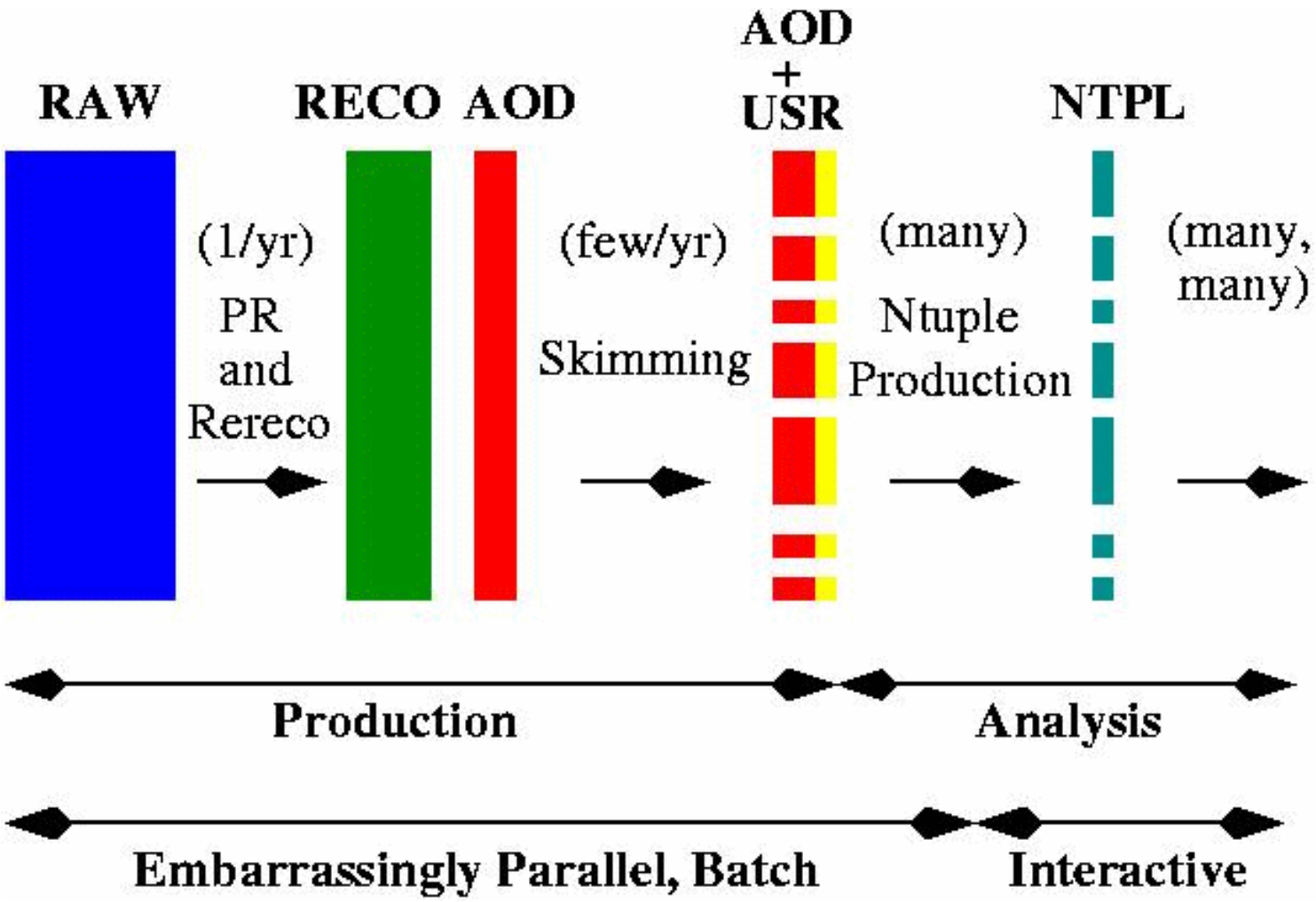
Distributed Data Management in HEP

Peter Elmer
Princeton University
CHEP06, Mumbai, India
15 Feb, 2006

Overview

- Working in an experiment, I find it difficult to talk about “data management” in isolation. I could talk about:
  - catalogs, SRM, data access protocols, SE's, quotas, scaling
- But what I really think about is:
  - Computing/Analysis/Data models, Access Patterns
  - Interactions with the workflow/workload management, etc.
  - What happens when things go wrong
- There is some bias towards an experiments point of view in this presentation (in particular BaBar and CMS)

Example: BaBar Data Reduction



Real data, similar picture for MC

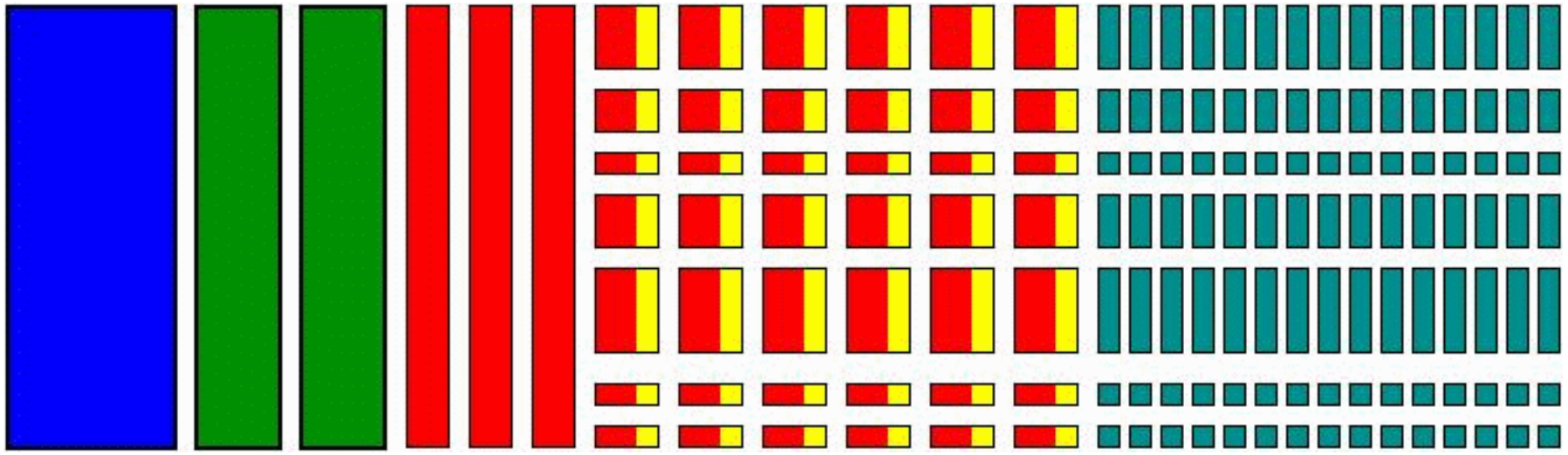
"Logical view" - mature experiment, "Factory"

Generic data reduction model

- Main characteristics:
  - Move common calculations upstream
  - Reduce data size *and* store calculations (with an obvious trade-off) to allow for more frequent, "chaotic" access downstream
  - Store enough useful quantities to support as large a fraction of analyses as possible
  - Each step prepares the data as best it can for subsequent consumers
  - Factorize out "time variation"/calibrations into separate database so that analysis user can use "datasets"

Datasets

"Dataset Bookkeeping View" - mature experiment



(Relative size indicates interest for analysis, rather than data size)

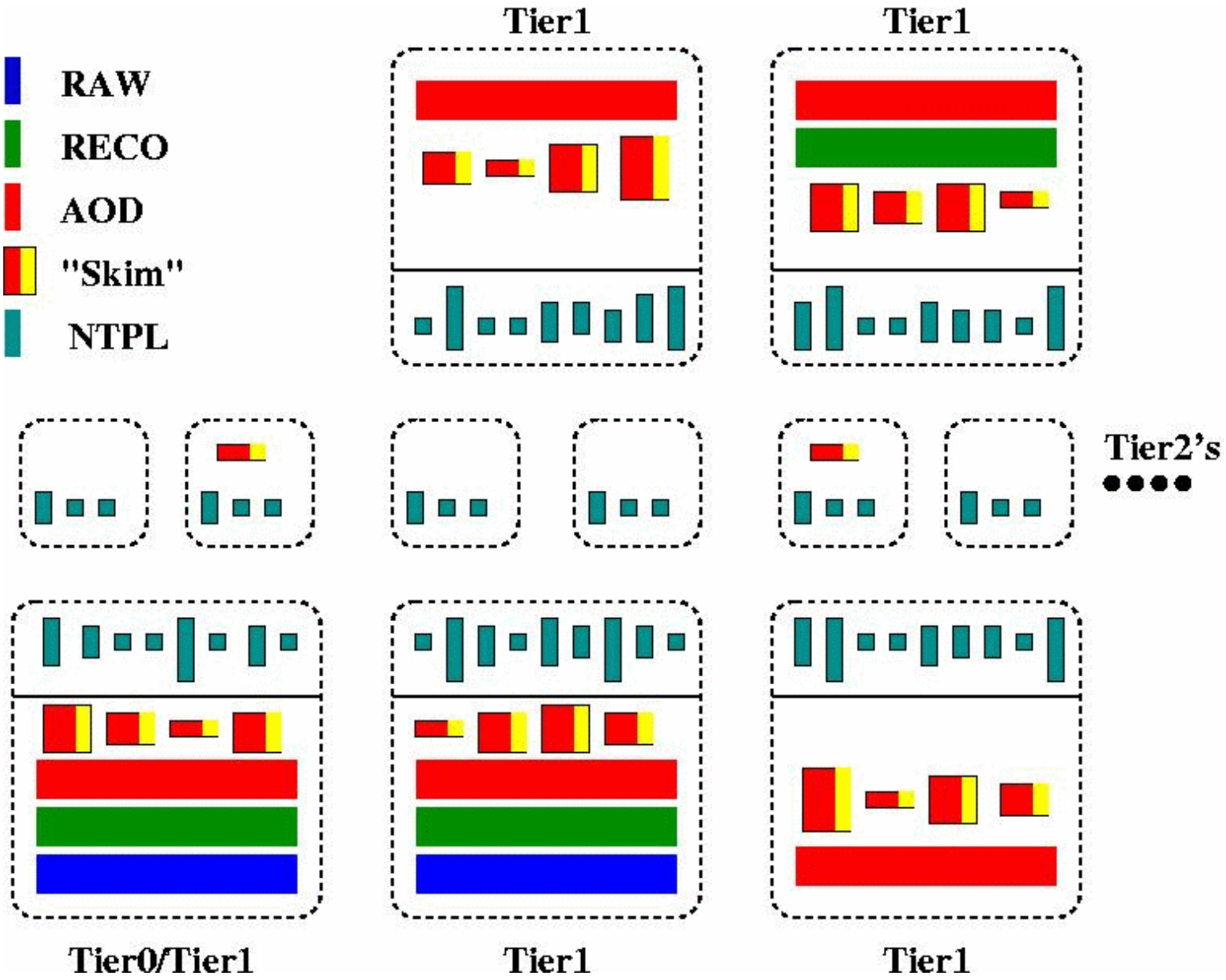
~1.2PB active data (+ ~1.0PB old/inactive), including MC

15 GB dataset bookkeeping, 40GB calibrations

BaBar Data Reduction

- These simple cartoons hide many, many DM things:
  - Overall dataset bookkeeping system, data access and storage
  - Files, file management, filesize management
  - Various flavors of "data handling" systems for each production step, significant struggle to make each work in its environment
  - Lots of error handling, file integrity checks, etc., etc.
  - Mature understanding of "RECO", "AOD" and the basic production framework to deal with evolving understanding of necessary "skims". Stabilization of access patterns.
  - Mature calibrations (such that with a large granularity the analysis user can view a "dataset" as such)

Distributed Data Management

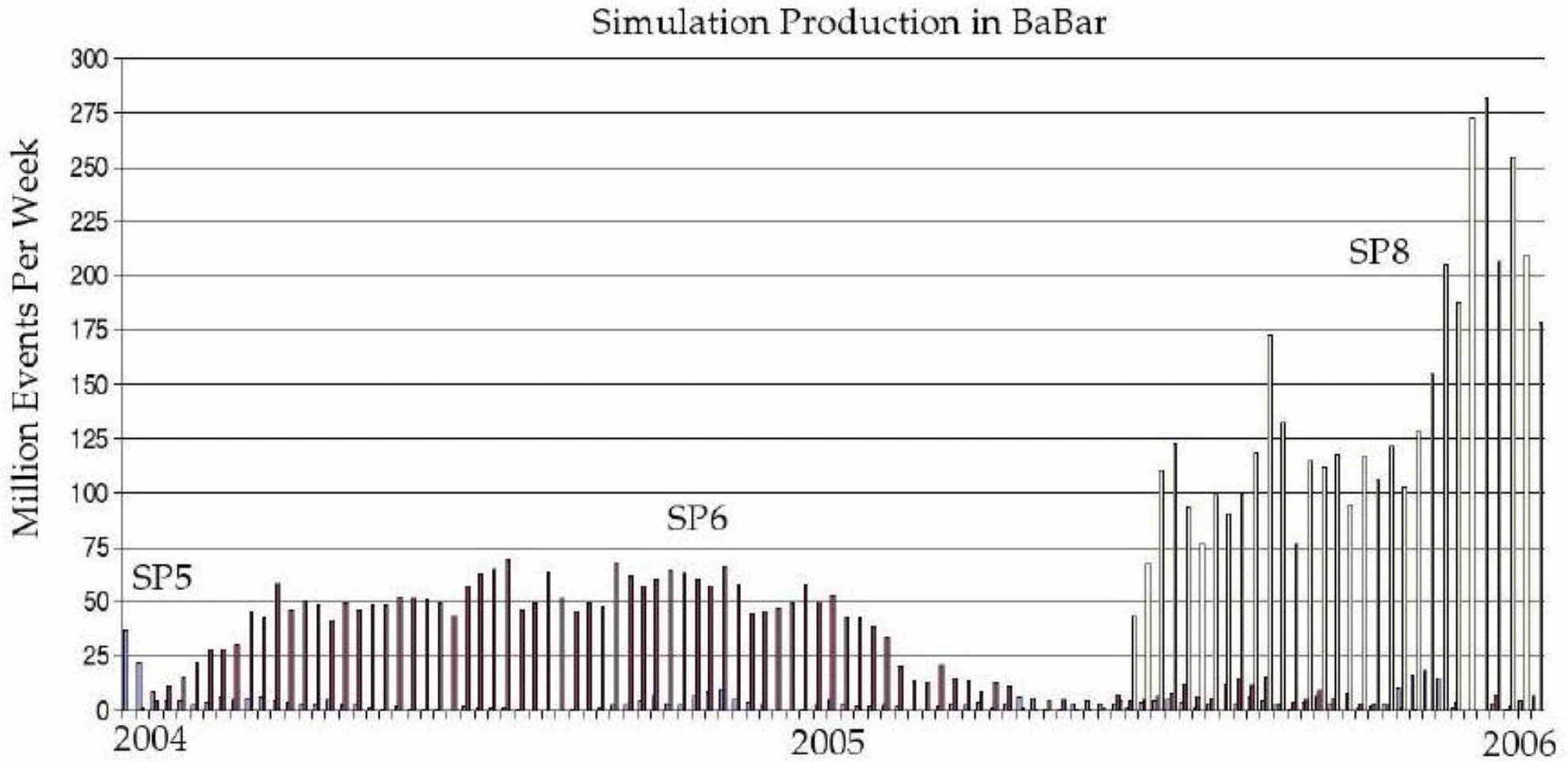


"Where do I do my analysis?" view

Data/MC/Skim Production systems

- Data handling systems “unto themselves”
- Input and output buffer sizes, I/O, etc. relatively well understood, including predictable scaling
  - Prompt Reconstruction
  - Monte Carlo Production
  - Skimming
- Internal, segregated bookkeeping systems to manage temporary outputs before merges. Eventual “publishing” into the global dataset bookkeeping used by other production systems and analysis.

Simulation production



Peak of 275MEvt/week in up to 25 sites

BaBar Data Management

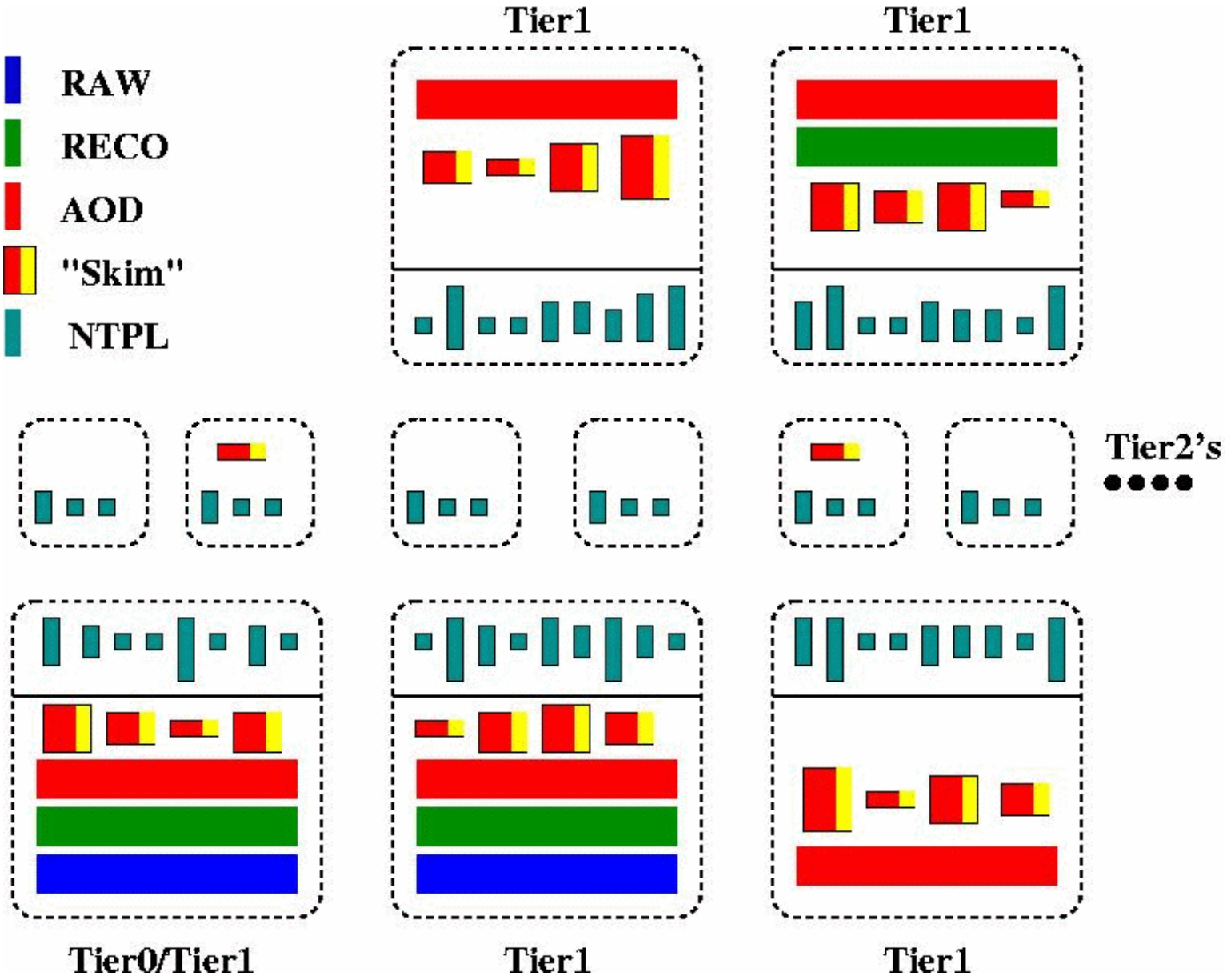
- The system was built around:
  - a difficult EDM initially and then a much simpler one
  - central dataset bookkeeping with replicas in some sites
  - local (“trivial”) file catalog used by jobs
  - event data storage with xrootd + MSS, some NFS for user data/ntuples
  - independent transfers (bbcp, bbftp, gridFTP) managed by custom tools
  - lots of local disk buffers for temporary pre-merge files produced by “production” (NFS + xrootd)

✔ Many things not “state of the art”, but produced physics results

BaBar DM after 6 years

- ✔ Relatively mature analysis and data reduction model
- ✔ Tier-1's used for “production” and user analysis
- ✔ Tier-2 use: pull model for analysis data, push (to Tier-1/SLAC) model for MC data produced at the Tier-2, push model for “Tier0/PR” in Padova to Tier-1/SLAC, push model for skimming from multiple sites, etc.
- ✗ No real regulation of the use of network bandwidth (not really needed), however Reconstruction and Skimming heavily locked into a few sites for other reasons
- ✗ Separate internal DM bookkeeping systems for “production”
- ✗ Inadequate data management for “user produced” data (custom ntuples/ROOT TTrees)

Distributed Data Management

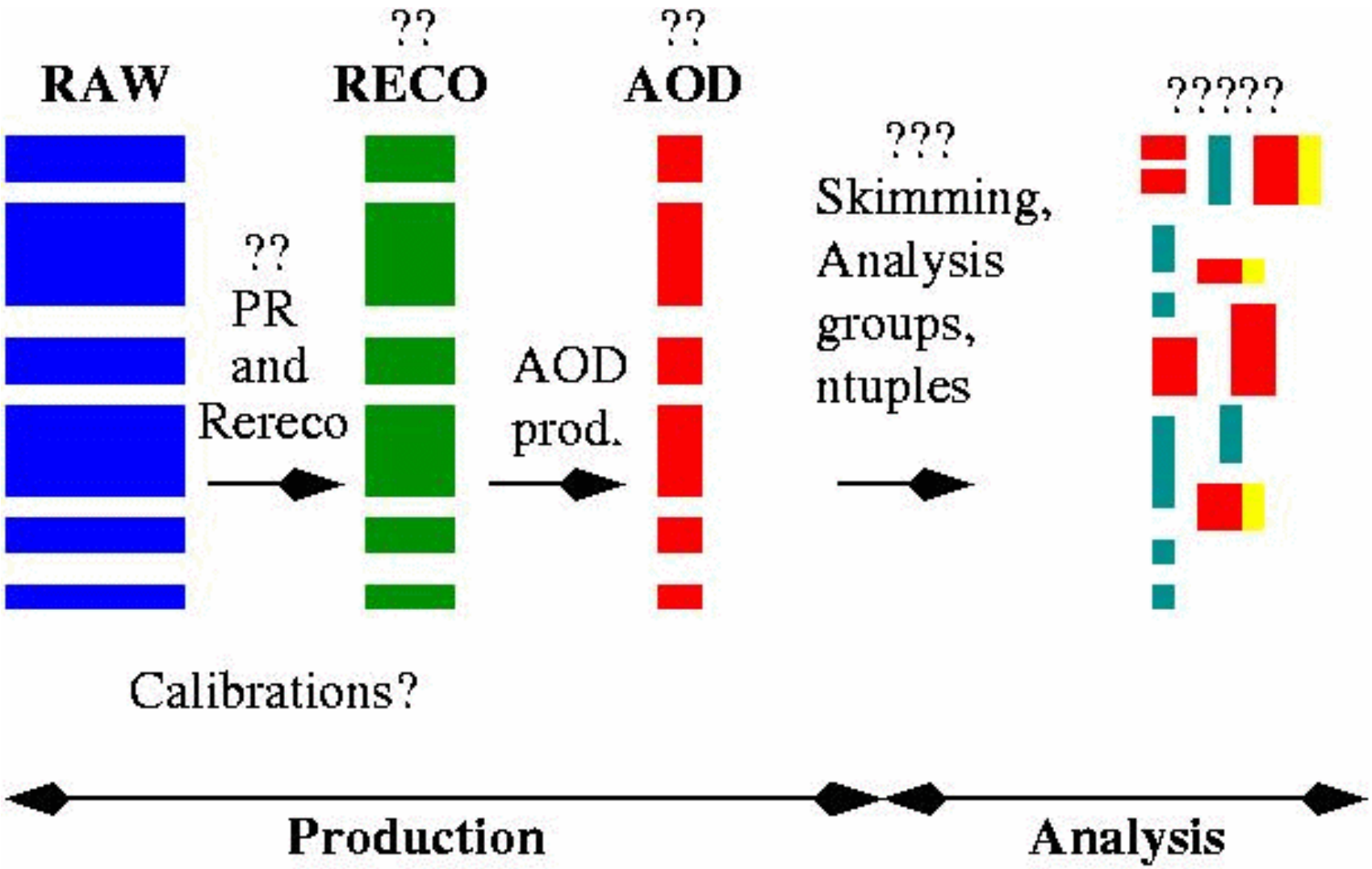


Note in particular the large amount of NTPL data scattered around the system

Why am I talking about this?

- So why this high-level review of a mature experiment?
- It is clear that it took *years* to climb our way up that mountain (this feeling is not unique to BaBar)
- The process will start anew very soon for the LHC experiments: how far back down the mountain will we slide?
  - Will the improvement and/or introduction of technologies make it easier this time?
  - Or will we benefit from being wiser as a community?
  - Where will the challenges be? What will blow up in our face?

Example: CMS Data Reduction

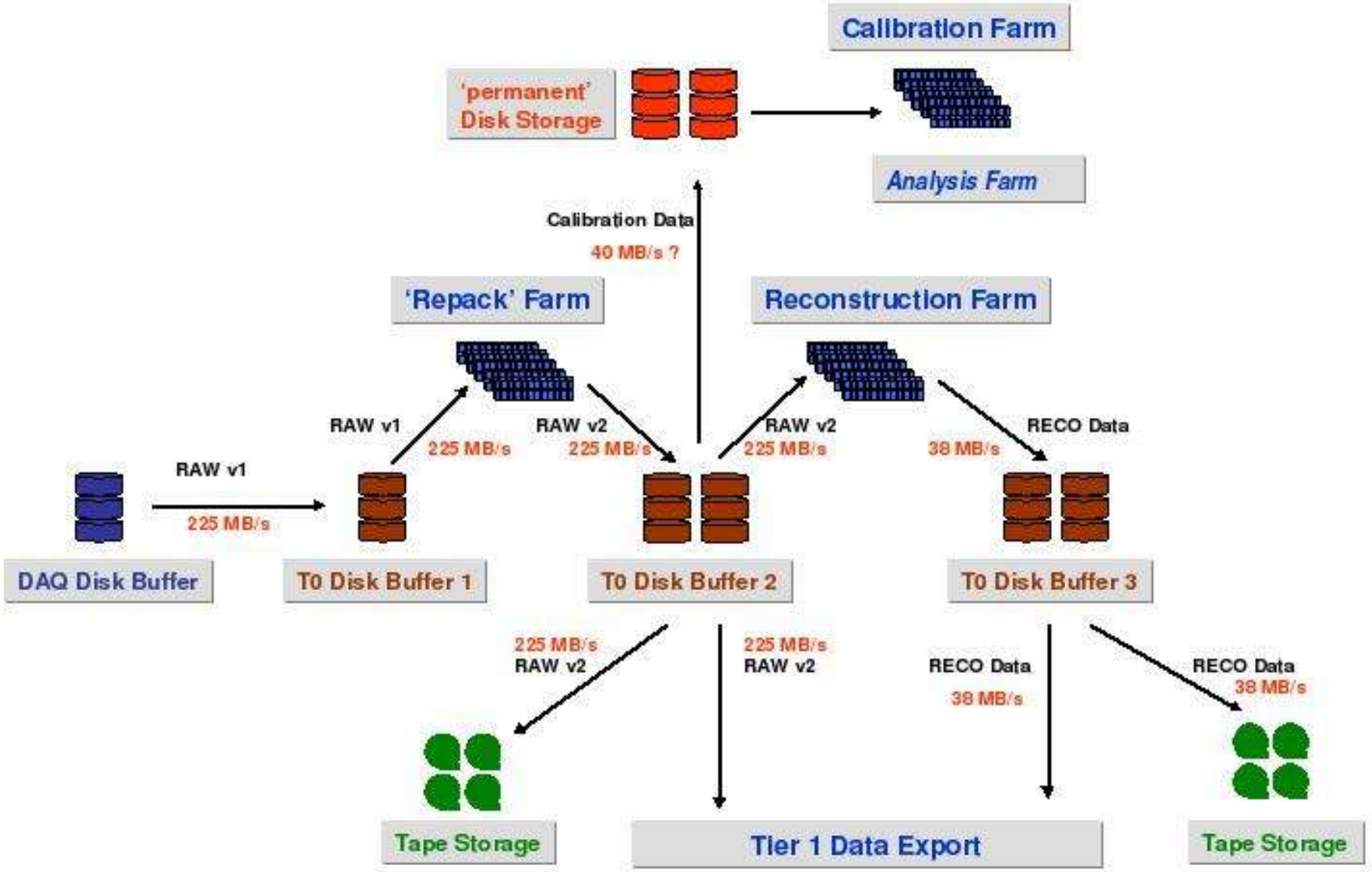


Back near the bottom of a mountain...

DM-related issues for the next years....

- RECO/AOD still being defined, will evolve significantly...
- Data access patterns not well understood, will evolve...
- Prompt Reconstruction/Tier-0 data handling
- Lots of detailed calibration work, reprocessings
- Prompt Calibration??? (Automated calibrations?)
- Learning to use of Tier-1's, learning to use of Tier-2's
- AOD productions, Skimming (foreseen “production”)
- True bulk user analysis?? Analysis model(s)?? (wide open)
- Some simplifying choices: very early streaming based on trigger selections, but even this brings some complications

Tier-0 model (in evolution)

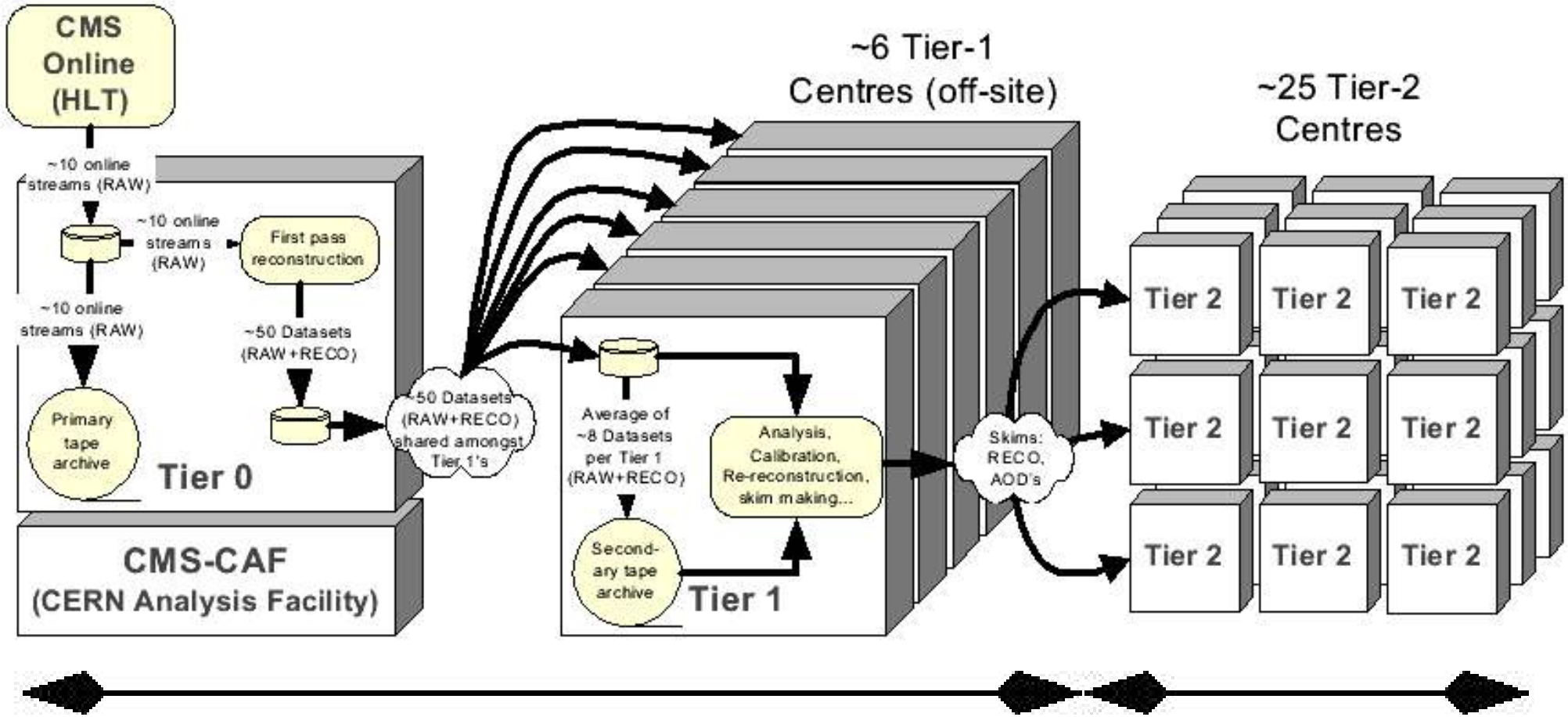


(Diagram from Bernd)

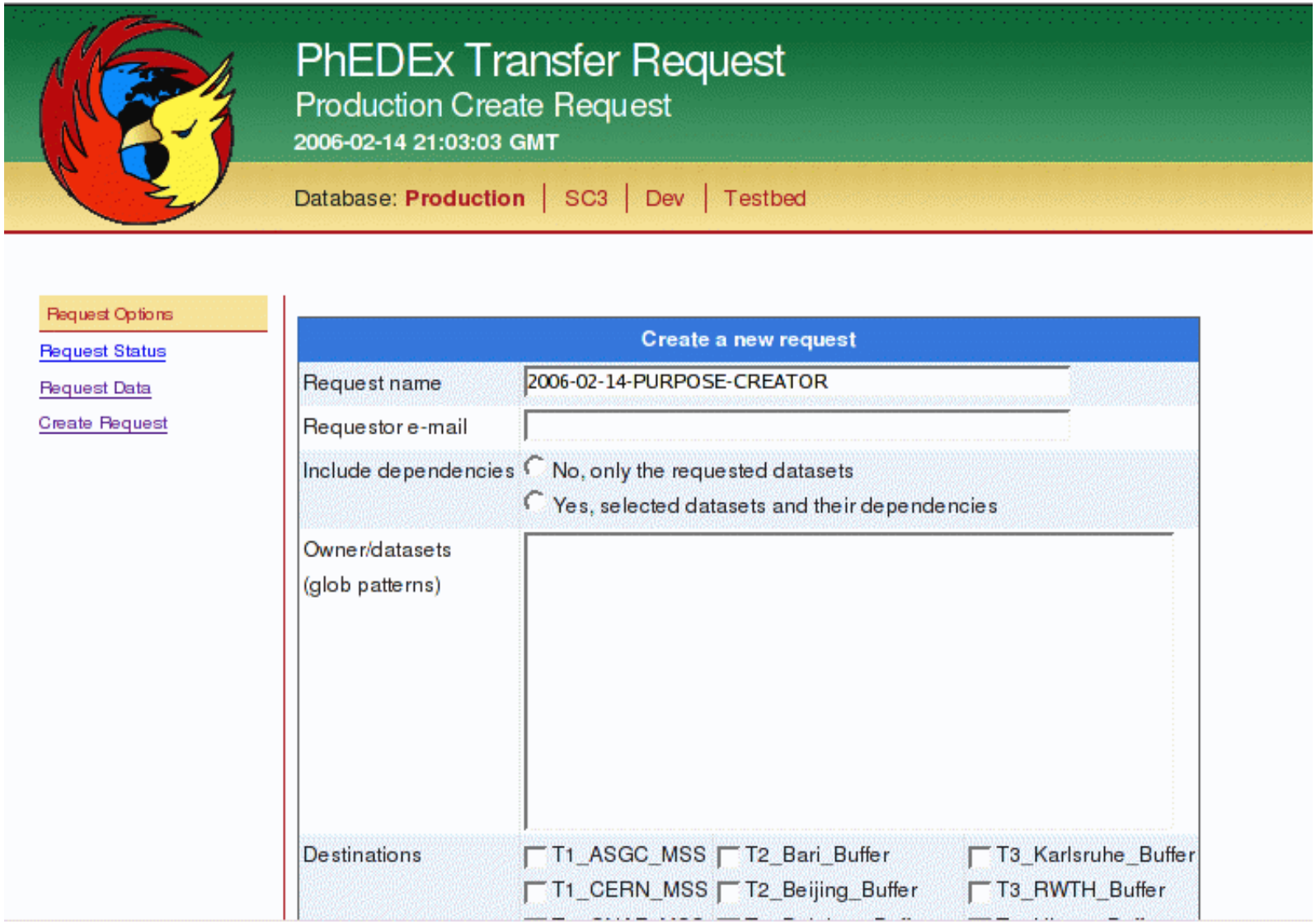
Tier-0 issues

- The place where everything starts is of course the Tier-0 and calibration farm, e.g. CERN CAF for CMS. This is the place where “data handling” becomes “data management”.
  - Filesize management: mismatch between file size from high level trigger (HLT) and the desired offline RAW file size
  - Sub-farm structure of HLT, managing many “primary streams”
  - Time and “lumi segment” ordering
  - Interactions with Prompt Calibration and calibration farm
  - Failover of Prompt Reconstruction, etc. to Tier-1's
  - I/O, latencies, buffer management come more complex

Distributed Data Management



Phedex – file transfer/placement



Builds on underlying services like FTS, etc.

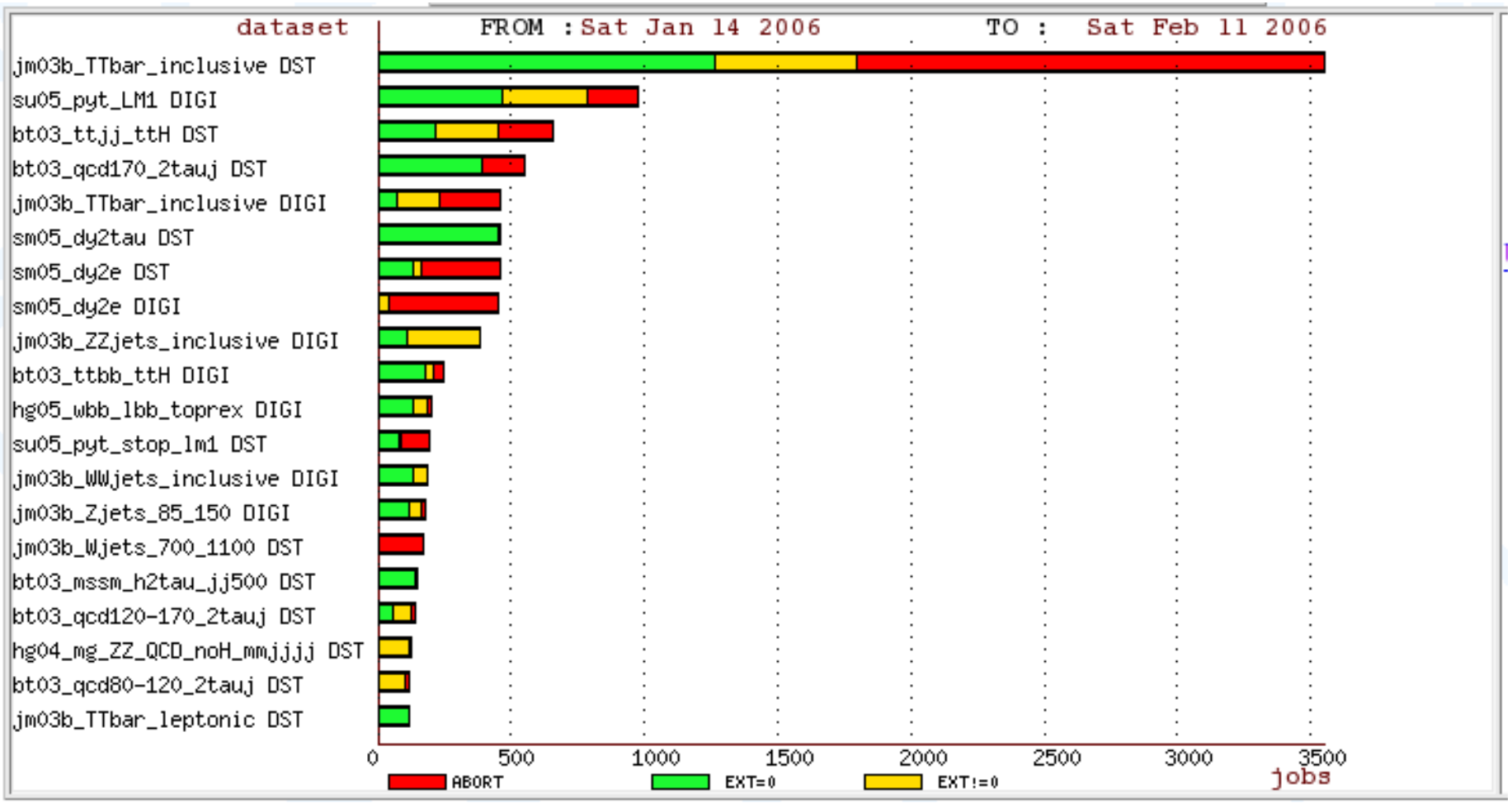
"PhEDEx high-throughput data transfer management system" - J. Rehn (389)

Phedex

Transfer request status					
Request	Destination	Known Files		Destination	
		Files	Size	Files	Size
2006-02-13-hit-gennai	T1_CNAF_MSS	–	–	–	–
2006-02-13-dst-gennai	T1_CNAF_MSS	–	–	–	–
2006-02-13-digi-gennai	T1_CNAF_MSS	–	–	–	–
2006-02-09-SUSYBSM-Charlot	T1_IN2P3_MSS	–	–	–	–
2006-02-09-EGAMMA-Charlot	T1_IN2P3_MSS	1002	102.7 GB	1002	102.7 GB
2006-02-08-LCGharvesting-JoseHernandez	T2_Spain_Buffer	30	21.8 GB	30	21.8 GB
2006-02-08-ForULangenegger-dfeich	T2_CSCS_Buffer	44	83.3 GB	44	83.3 GB
2006-02-06-transfer-test01-stdweird	T2_Belgium_Buffer	4	7.9 GB	–	–
2006-02-05-Pileup-SS	T1_FNAL_MSS	45	83.2 GB	45	83.2 GB
	T2_Caltech_Buffer	45	83.2 GB	–	–
2006-02-01-PUforDESYPROD-Rabbertz	T2_DESY_MSS	46	86.1 GB	46	86.1 GB
2006-02-01-LCGproduction-JoseHernandez	T1_PIC_MSS	46	86.1 GB	46	86.1 GB
	T2_Spain_Buffer	46	86.1 GB	46	86.1 GB
2006-01-27-TTbar-filippidis	T2_Demokritos_Buffer	827	1.5 TB	–	–
2006-01-26-gennai	T1_CNAF_MSS	198	93.0 GB	–	–
2006-01-26-LCGharvest-JoseHernandez	T2_Spain_Buffer	30	16.2 GB	30	16.2 GB

Has potential to really enable in particular the Tier-2 sites early

Dataset Access Patterns



Understand dataset access patterns (analysis) by instrumenting the distributed analysis toolkit (CRAB in CMS)

Dataset/File Access Patterns

BaBar data access monitoring

Basic view

Top performers

List active
- users
- skims
- files
- servers
- clients
- jobs

Query by
- user
- skim
- file
- server
- client
- job

Common queries

Xrootd statistics

Table rows: 5 | Time Period: Last Week | Site: SLAC | Update

Top active users

User Name	Now			Last Week			
	Number of Jobs	Number of Files	File Size [MB]	Number of Jobs	Number of Files	File Size [MB]	MB Read
bbrprod	928	68	24,697	48,214	5,257	1,231,228	1,092,845
bbrskim	463	230	265,327	53,725	4,344	4,453,310	3,324,473
zhangjl	149	17	19,868	3,870	357	288,594	309,885
denardo	132	12	18,547	5,645	543	720,976	7,406
steven	91	12	13,285	5,924	1,187	1,295,120	2,183,717

Hottest skims

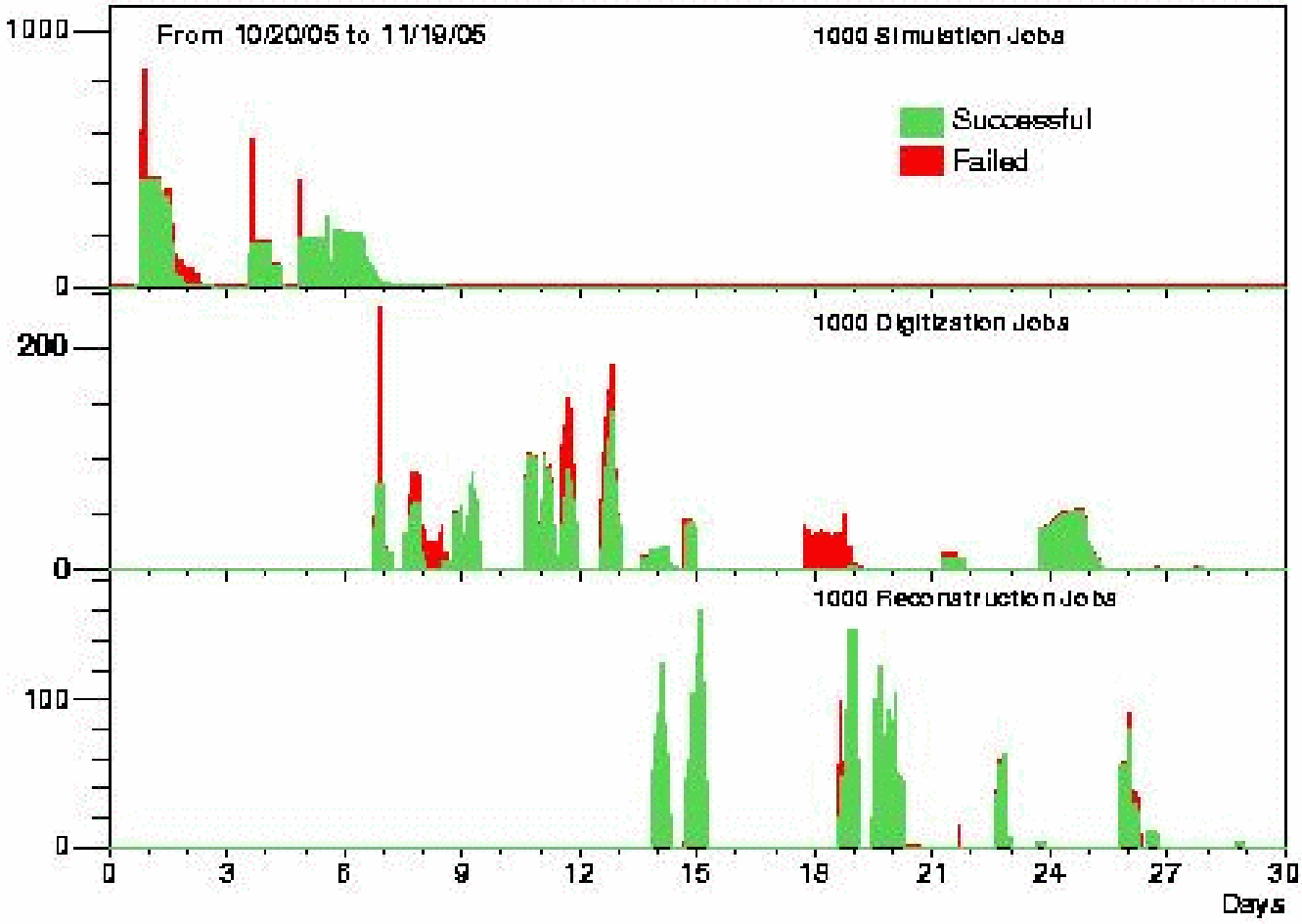
Skim Name	Now				Last Week				
	Number of Jobs	Number of Files	File Size [MB]	Number of Users	Number of Jobs	Number of Files	File Size [MB]	Number of Users	MB Read
AllEvents	287	499	675,757	13	25,714	1,387	2,088,780	21	2,095,487
BToDlnu	286	199	311,984	6	2,404	218	328,183	5	349,171
BSemiExcl	228	253	297,878	13	17,510	2,659	2,825,392	12	4,511,073
BchToDKCabNonCP	144	20	21,716	1	3,879	357	288,594	1	309,957
InclK0s	104	392	546,012	7	15,430	3,665	4,674,952	9	2,550,511

Understand dataset and file access patterns by user by instrumenting data access (xrootd) server (includes I/O information, multi-site)

User produced data

- Want to provide storage, bookkeeping, backup and (eventual) accessibility to the entire collaboration to this data
- Competing interests: people will do what they need to do to get work done, but often (afterwards) realize that they also want more “data management”
- Includes filesize management
- By design much of this will be in the Tier-2's
- Depends heavily on ease-of-use of distributed analysis tools, which often provide the coupling to DM

MC Production (on the Grid)



Some problems here are "grid" related, but many are related to the difficulty of the data management

Can be compensated by massive amounts of manpower, but this isn't where we want to be....

Comment on file sizes

- Larger file sizes still desirable to deal with storage system, catalog and other scaling limitations, however there does not seem to be an obvious, universal solution to push small-file merging into storage systems at all sites
- Filesize management (i.e. merging of files) is a clearly needed part of experiment workflows
- The 2GB limit is a thing of the past, in principle everything should now support >2GB files
- Should be tested this during SC4: (CMS request)
  - Basic test of moving single >2GB files to all sites/systems
  - Larger test of moving datasets of 10GB files, say, to all sites/systems

DM error handling and reporting

- Deserves a talk in itself (but might be really tedious)
- missing files discovered by analysis/production jobs, broken tapes, corrupted files, etc.
- Fault handling:
  - within access/storage system
  - within the applications
  - within Grid workload management system
  - within experiment workflow management system
- Automated reporting needed...

From challenges to "The Challenge"

- Data/service challenges are very useful exercises for testing functionalities and gaining experience, but fundamentally they do not (or have not yet) tested:
  - turning "challenge" systems into 24x7 production ones
  - shocks to the system (plain failures of parts of the system)
  - evolution of requirements, discoveries!?!
- We have also resource limitations and the practical problems of real data (calibrations not ready/understood, new reconstruction version not ready, "blocks of runs" with various calibration and reconstruction problems, plain old bugs, etc.)
- The most "interesting" aspect of any system is how it fails (or at least what happens when it is pushed to its limit)

Conclusions

- Lots of exciting DM-related work in the next 3-5 years
- Lots of reality checks as we transform technologies into real systems
- The DM both shapes and is shaped by the evolving access patterns and analysis models
- By CHEP07 I'm sure we'll see the “first lessons”
- By CHEP09, we'll hopefully actually have some real experience with these systems and have some of the first steps in the “data reduction” scheme down