Future Oracle use by CMS Offline DM/WM management

PhEDEx, DBS, T0AST…

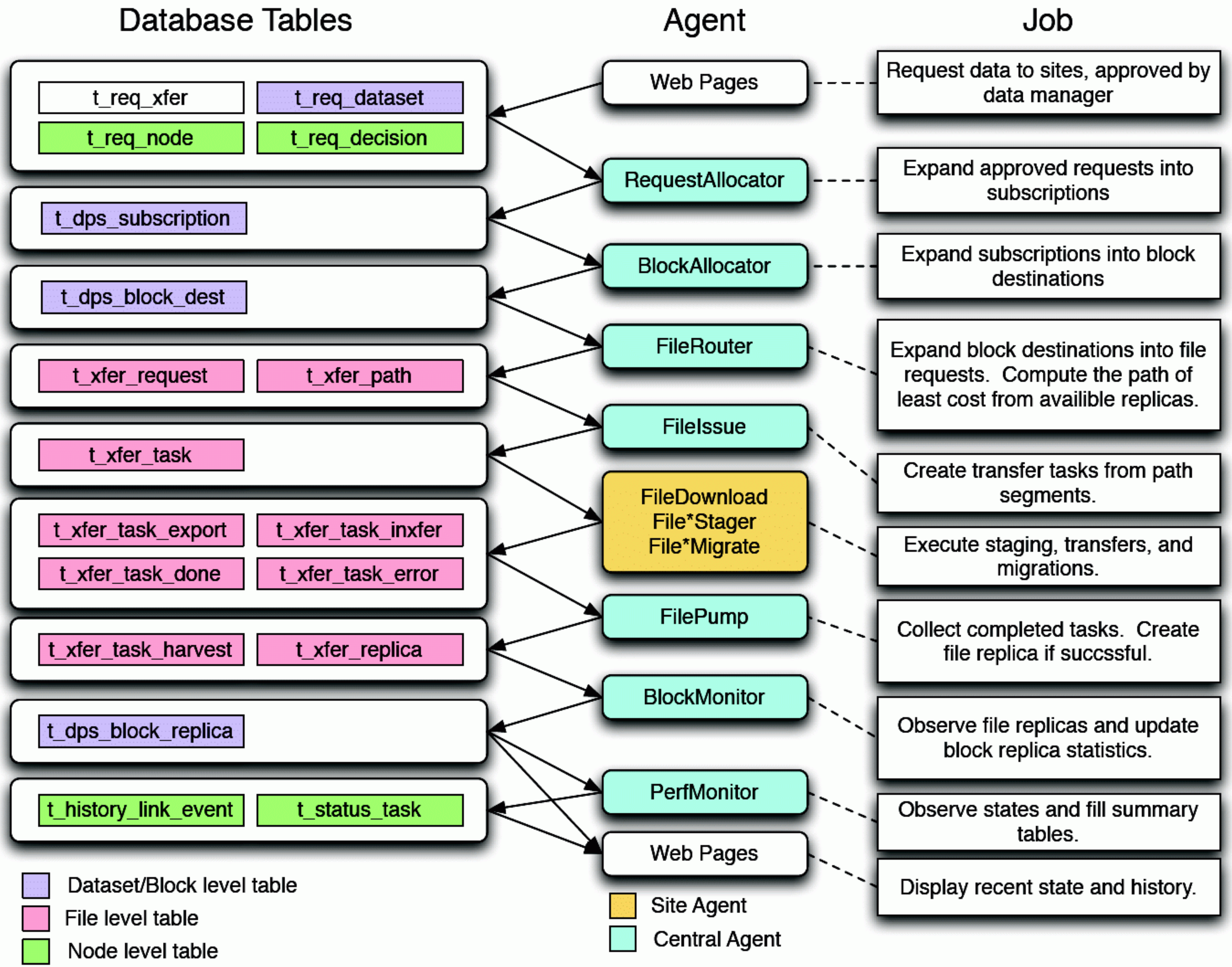
This talk

- Oracle-based Offline DM/WM applications
  - What they are, what they're for
  - Who uses them and how
  - How we expect them to change in the future

- PhEDEx: Physics Experiment Data Export
  - Manages distribution of all data for CMS
- T0AST: Tier0 Activity State Tracker
  - Manages the Tier0 workflow
- DBS: Dataset Bookkeeping Service
  - Relationships between datasets, what the data is

PhEDEx

- Manages bulk data transfers & deletions
  - Reliability, robustness, performance, scale
  - Data-placement policy, data consistency, custodiality
  - Knows where all the data is, all the time
- ‘central’ agents, site agents
  - Direct DB connection, one agent per task
- Data-service and website
  - Interaction with users (make transfer requests)
  - Other components of CMS DM/WM
    - Tier0, MC production, external monitoring (site/group-specific)



Database Tables

Agent

Job

t_req_xfer
t_req_dataset
t_req_node
t_req_decision

t_dps_subscription

t_dps_block_dest

t_xfer_request
t_xfer_path

t_xfer_task

t_xfer_task_export
t_xfer_task_inxfer
t_xfer_task_done
t_xfer_task_error

t_xfer_task_harvest
t_xfer_replica

t_dps_block_replica

t_history_link_event
t_status_task

Web Pages — Request data to sites, approved by data manager

RequestAllocator — Expand approved requests into subscriptions

BlockAllocator — Expand subscriptions into block destinations

FileRouter — Expand block destinations into file requests. Compute the path of least cost from availible replicas.

FileIssue — Create transfer tasks from path segments.

FileDownload
File*Stager
File*Migrate — Execute staging, transfers, and migrations.

FilePump — Collect completed tasks. Create file replica if succssful.

BlockMonitor — Observe file replicas and update block replica statistics.

PerfMonitor — Observe states and fill summary tables.

Web Pages — Display recent state and history.

🟪 Dataset/Block level table

🟥 File level table

🟩 Node level table

🟨 Site Agent

🟦 Central Agent

PhEDEx in numbers

- ~16 million files, transfers >= 200 TB/day
- over 100 sites (T0, T1, T2, T3)
- Database
  - Currently occupies ~50 GB
    - Grows at ~2GB/month, likely to continue
    - Should be ~linear with data volume
  - CPU use moderate
    - Should go with active transfer-volume
    - Scalability of central agents perhaps our biggest concern

PhEDEx scalability

- Performance tested with the LifeCycle agent
  - Custom test-agent drives accelerated lifecycle
  - Verifies schema, agents, at >10x ‘today’ levels
  - Maintain confidence that we are >1year away from (known) problems
  - Identify problematic SQL or agent behaviour early
    - Increasingly find bottlenecks are in the agents, not the DB
  - Useful for testing new DB hardware (11gr2)

T0AST

- Tracks files, batch jobs, & metadata
- Drives the T0 processing and data-handling
- Fed by information from several sources
  - Cessy -> Meyrin transfer system injects files
  - StorageManagerDB & PopConLogDB trigger the T0 to start processing
  - ConfDB provides HLT/trigger/dataset mapping
- Injects data into PhEDEx and DBS

- Prevent inconsistent DB states
  - Expected software crashes, batch failures, transfer problems,
  - Esp. in early days, but can happen at any time
  - SQL-surgery should not become the norm
- Designed to crash and fail quick and clean
  - Fix problem at source, restart the crashed component
    - Largely successful with this strategy
- T0Mon, data-service for monitoring
  - Used inside & outside the T0 (express visualisation…)

- Small number of clients
  - But very active on the database!
- Periodically backup and wipe the DB
  - No need to maintain history
  - 4th iteration in 2 years
    - 25 GB at last cleansing. 7 GB in 3 weeks since
- Test schema by replays on separate instance
  - Also test schema-evolution, SQL optimisation
    - Rely on Oracle stats for guidance
- Schema stable, no major changes foreseen
  - Expect client-code to evolve more than schema

- Driven by data-rate
  - Bandwidth, but also rate of files
    - 0.6M files per day during running (0.5M from detector)
    - 15K files per day output to PhEDEx and DBS
  - Processing reduces file-rate
- Number of primary datasets => #workflows
  - Increases complexity, volume of metadata
- Row-churn
  - Metadata updates with high turnover
  - Caching cannot help
- Table-growth, slows down queries over time
  - Optimise for recent data, helps, but not perfect

DBS

- Catalogue of all CMS data
  - How it was produced, what it contains
  - Starting point for data-discovery for physicists
  - 85 thousand datasets, 18 million files
- Interacts with…
  - SiteDB for user auth, PhEDEx for block-location
  - DAS (WMAgent) for MC production & reprocessing
    - Accounts for 2/3 of current activity
  - CRAB for user-analysis jobs

- DB is ~200 GB
  - Will grow ~linear with data-volume
  - Need to maintain history
    - Never delete information from DBS
- Variable workload, factor 10 or more
  - Clients distributed everywhere around CMS
  - Work-cycles come and go (MC, real data)
  - Difficult to set performance targets

- Schema migration non-trivial
  - Currently in the middle of a major re-design
    - DBS2 => DBS3
  - Needs integration with several client tools
  - Re-deployment of many DM/WM components simultaneously
- DBS3 testbed ‘soon’
  - Run validation, scale, performance tests
  - Integration tests with clients
  - Query-optimisation, well advanced
  - Release data contingent on outcome of tests,
    - Reasonable to expect by end of this year

Summary

- PhEDEx:
  - Mature, slow evolution of schema
  - Can test schema under realistic circumstances
- T0AST:
  - Mature, slow evolution of schema
  - Re-initialise to limit size or for schema updates
- DBS:
  - Major schema update this year, stable after that
  - Most visible of the three, less predictable loads