

CMS status and 2011 plans

Efficiency

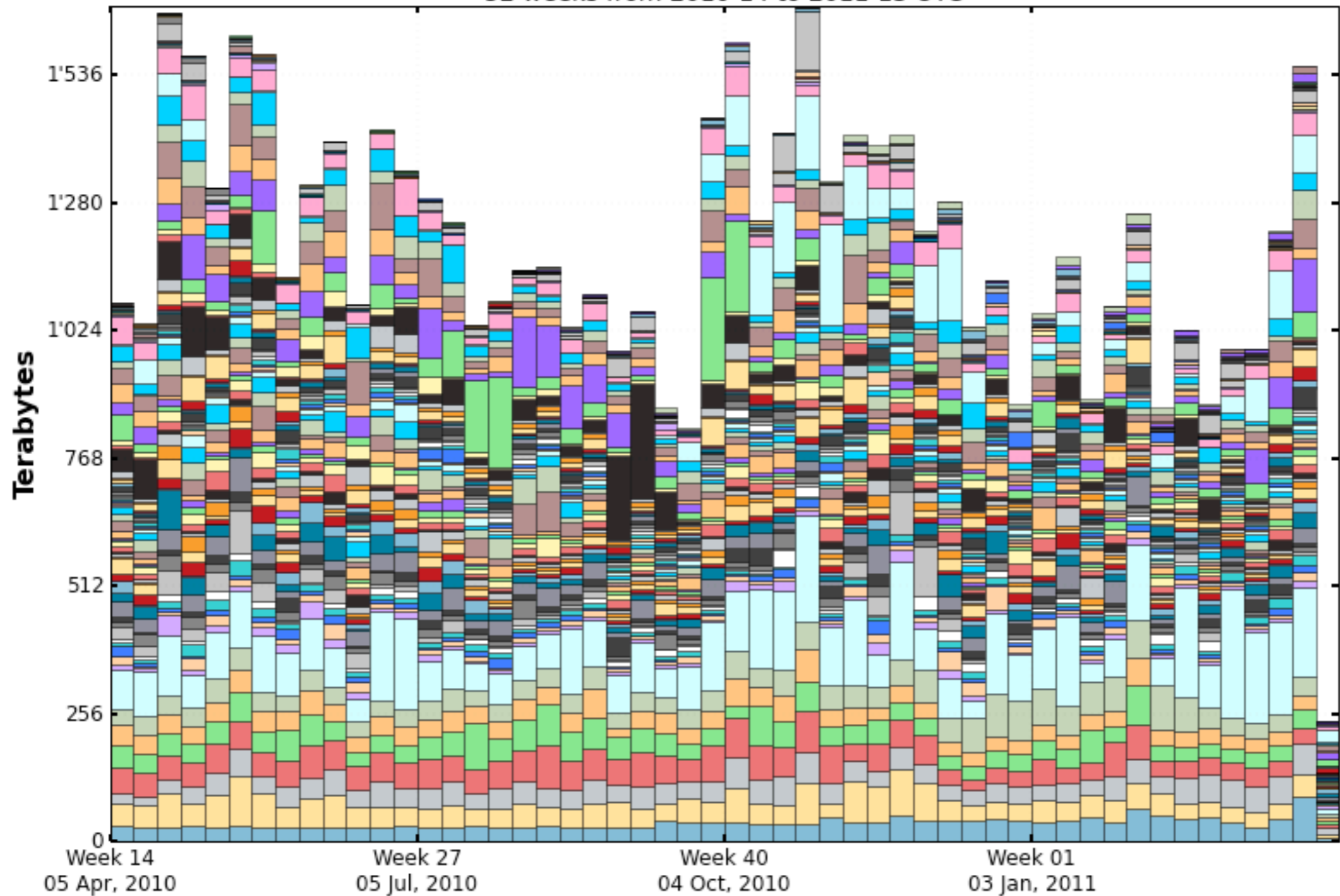
2010 Run

- Generally very successful
- CMS computing infrastructure stood up to first data well
- Have sacrificed resources to get things done
- e.g. lots of copies of data, lots of reprocessing, lots of stressed people

Data

Weekly CMS PhEDEx transfer volume, Debug + Production

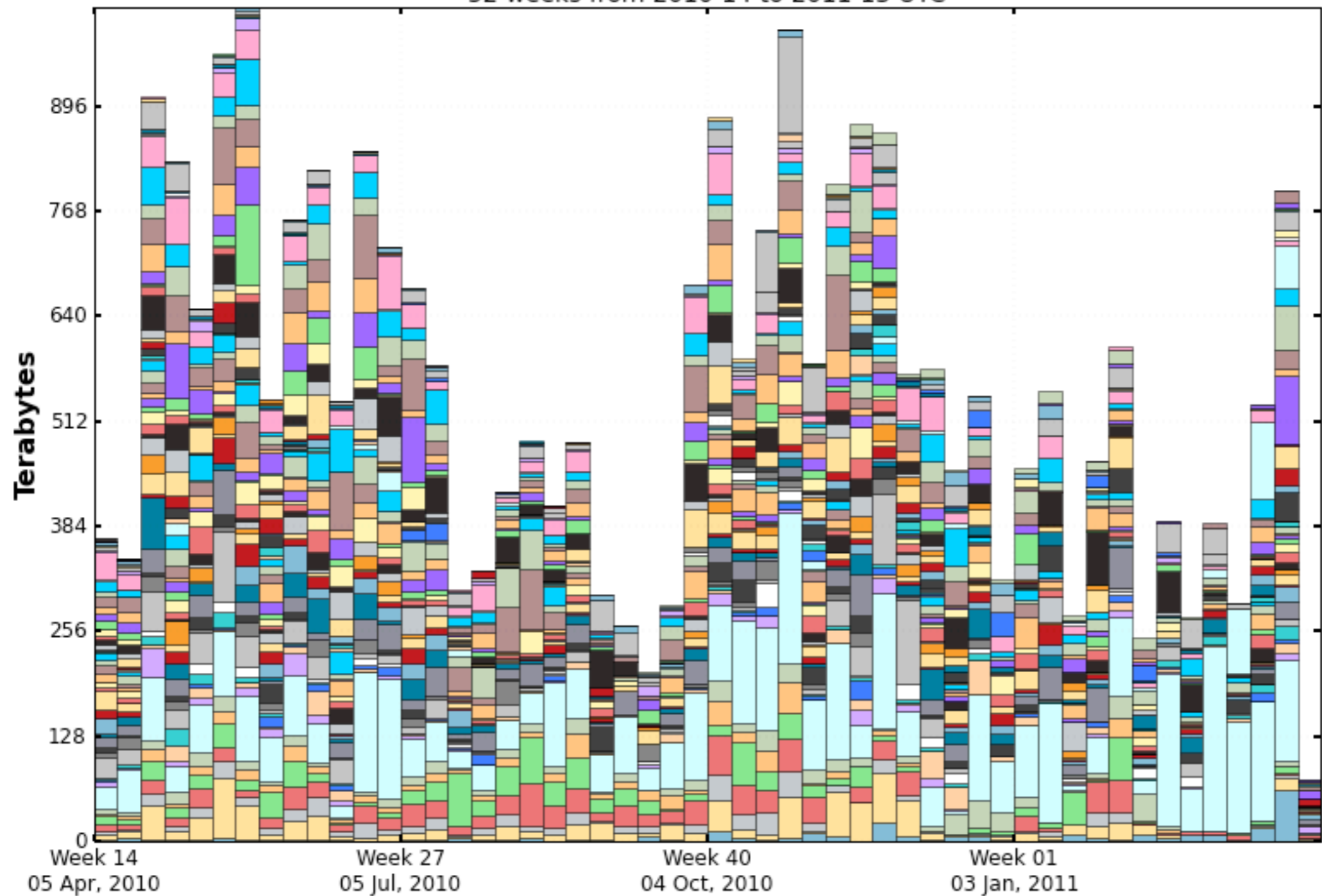
By destination storage node for non-tape storage only
52 weeks from 2010-14 to 2011-13 UTC



Data

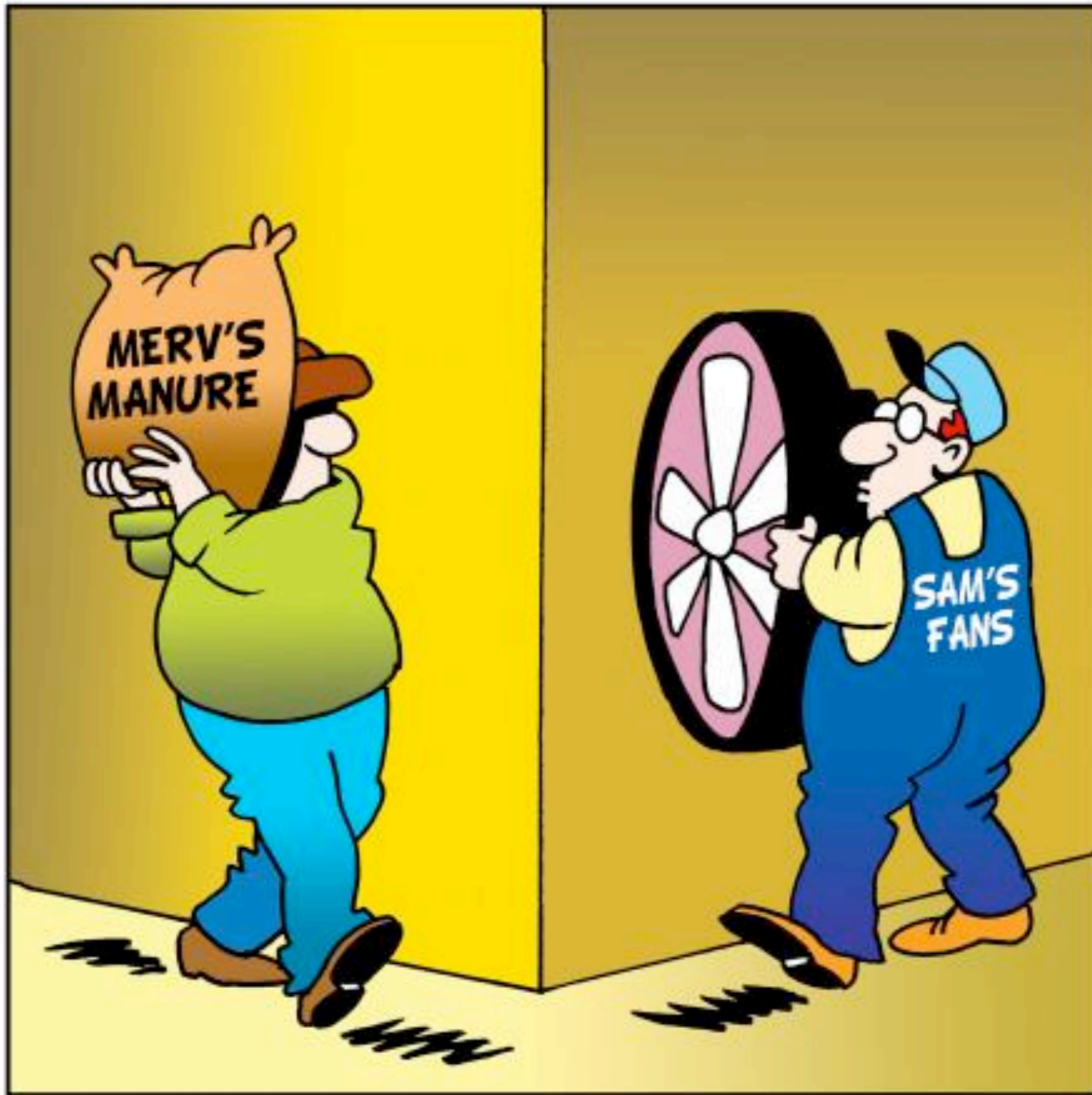
Weekly CMS PhEDEx transfer volume, Production

By destination storage node for non-tape storage only
52 weeks from 2010-14 to 2011-13 UTC



2011 Run

- Will start to be resource constrained



Where we are inefficient

- User job stage out
- Uncoordinated tape reads
- Meta data look up
- Quality control of middleware, experiment infrastructure & experiment software
- Perceptions of Ops time, cost of Infrastructure

User job stage out

- Synchronous stage out from WN results in:
 - Locking WN while staging out data - potentially spend more time on WAN IO than CPU
 - Transfer over untested/inefficient/unmanaged links
 - DDOS of sites
- Solution is Asynchronous Stage Out
 - Write to local SE, transfer via FTS
 - Should reuse tested infrastructure

Tape reads

- This issue is partially a T I configuration issue
 - Some sites suffer it more than others
 - RAL has so far avoided it by throwing resource (disk/people) at the problem
- Commissioning a StageManager tool to provide automated tape recall to DataOps
 - Developed by James/Simon/Manny

The implementation



DataOps

Request script/web interface (POST)

Operator interface enters a request (at block or dataset level) into the system for one or more T1 sites.

Currently only a command line tool. No request approval or long term tracking - this is not a user tool.

This can be reused as an API from the WMSystem - e.g. after a ReReco request is approved trigger a stage.



T1 Firewall

Replicate: PULL
in work

Replicate: PUSH back
results



Site Agent

Query

Stage Request

Status Request



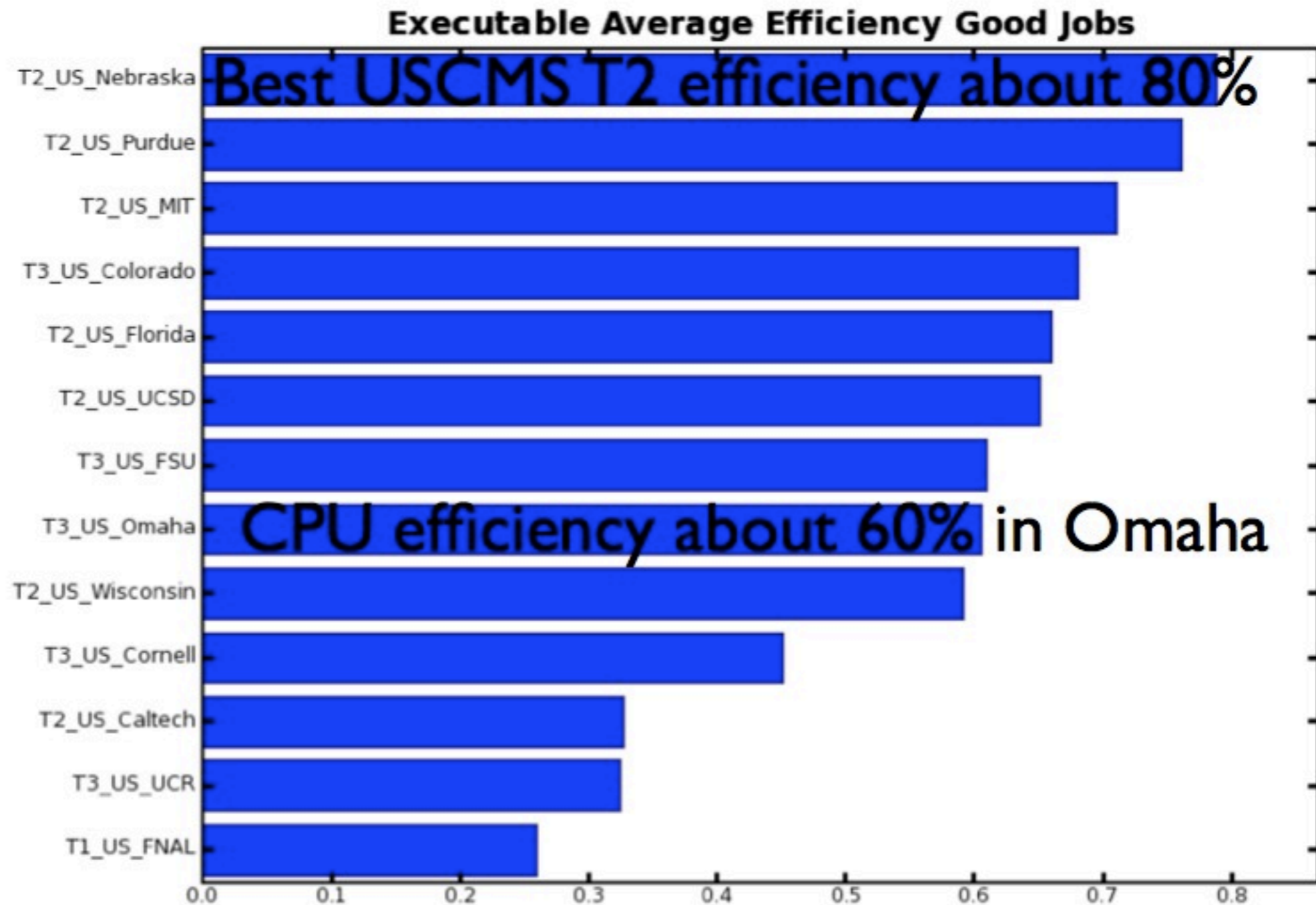
Successful stages are accounted for and removed from the system. If you want the file to come back because you took too long to submit jobs you need to make a new request.

Site agent sees request in CouchDB, contacts PhEDEx to get the data at the site matching the request and triggers site provided stager scripts.

WAN access of data

- Computing model predicated on poor networks (10 years old)
- 10 gig pretty standard, job IO rates not spectacular (1-5MB/s/core)
- Test bed uses xrootd with HDFS used as cache
- WAN access of some portion of data possible, less impact on efficiency than you'd expect

Example



The IO problem

- WAN access is fine while we process in the <GB/s regime
- LAN access is fine while we process in the 1-10 GB/s regime
- How do we process at higher speeds?
- When do we need to?

Meta data

- Currently record >6million files in
~1 million file blocks
- Intention was for 1000 files/block
- “Open” queries against DBS very inefficient, addressed with DBS3
- When will we out grow Oracle?

Quality control

- Testing a new release of any component is laborious - which is what you'd expect
 - Compounded by poor change management, poor documentation
- Common problem, not well addressed in CMS (at least)
 - No real Continuous Integration/Deployment
 - Middleware and Infrastructure don't lend themselves well to this as implemented - weird/unexpected couplings
 - Testing at scale is difficult, phased roll out might help
 - Unit test coverage <<100%
 - Integration effort significantly smaller than you would find in industry

Operational/ Infrastructure costs

- Costs not exposed to end users
 - Expectations \propto capabilities of home media centre
- Small changes in organisation have potentially large impact on operations
- Results in:
 - Low quality infrastructure SW - “*patch it in DMWM*”
 - Unreasonable expectations on Ops - e.g. requests to restart large MC production, on Christmas eve

Summary

- Technical issues understood and being addressed
- Sociopolitical issues more complicated to resolve and introduce more instabilities than technical issues

