Big PanDA Pilot for Titan. Recent changes

Danila Oleynik
UTA / JINR
05/06/2014

PanDA Pilot for Titan LCF recent changes

- Latest version of PanDA Pilot (PICARD 59aRC4a) integrated with Titan LCF:
  - Added processing of some special exceptions, related with local batch system (like user interruption of payload execution)
  - Implemented method for publishing status of payload in internal batch queue

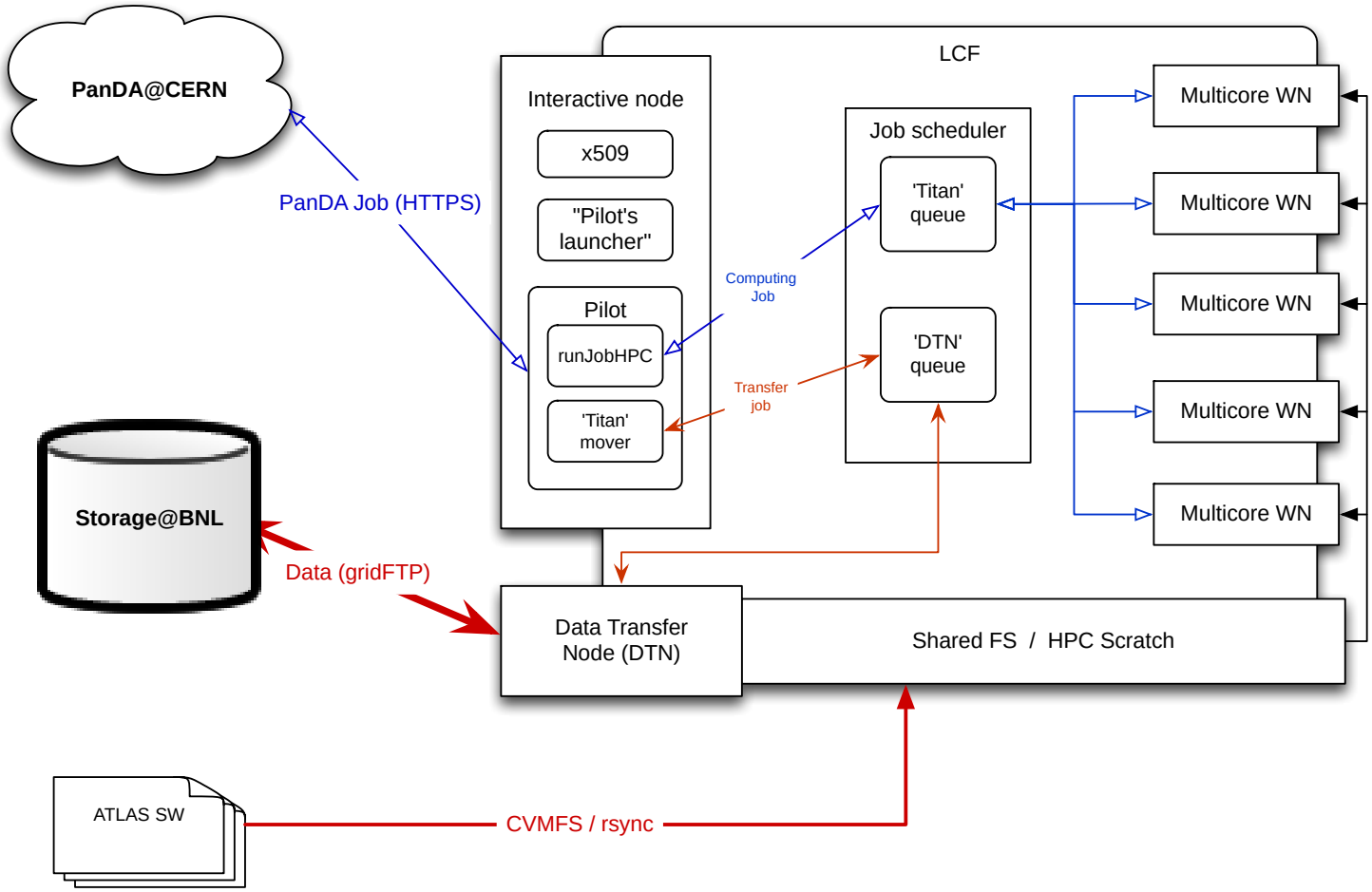
Data management issues

- Transfers data to(from) remote SE (at least BNL)
  - Due to recent changes on interactive nodes gridftp not working properly anymore
  - Data Transfer nodes not affected and works good (with highest throughput)
- Exercises with realistic payloads raise situations with jumbo outputs (dozens of Gb)

Data Transfer Nodes

- OLCF recommend solution for remote data transfers:
  - The DTNs have been tuned specifically for wide area data transfers, and also perform well on the local area. They are recommended for data transfers as they will, in most cases, improve transfer speed and help decrease load on computational systems' login nodes.
- DTN usage modes
  - Interactive access. Works good for single transfers and for debugging. Not so good for software integration, due to OTP authorization required
  - Batch access. Data transfer nodes can be accessed from Titan's batch system by submitting a batch job targeting the 'dtn' partition. Preferable for implementation.

Involving DTN in PanDA Pilot workflow



“Titan” mover. Current state.

- Both modes of usage of DTN were successfully tested
- Batch access mode was chosen for software integration
- Intercommunication with batch system realized through SAGA API
- In progress:
  - Proper declaration of calls of methods (for proper integration with Pilot)
  - Interpretation of results of transmission (based on parsing of batch job output files)

Realistic payload. Output issues.

- Current algorithm for MPIzation of job may cause generation of huge amounts of data.
- Amount of data will have linear depends from allocated cores (with minimal factor 16)
- During initial tests, output easily reached dozens of Gb, with thousands of directories/files
  - Packing and clearing of this data by pilot took hours

Possible solutions for managing of output

- Upper limit of number of allocated cores
  - Processing of this parameter already implemented
  - Initial tests shows no depends between number of allocating nodes and waiting in scheduler queue
- Set of recommendations for transformations developers
  - Minimization of disk operations
  - Proper using of MPI benefits (real CPU intensive algorithms)