Integration of GPU and Container with Distributed Cloud for Scientific Applications

Academia Sinica Grid Computing
Edward Wu

Academia Sinica Grid Computing (ASGC)

- T1/T2 site in WLCG in Taiwan
- We also build a computing platform for science named
  DiCOS: Distributed Cloud Operating System
- We support various scientific computing needs

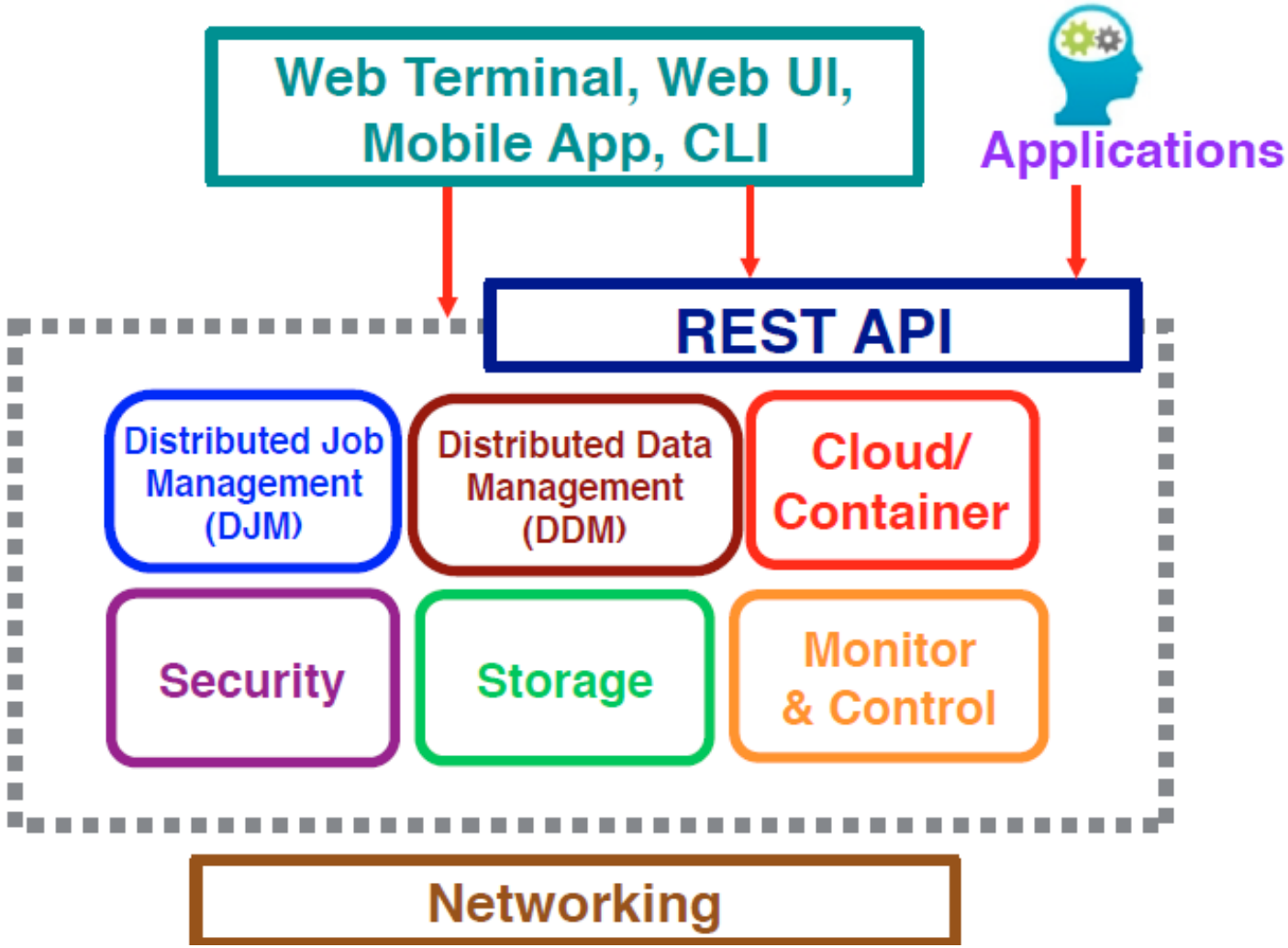
Application (Science)

- ATLAS (高能物理)
- Alpha Magnetic Spectrometer (AMS, 粒子天文)
- KAGRA, VIRGO (重力波)
- TEXONO (微中子)
- World Wide Grid Computing (CERN)
- Advanced Networking (iCAIR)
- Proton Therapy (NCU, CGU/CGMH)
- Disaster Mitigation (NCU, Southeast Asia)
- Soundscape (Southeast Asia)
- Earth Science
- Cryo-EM (2017)
- Computational Biology (2017)
- Bioinformatics (U. Chicago)

DiCOS Software Stack

Core software had been used by World Wide Grid (WLCG) connecting 300+ sites and processing over 100PB data per year for more than 5 years.

In addition to the core components, ASGC develops web and mobile interface, API, parallel computing support over Openstack, automatic installation system, private cloud integration, Container support etc.



Integration of GPU and container into DICOS

- Container:
  - A fast way to build customized computing environment, making it easier to control/debug computing environment
  - Extend the job types we can support
- The requirement of GPU in scientific computing is increasing:
  - Tenserflow (Institute of Information Science in Academia Sinica)
  - Gravity wave (iKAGRA)
  - Lattice QCD (Institute of Physics, Academia Sinica)
  - CryoEM (Institute of Biological Chemistry, Academia Sinica)

First case: Genomic analysis

- Biomedical Data Commons (BDC) under Center for Data Intensive Science (CDIS) in University of Chicago
- Data commons co-locate data, storage and computing infrastructure with commonly used software services, tools & apps for analyzing and sharing data to create a resource for the research community.*
- We provide DICOS computing platform to support RNA/DNA sequencing analysis from GEUVADIS and 1000 GENOMES data set.

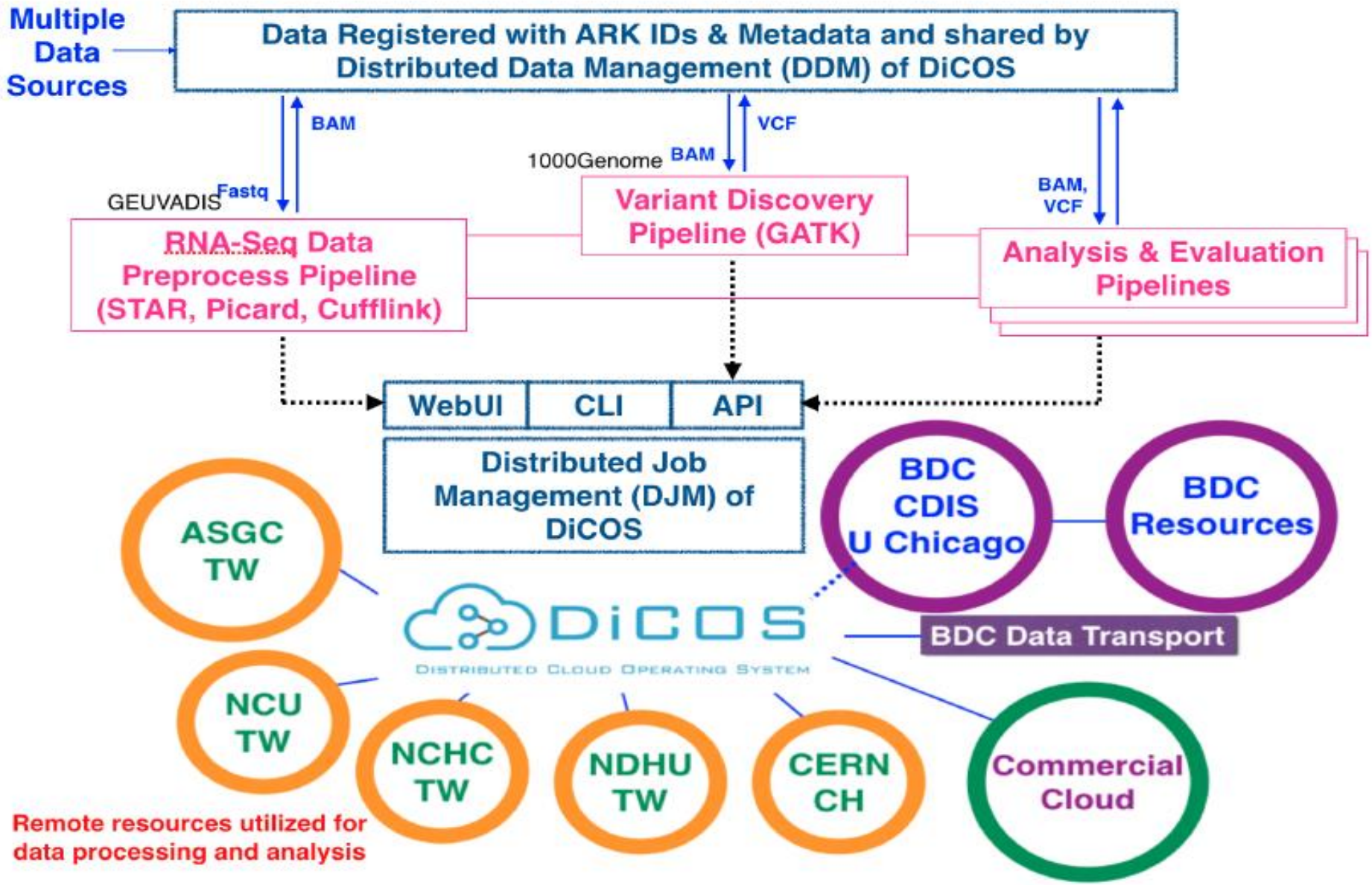
*Robert L. Grossman, Allison Heath, Mark Murphy, Maria Patterson and Walt Wells,
A Case for Data Commons Towards Data Science as a Service, IEEE Computing in Science and Engineer, 2016.

User requirement for container

- Data Confidentiality is much more emphasized for biomedical data
- Customized environment for different jobs
- Independent environment for jobs to prevent environmental issues

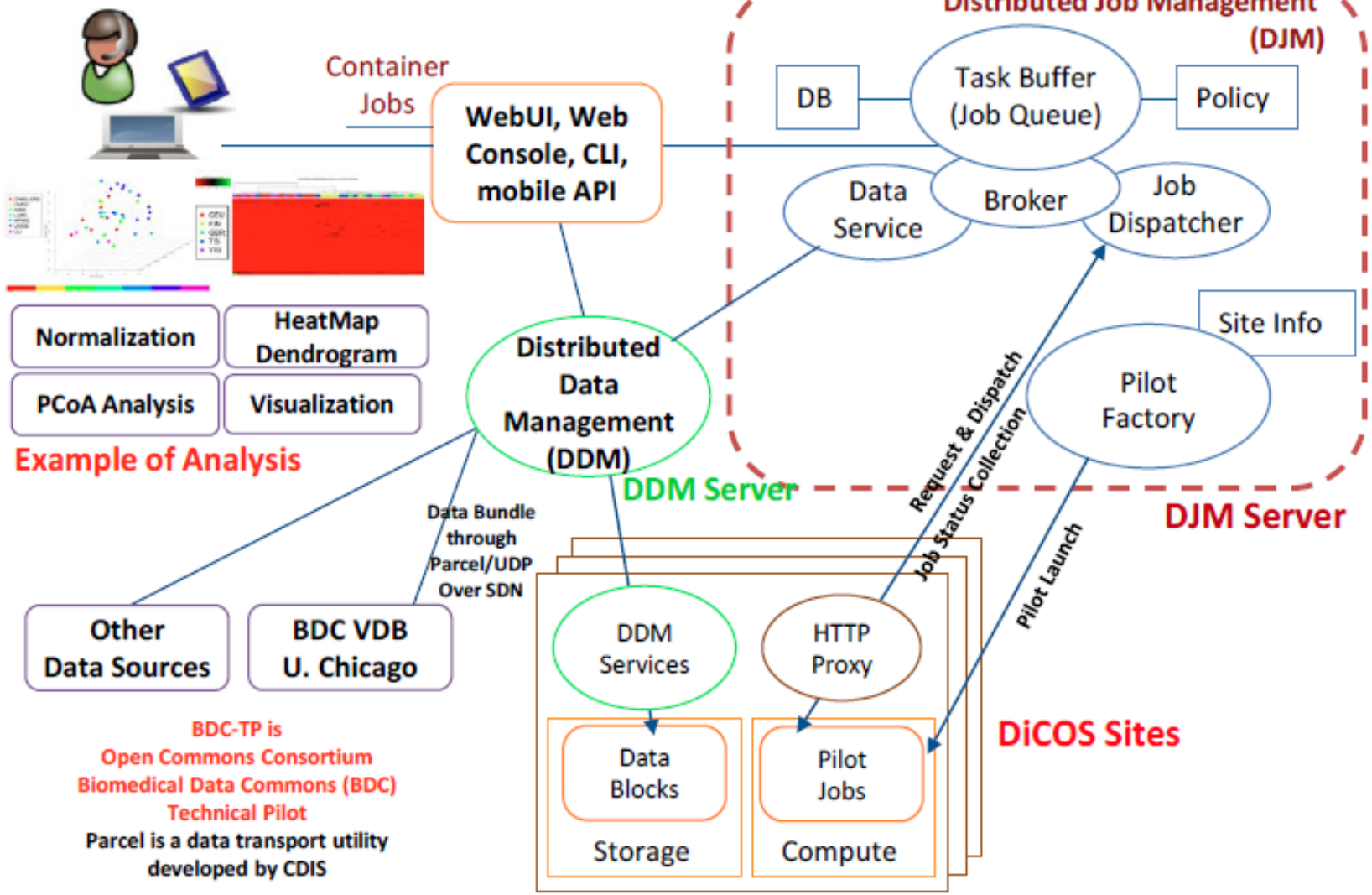
We collaborate with U. Chicago to support genomic analysis

DiCOS Supports Various Pipelines of Bioinformatics Analysis by Scalable & High-Throughput Distributed Infrastructure (testbed in BDC-TP and ASGC)

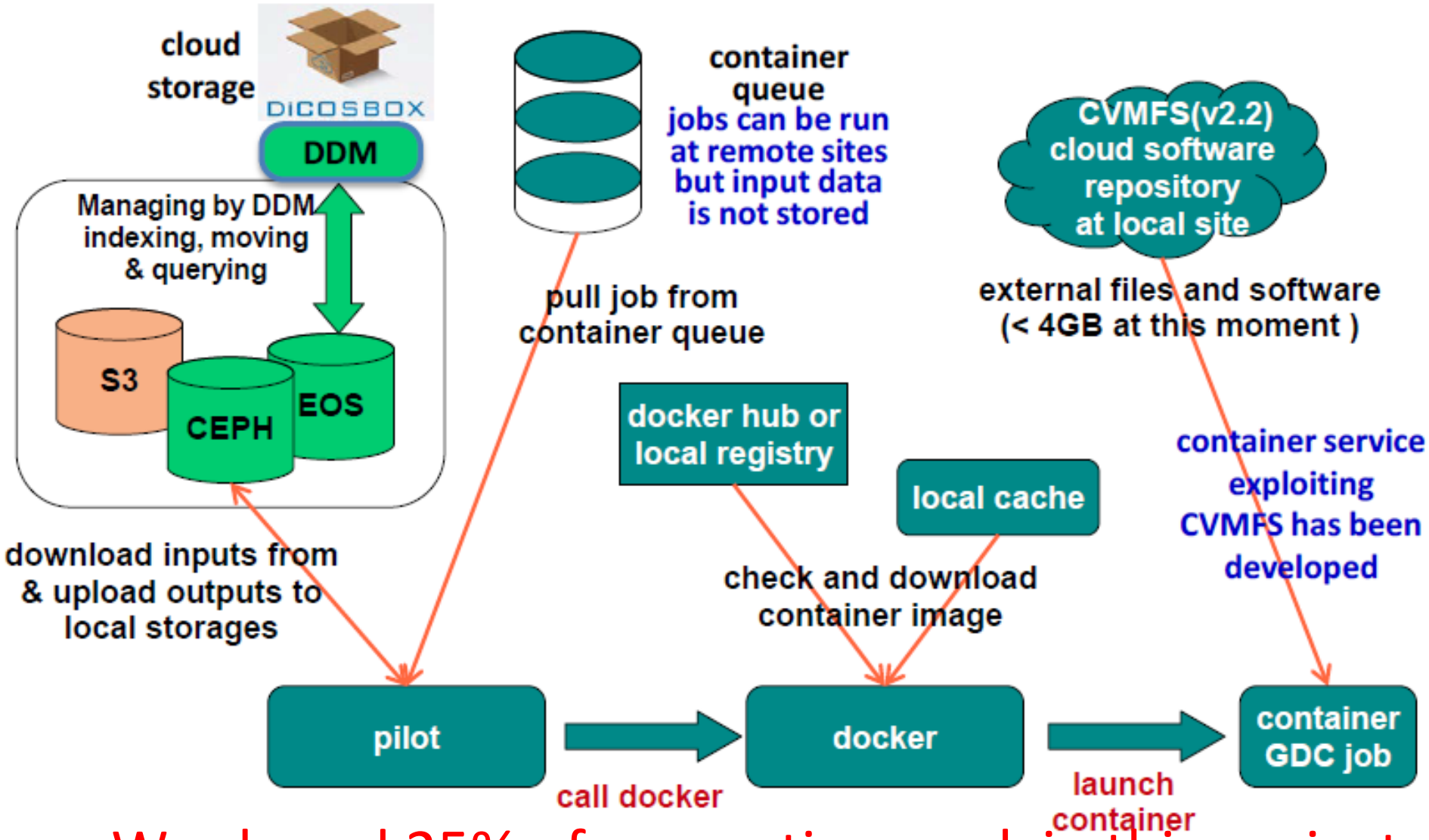


DiCOS for BDC-TP: RNA-Seq Data Analysis Scenario

Job is run at best available site and data is moved transparently by File Transfer Service (FTS)



Pilot Integration with Container



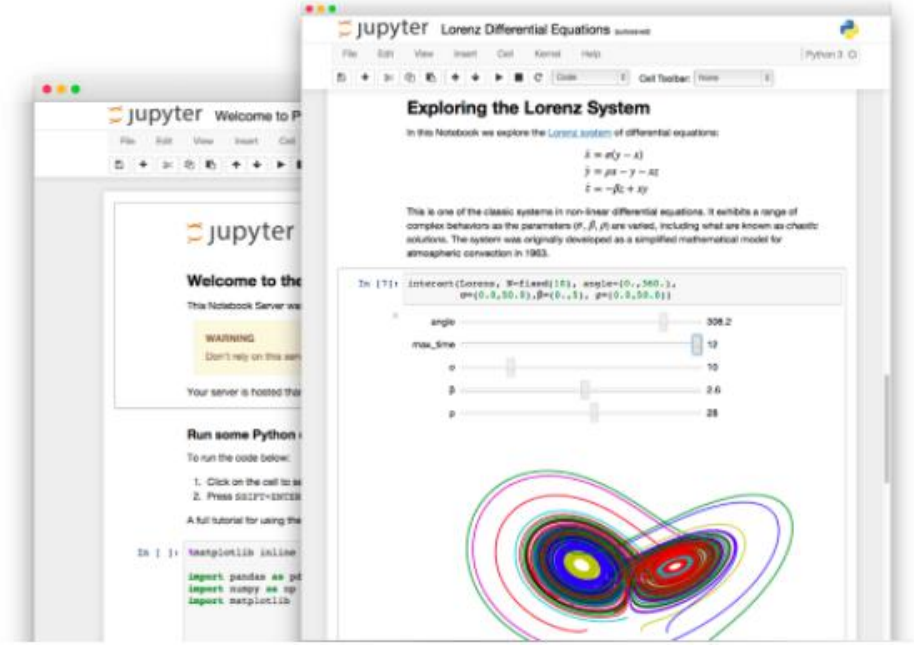
We shared 25% of computing work in this project

ASGC GPU farm

- Two models available
- Consumer GPU
  - GTX 1080Ti: 64 in total.
  - 8 * GPUs per compute box.
  - 128GB system memory.
  - 1TB SSD local hard-drive.
- Server GPU
  - Tesla p100: 16 in total
  - 4 * GPUs per compute box.
  - 128GB system memory
  - 1TB SSD local hard-drive
  - NV-link
- 8 * v100 model is on-going.

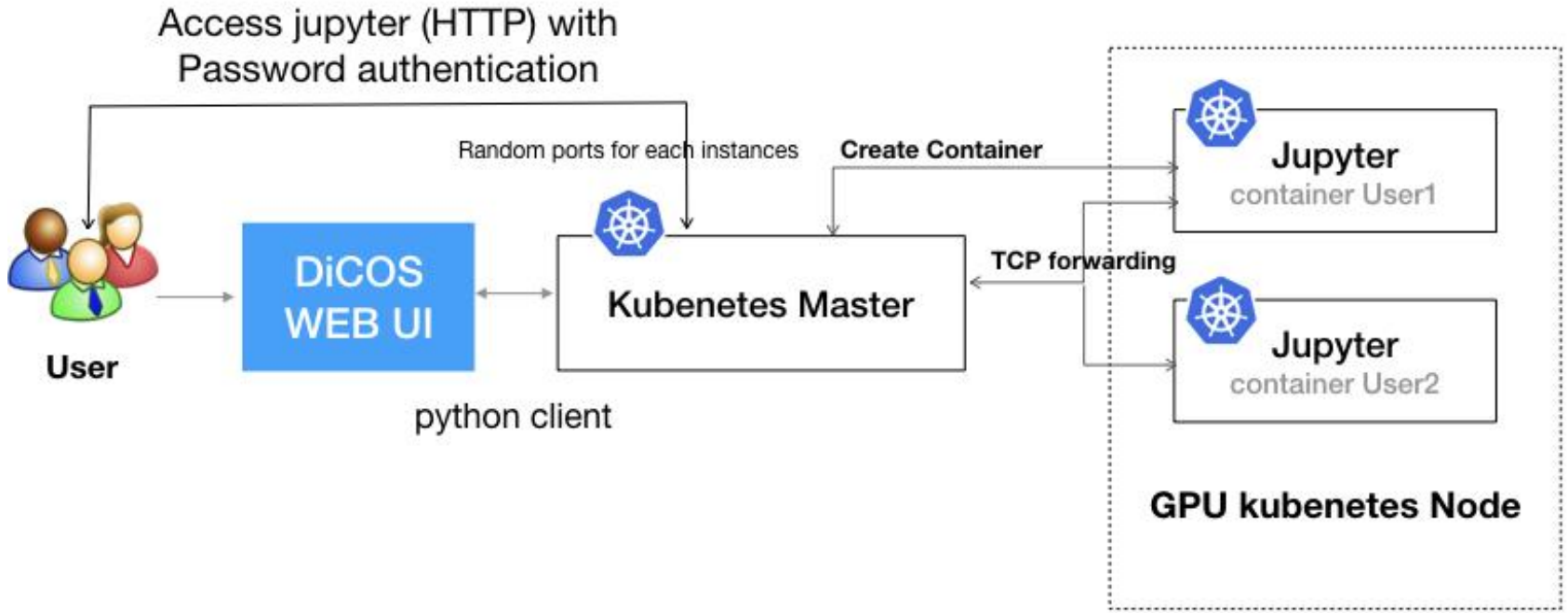
Popular notebook for research
-- Jupyter Notebook

- Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text
- It have been popular in data science and machine learning.

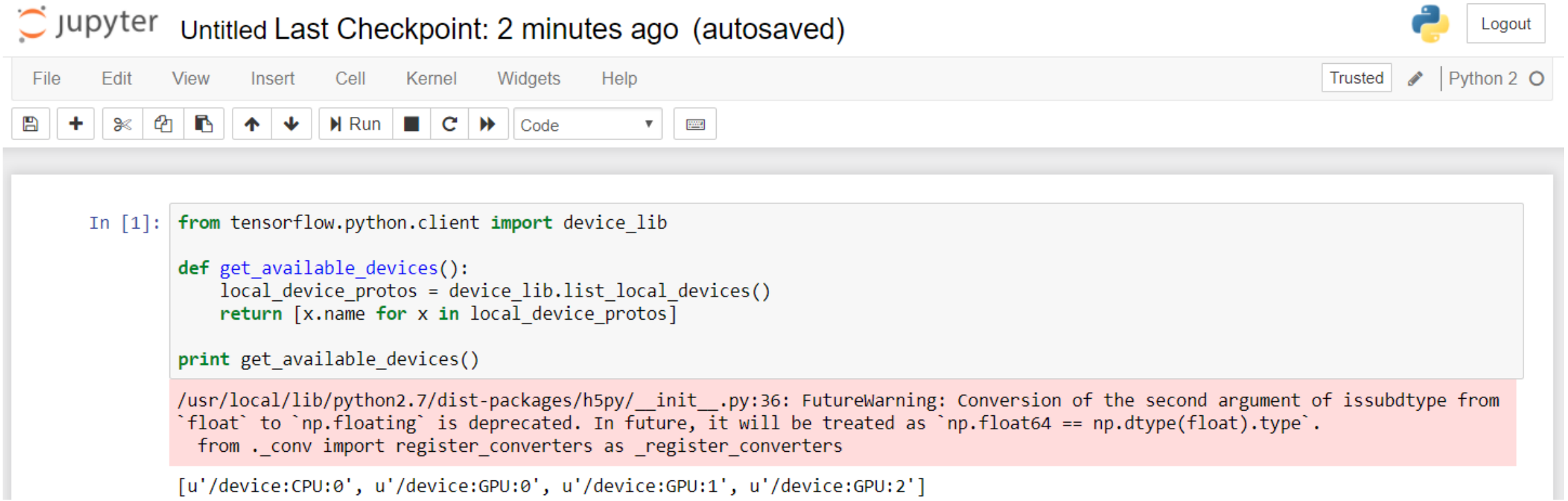


Jupyter notebook on GPU

- User can use jupyter notebook running on GPU on Web-UI
- Deep learning libraries - Keras, Tensorflow, OpenCV, etc.
- One Master with 12 Nodes with 8 GPU each



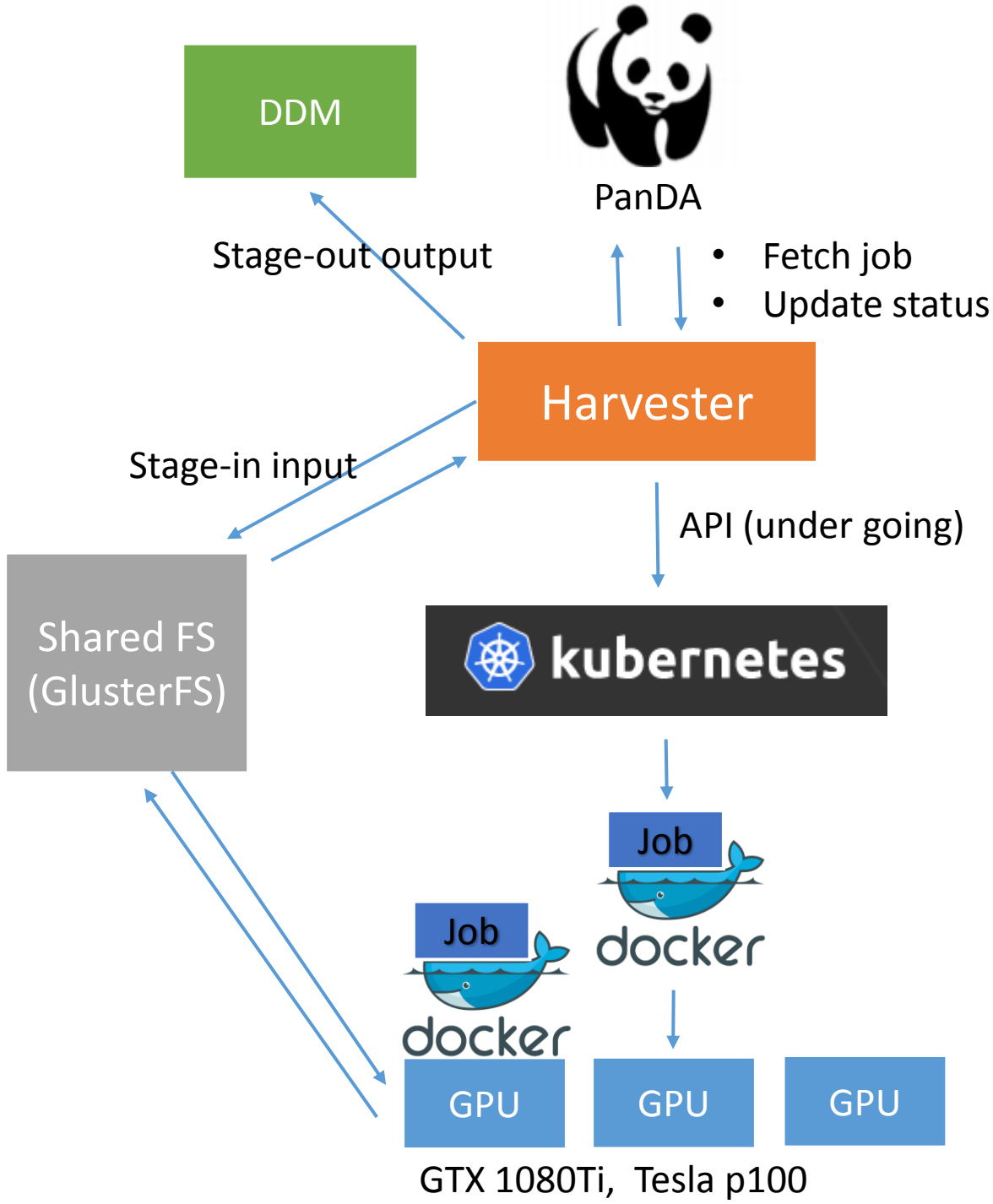
Check GPU from Jupyter



Future: GPU queue for grid computing

- For Harvester:
  - Push jobs into cluster. Unlike pilot, which pull jobs.
  - Harvest can optimize resource allocation among differences of architectures
  - For example, it can assign number of CPU/GPU for different MPI jobs
  - Various (new) workflows, such as job/event-level late-binding and jumbo jobs

- The integration between Harvester and kubernetes is developing in cooperate with people in Harvester team.



Conclusion

- Take advantage of container, we make our computing platform more flexible to different computing requirements.
- To fix the need for GPU computation, we provide a Jupiter notebook service on GPU for science.
- A GPU queue in DICOS is under going.

Questions ?
Thank you for attention.