Data Management at LHC:
some food for thoughts in SuperB

Daniele Bonacorsi
[deputy CMS Computing coordinator – University of Bologna, Italy]

Introduction

LHC is taking (and producing derived) data, the "system" is digesting it all

- Improvement and evolutions are always possible and in some areas actually needed
  - see Claudio's talk
- Anyway, the experiments are handling the data at the required scale

Data management (was and) is a challenge.

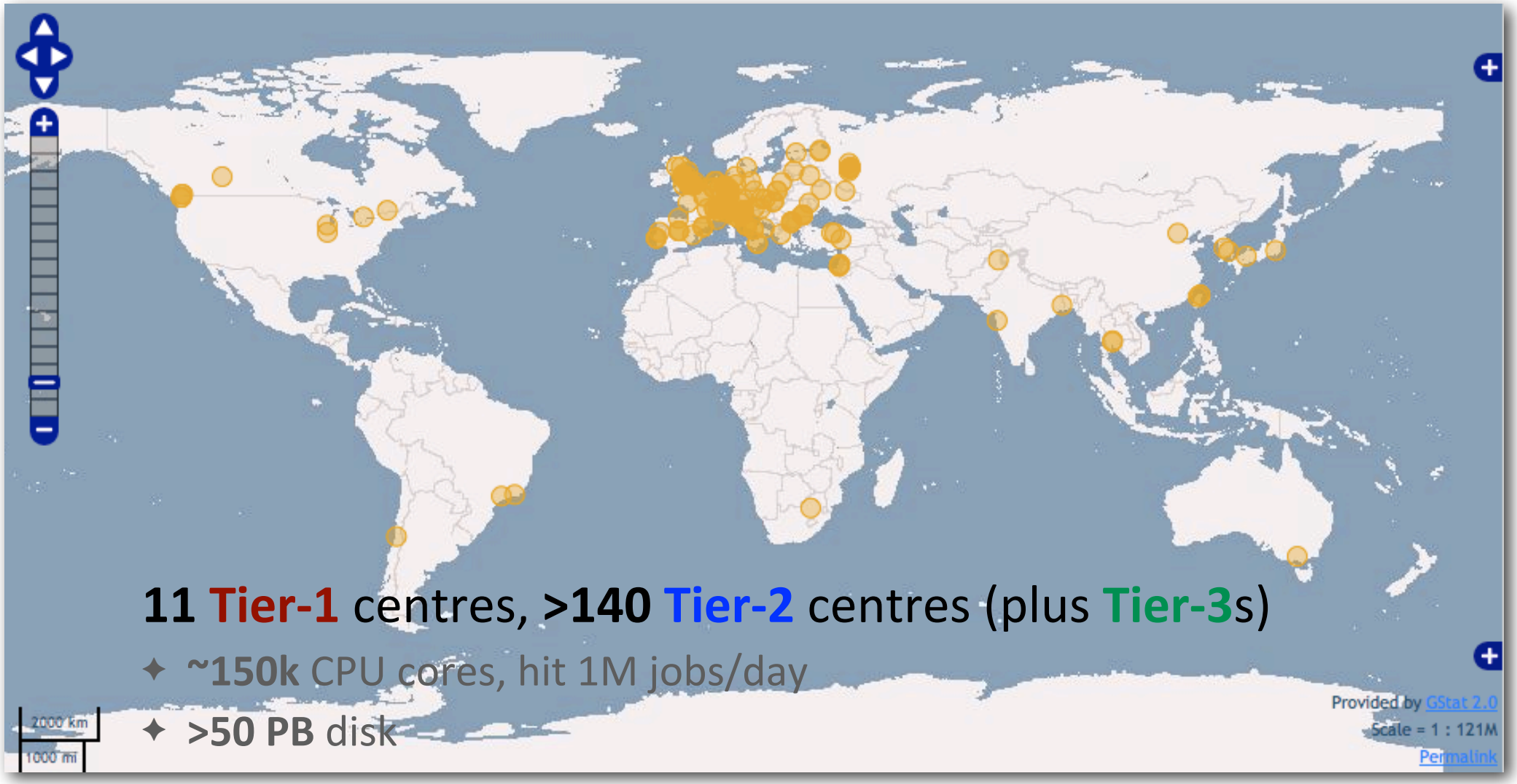
- In the design and implementation phases as well as in the operation phase
- Lots of data + stringent requirements
  - transfer efficiency, access performances, ...
- The LHC computing models differ (despite many somehow common basic principles)
  - Architecture, as well as strategic choices in adoption of middleware, tools, or actual implementation of solutions
- The sites differ as well
  - And their strategic choices, e.g. storage solutions
- Ultimately, data management becomes a storage management challenge

No LHC experiment has THE solution for you (Super-B).

But all experiments might have lesson learned (aka: food for your thoughts)

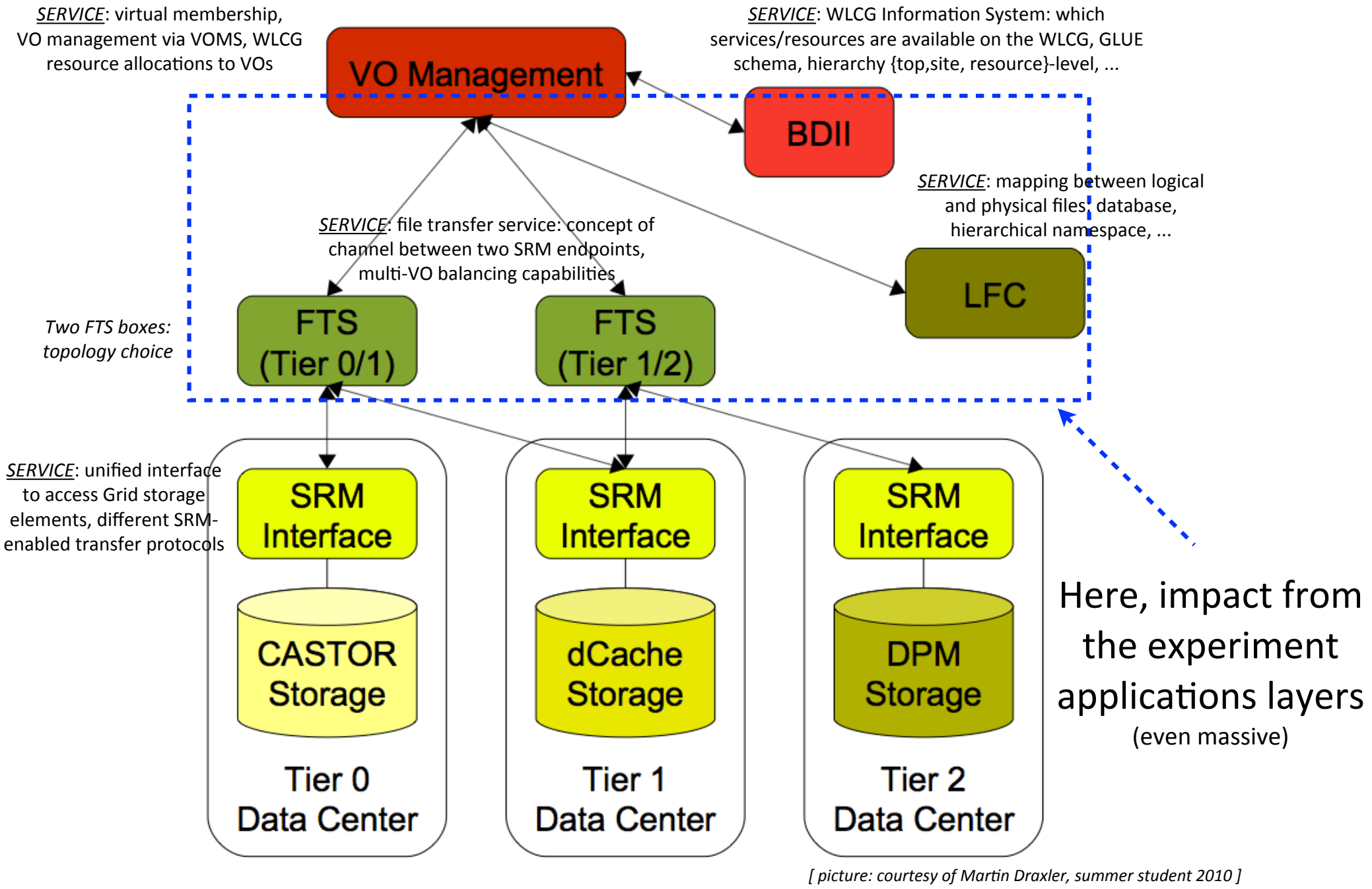
- design/implementation experience _and_ operational experience
- working solutions in production at LHC experiments may be interesting for your evaluation

Ingredients [1/2]



Ingredients [2/2]

CAVEAT: simplified picture.
And gLite middleware only.



Here, impact from the experiment applications layers
(even massive)

[picture: courtesy of Martin Draxler, summer student 2010]

Tapes (and not Disk)

Typically, HEP experiments rely quite heavily on hierarchical MSS

- really large datasets sit only in portions on disk, the rest is on tape
- organized data access is done on large centers with tapes (Tier-{0,1} in MONARC jargon)

The existence of data on tape enforces its custodiality

- "cold" vs "hot" copies as well as actual number of copies, depend on each Computing Model

The use of tape systems evolved over time

- Robots mount the tapes (originally mounted by humans)
- Done upon request, to a relatively large capacity disk buffer, whose management is crucial

"Mount →Read/Write→Unmount" cycles must be minimized

- Large files. Storage classes. Tape families. And many more lessons learned...

Latency for mounting encourages customers to be careful in:

- in designing a system layout
- in placing data on disk and/or tape
- in planning the computing operations

Tape vs. Disk ?

Changes of scale, cost, environmental conditions

- Decrease in the cost of disks and technology to run large disk systems
  - LHC data being accessed from 2011 on could (should) be mostly on disks
- Growing data volume to be archived on tapes (see e.g. CERN, FNAL, BNL)
  - migrate facilities to higher capacity tapes (next step: 5 TB/tape)

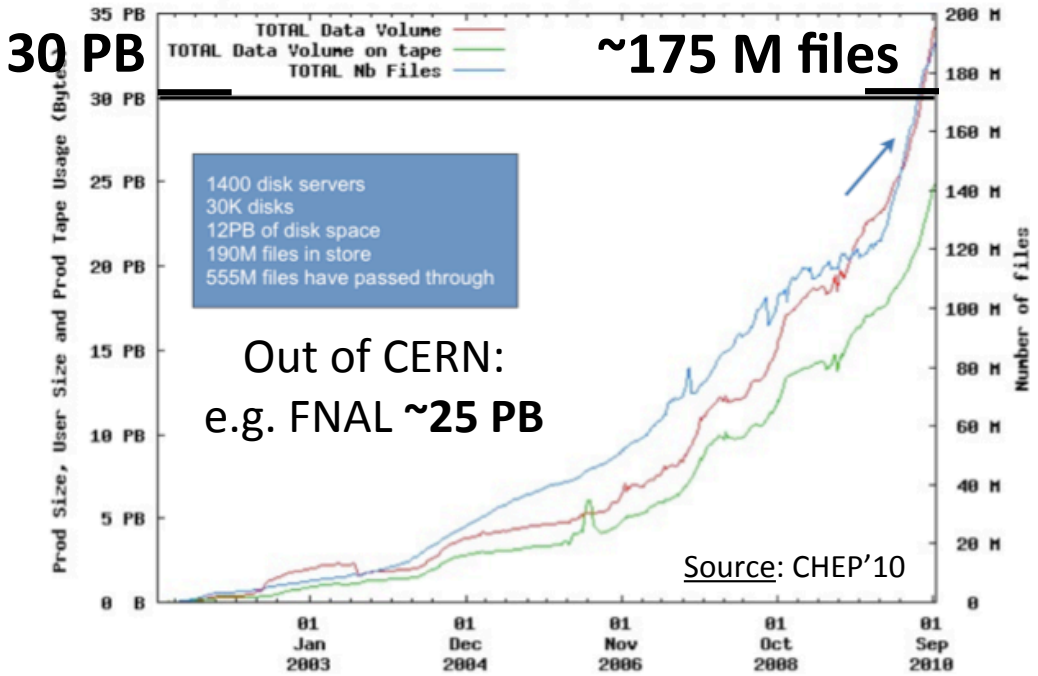
Visible trend in all experiments, and work in progress

- Disk evolves from "~10% caches" to covering a prominent data hosting role
- "Use tapes as tapes" paradigm. Tapes evolves to pure LHC data archives

	ALICE	ATLAS	CMS	LHCb
T0 Disk (TB)	6100	7000	4500	1500
T0 Tape (TB)	6800	12200	21600	2500
T1 Disk (TB)	7900	24800	19500	3500
T1 Tape (TB)	13100	30100	52400	3470
T2 Disk (TB)	6600	37600	19900	20
Disk Total (TB)	20600	69400	43900	5020
Tape Total (TB)	19900	42300	74000	5970

	DZero	CDF
T0 Disk (TB)	~500	~500
T0 Tape (TB)	5900	6600

Credits: Ian Fisk (CHEP'10, MSST'11)



Disk (and not Tape)

Most of LHC analysis is done (physically or logically) far from tape

- T2 (also T{3,4}) centers, with no tapes
- Or T{0,1}, but with care to protect tape systems from uncontrolled accesses

This brings to a more rigid (logical) separation between:

1. where you archive the data
2. where you access the data

At a large extent, disk-only resources are **T2 centers**

- at the end of 2011, >60 TB of T2 disk in LHC (spread over ~140 T2s)
- vary from smaller - $\mathcal{O}$(10s TB) - to larger - $\mathcal{O}$(1) PB
  - then, you have T3s, many, and some are bigger than the average T2...
- growing number of options to manage such (potentially large) disk space
  - dCache, DPM, hadoop, Lustre, GPFS, and more ...

more info in 1 slide

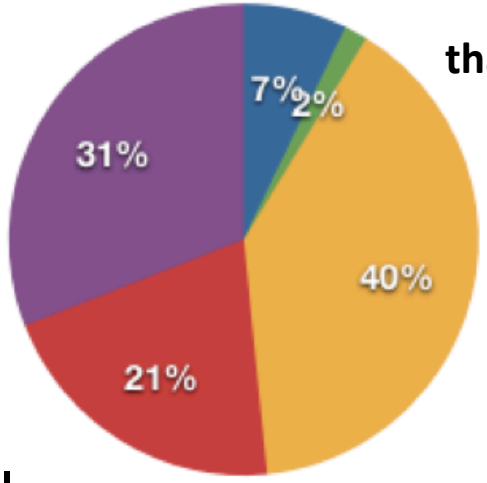
Of course, T2s must be reachable and usable

- At the moment, we still send jobs where the data is
- So data placement is (still) our “step 0”

more info in 2 slides

Storage solutions:

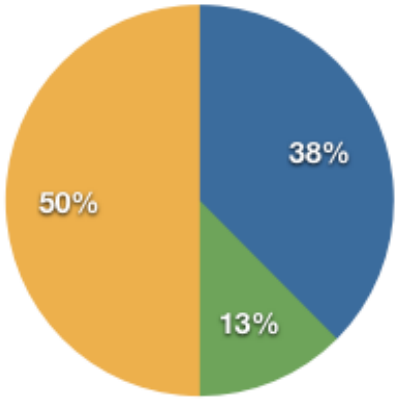
taking the CMS case as an example



Breakdown of storage solution at sites that are nodes in the CMS PhEDEx topology

- # Tiers with Castor
- # Tiers with StoRM/GPFS
- # Tiers with dCache
- # Tiers with DPM
- # Tiers with Disk

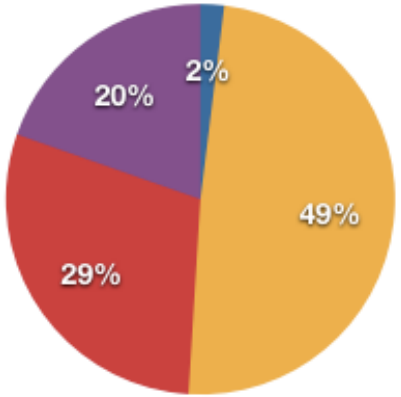
T1 only



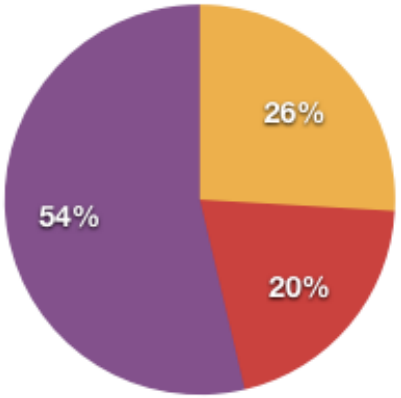
CERN	Castor

ASGC	Castor
CNAF	StoRM/GPFS
FNAL	dCache
IN2P3	dCache
KIT	dCache
PIC	dCache
RAL	Castor

T2 only



T3 only

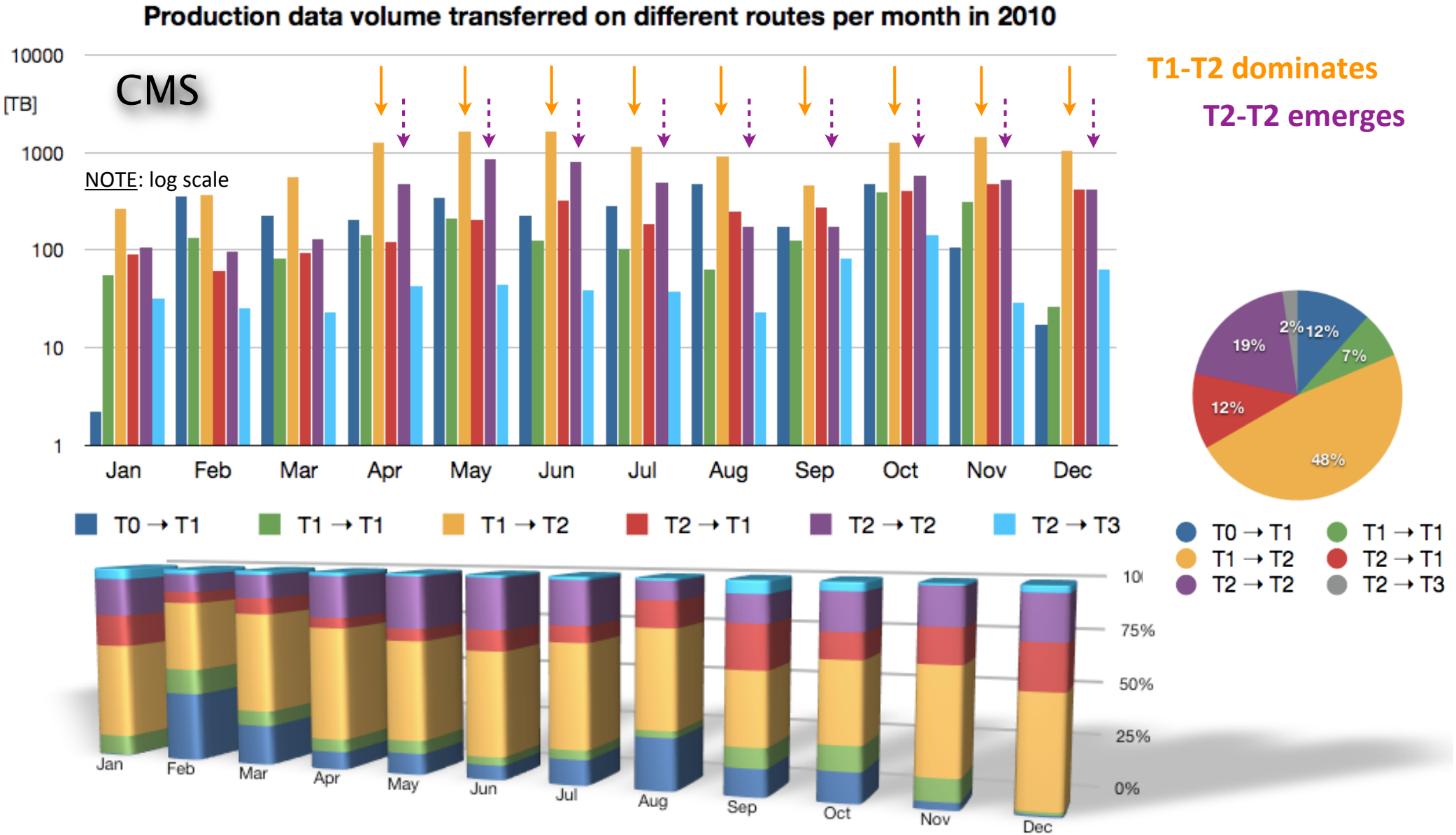


Storage is **not** the same at the same Tier level (true for all experiments)

Any LHC data management system must face this level of heterogeneity

- ✦ strengths and weaknesses might be covered in other talks at this workshop

2010 Tx-Ty traffic breakdown in CMS



ALICE is basing on non-locality of data (see later). ATLAS is relaxing its original "regional clouds" as well (see later). More dynamic data placement systems emerge.

Network

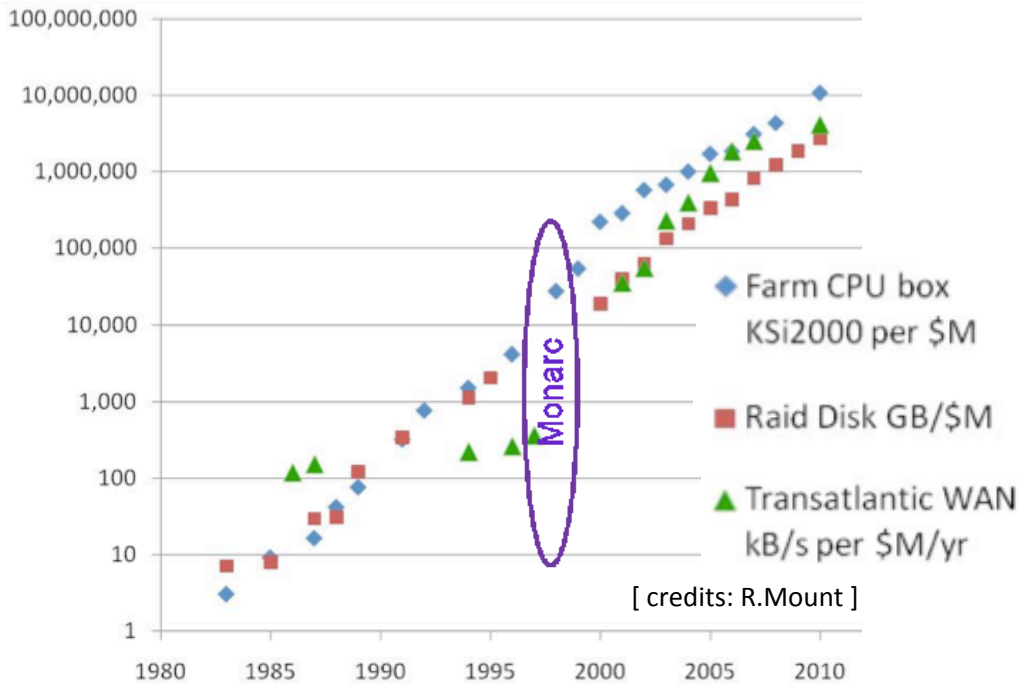
This evolves with both the user needs ***and*** the network capabilities.

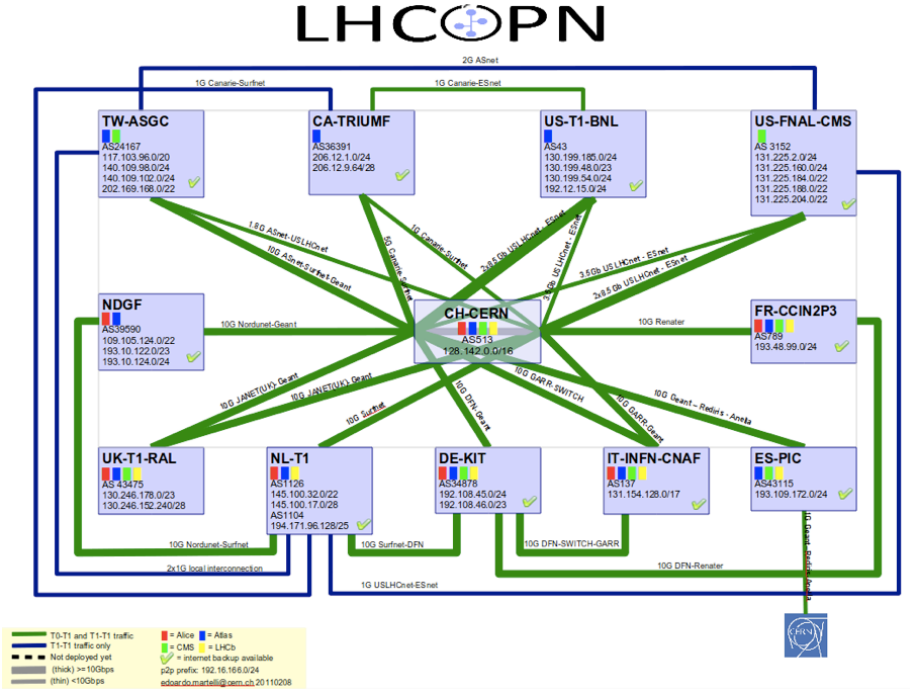
LHC Computing models are based on the MONARC model

- Network perceived as insufficient and/or unreliable
- Data pre-placed. Jobs sent to the data. Multiple copies were felt as better than more transfers

MONARC was developed more than a decade ago

- It served the community remarkably well, evolutions in progress now (e.g. WAN access)





A fully redundant **LHCOPN** for T0-T1 (and T1-T1)

And soon: "**LHCONE**" for T2/3

Some peculiarities by experiment.

ALICE

As from the original model:

- T0: first pass reco, calibration and alignment; it stores 1 copy of RAW, calibration data and first-pass ESDs
- T1: reco and scheduled batch analysis; it stores second collective copy of RAW, 1 copy of all data to be kept, disk replicas of ESDs and AODs, replica of calibration data
- T2: simulation and end-user analysis; it stores disk replicas of AODs and ESDs

Distinction among Tier roles is becoming more shaded, though

- except for reco, *everybody does everything if needed or possible*

Updated ALICE Computing Model parameters

	pp/event	PbPb/event
CPU reco (KHEP06×s)	0.07 (+10%)	9.75 (+71%)
CPU MC (KHEP06×s)	1.30 (+40%)	150.00 (+4%)
Raw size (MB)	1.3 (+18%)	12.5 (0%)
ESD size (MB)	0.08 (+37%)	1.20 (-65%)
MC Raw size (MB)	0.4 (0%)	61.5 (0%)
MC ESD size (MB)	0.26 (0%)	50 (0%)

F. Carminati

ALICE

AliEn as a common front-end for all distributed resources

- Using transparent interfaces to different Grids where needed

Resources are shared

- No "localization" of data. Prioritization of jobs in the Central Task Queue

Data access only through the GRID and AAF (ALICE Analysis facility)

- No backdoor access to data. No "private" processing on shared resources. No "private" resources outside of the Grid

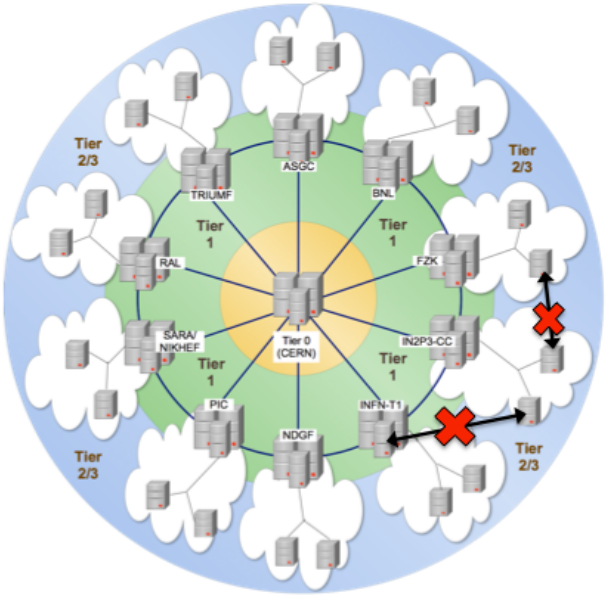
Data Management "centralized"

- Central File Catalogue: central DB of all files produced
- Xrootd as uniform access protocol
  - across sites, storage architectures, use cases
- Central transfer queue for transfers
- ...

ATLAS

Original Computing Model: ~ similar to CMS

- one of major differences: the cloud concept
  - some Tier-Tier data exchanges are (were) forbidden: need a T1
  - MC confined within a cloud



Data Management:

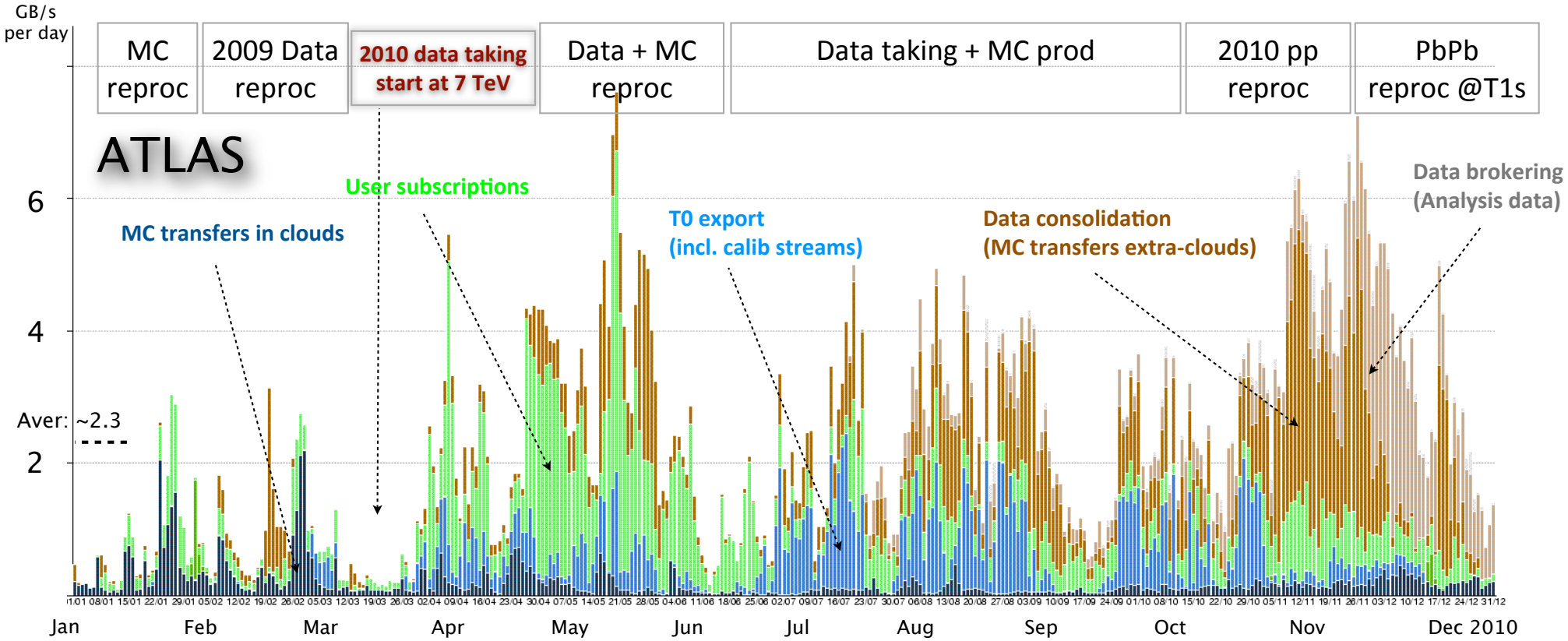
P2DP

- First, PanDA triggers dataset replication to T2s upon access requests at T1s
  - jobs continue to arrive at T1s until a replica is available somewhere else at the T2 level
- Now, extended also to T1s
  - work in progress to go towards a regional-cloud-less model...

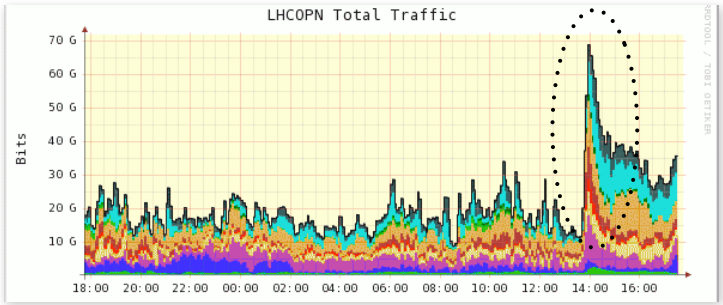
Popularity

- monitoring most popular sites, data types, datasets
- take corrective actions (e.g. data replication)

ATLAS



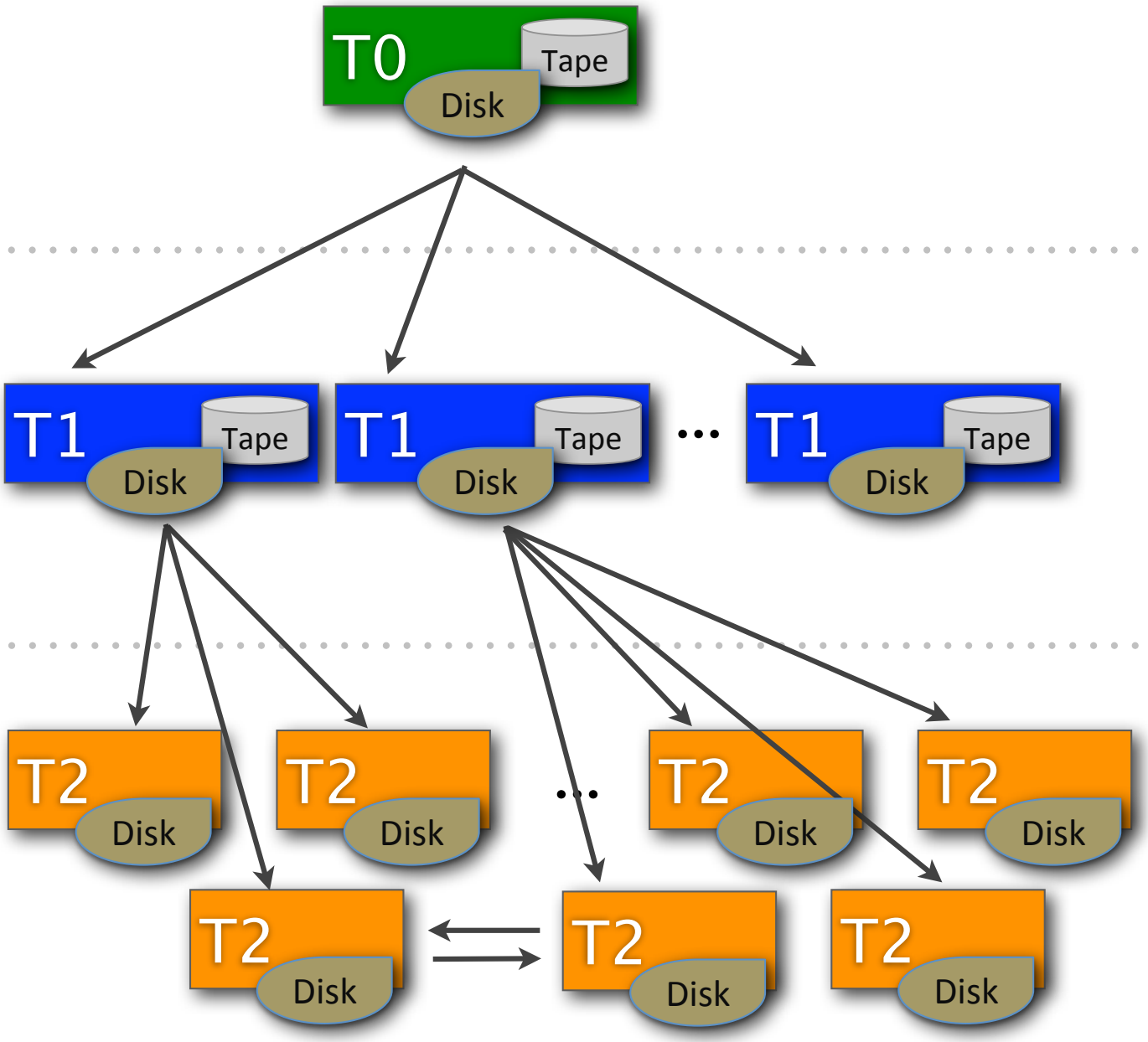
Being ATLAS (and CMS) experiments build on a model that discouraged from relying too much on the network, both use it pretty heavily...



Traffic on OPN measured up to 70 Gbps

- ATLAS massive reprocessing campaigns

CMS



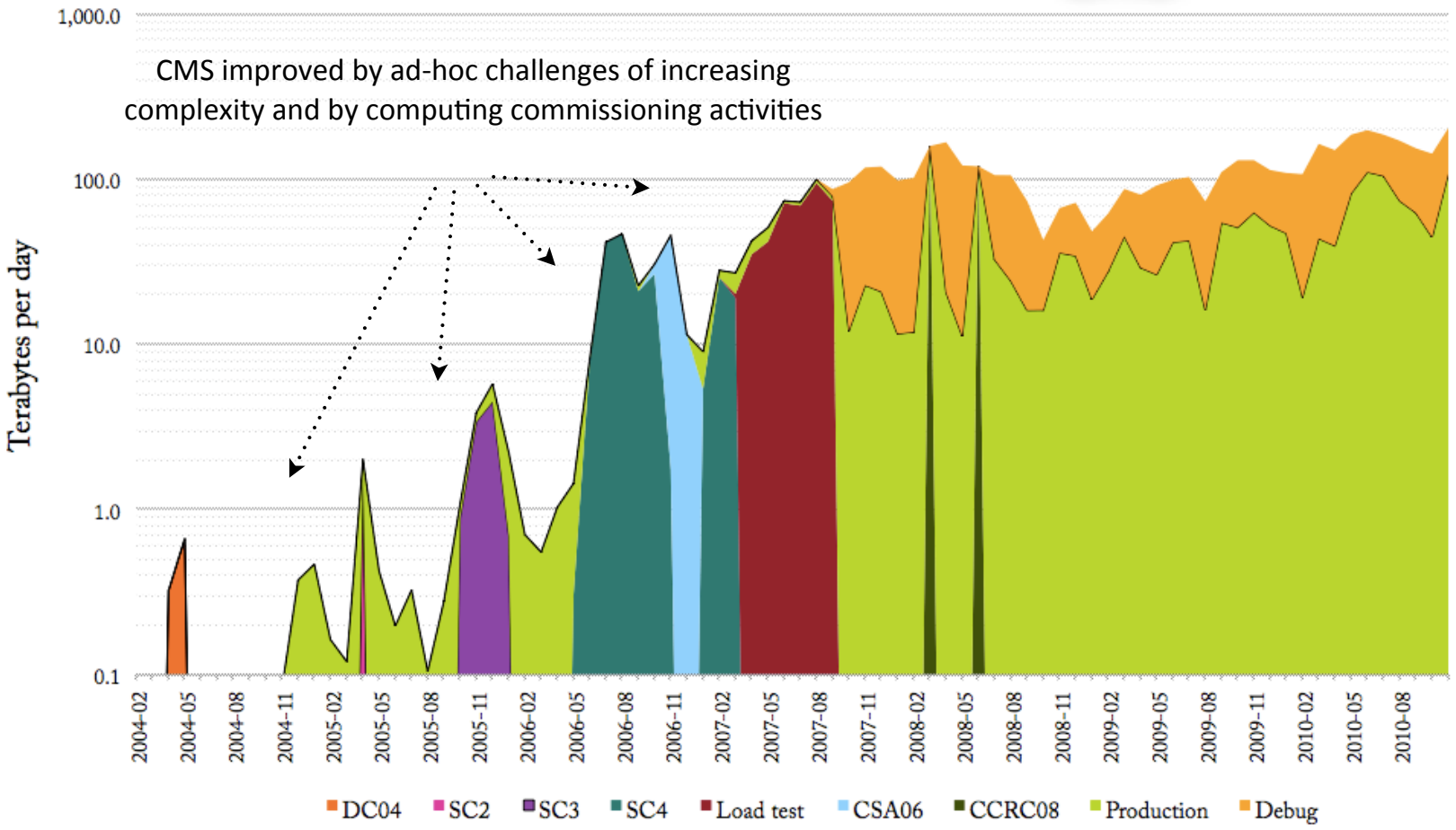
- Prompt processing
- Archival Storage
- Data serving

- Organized processing
- Storage
- Data serving
- Production

- Analysis
- Production

CMS

NOTE: log scale



CMS PhEDEx

PhEDEx is ***not*** a CMS-specific tool: reliable, scalable dataset replication system

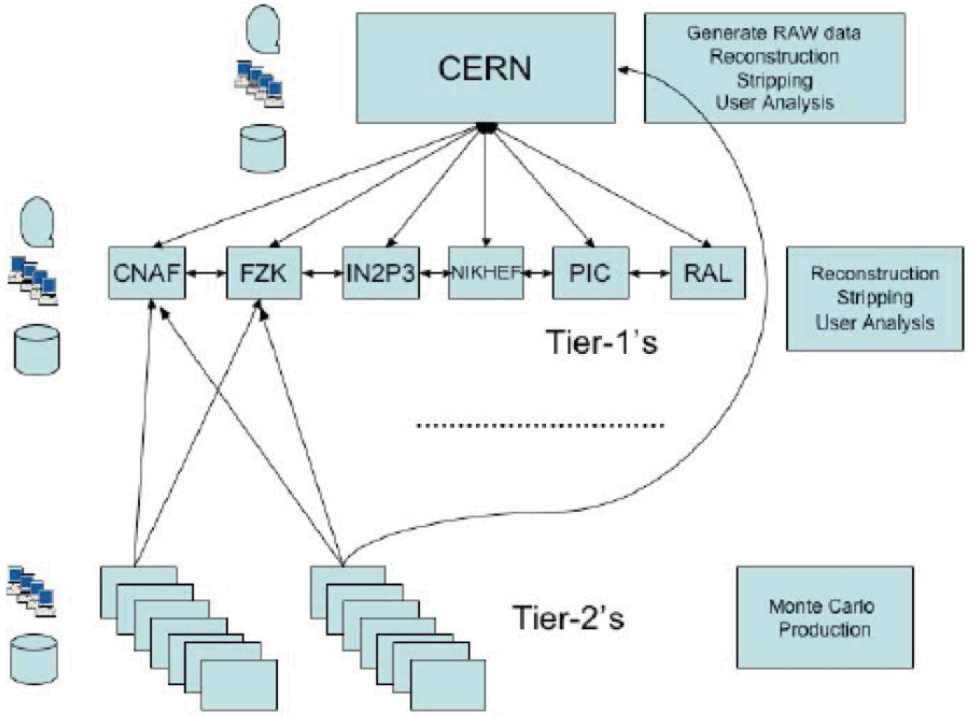
- ✦ It's sustaining up to >200 TB/day of production transfers on the overall topology
- ✦ 100% transfer efficiency, very low transfer latencies, among a complete Tier-{0,1,2,3} transfer topology

If interested, it can be adapted to the needs of other experiment communities

LHCb

Different scale wrt e.g. ATLAS/CMS

- 35 kB RAW, trigger rate 2-3k evts/s
  - 25 kB RDST (aka ESD), 85 kB (aka AOD)
- aggregate rate out of CERN ~40 MB/s
  - in 2010: 155 TB replicated to T1s
- typical RECO time: ~12 HS06s/evt



RECO (first pass) needs T1s also

- 24 kHE06 for reco, typically 2k CPU slots, CERN alone is not enough

Analysis

- most problems come from the data management sector
  - SE accessibility, scalability (load), reliability restrict the # of usable sites
  - use also T1s for analysis

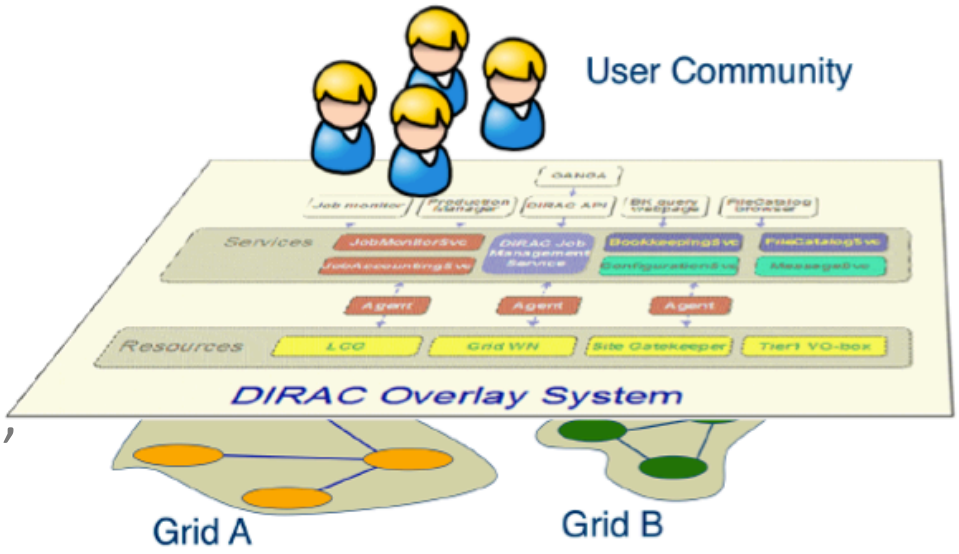
Simulation

- use all possible not-T1 resources for simulation

LHCb

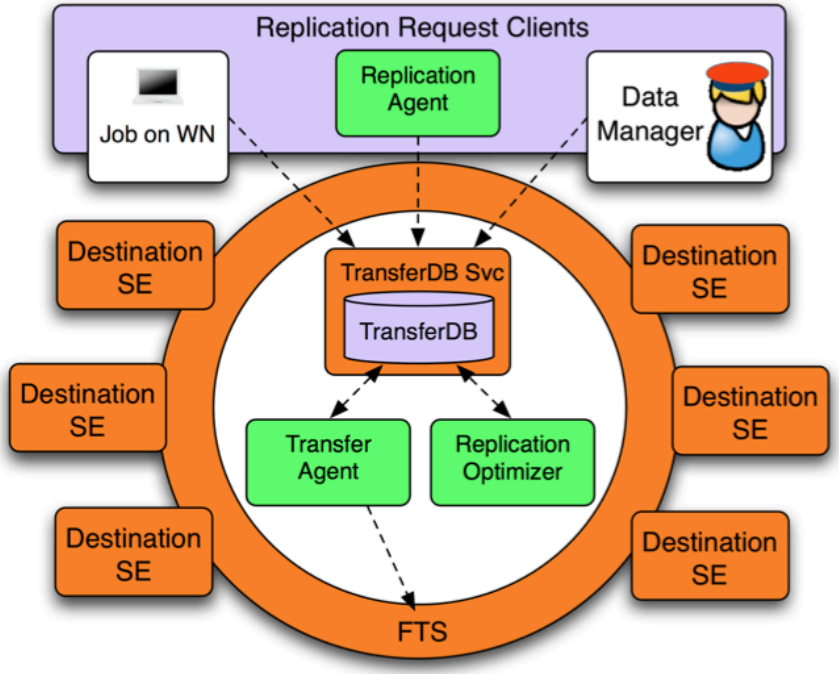
LHCb data management deeply integrated into **DIRAC**

- Services and agents of DIRAC overlay resources
- transparent use of different Grids
- integration of non-Grid resources (local, clouds, batch systems, ...)
- Grid compliant security framework (OpenSSL with X509 certificates)



Bulk data replication

- transfer requests aggregated and centrally managed (TransferDB)
- Transfer Agent polls DB, get bulk requests, submits and monitor via FTS CLI
- Much more...



Some work in progress

[Caveat: some highlights, not a full list of course...]

ALICE

- More files than ever anticipated, mainly MC (25 10^6 files in 2010 catalogue)
- Jobs are becoming more complex and demanding (-> analysis trains)

ATLAS

- T2D concept and transfers inter-clouds
- refine and improve the P2DP model

CMS

- migrate RECO->AOD, scrub AOD size, less AOD replicas, reduce disk needs
- full-mesh done. Now work on WAN access possibilities, data popularity, ...

LHCb

- Reduce the DST replicas (from all T1s to only 4 T1s)
- Increase in disk needs (-> more aggressive clean-up policies)

One word on "evolution"

The evolution in WLCG data management over last years is a story of oscillations between **STRUCTURE** and *flexibility*

Credits: input from (and discussions with) Ian Fisk

- FTS channels added some structure/control to point-2-point transfers
- ALICE remote access goes towards more flexibility
- Systems like CMS PhEDEx added "structured flexibility" on top of FTS
- post-MONARC network models adds flexibility to the original hierarchical model
- full-mesh T2-T2 traffic (CMS) or inter-cloud T2D traffic (ATLAS) add flexibility
- LHCONE sets up a "structure" for the (otherwise unpredictable on GPN) T2 transfer topology

You can find better examples (or disagree on some of these). But what's "evolution"?

ev•o•lu•tion |ˌevəˈlo͞o SH ən|

noun

1 the process by which different kinds of living organisms are thought to have developed and diversified from earlier forms during the history of the earth.

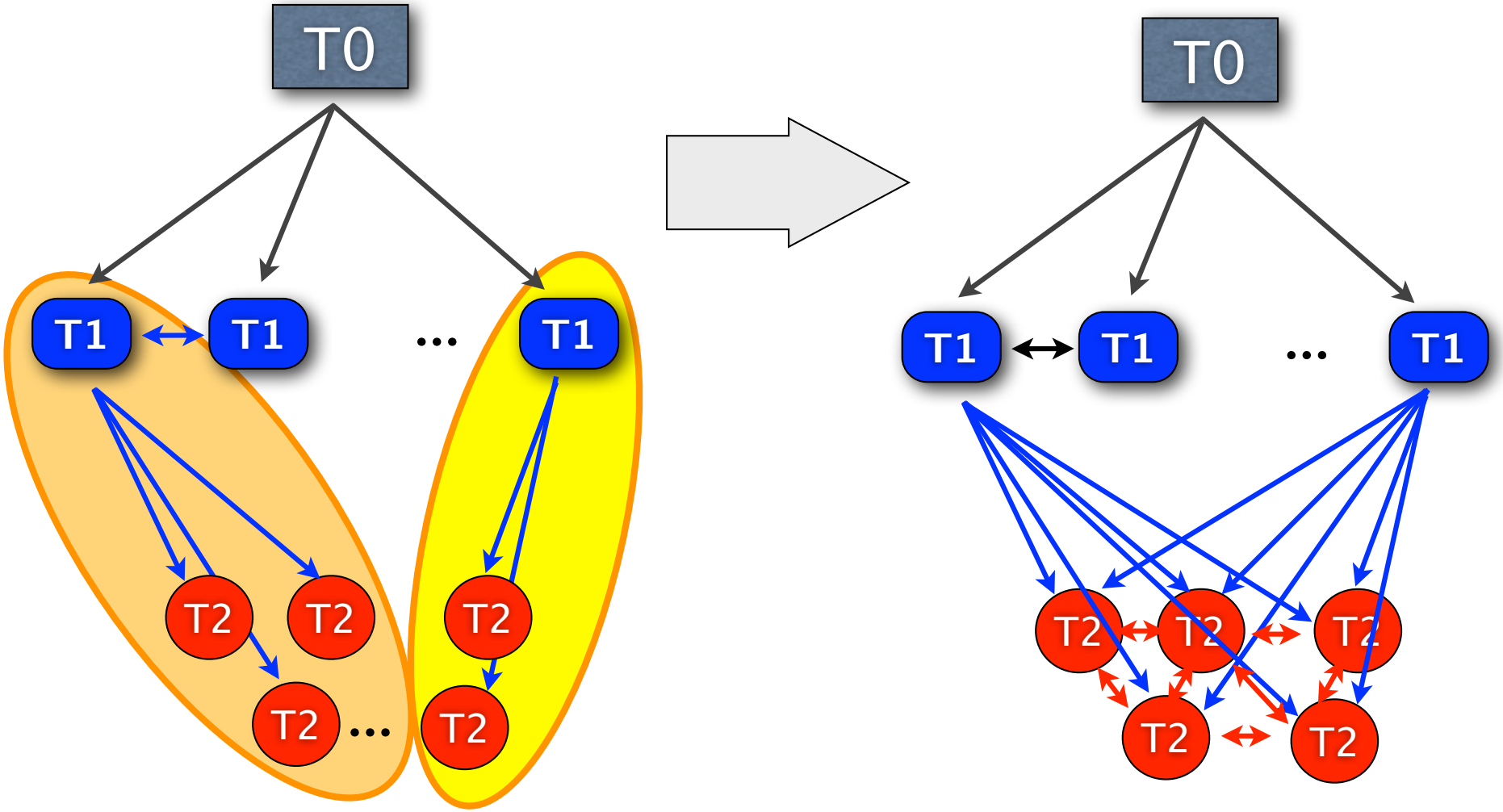
...

ORIGIN early 17th cent.: from Latin ***evolutio(n-) 'unrolling,'*** from the verb ***evolvere*** (see EVOLVE). Early senses related to physical movement, first recorded in describing a tactical "wheeling" maneuver in the realignment of troops or ships. Current senses stem from a notion of "opening out" and "unfolding," giving rise to a general sense of [development.]

It seems to not necessarily mean rushing towards complexity, at least not more than it just implies *doing the right moves towards satisfactory answers to concrete questions*.

One data management “evolution”

Aka “Leave open diagonals for your Bishops”



It’s not done for the sake of flexibility in itself. Or because it’s more “beautiful”.

We measured it could enable us with power to serve analysis groups better.

Outlook

Unrealistic (impossible?) to discuss data storage and management in a distributed environment without also talking about **networking** and **access**

- Sites are not to be treated independently, but as part of a coherent system

LHC experiments can manage their data

- Able to **store** data
- Able to **transfer** data
- Able to **access** data - both for organized and "unpredictable" workflows

Some directions emerge

- Use tapes as tapes. Use disk more but better. Rely now on networks. Etc...

Some work (ok, "evolution"...) in progress

- modification in the access paradigms and the data management could have interesting gains in efficiency

New experiments could get a two-fold input from LHC experiences

- in designing an architecture that incorporates the (most painful) lessons learned
- about existing tools/solutions (easily) exportable and adaptable to new environments