Distributed Computing Infrastructure for HERD Experiment

Xuantong Zhang, Xiaowei Jiang, Qingbao Hu (Speaker)

Computing Center of IHEP, CAS

Outline

HERD Introduction

Computing model for HERD

Computing infrastructure

Storage infrastructure

Network and data transfer

Authentication and Authorization

Summary

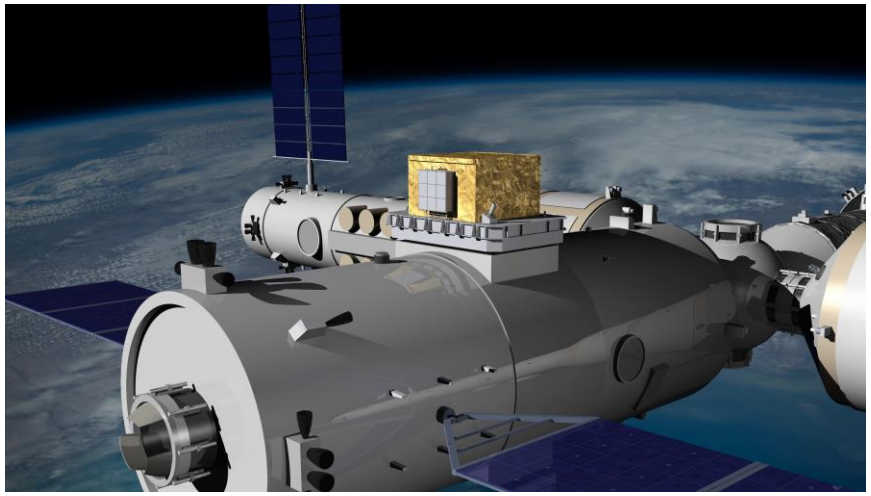
HERD & Its Computing

HERD Experiment

High Energy cosmic-Radiation Detection facility (HERD),

- A space particle astrophysics experiments, will run in the Chinese Space Station for >10 years since 2027.
- HERD is a international collaborated experiment with around 47 institutes, labs and universities from China, Italy, Switzerland, Spain and Sweden.

Science Goal	Type	Contribution to Physics	Methods
Precision measurement of cosmic ray electron flux and dark matter search	Core	Key contribution to solve one of the most important puzzle for astronomy and physics: dark matter	Precision flus measurement of high energy electron and gamma.
Origin, acceleration and propagation of cosmic rays	Core	Key contribution to the origin of cosmic rays	Measurement of cosmic ray nuclei up to Z=28 to the highest energy
High energy gramma rays all-sky survey and monitoring		Search and identify gamma ray source , understand the physics of extreme conditions in the universe; search for new physical signals	Wide energy range, High precision measurement of Gamma rays



HERD Computing Requirement

Storage resources:

- >30PB in 5 years,
- >90PB in 10 years.

Computing resources:

- ~7500 CPU cores in 5 years,
- ~16000 CPU cores in 10 years.

Network and data transfer:

- 10-100 Gbps.

Data type	Data size（PB）			Computing（CPU Core）		
	5 years	10 years	Site	5 year	10 year	Site
Flight Data	2	6	T0, T1	-	-	T0
Standard Reconstruction	2.5	7.5	T0, T1	200	400	T0
Data transmission control system	1	2	T0	300	600	T0
PassN reconstruction	5（2 version）	15（2 version）	T0, T1	1000	3000	T0
Simulation data	5	15	T0, T1	4000	8000	T0
Analysis Data	2	4	T1	2000	4000	T1
Summary	15.5+16.5	45.5+47.5		7500	16000	

Data processing challenges:

- Need to distribute RAW data from China Space Station to CN and EU data centers.
- Need to schedule multiple data process tasks among CN and EU data centers.
- Need to provide uniform user authentication and resources permission management system.

So, we need to build infrastructure for HERD computing.

3 Layers of HERD Computing

User Interface:

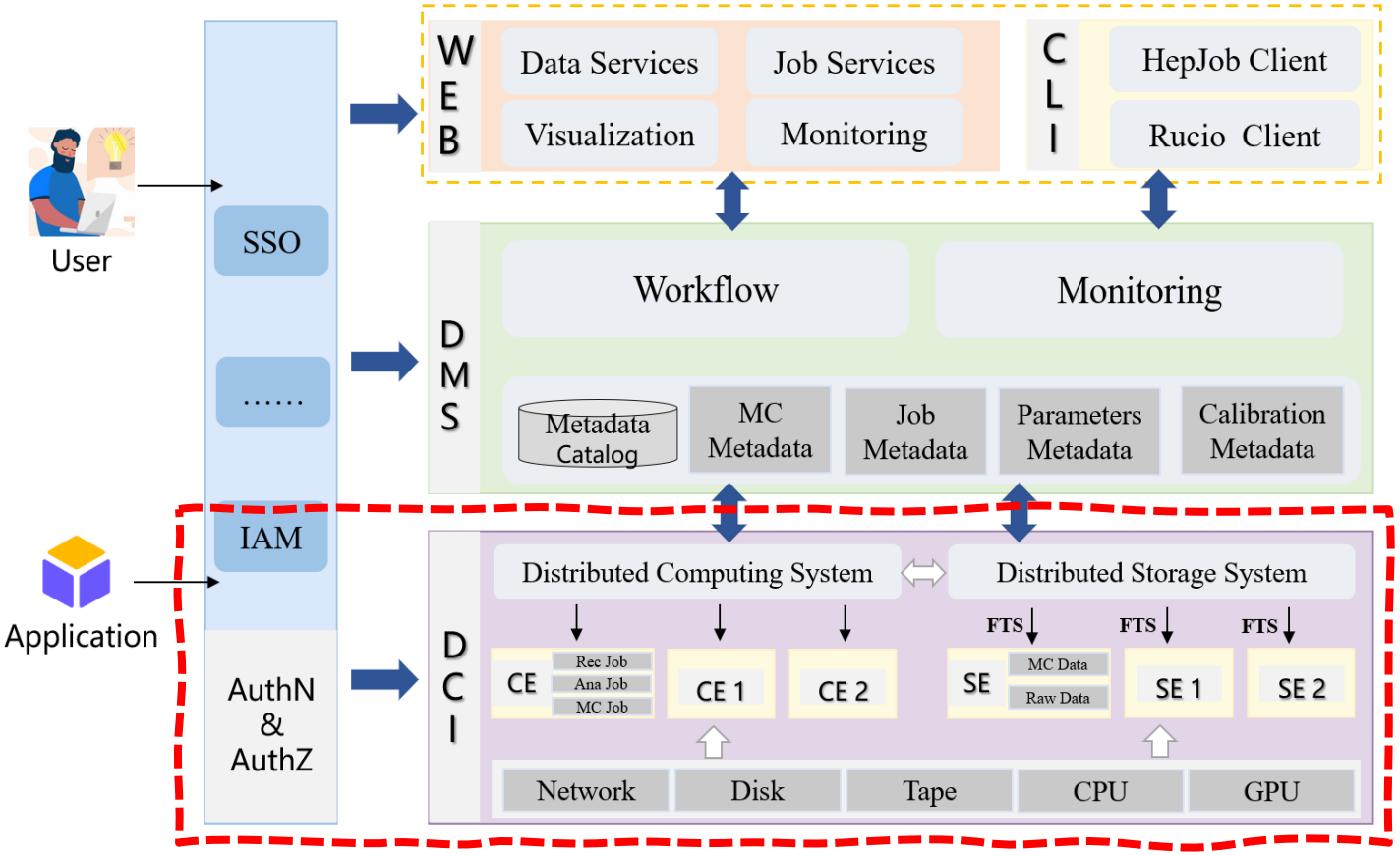
- Provide web UI and CLI for user.
- Trigger data process, analysis, monitoring task by user.

Data Management System (DMS):

- Manage data processing tasks.
- Provides metadata database.

Distributed Computing Infrastructure (DCI):

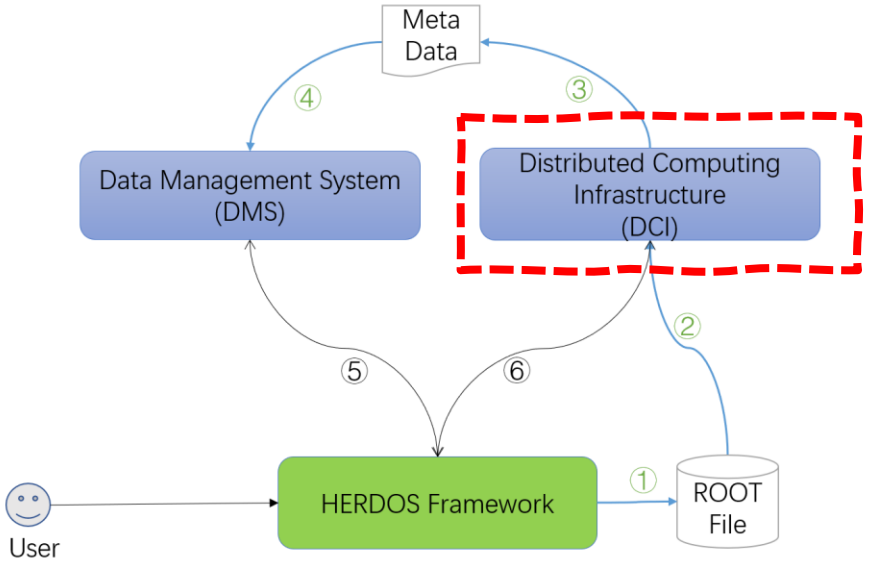
- Manage computing/storage resources.
- Executing data processing tasks.

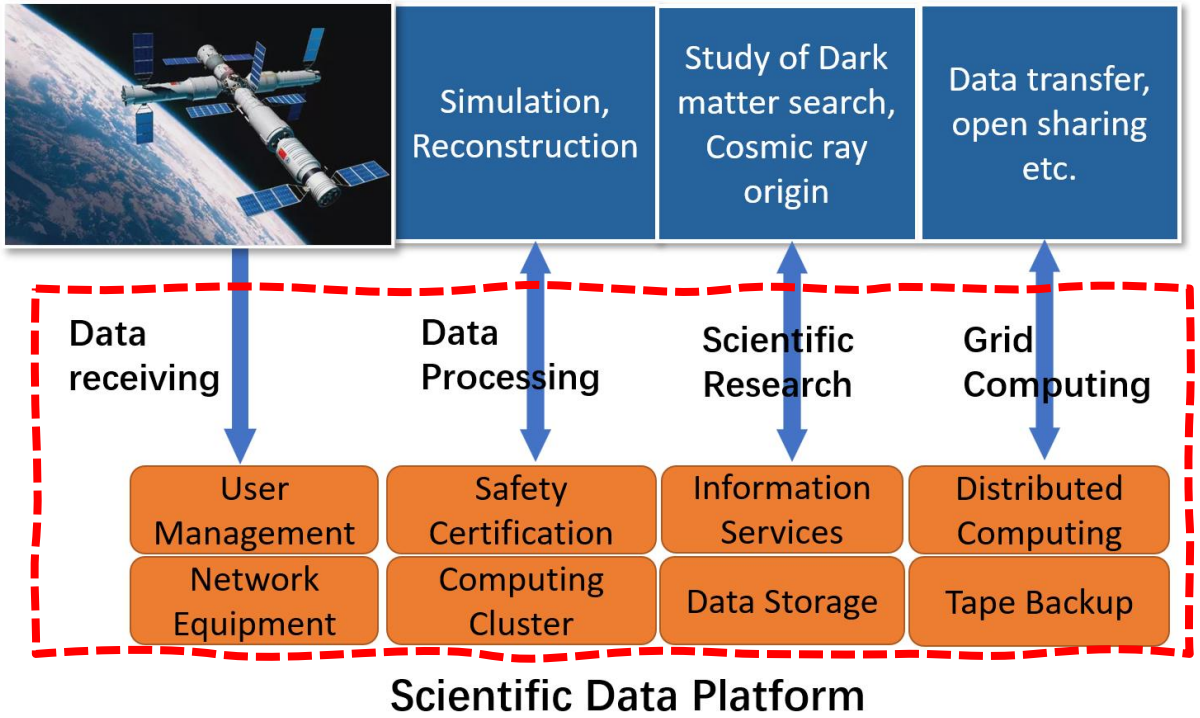


Distributed Computing Infrastructure

HERD DCI is a distributed computing system for,

- **Data processing** -> Distributed computing system
- **Data access** -> Distributed data management system
- **Data distribution** -> Network and data transfer system
- **Data privilege management** -> Authentication and authorization system



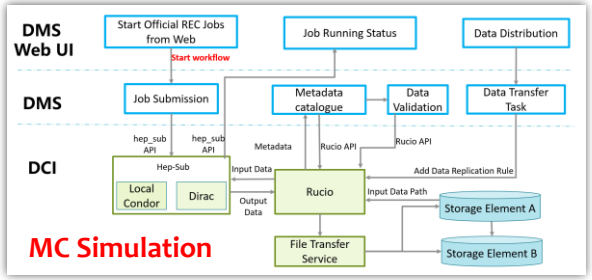


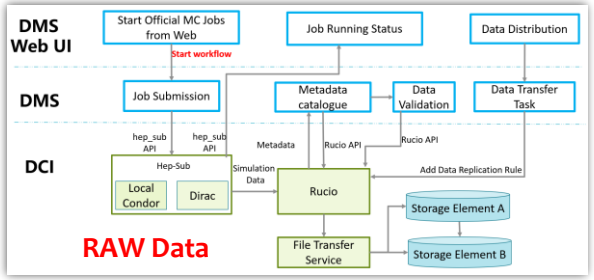
Computing Model of HERD

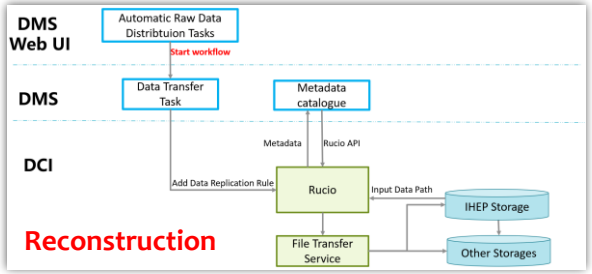
DCI supports all data processing with computing and storage resources.

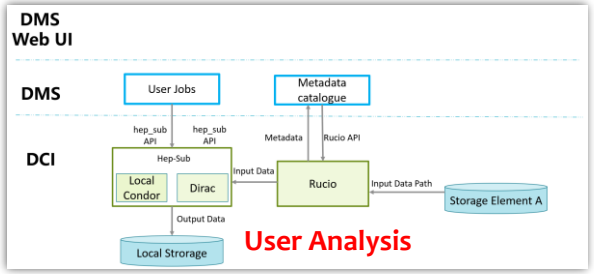
Multiple data process workflows,

- MC Simulation,
- RAW Data,
- Reconstruction,
- User Analysis.



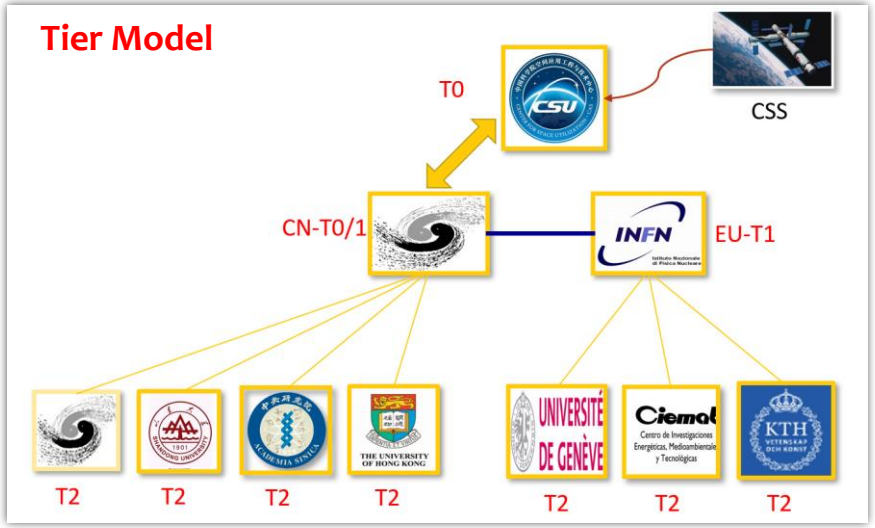






Tier model for data process,

- **Tier-0 sites: Central site.**
  - CSU: Raw data acquisition from Space Station.
  - IHEP: All types data storage and data distribution source.
- **Tier-1 site: Regional center site**
  - IHEP(CN-T1)/INFN(EU-T1): SIM and REC data storage, computing resources.
- **Tier-2 site: simulation data processing**



Subsystems in DCI

Distributed Computing,

- Manage distributed computing resources and execute computing tasks.

Distributed Storage,

- Manage distributed storage and storage raw and processed data.

Data Transfer and network,

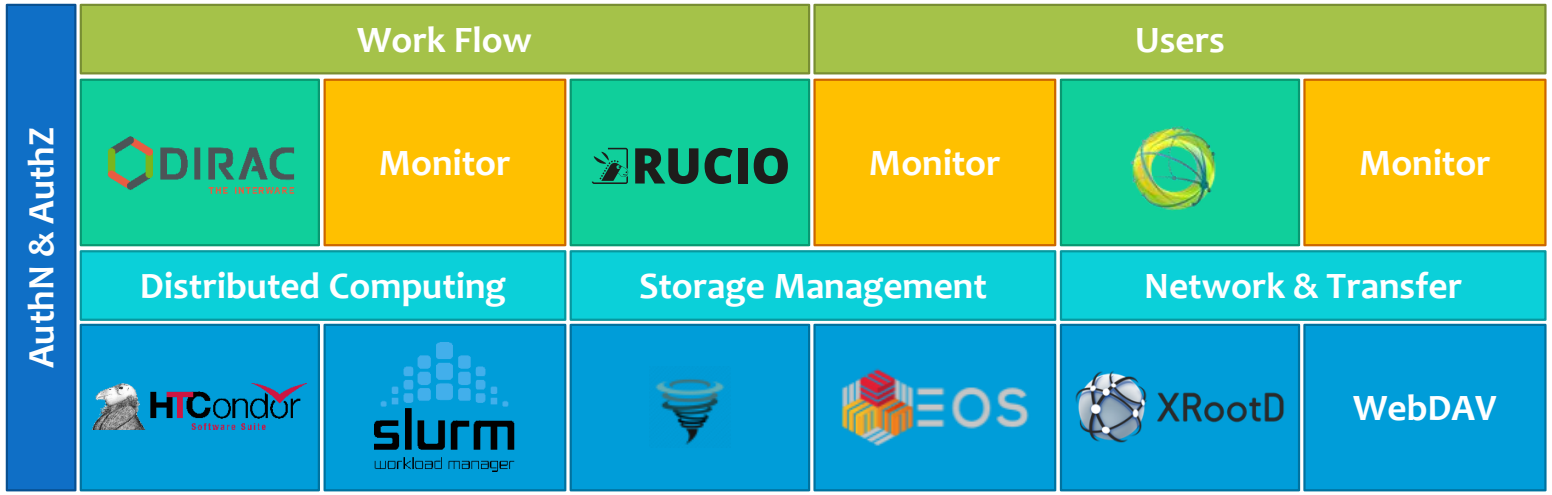
- Manage data transfer tasks.

AuthN & AuthZ,

- Manage user permissions in data processing.

Resources,

- Computing clusters and disk/tape storage in each data centers.



Computing System for HERD

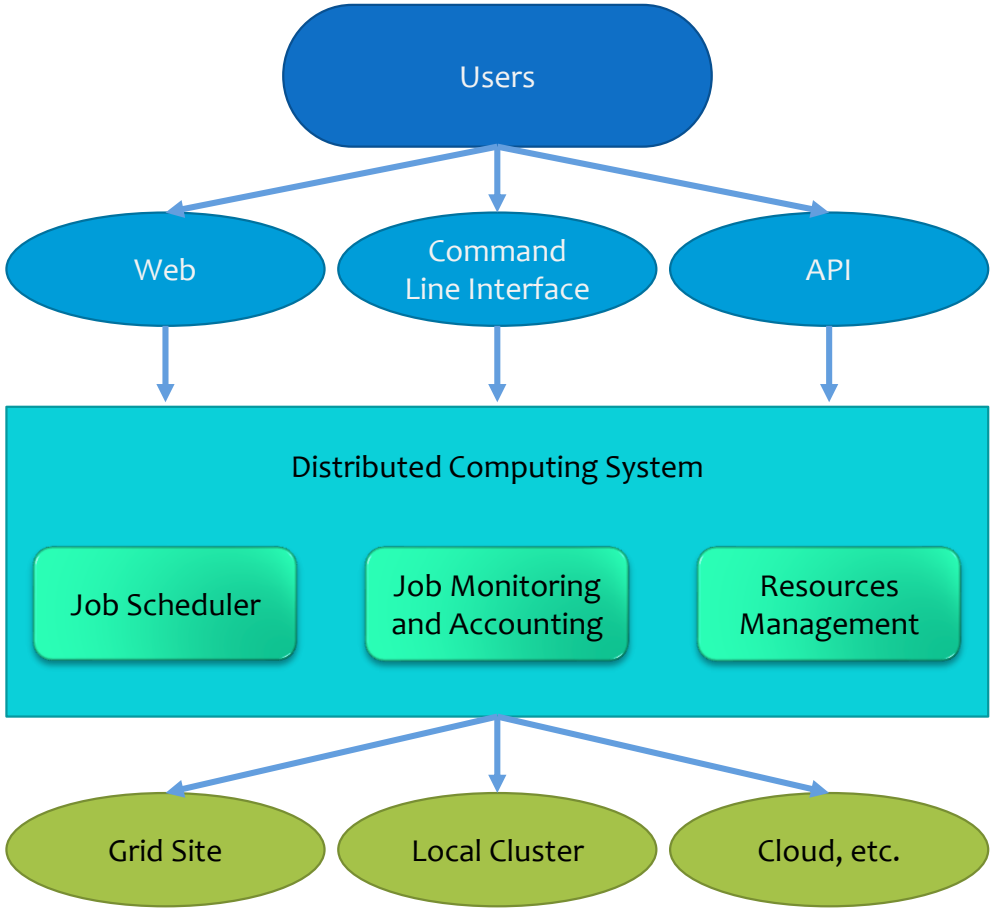
Computing Management Design

To users:

- **Unify computing sites** with heterogeneous computing systems.
- Allow to use HTCondor, Slurm, cloud computing, supercomputing, local cluster, etc. **based on data processing tasks**.
- Supply **unified job management interface**.
- By Web, Command line interface and APIs.

To computing sites:

- **Schedule jobs** in computing resources.
- **Optimize jobs distribution** among sites.
- Monitoring computing resources status.
- Generate site reports and accounting sites usage.



Structure of Computing System

“One entrance, all computing tasks”

Distributed computing system

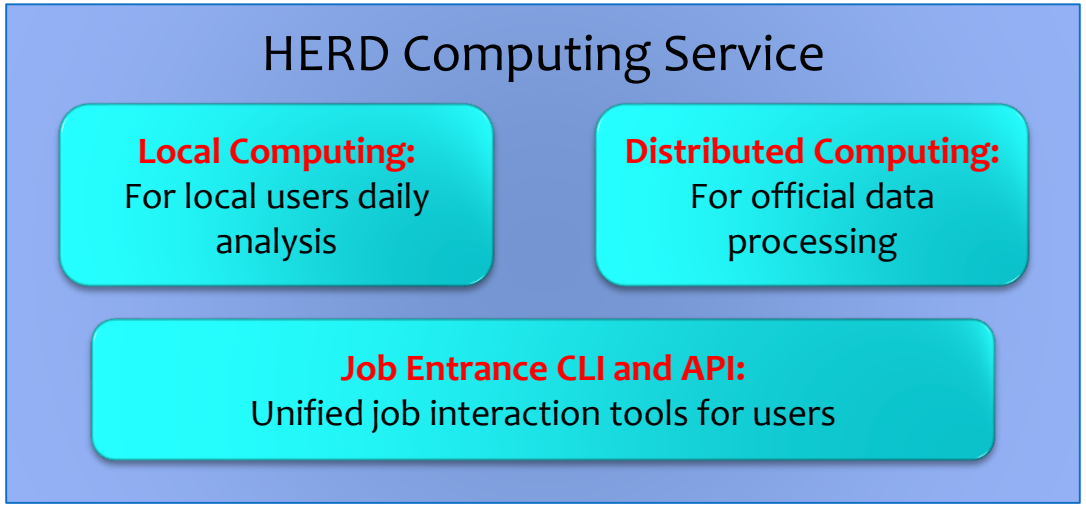
- Distributed computing resources around the world.
- For official data processing and special tasks.

Local computing system

- Local computing resources in each data center.
- For local users analysis tasks.

Unified job entrance tools

- Provide a unified job entrance tool for all type of data processing and analysis jobs.



Distributed Computing – DIRAC

DIRAC: Distributed Infrastructure with Remote Agent Control

- First developed for LHCb. Widely used in Belle2, ILC, CTA. In IHEP’s experiment: JUNO, CEPC.
- For HERD, we use it to **manage distributed computing job scheduling**.

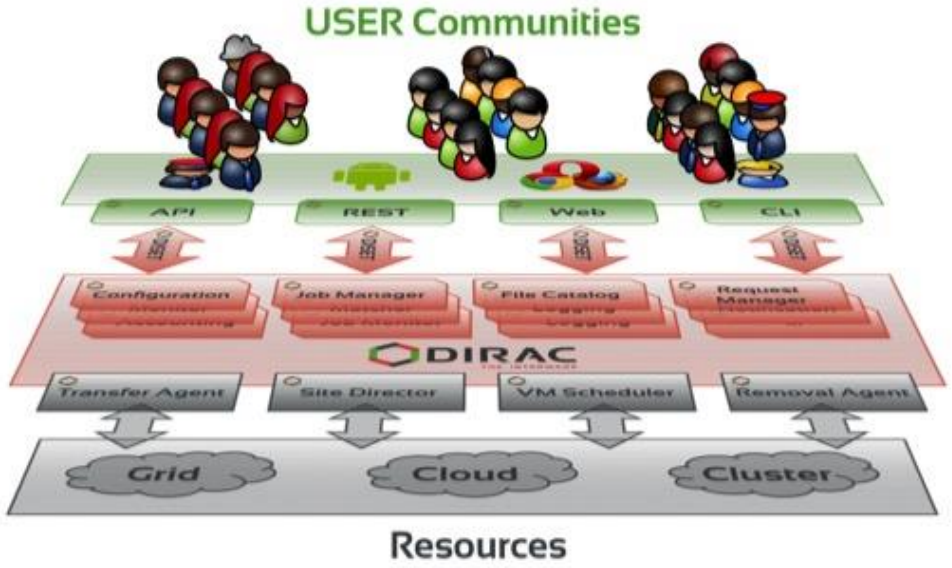
Middle layer between users and resources.

- User interface: API, REST, Web, CLI.
- Computing management: Grid, Cloud, Cluster.

Job management in DIRAC:

- Job submit: JDL job.
- Job schedule: Pilot job.

```
JobName = "Simple_Job";
Executable = "/bin/ls";
Arguments = "-ltr";
StdOutput = "StdOut";
StdError = "StdErr";
OutputSandbox = {"StdOut","StdErr"};
```



Local Computing – HTC/HPC

Type of HERD computing jobs includes,

- **Single-core job or multi-core job within one node**: simulation, reconstruction, analysis,
- **Multi-core job on multi node or GPU job**: part of reconstruction, AI training.

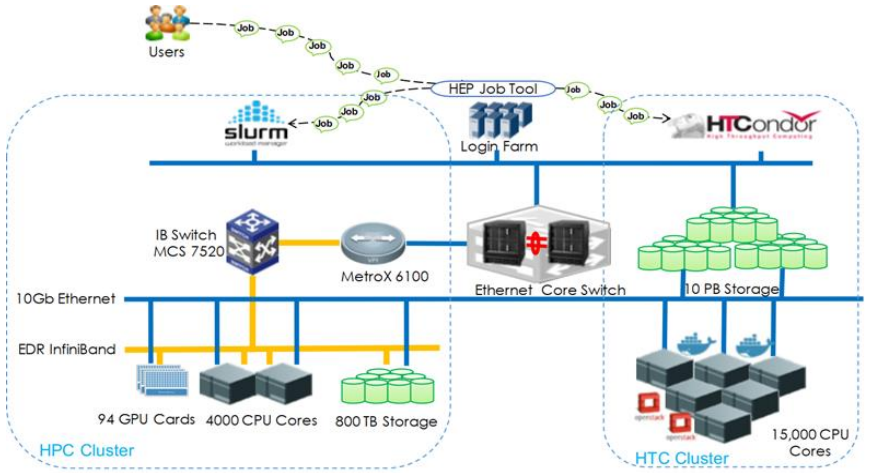
High throughput computing: single-core or multi-core within one node,

- HERD **data processing** is a typical high throughput computing with hundreds of thousand jobs,
- HERD HTC cluster is **based on HTCondor** which is a open-source high throughput computing software suite.

High performance computing: big multi-core job or GPU job

- Part of HERD **reconstruction data processing** is using AI to driven,
- HERD HPC cluster is **based on Slurm**.

dHTC for share resources between HTC and HPC



Job Entrance – HepSub Tools

One Job Entrance is a job APIs, based on HepSub tools:

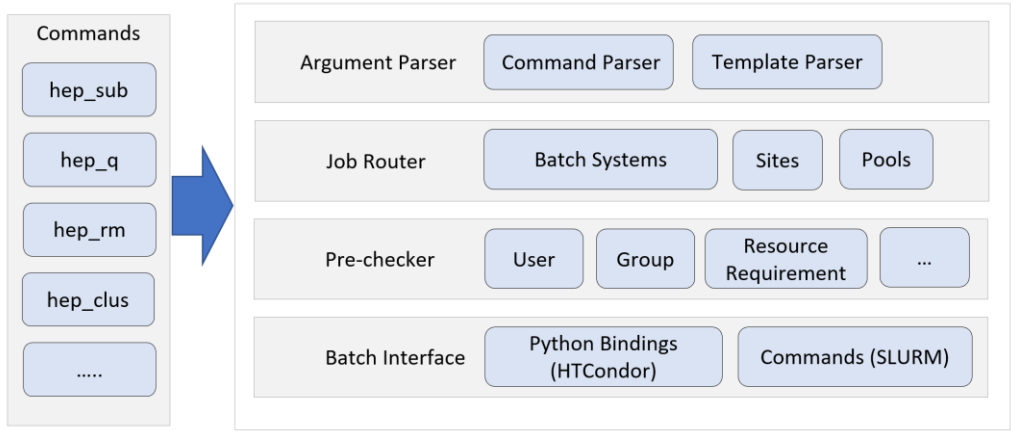
- Support **unified job submission endpoint and interface**, no matter of user jobs or production jobs.
- Support **both Grid jobs (DIRAC jobs) and cluster jobs** submission.
- Support **Grid data access and management**.
- Flexible enough for **integrating to other job services**, such as HERD data production system service, authentication system, monitoring and accounting system, common user interfaces…

HepSub is a job submission tool developed by IHEP,

- Already unified HTCondor/Slurm cluster job operations: submission, query, remove, etc.

HepSub is under development to extent to submit DIRAC jobs.

- To **support DIRAC JDL** format translation,
- To **support IAM** with X.509 certificate and Sci-Token authentication.



Storage Management System for HERD

Storage Management Design

HERD Grid storage management,

- To **produce and distribute data** from distributed computing and storage sites.
- To **manage distributed data access quests** from DMS system.
- Based on **Rucio system**, a popular grid data management system in HEP.

Storage management services manages data production,

- **Raw data distribution**, from IHEP to Chinese and European Storage Sites.
- **Processed data distribution**, replicate among Tier1 and Tier2 sites.
- **Official data operation**: adding, deleting, modifying, querying in distributed storage sites.

For normal users,

- Supply data access in developed HERD Software APIs and CLI command, normal user could get official processed data by HERD Software.

Structure of Storage Management

"One API, all data management tasks"

Storage Management System:

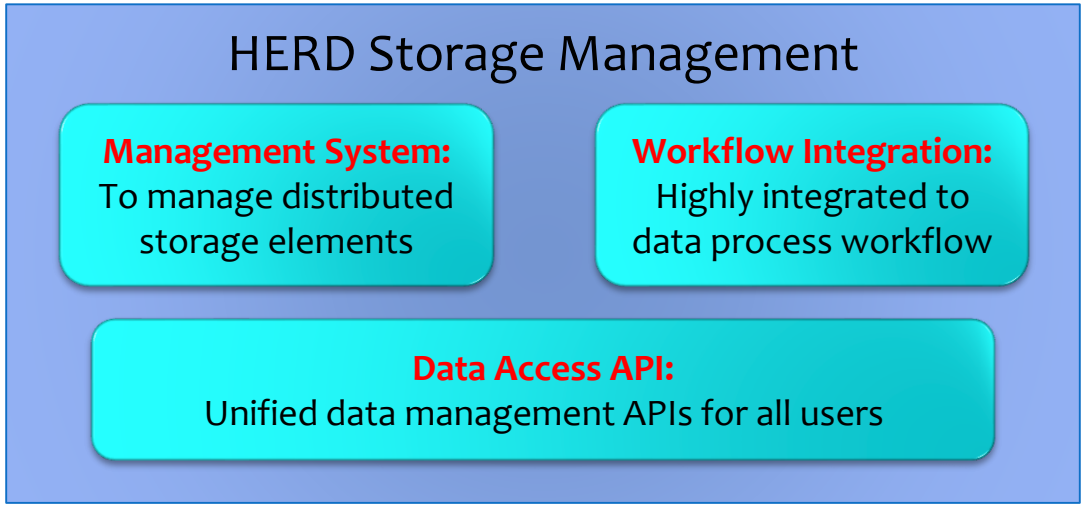
- **Based on Rucio** to manage distributed disk and tape storages,
- Develop **HERD customized file catalog policy** and **HERD Rucio plugins**.

Workflow Integration:

- Highly **integrated to HERD software** and data process workflow.
- Allow to be used in all tasks.

Data Operation API:

- Develop a **API for data operation** for both users and workflow tasks.



Distributed Storage Management – Rucio

Rucio with customized grid data file catalog namespace,

- **To make data logic name closer to local data, follow normal POSIX rules, has 3 types:**
  - Data container, to contain other containers and datasets,
  - Dataset, to collect files,
  - Data file, basically ROOT files.
- **Scopes are working as data status zones, so data types could be distinguished by its name,**
  - Temp, Valid, Corrupt, etc.

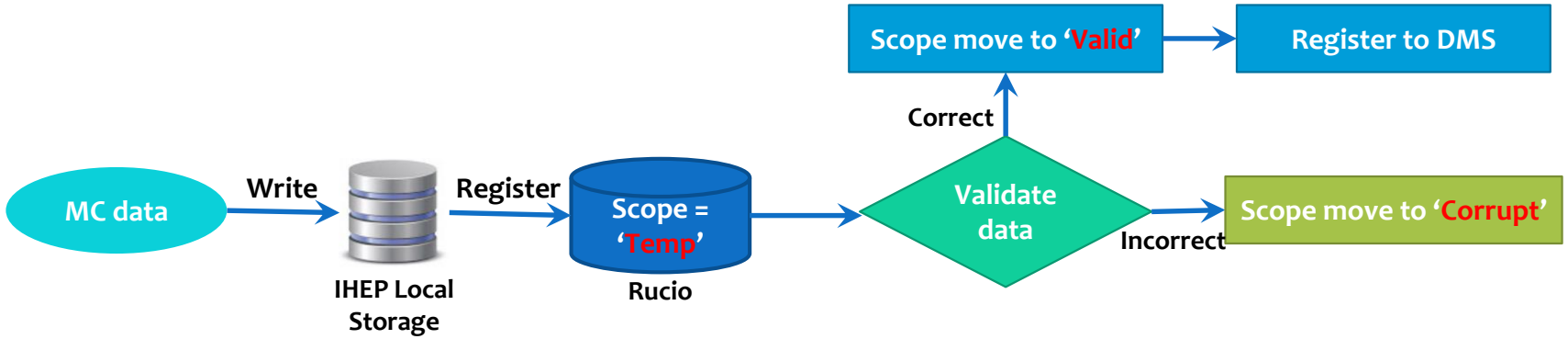
```
+-------------------------------------------------------------------------------+--------------------+
| SCOPE:NAME                                                                    | [DID TYPE]         |
|-------------------------------------------------------------------------------+--------------------|
| temp:/herd/user/z/zhangxt                                                     | DIDType.CONTAINER  |
| temp:/herd/user/z/zhangxt/                                                    | DIDType.DATASET    |
| temp:/herd/user/z/zhangxt/opt/herd/proton-center-E2.7-1_20TeV-34621161.0.root | DIDType.FILE       |
| temp:/herd/user/z/zhangxt/output1-test.g4mac.root                             | DIDType.FILE       |
+-------------------------------------------------------------------------------+--------------------+
```

Namespace Component	HERD Namespace Policy
Name	Linux-like directory and file path
Scope	Defined as data status in data flow(Temp, Valid, Corrupt)
Dataset	Collection of all Files in a directory
Container	Collection of all sub-directories (=datasets) in a directory

HERD Dataflow Integration

HERD data processing are all based on workflow,

- **For Workflow system:**
  - Synchronize file catalogs from Rucio.
  - Trigger data processing jobs. In these jobs, data are downloaded/uploaded by Rucio.
  - Trigger data validation and data distribution in Rucio.
- **MC dataflow as an example:**
  - Register all raw MC data to 'Temp' scope,
  - Data validation program use APIs to validate whether data are good.
  - If good, move scope to 'Valid', then provide it to metadata registering.
  - If not good, move scope to 'Corrupt' scope, waiting for deletion.



HERD Data Operation API

We are developing a HERD workflow-oriented API,

- **For both experiment software data access in jobs and workflow.**
- **Merged to HERD software and workflow system.**

API can provide methods for:

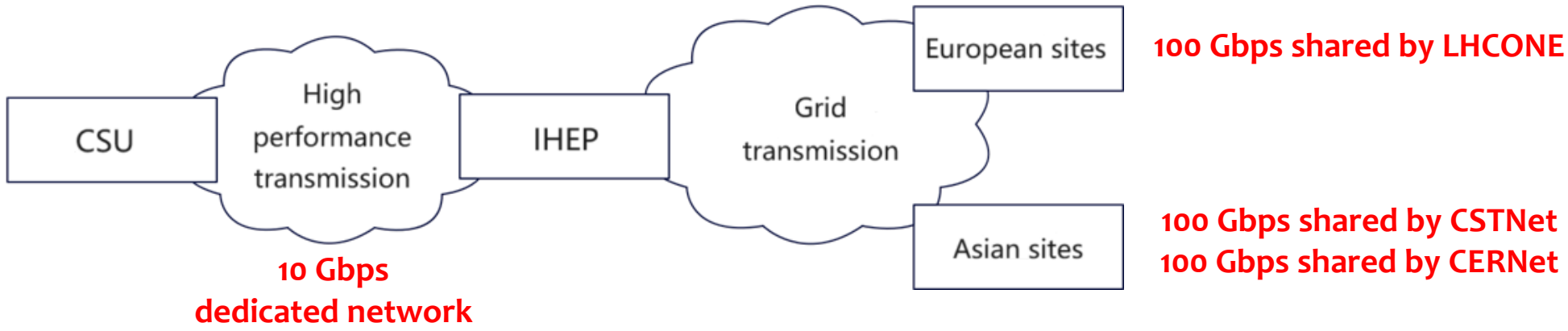
- **Formatted metadata methods for workflow system, keys includes:**
  - Production batch, log file path, job finished time, etc.
  - Which could not got directly from remote jobs.
- **Method not directly provided from Rucio commands:**
  - Scope modification.
  - File removal.
  - Batch files upload with divided backend jobs or submit to local computing cluster.
  - Automatic container creation based on ‘HERD’ policy.
- **Some daemons:**
  - Automatic account synchronizer from IHEP-SSO and HERD-IAM.
  - Automatic register and rules creation (under development).
- **Other common Rucio methods but packaged in a better model for HERD production.**

Network and Data Transfer in HERD

Network and Data Transfer

Network link should be established between HERD data centers,

- A high-performance data transmission network of **no less than 10Gbps** from CSU to IHEP.
- Network between IHEP and other European sites share a **100 Gbps link by LHCONE**.
- IHEP and other Chinese sites shared **100 Gbps link by CSTNet and CERNet**.



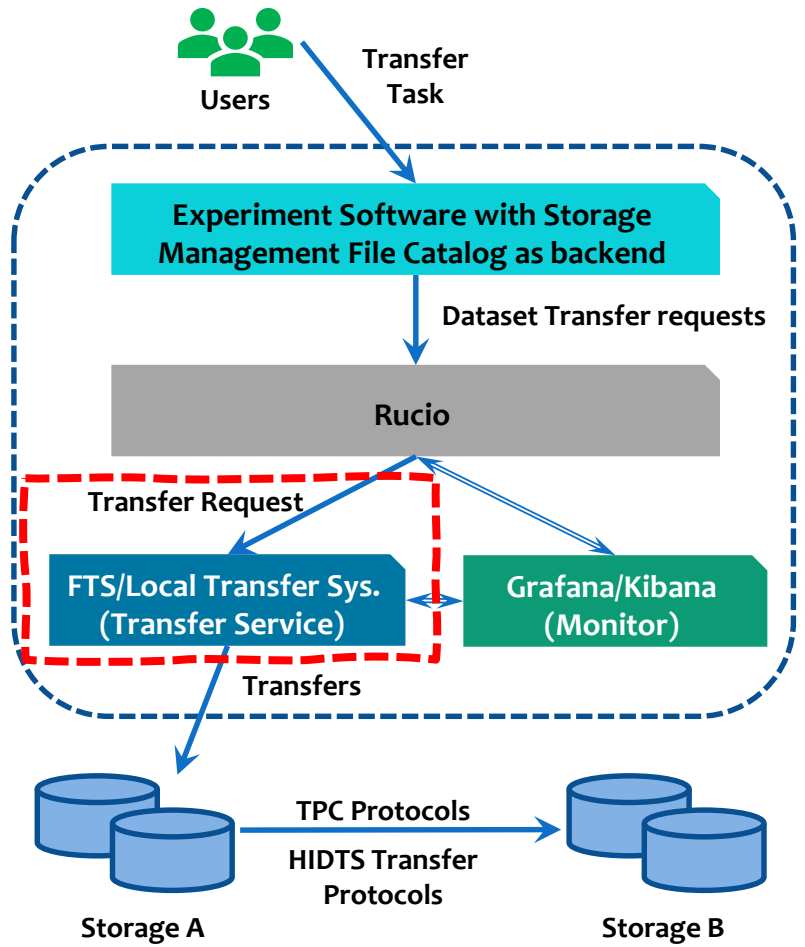
Rucio Trasnfer Plugins

We are also developing some Rucio Transfer Plugins for IHEP local data transfer system (HiDTS).

- **Working as another FTS plugins but for HiDTS, an IHEP self-developed data transfer system.**
  - HiDTS uses a commercial data transfer system as backend,
  - But IHEP-CC developed a RESTful API for HiDTS, allowing user submits local storage data transfer.
- **We are developing the plugins with allow Rucio use HiDTS as transfer system.**

So that we could support more local storages,

- **IHEP has lots of storage sites not supporting normal protocol such as Xrootd or WebDAV…**
- **Serving for future non-WLCG type experiments or big science devices.**



Authentication and Authorization in HERD

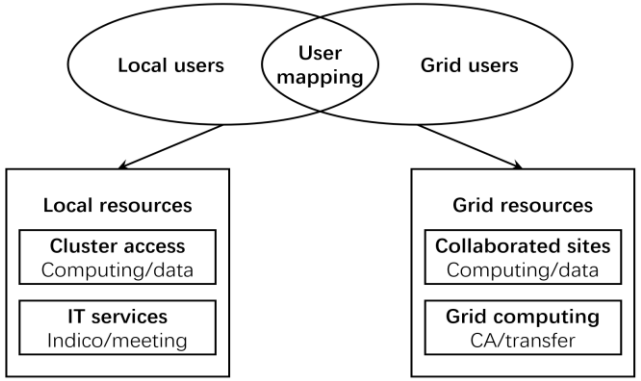
Authentication For HERD

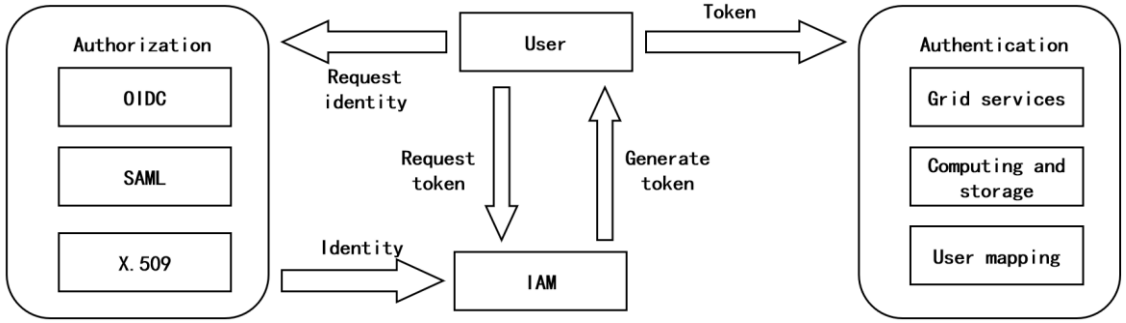
HERD follows WLCG standard and development tendency:

- Based on **Certificates and WLCG Sci-Tokens**.
- Both INFN and IHEP already support WLCG authentication standard.

So, Indigo-IAM for HERD,

- It is the suggested Grid user management system by WLCG.
- Support user management, Access control, Authentication, Auditing and monitoring.
- Will be highly integrated to data processing with Grid resources.
- Already support user authentication by INFN and IHEP SSO with eduGain.





Summary

Summary

HERD experiment plan to processed 90PB dat in next 10 years since 2027.

- Data centers from China and Europe will take the tasks of data processing.
- HERD distributed computing system is designed to unify and manage distributed computing and storage in data processing.

“One entrance, all computing tasks”

- Job entrance tools is developed to manage jobs of all type of tasks to all type of computing clusters.

“One API, all data management tasks”

- Storage management API integrated to HERD software and workflow is developed to provide storage operation methods.

Transfer and authentication system follows WLCG tools and standard.

- Meanwhile a Rucio transfer plugin based on non-WLCG transfer system is developed.

Thanks for your attention

Backup