



# PhEDEx and BoD

Use-case, requirements, API...



- Based on <https://twiki.cern.ch/twiki/bin/view/Main/PhEDExAndBoD>
- PhEDEx: data-placement for CMS
  - T0 -> T1: custodial data
    - Primary use-case for investigation/prototyping
  - T2 -> T1: harvest MC production
  - T1->T2, T2->T2: placement for analysis
  - #nodes, time-profile, concurrency vary considerably
- First version released in 2004
  - A time when the network expected to be the bottleneck
  - *Assume* network would fail, use robust backoff, probe, retry...
  - Now, network is *most* reliable component (c.f. storage, MSS, people)
  - => time to change the model?



- Three instances of PhEDEx, Prod/Dev/Debug
  - Each has own set of agents (central mgmt, per-site)
  - Up to 12 TB transfers queued per (src,dst) pair
    - Central agents maintain queues, site agents pull queue and report back on progress



Average rate last year	Production	Debug	Total
T0 -> T1	230 MB/sec	100 MB/sec	330 MB/sec
T2 -> T1	190	200	390
T1 -> T2	620	230	850
T2 -> T2	260	180	440

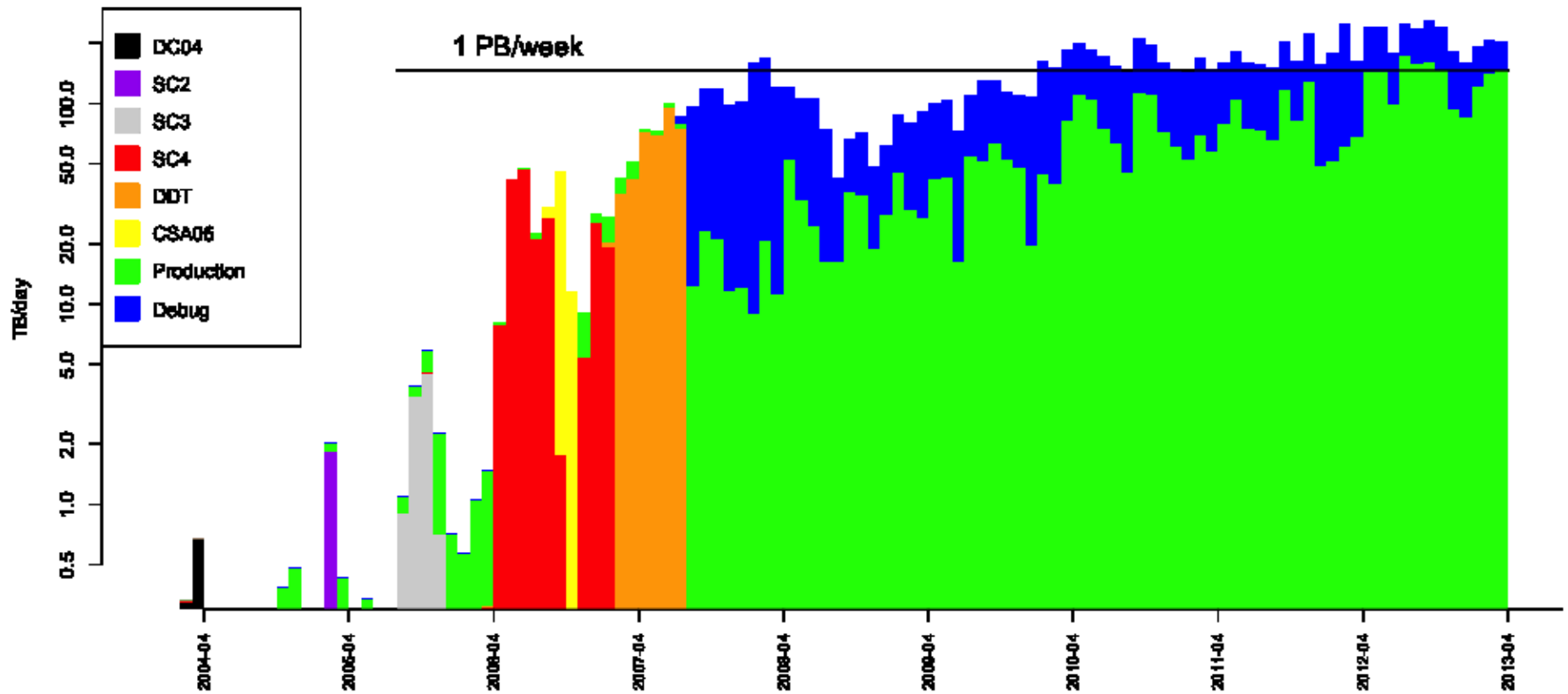
Production instance is real data

Debug instance is for commissioning and link-tests

- separate instances => separate sets of PhEDEx agents.

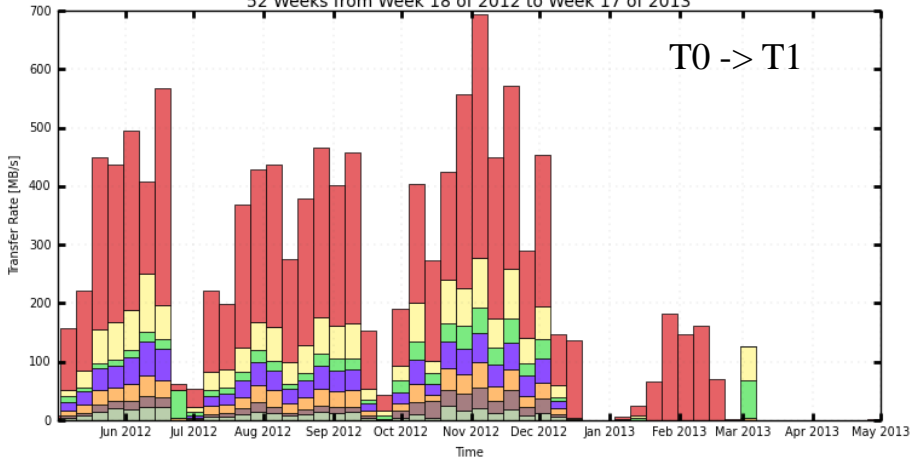
Why so much traffic in Debug? I don't know...

Average rate ~ 2 TB/sec CMS-wide, sustained over last 3 years



### CMS PhEDEx - Transfer Rate

52 Weeks from Week 18 of 2012 to Week 17 of 2013



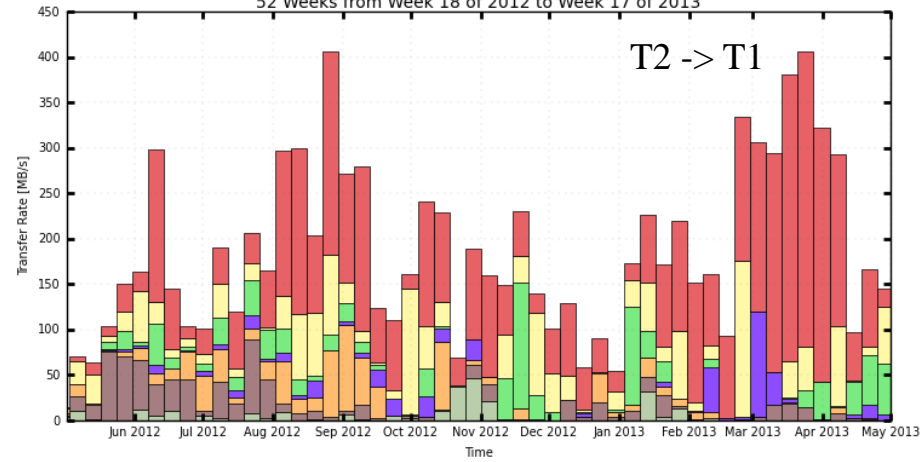
T0 -> T1

- T1\_US\_FNAL\_Buffer
- T1\_IT\_CNANF\_Buffer
- T1\_DE\_KIT\_Buffer
- T1\_UK\_RAL\_Buffer
- T1\_FR\_CCIN2P3\_Buffer
- T1\_TW\_ASGC\_Buffer
- T1\_ES\_PIC\_Buffer

Maximum: 693.13 MB/s, Minimum: 0.00 MB/s, Average: 227.79 MB/s, Current: 1.73 MB/s

### CMS PhEDEx - Transfer Rate

52 Weeks from Week 18 of 2012 to Week 17 of 2013



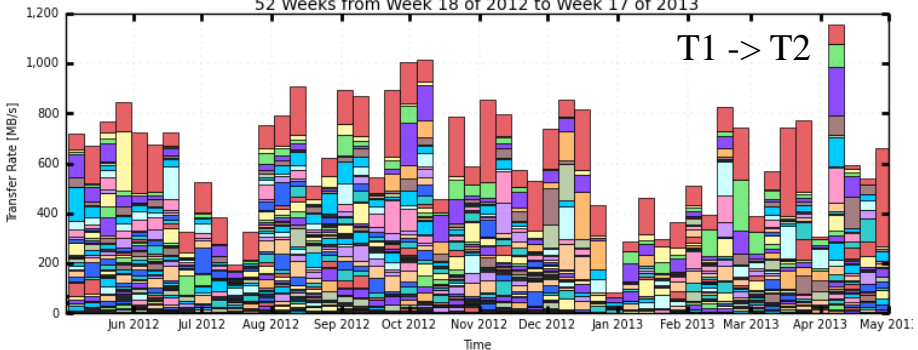
T2 -> T1

- T1\_US\_FNAL\_Buffer
- T1\_IT\_CNANF\_Buffer
- T1\_DE\_KIT\_Buffer
- T1\_ES\_PIC\_Buffer
- T1\_UK\_RAL\_Buffer
- T1\_FR\_CCIN2P3\_Buffer
- T1\_TW\_ASGC\_Buffer

Maximum: 406.43 MB/s, Minimum: 15.06 MB/s, Average: 185.57 MB/s, Current: 144.50 MB/s

### CMS PhEDEx - Transfer Rate

52 Weeks from Week 18 of 2012 to Week 17 of 2013



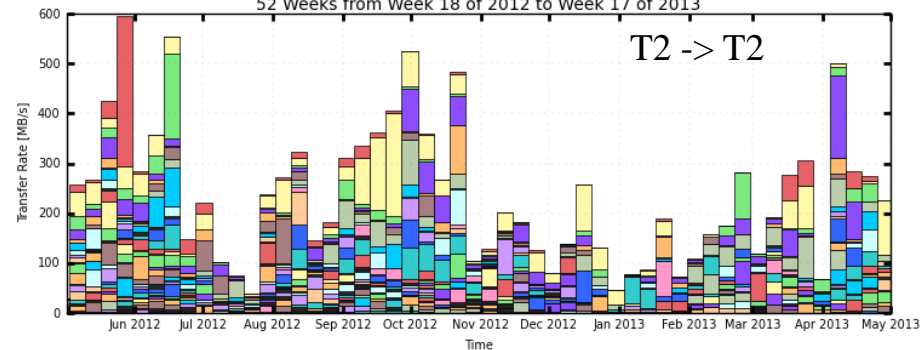
T1 -> T2

- T2\_CH\_CERN
- T2\_US\_Purdue
- T2\_IT\_Rome
- T2\_US\_Nebraska
- T2\_DE\_RWTH
- T2\_US\_Florida
- T2\_FR\_METU
- T2\_FR\_CCIN2P3
- T2\_RU\_IHEP
- T2\_RU\_INR
- T2\_BE\_IHHE
- T2\_KR\_KNU
- T2\_FR\_GRIF\_IRFU
- T2\_US\_Nebraska
- T2\_DE\_RWTH
- T2\_US\_Florida
- T2\_FR\_METU
- T2\_FR\_CCIN2P3
- T2\_RU\_IHEP
- T2\_RU\_INR
- T2\_US\_UCSD
- T2\_US\_Wisconsin
- T2\_FR\_GRIF\_LLRC
- T2\_US\_Caltech
- T2\_IT\_Legnano
- T2\_AT\_Vienna
- T2\_CN\_Beijing
- T2\_UA\_KIPT
- T2\_HU\_Budapest
- T2\_RU\_ITEP
- T2\_DE\_DESY
- T2\_US\_MIT
- T2\_UK\_London\_IC
- T2\_UK\_London\_Brunel
- T2\_PL\_Warsaw
- T2\_IN\_TIFR
- T2\_RU\_SINP
- T2\_BR\_UERJ
- T2\_UK\_SGrid\_Bristol
- T2\_GR\_Ioannina
- T2\_US\_Vanderbilt
- T2\_UK\_London\_IC
- T2\_US\_Florida
- T2\_RU\_JINR
- T2\_UK\_SGrid\_RALPP
- T2\_BE\_UCL
- T2\_CH\_CSCS
- T2\_PT\_NCG\_Lisbon
- T2\_US\_Purdue
- T2\_FR\_GRIF\_LLRC
- T2\_US\_Caltech
- T2\_KR\_KNU
- T2\_FR\_CCIN2P3
- T2\_AT\_Vienna
- T2\_US\_Vanderbilt
- T2\_HU\_Budapest
- T2\_US\_UCSD
- T2\_IT\_Legnano
- T2\_DE\_RWTH
- T2\_RU\_SINP
- T2\_FR\_IPHC
- T2\_ES\_CEMAT
- T2\_CN\_Beijing
- T2\_PT\_NCG\_Lisbon
- T2\_US\_Vanderbilt
- T2\_UK\_SGrid\_Bristol
- T2\_US\_UCSD
- T2\_IT\_Legnano
- T2\_DE\_RWTH
- T2\_RU\_SINP
- T2\_FR\_IPHC
- T2\_ES\_CEMAT
- T2\_CN\_Beijing
- T2\_PT\_NCG\_Lisbon
- T2\_US\_Vanderbilt
- T2\_UK\_SGrid\_Bristol

Maximum: 1,156 MB/s, Minimum: 71.65 MB/s, Average: 620.71 MB/s, Current: 623.83 MB/s

### CMS PhEDEx - Transfer Rate

52 Weeks from Week 18 of 2012 to Week 17 of 2013



T2 -> T2

- T2\_CH\_CERN
- T2\_US\_Wisconsin
- T2\_UK\_London\_IC
- T2\_US\_Florida
- T2\_RU\_JINR
- T2\_UK\_SGrid\_RALPP
- T2\_BE\_UCL
- T2\_CH\_CSCS
- T2\_PT\_NCG\_Lisbon
- T2\_US\_Purdue
- T2\_FR\_GRIF\_LLRC
- T2\_US\_Caltech
- T2\_KR\_KNU
- T2\_FR\_CCIN2P3
- T2\_AT\_Vienna
- T2\_US\_Vanderbilt
- T2\_HU\_Budapest
- T2\_US\_UCSD
- T2\_IT\_Legnano
- T2\_DE\_RWTH
- T2\_RU\_SINP
- T2\_FR\_IPHC
- T2\_ES\_CEMAT
- T2\_CN\_Beijing
- T2\_PT\_NCG\_Lisbon
- T2\_US\_Vanderbilt
- T2\_UK\_SGrid\_Bristol
- T2\_US\_UCSD
- T2\_IT\_Legnano
- T2\_DE\_RWTH
- T2\_RU\_SINP
- T2\_FR\_IPHC
- T2\_ES\_CEMAT
- T2\_CN\_Beijing
- T2\_PT\_NCG\_Lisbon
- T2\_US\_Vanderbilt
- T2\_UK\_SGrid\_Bristol

Maximum: 596.80 MB/s, Minimum: 20.05 MB/s, Average: 222.12 MB/s, Current: 226.47 MB/s

## Transfers by destination, last 52 weeks, Production instance



- Initial use-case: T0 -> T1 transfers
  - Rates, profiles, #links more stable
  - Easy to model (e.g. ANSE PhEDEx testbed)
  - Delays@T0 -> bigger margins/buffers, less sleep
- T2 -> T1? MC upload, perhaps less important
- T1->T2 & T2 -> T2?
  - Analysis flows, physicists waiting for data!
  - More determinism here would be well received
  - Much harder to understand/model
  - What metrics to use to measure success?
    - Impact on data flow  $\neq$  impact on analysis
  - Not considered: AAA, popularity svc, JIT placement



- 4 places to couple PhEDEx to BoD
  - Per-transfer (e.g. FDT) -> not really interesting
  - Per-link (FileDownload agent) useful for T0!
  - Per-instance (FileRouter agent)
  - CMS-wide, across all three PhEDEx instances
- Circuits managed *by* or *for* PhEDEx => I don't care
  - PhEDEx provides hints or requests to a service, can react to response or notification that a link is oversubscribed or saturated
- No circuit? Continue anyway on GPN (1<sup>st</sup> order)
  - Circuits can augment throughput, not change workflow
  - Creation, teardown, failure of circuit transparent to PhEDEx. Ongoing transfers may fail, but PhEDEx will retry as always
  - *Expect* circuit failure/removal or circuit reservation failure as 'normal' business, or BoD does not belong in PhEDEx stack





- Units: (TB, hours), not (GB, minutes)
- Basic requirement: use a circuit if doing so will significantly improve average performance on this scale
  - Implies a whole bunch of monitoring & feedback
- Budget? Be able to cope with refusal to create a circuit
  - Higher priority requests from other CMS activities?
  - Saturation of a VO share on a link?
  - Fair-use policies averaged over time?
  - Max number of allowed circuits reached? Per time-interval?
- CMS must be able to prioritise circuit requests
  - Higher-priority request displaces existing lower priority circuits?



- What to ask for...?
  - Minimum bandwidth:
    - PhEDEx maintains its own performance history. If a circuit can't improve on that, don't create one.
  - Maximum bandwidth:
    - Don't exceed what I can do to/from disk
    - Don't exceed output capacity of T0...
    - Want to leave capacity for other usage/users
  - Min/max data-volume
    - Choose bandwidth x duration to fall into this range
    - Below this I don't want to pay the cost. Above it I cannot keep the pipe full, I don't have enough work in my queue (yet?)



- Priority?
  - Allow eviction of existing circuits of lower priority
  - Eviction implies ownership – don't evict circuits belonging to other entities
    - Even other entities within CMS? Implies fine-grained authorisations, probably managed within CMS
- Circuit identity?
  - Allows requests like: replace this circuit with a new one with different specification if you can, but keep the existing circuit if you cannot.
  - Useful when my work queue gets extended before I'm finished processing it (this may be the norm)
- Start time?
  - Are bookings JIT, or in advance? PhEDEx may know hours in advance, or can adapt if booking not possible for several hours



- What to provide...?
  - newCircuitID = request\_circuit(minBW, maxBW, minGB, maxGB, priority, me, circuitID)?
- State information
  - get\_my\_circuits(me)
    - Needed if I lose state after process/machine restart
  - get\_circuit\_state(me,circuitID)
    - Find out if I am using my circuit(s) efficiently
  - get\_global\_state(me)
    - Can I ask for more bandwidth/data-volume?
    - ‘me’ => restrict to my VO
- API or RESTful service? Prefer the latter



- Summary:
  - No hard requirements (yet!)
    - Need to learn what's feasible/useable, operations models etc
  - PhEDEx has 3 use cases with different features
    - #circuits, topologies, time-evolution
    - Scales: hours, TB, nothing smaller
    - Start with T0 -> T1s
    - Ultimate goal is to support analysis flows too
  - RESTful service
    - Augment existing capabilities with circuits
    - Expect occasional failure or refusal from service
    - Need priority (& ownership?)
    - Budget/share-management? Who provides that?