Dataset-based High-Level Data Transfer System in BESDIRAC

Tao Lin and Xiaomei Zhang
Institute of High Energy Physics

Abstract

Data Transfer is an essential part in grid. In the BESIII experiment, the result of Monte Carlo Simulation should be transfered back from other sites to IHEP and the DST files for physics analysis should be tranfered from IHEP to other sites. A robust transfer system should make sure all data are transfered correctly. In this poster, the design and implementation of a Dataset-based Data Transfer System will be shown.

1 Introduction

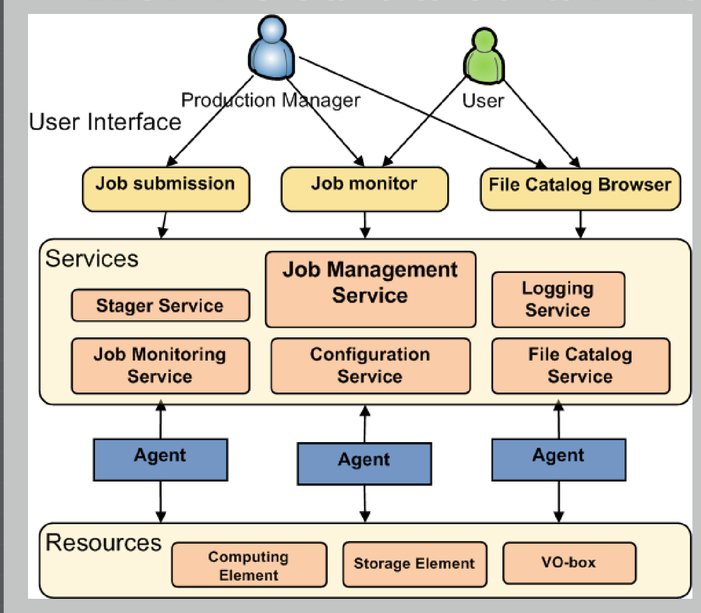
- BESIII experiment is a general purpose experiment for studying electron-positron collisions at BEPCII.
- The BESIII data production uses both a local cluster model and a distributed computing model.
- DIRAC is a solution for the distributed computing.
  - Monte Carlo production in remote sites;
  - Reconstruction in IHEP;

2 Why need a dataset-based Transfer System?

- What will be transferred?
  - The result of Monte Carlo Simulation
  - The files for physics analysis
- Real big data!
- But, poor network connectivity!
- If there is a transfer system:
  - user don't need to wait any more;
  - user can retransfer failed files easily.

3 Developing in DIRAC & BESDIRAC

- DIRAC consists of cooperation distributed *services* and light-weight *agents* delivering the workload to the Grid Resources.
- We can *reuse* the most functionalities supplied by DIRAC.
- BESDIRAC is an extension to DIRAC for BESIII specified.



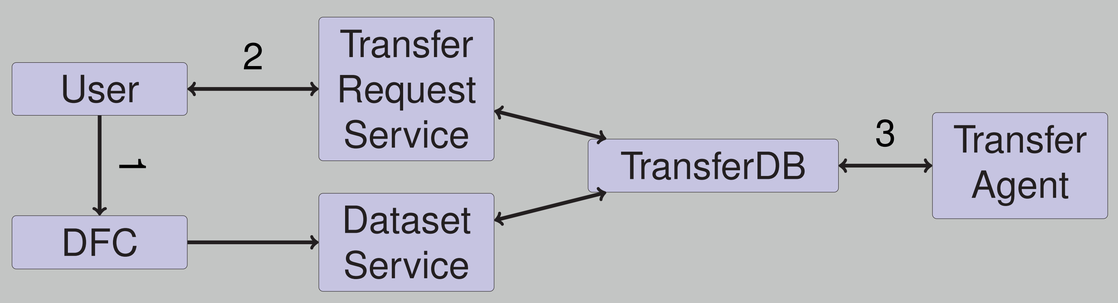
DIRAC follows the `Service Oriented Architecture (SOA)` paradigm, accompanied by a network of lightweight distribued agents which animate the system.

- User Interfaces
- Services
- Agents
- Resources

4 Overview of Transfer System

Transfer Agent is the scheduler.
Transfer Request Service is to create, kill, retransfer and monitor the transfer requests.
Dataset Service is for the dataset management.
Transfer DB is the shared memory.

5 The workflow of Transfer System

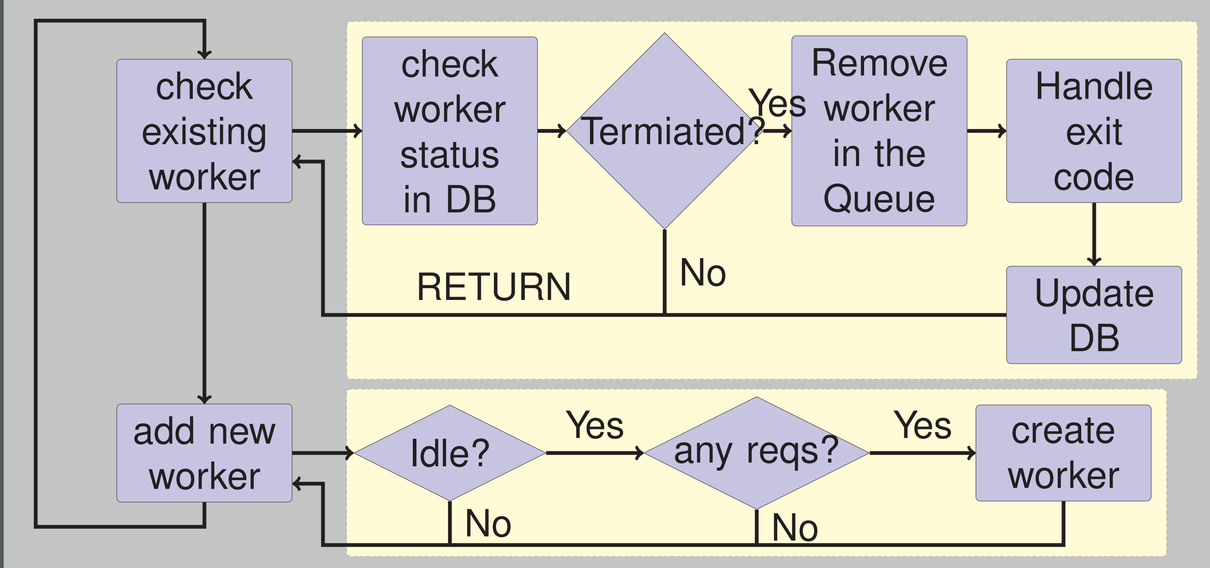


1. User create a snapshot of the file list in DFC, which is registered in the Dataset Service.
2. User create or modify or monitor the transfer request.
3. Transfer Agent will transfer these files in the Database.

6 Transfer Agent

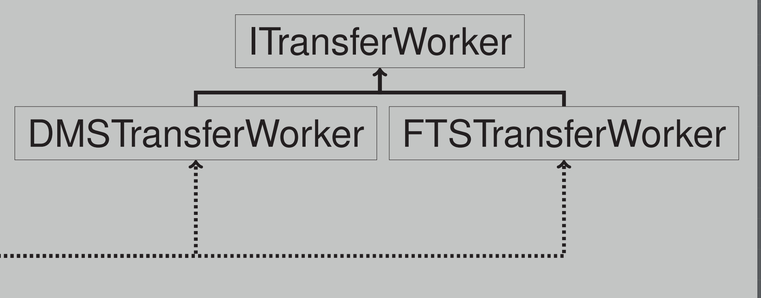
- In `DIRAC`, `AgentModule` is the base class for all `Agent`s. The derived classes should implement:
  - `initialize`
  - `execute`
  - `finalize`
- `TransferAgent` implements the `non-blocking` scheduler in the `execute` method.
- In fact, we create several `sub processes` to run the transfer commands. The scheduler uses the `async I/O` to communicate with the `sub processes`.
- To support multiple transfer protocols, we use a `TransferFactory` to create `TransferWorker`s.

7 The workflow of Transfer Agent



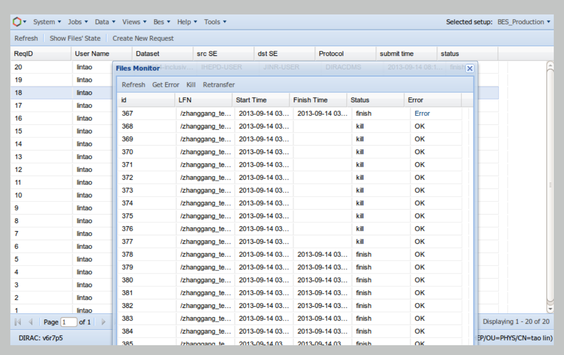
8 Transfer Factory

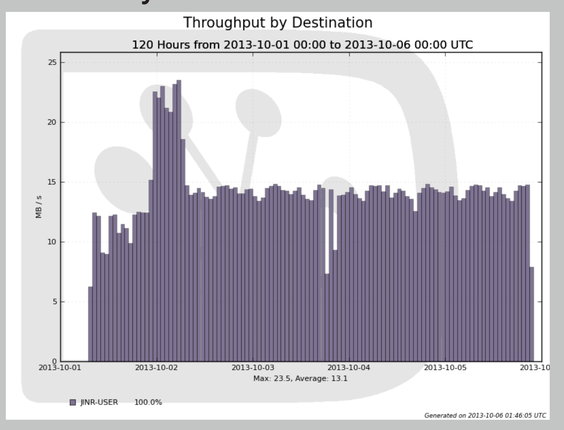
Transfer Factory will create the specific worker according to the protocol.



9 Web Portal and Accounting

- Extensions can integrate with DIRAC easily.





10 Conclusion and Outlook

- Conclusion
  - The design and implementation of BESDIRAC Tranfser System was presented.
  - The building of the prototype system makes us earn the experience to deal with DIRAC.
  - DIRAC is flexible to extend its functionality.
- Outlook
  - Work on BESDIRAC Transfer system is ongoing.

lintao@ihep.ac.cn