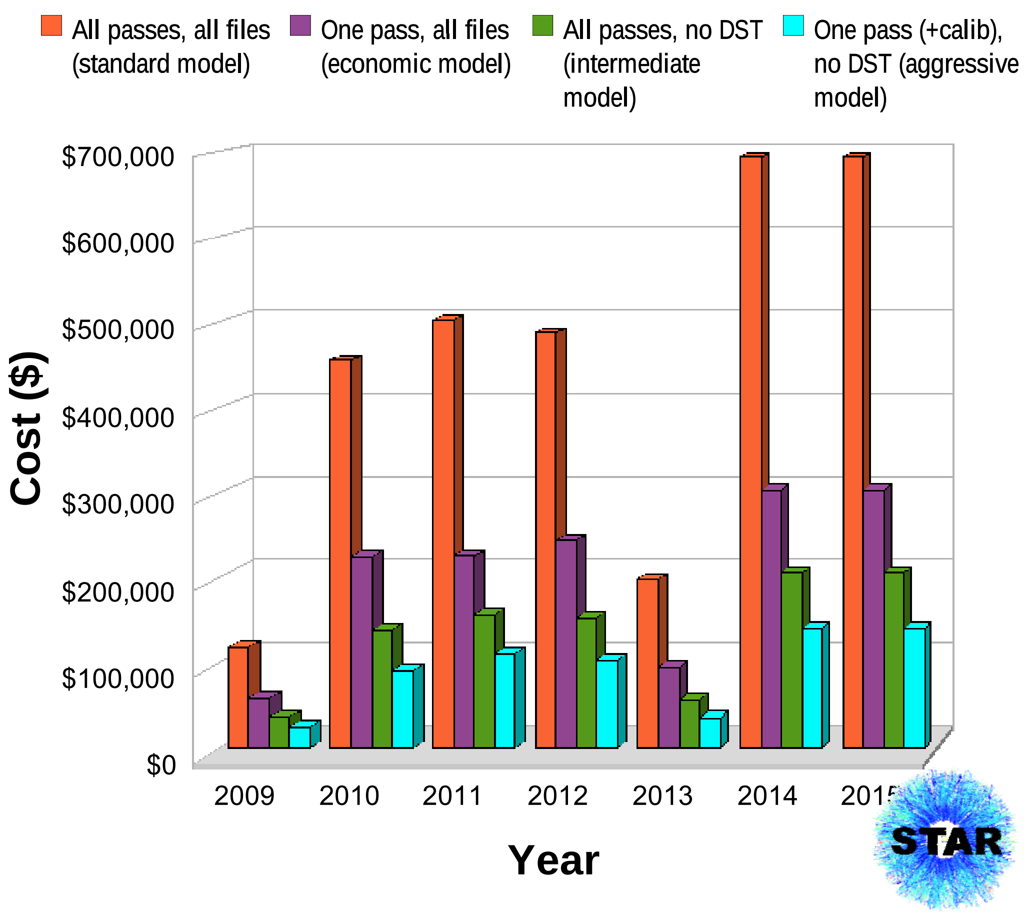
Composition, Evolution & code growth

- Overall
  - C++ dominates
  - FORtran (MORtran) is second - Geant3 legacy
- STAR: Mandate to have all data reproducible to the exact
  - Very rare code are removed – code change with compilers
  - Geant3 phasing out

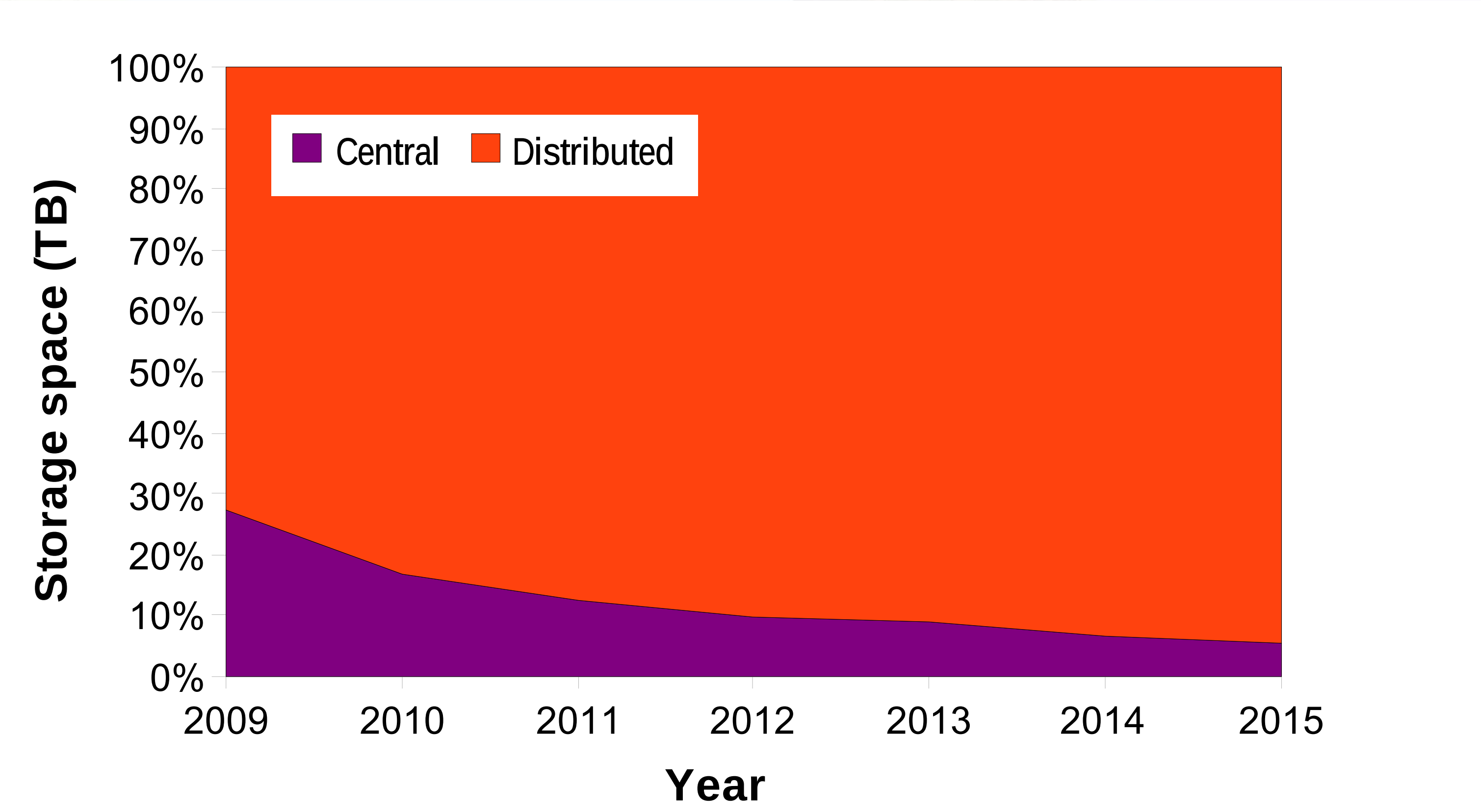


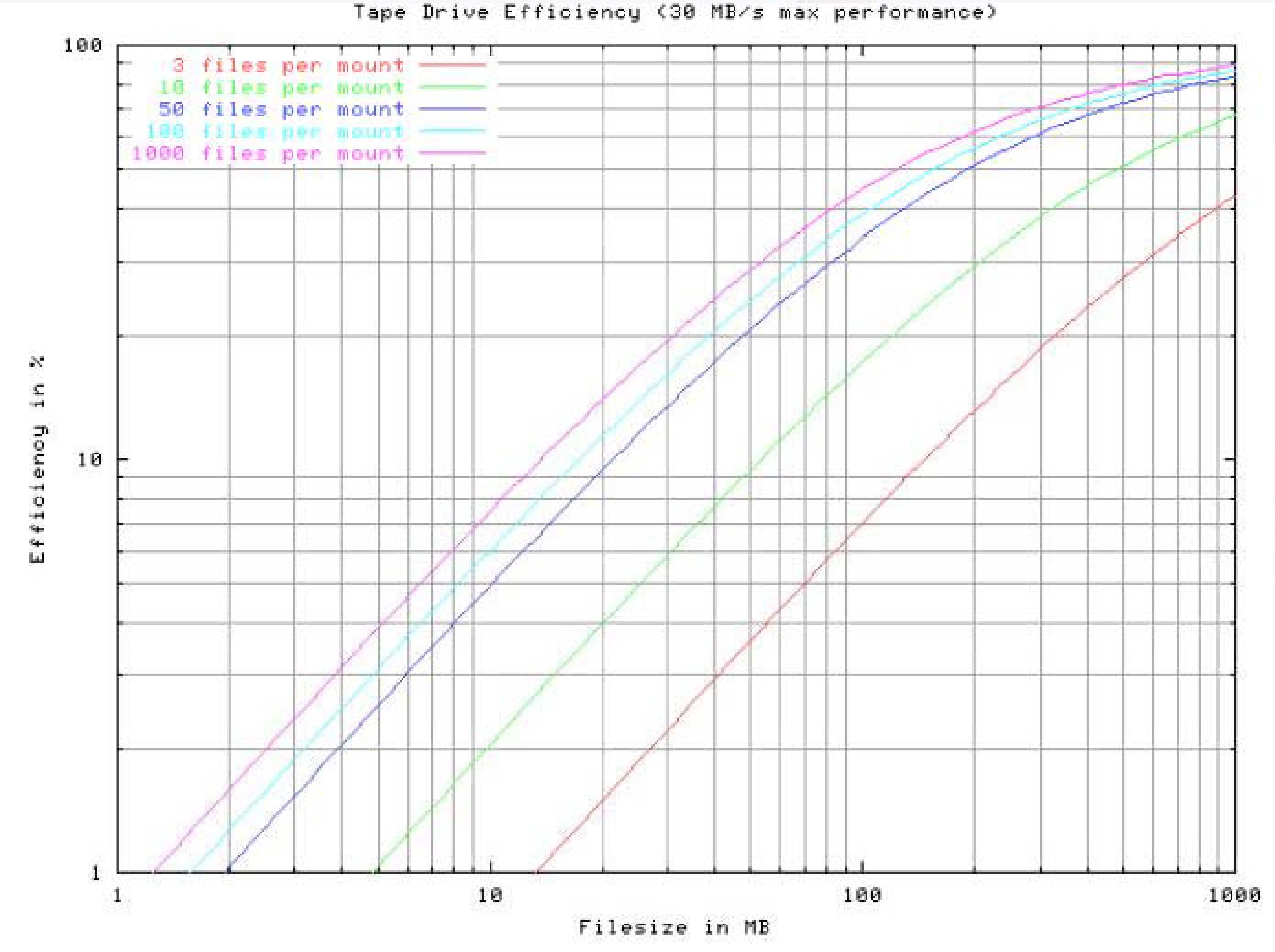
Cost saving – DST drop?

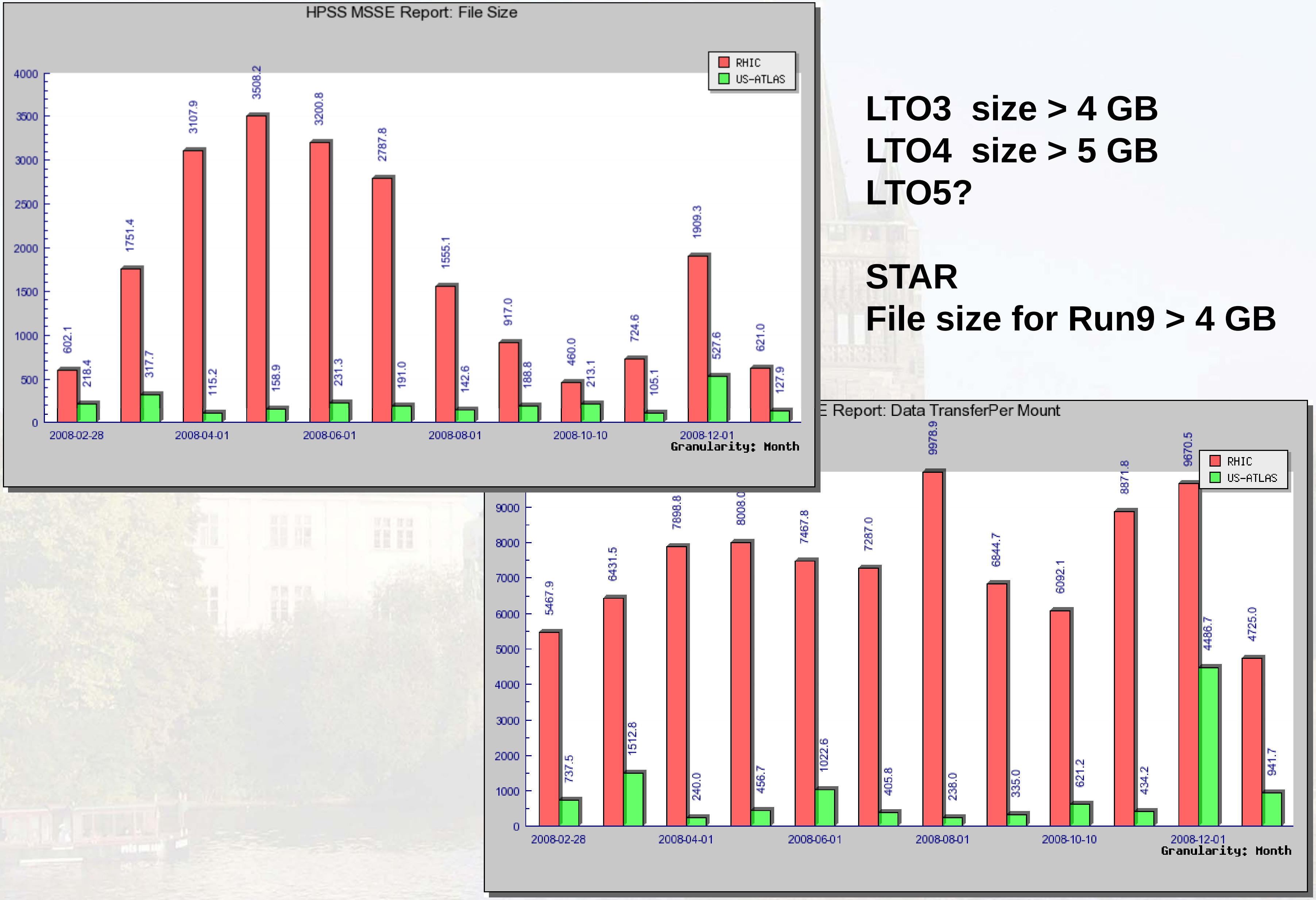
- Business as usual
  - 1M$/year of tape
- One pass data only
  - ½ the cost
- Aggressive cost saving
  - 1 pass + 0.2 calibration
  - No DST
  - Very much viable
- STAR choice
  - All passes MicroDST
  - 1/10th or less DST
  - Delete previous pass DST



Relative ratio Centralized/distributed disk in STAR, 2009-2015



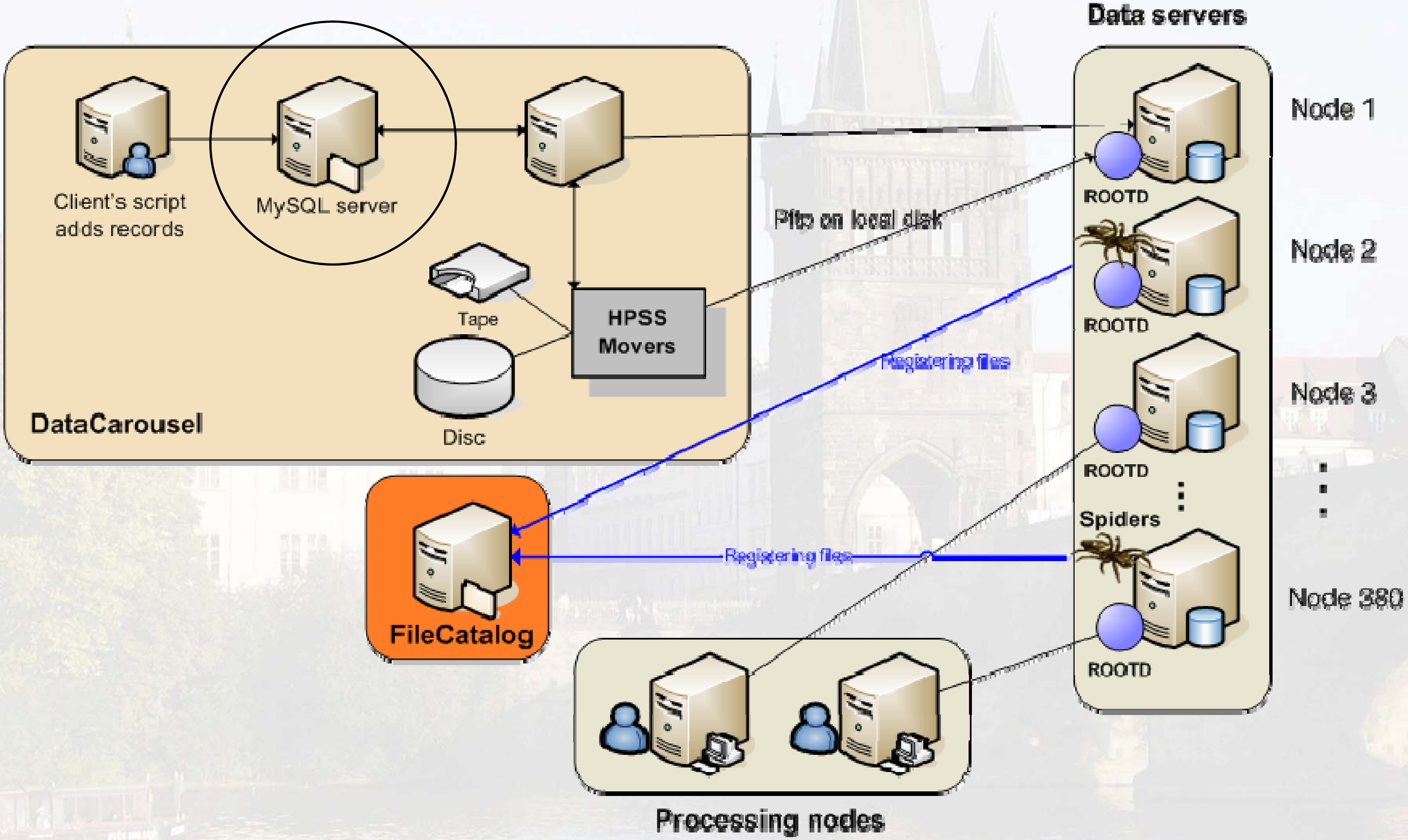

Tape Drive Efficiency (30 MB/s max performance)
3 files per mount
10 files per mount
50 files per mount
100 files per mount
1000 files per mount
Efficiency in %
100
10
1
1
10
100
1000
Filesize in MB

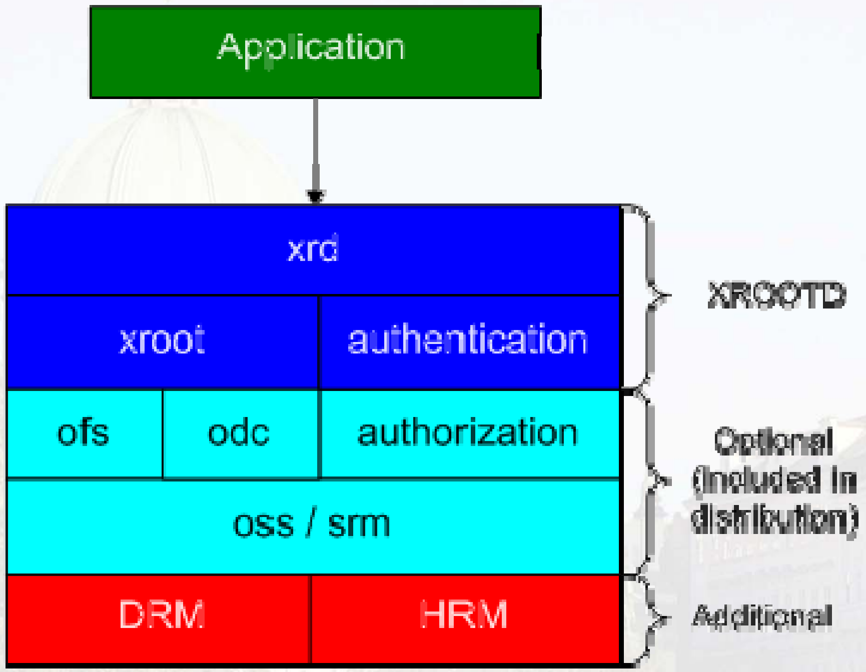


LTO3 size > 4 GB
LTO4 size > 5 GB
LTO5?

STAR
File size for Run9 > 4 GB

The DataCarousel





Also proposed a similar “principle” using SRM DRM/HRM for Scalla/Xrootd

Grid data access on widely distributed worker nodes using scalla and SRM
P. Jakl *et al* 2008 *J. Phys.: Conf. Ser.* **119** 072019

Available since 2008.

Analysis train and analysis taxis

- Since ~ summer '06
  - Add all existing distributed disk space into dCache pools
  - Stage and pin files that are in use (once!)
  - Close dCache to general use, only users phnxreco (mostly write) and anatrain (read/write) have access: performance when open to all users was disastrous - too many HPSS requests, frequent door failures, …
  - Users can "hop in" every Wednesday, requirements are: code tests (valgrind, insure), limits to memory and CPU time consumption, approval from WG for output disk space
  - Typical time to run over one large data set: 3-6 days
- Currently used by ~300 different PHENIX Analysts

The data

Entire data set(s) staged from HPSS into dCache disk (once) and kept there

New rides start every week

Condor jobs are submitted for each "fileset" (~10GB chunk of input data), which is then copied from dCache into the local area of the executing node

All the modules that need a given fileset run over it

Database keeps track of failed jobs for each module, which are then resubmitted

The process

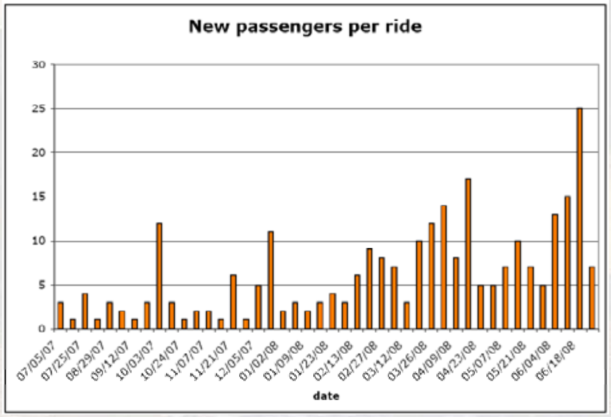
Develop and test analysis code using small central disk-resident sample

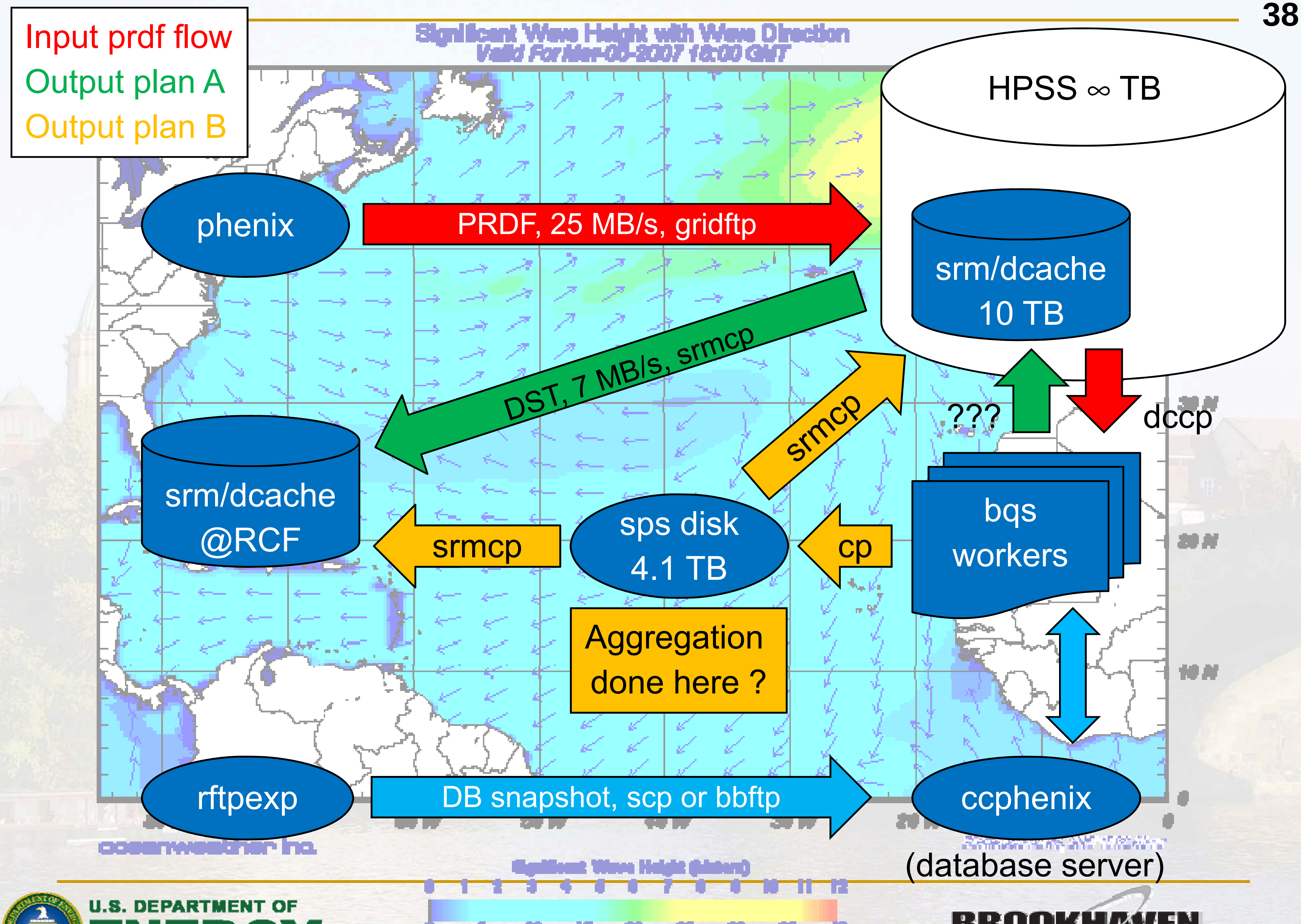
Get approval from WG for usage of space for analysis output

Check-in the code and fill a web-based form to get a ride

Can check on the status of each module's progress online, as well as deactivate/reactivate it

The users




Significant Wave Height with Wave Direction
Valid For Mar-03-2007 18:00 GMT
oceanweather inc.
Significant Wave Height (meters)

Input prdf flow

Output plan A

Output plan B

HPSS ∞ TB

srm/dcache 10 TB

phenix

PRDF, 25 MB/s, gridftp

DST, 7 MB/s, srmcp

srm/dcache @RCF

srmcp

sps disk 4.1 TB

srmcp

cp

???

dccp

bqs workers

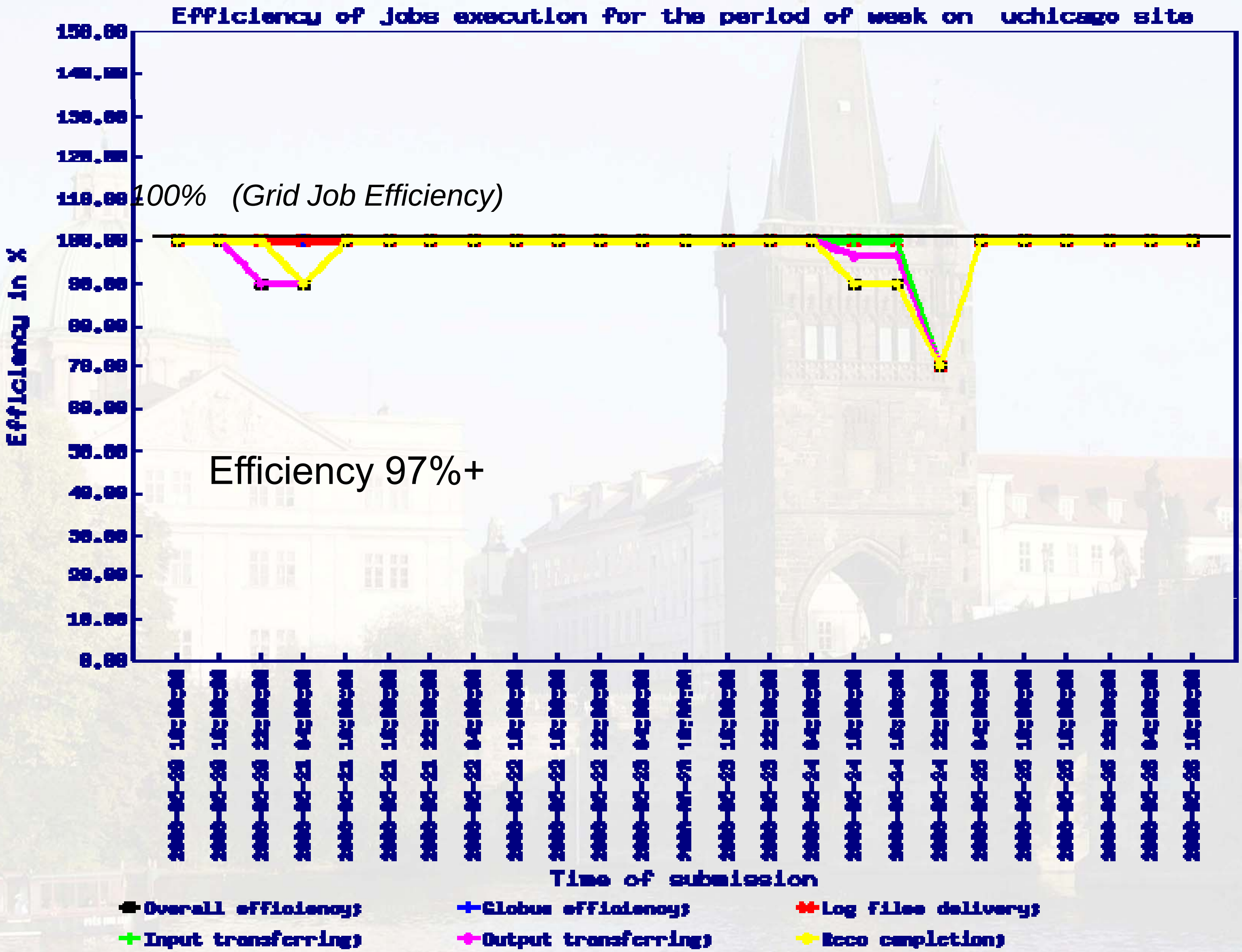
Aggregation done here ?

rftpexp

DB snapshot, scp or bbftp

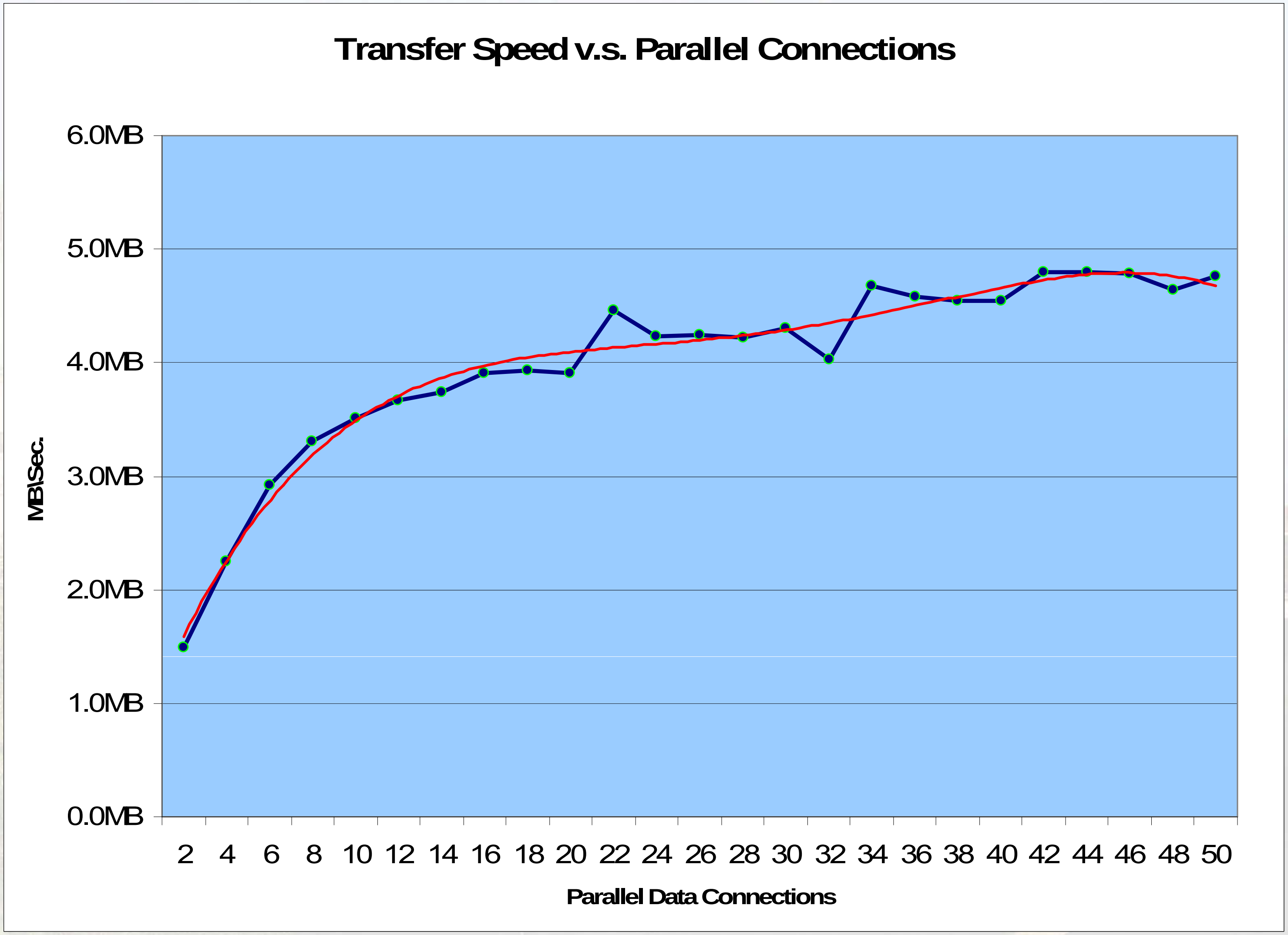
ccphenix

(database server)



100% (Grid Job Efficiency)

Efficiency 97%+



Transfer speed from one node on Amazon/EC2 – scale linearly
Up to 20 nodes (test done in 2008)