



Introduction to Asi@Connect/TEIN*CC & APAN APRP WG Activities

KISTI/KREONET
Jeonghoon Moon
9th Oct 2023



1. Overview of the project

- Overview & Background

2. Activities of the project

- Activity 1
- Activity 2
- Activity 3

3. APAN & APRP WG

- APAN 56th Meeting
- APRP WG

4. Future plan & Conclusion

Overview of the project

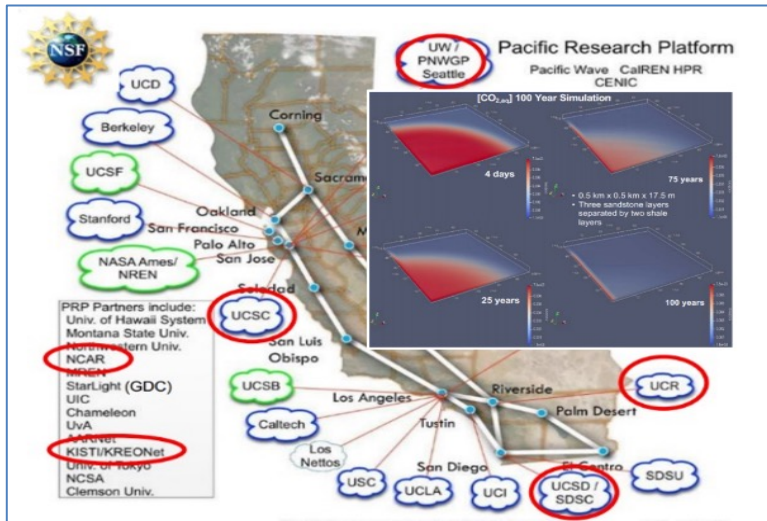
[1] Overview & Background

Overview of APRP Project

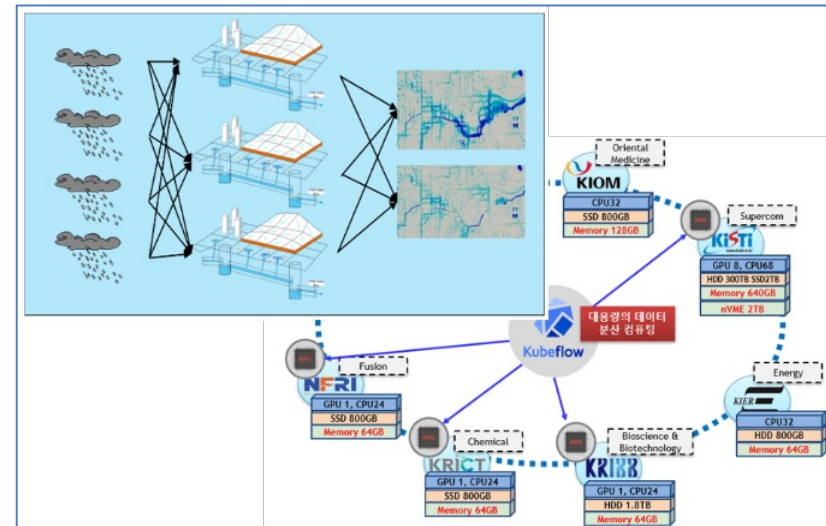
- **Title : Building a high bandwidth distributed HPC**
- **Work Package : WP4**
- **Participants : Korea, Australia, Malaysia, Pakistan**
- **Duration : 1st April 2022 – 31st July 2023**
- **Budget : 150K Euro**
- **Lead & Co-applicant :**
 - Jeonghoon Moon : Korea Institute of Science and Information Technology(KISTI)/KR
 - Andrew Howard: National Computational Infrastructure(NCI)/AU
 - Mohammad Asif Khan : Perdana University/MY
 - Nor Asilah Wati Abdul Hamid : University Putra Malaysia(UPM)/MY
 - Syed Asif Raza Shah : Sukkur IBA University(SIBAU)/PK



- Increasing big data → Required high throughput/capacity network → Big data super highway
- Lack of IT infrastructure → Computing resources → Distributed/Shared HPC
- Linked ScienceDMZs : Research Platform + Kubernetes = High bandwidth distributed HPC



Example1: Using distributed computing resources over US NSF funded PRP and Computing for analysis of CO₂ sequestration



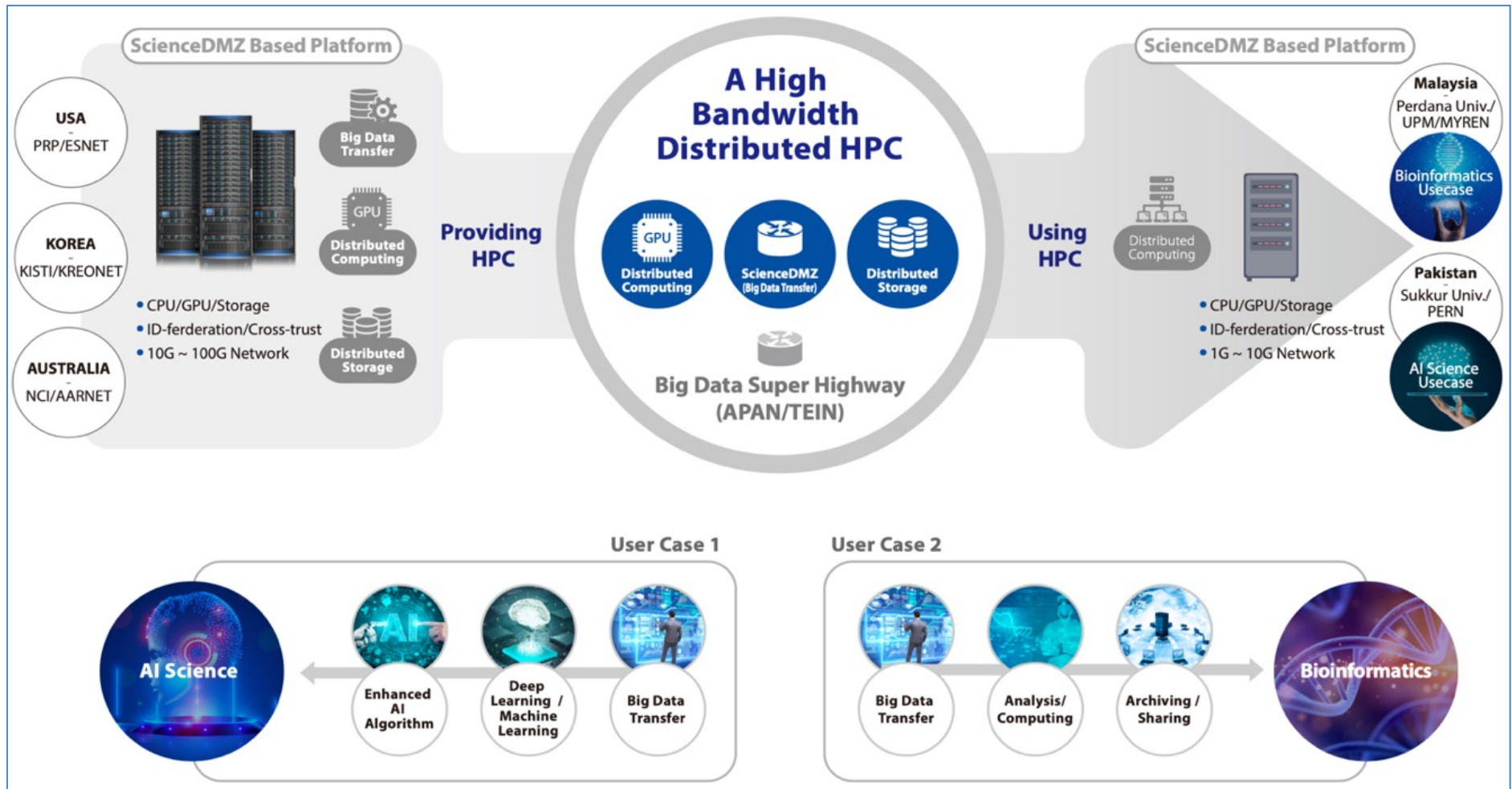
Example2: Using distributed computing resources and LSTM data over Korea government project for climate analysis by KRP

PRP→NRP→GRP→APRP→Asi@Connect project

Activities of the Project

- [1] Activity 1
- [2] Activity 2
- [3] Activity 3

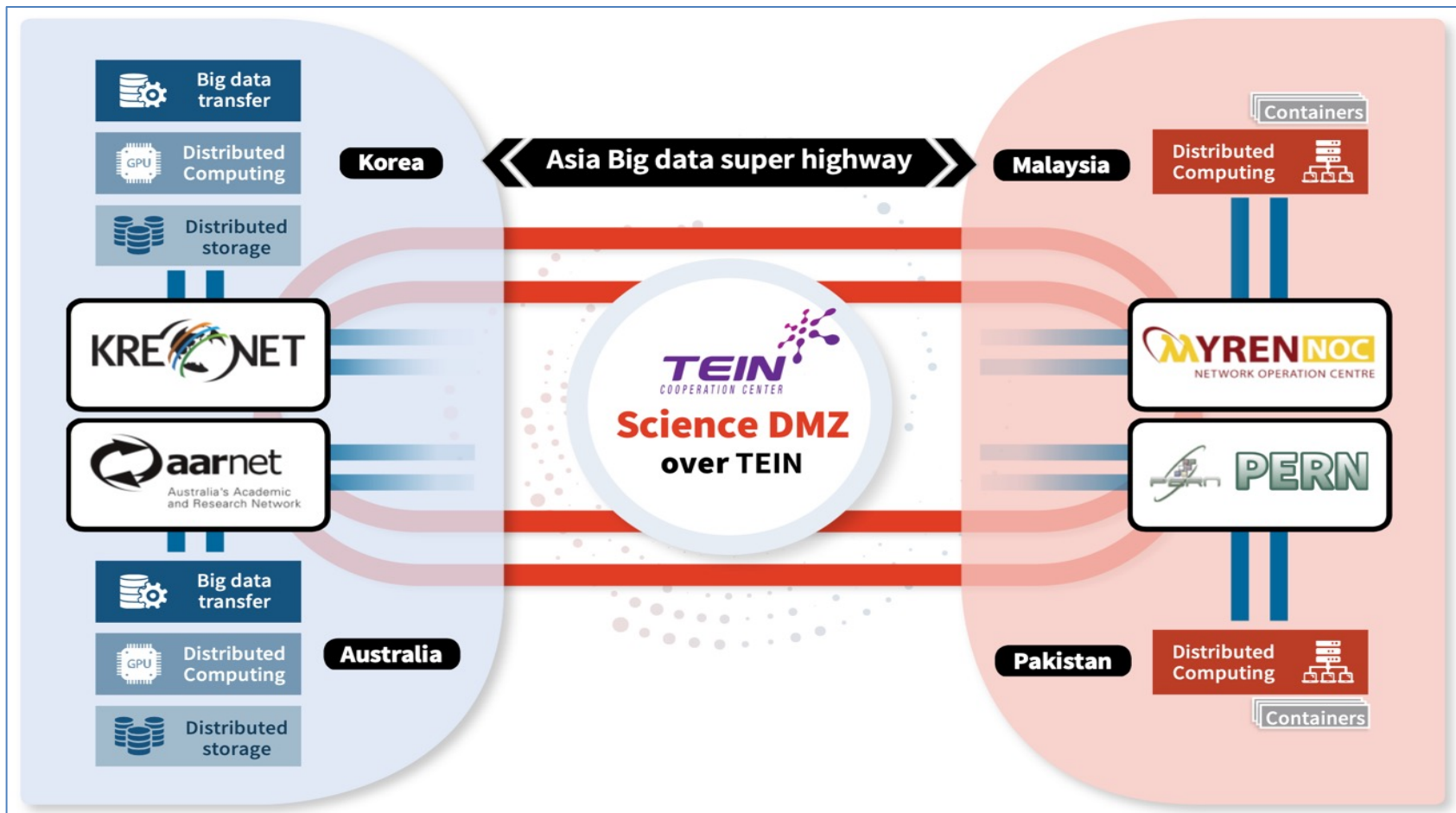
- ScienceDMZ-based high bandwidth networking architecture (participating countries via TEIN)
- Interconnect available computing recourses in a distributed environment
- Resource of computing servers will be managed by container cloud technology



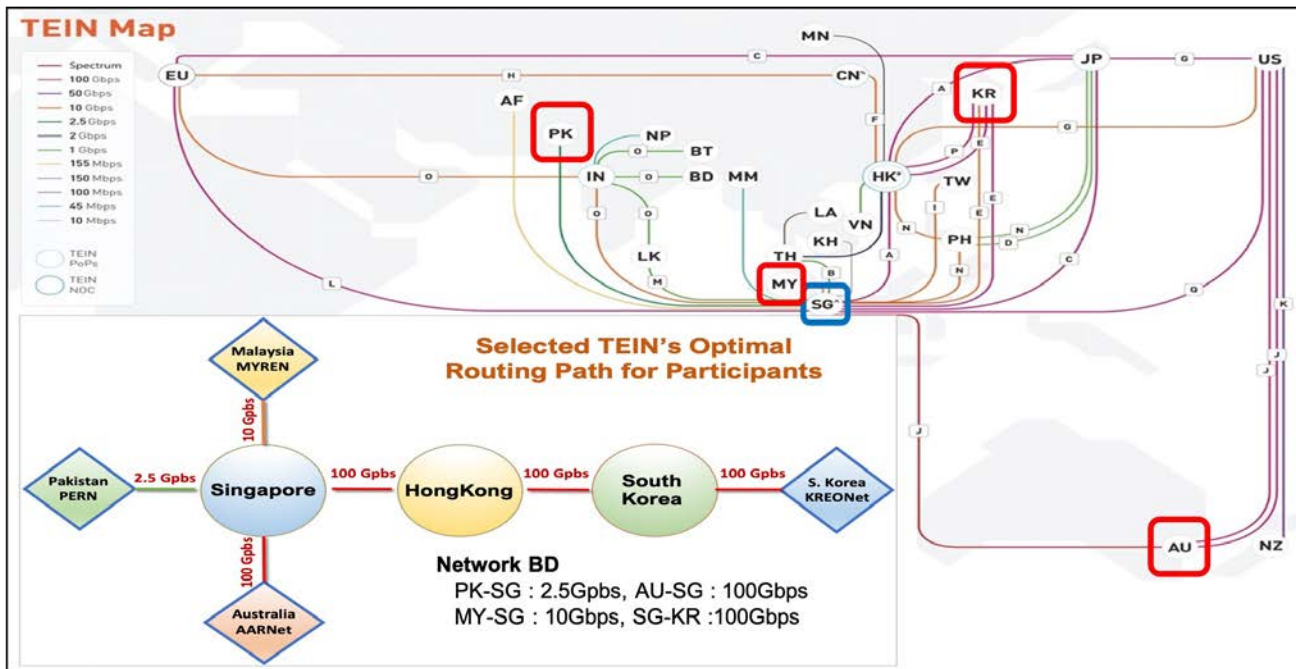


- **Activity1: Building a big data super-highway based on TEIN**
 - KREONET(KR), AARNet(AU), MYREN(MY), PERN(PK)
 - Deploy ScienceDMZ/DTN
 - Big data transfer via ScienceDMZs
- **Activity2: Building a distributed HPC platform**
 - Research Platform by Kubernetes based on container
 - CPU/GPU/Memory and Storage
- **Activity3: Presenting pilot use cases**
 - Bio-Science (MY)
 - AI-Science (PK)

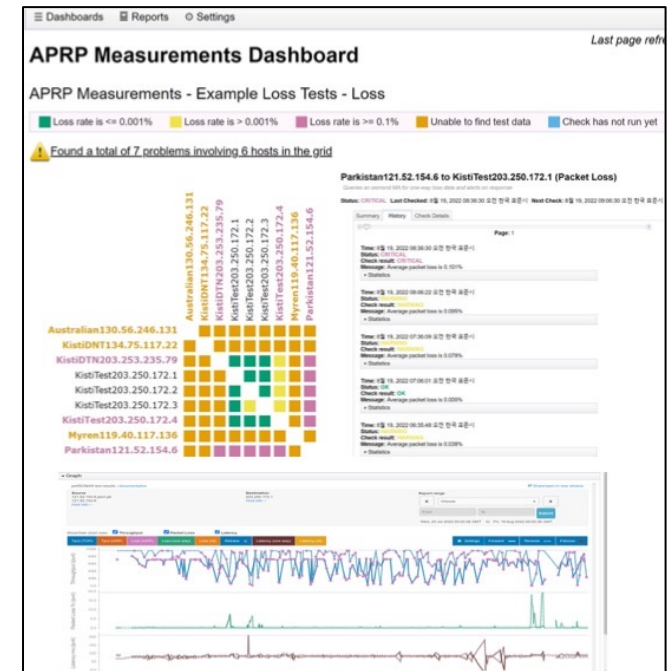
- **Activity1: Building a big data super-highway based on TEIN**
 - Interconnecting participant NRENs via TEIN (KROENET, AARnet, MYRen, PERN)
 - Installing DTN at each participant institutes (ScienceDMZ)
 - Measuring the ScienceDMZ using a monitoring system (PerfSONAR)



- **Activity1: Building a big data super-highway based on TEIN**
 - Interconnecting participant NRENs via TEIN
 - Installing and adding local DTNs
 - Complete setup PerfSONAR

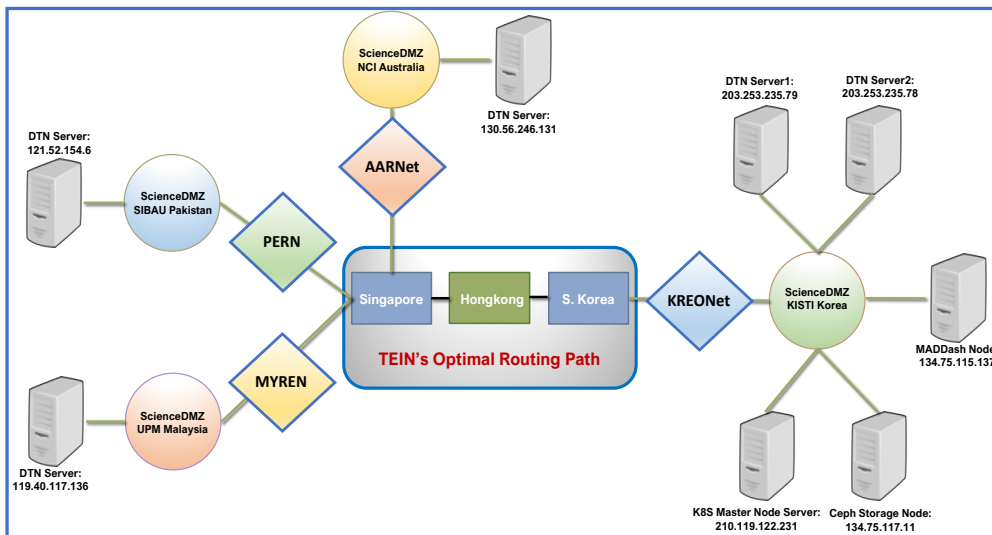


NREN topology & Bandwidth

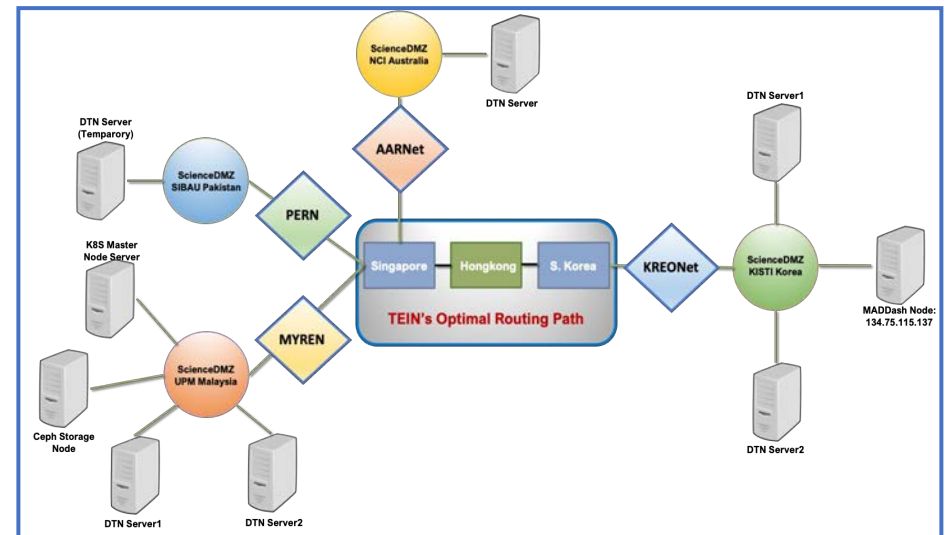


PerfSONAR

- Installing and adding local DTN
 - Configuration each DTNs base on the local network topologies
 - Re-locate DTNs and network paths



The previous end-to-end network topology of temporary DTN servers



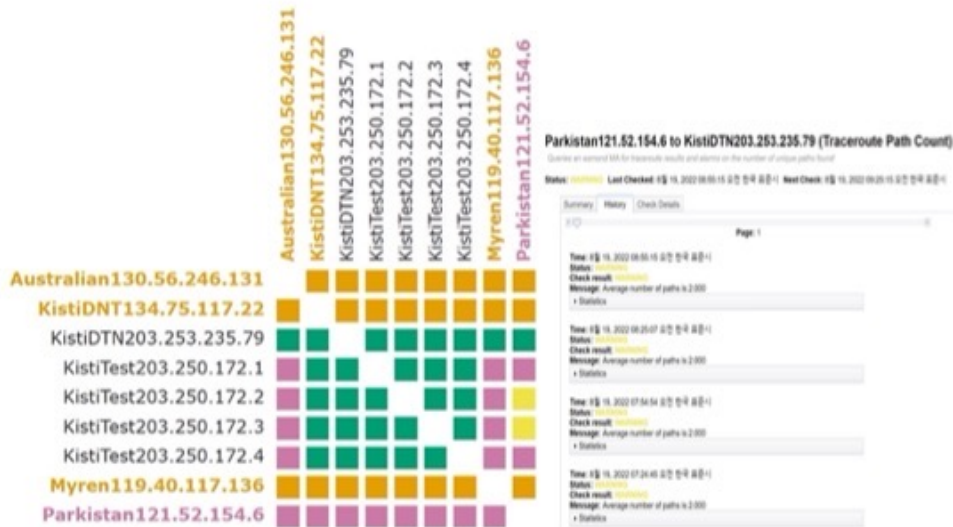
The updated end-to-end network topology of DTN servers at participating countries

- Measuring the ScienceDMZ using a monitoring system
 - PerfSONAR – Measurement of Network performance and status
 - Docker-based perfSONAR install and adding a MaDDash

APRP Measurements - Example Traceroute Tests - Path Count

■ Paths = 1 packets
 ■ Paths > 1
 ■ Paths > 2
 ■ Unable to find test data
 ■ Check has not run yet

! Found a total of 5 problems involving 4 hosts in the grid



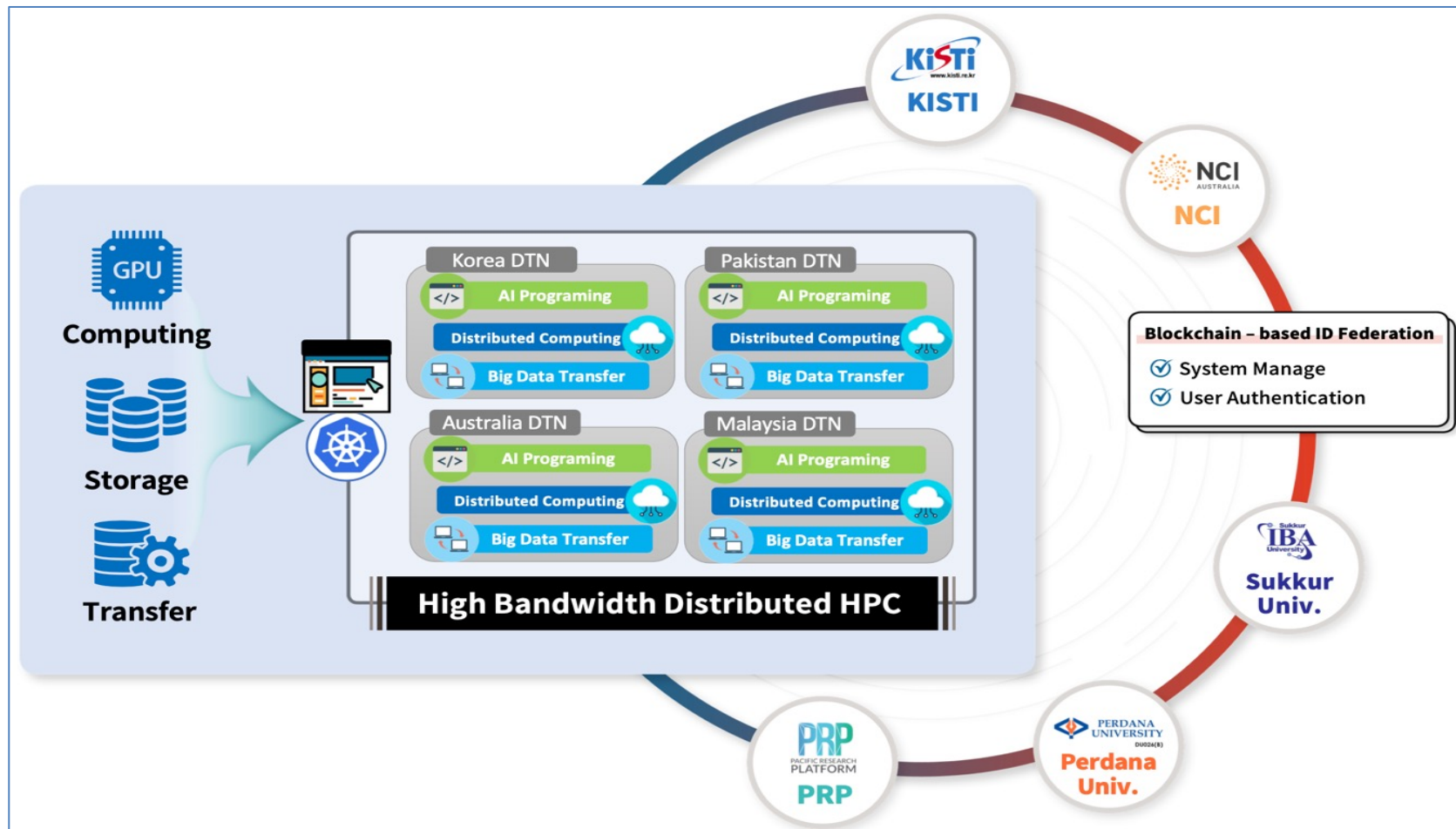
MaDDash Dashboard: Example of Traceroute Tests among DTN servers



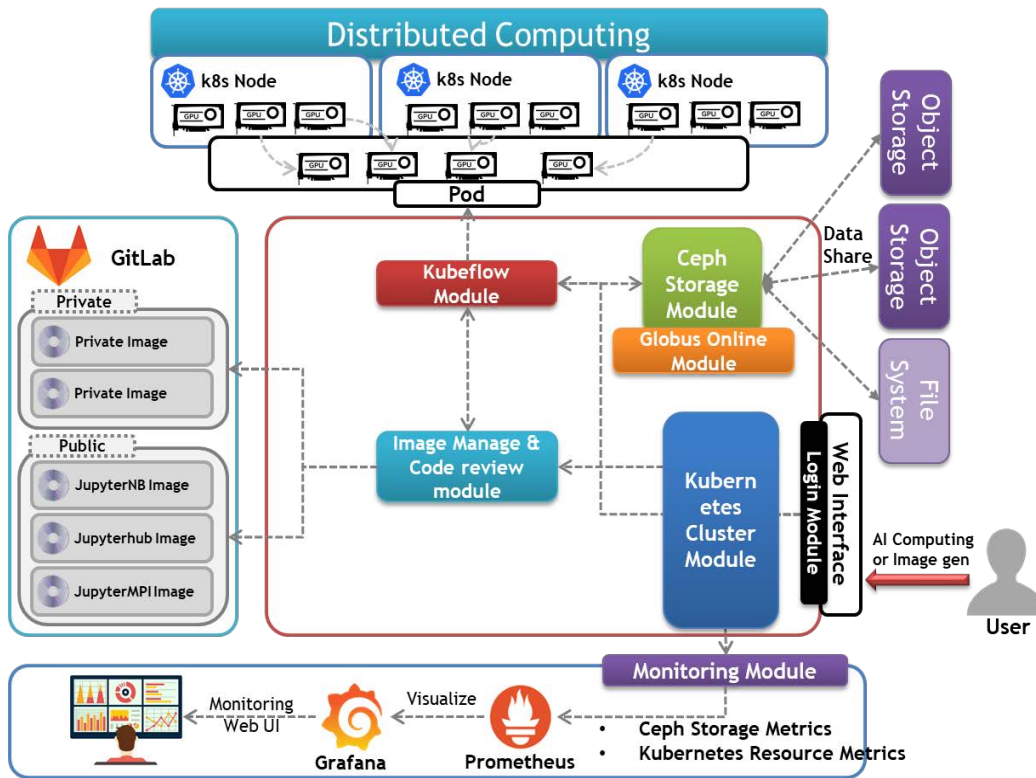
Docker-based PerfSONAR test-points running at participating institutions

- **Activity 2: Building a distributed HPC platform**

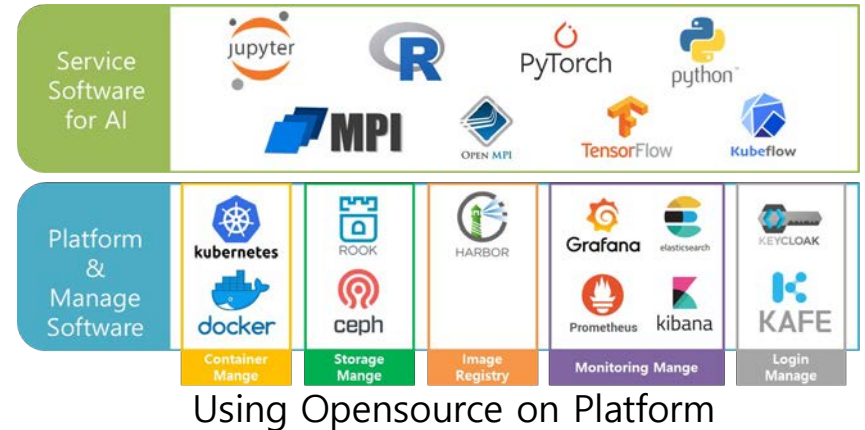
- Building a platform to manage CPU/GPU computing resources, including storage by Kubernetes
- Building block-chain based ID federation
- Building an easy user interface



- **Building a platform to manage CPU/GPU computing resources, including storage**
 - Building a platform to manage CPU/GPU computing resources, including storage by Kubernetes
 - Kubernetes servers specification
 - Using open source



Asia-Connect Platform Architecture



Using Opensource on Platform

Server Role	GPU	CPU	Mem ory	Storage	Network
Master Node	No GPU	Intel(R) Xeon(R) Gold 622 6R CPU @ 2.90GHz 32-Core Processor	DDR4 2 51GB	HDD 480GB	1GB
Pakistan Node	NVIDIA A40 1G PU	AMD EPYC 7502 32-Core Processor	DDR4 188GB	SSD 960GB NVMe 960GB	1GB
Malaysia Node	NVIDIA A40 1G PU	AMD EPYC 7502 32-Core Processor	DDR4 188GB	SSD 960GB NVMe 960GB	1GB
Storage Node	No GPU	AMD EPYC 7402 24-Core Processor	DDR4 188GB	HDD 58TB	100GB

Kubernetes Servers Specification

Opening ceremony



MoU

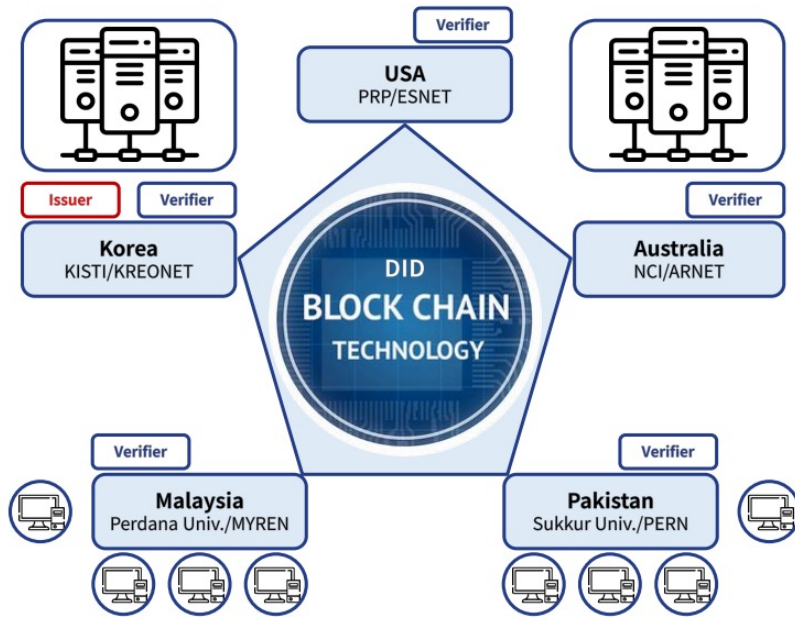


Built Platform at UPM

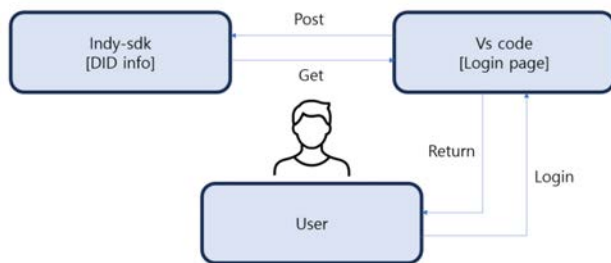


Opening ceremony & Workshop





Asi@Connect Platform Block chin Diagram



A simple process

Function	Description
open_wallet	Call the wallet.open_wallet function provided by the Indy SDK to open the wallet. If the wallet doesn't exist, create it and open it.
get_did_info	Call the Indy SDK's did.list_my_dids_with_meta function to retrieve information about the user's Distributed Identifier (DID) from the wallet. The user's DID information is returned in JSON format.
run	It runs the open_wallet and get_did_info functions. It also opens the wallet, retrieves the DID information, and stores it in the global variable did_info.
compare_did_info	Compares the DID and verkey information entered by the user with the DID information stored in did_info. Returns True if the entered DID and verkey exist in did_info.
route '/'	Provide a login form on a web page that requires the user to enter their DID and Verkey. When the user tries to log in after entering the DID information, the / route is called with the POST method. This calls the compare_did_info function with the user's DID and Verkey as arguments and tries to authenticate. Depending on the result of the compare_did_info function, the user is connected to the Naver homepage if the authentication is successful, or redirected to the Google homepage if it fails.

Table Core functions in webpage code

1. **Source Code development for AI model** ←
2. **Create image importing the source code** ←
3. **Distributed Computing Yaml file generation and implementation** ←
4. **Check the Distributed Computing result** ←
5. **Next job send a yaml file to kubelet** ←

How need resources?

GPU 개

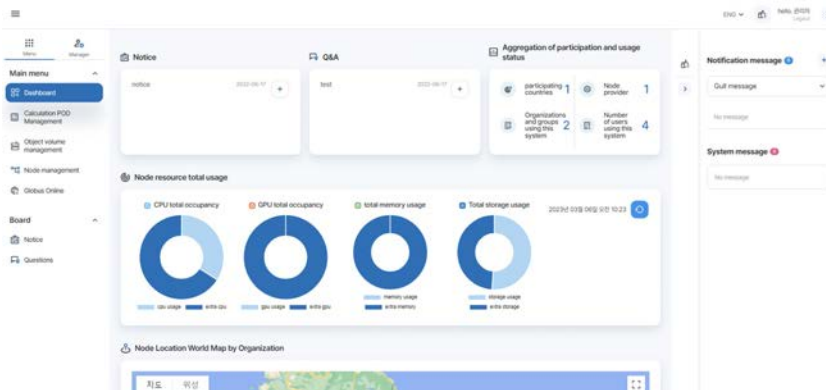
CPU Core

Memory MB

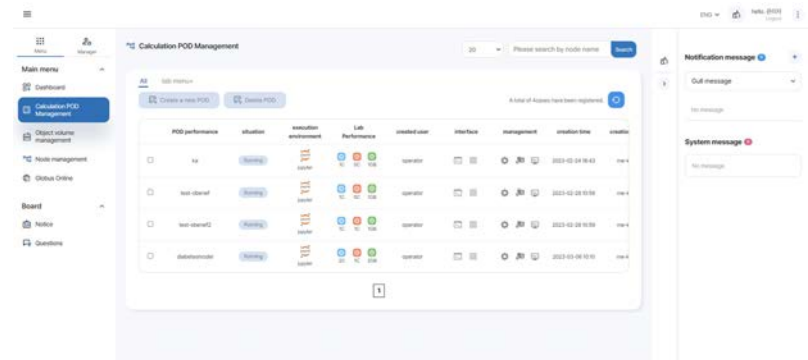
Select your image or generation

- image1
- image2
- Generation

Attach the source code

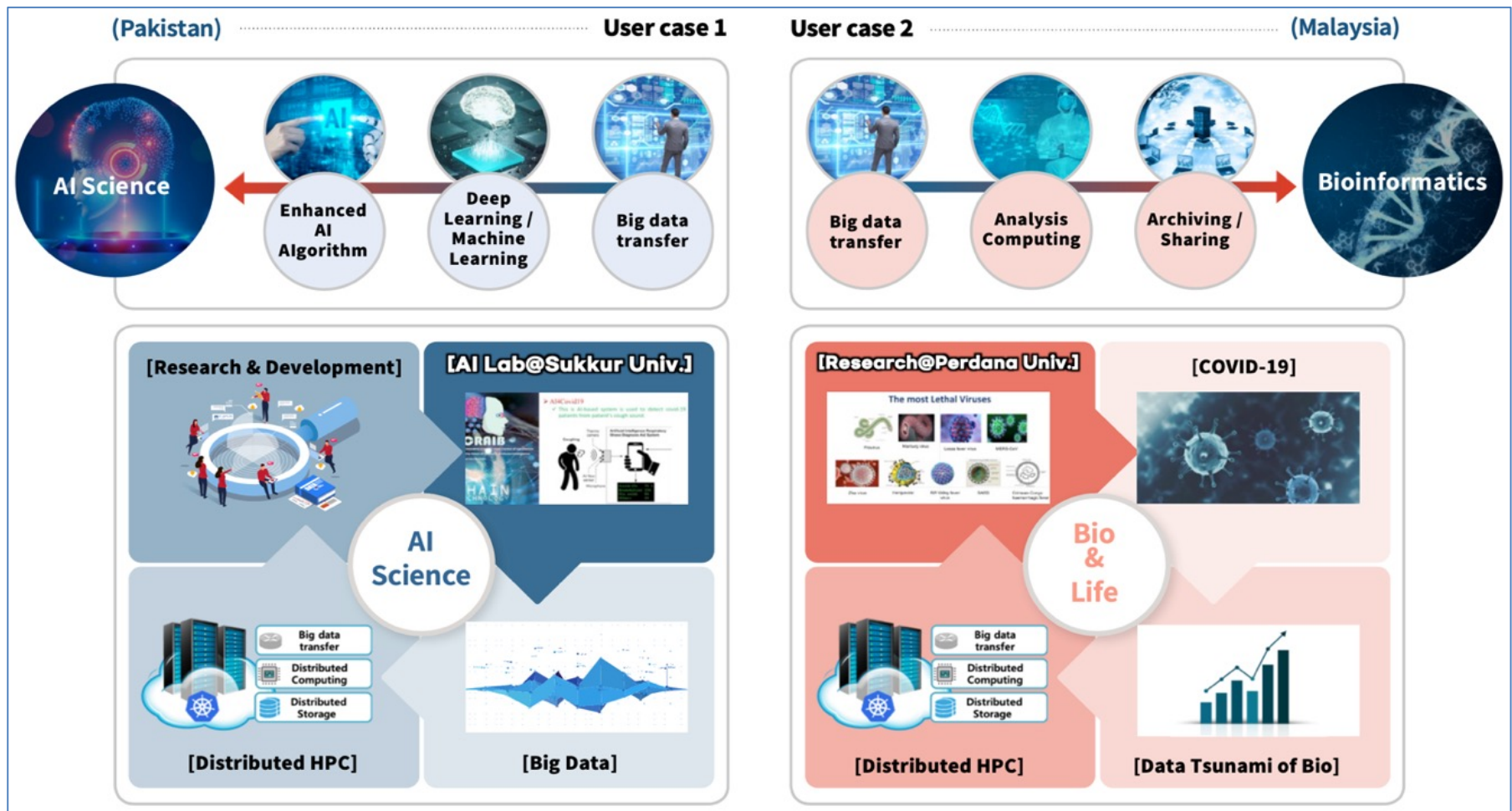


Check the resources



Status of the selected pod

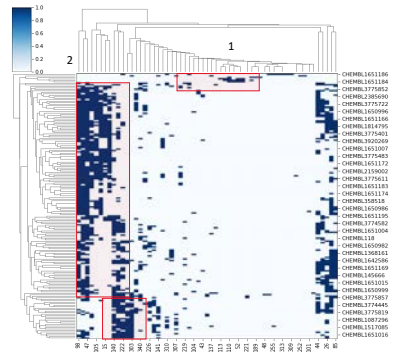
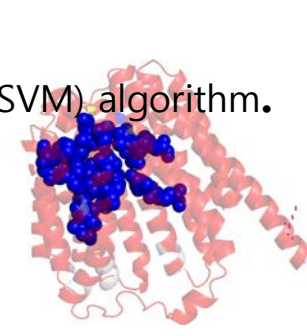
- **Activity 3: Presenting pilot application use-cases**
 - Bioinformatics-based Science use-cases
 - AI-based Science use-cases



- **Activity 3: Presenting pilot application use-cases**
 - Bio-Science use-cases by Asif Khan(Perdana Univ. Malaysia and Turkey)
 - Multiple sequence alignment (MSA) of large number of sequences on HPC
 - Applying distributed HPC in Drug Discovery and Molecular Simulation
 - AI-Science use-cases by Asif Raza (Sukkur-IBA Univ. Pakistan)
 - Multiple object tracking for aerial view images
 - Distributed training(MNIST)/Support Vector Machine (SVM) algorithm.
 - Running Distributed MPI job



Multiple Object Tracking for Aerial View Images



Molecular Simulation: Binding Residues & Characteristics for 47-PHE, 19-ILE, 98-ARG, 140-PHE, 23-ILE, 22-VAL



1M sequences, with 1.5 TB RAM, 100 CPUs 1 hour by RefAligner



- Distributed MNIST (Modified National Institute of Standards and Technology database)**

```

kisti@asia-k8smaster/home/kisti/asif/tensorflow-kubeflow-examples
apiVersion: "kubeflow.org/v1"
kind: "TFJob"
metadata:
  name: "dist-mnist-for-e2e-test"
spec:
  tfReplicaSpecs:
    PS:
      replicas: 1
      restartPolicy: Never
      template:
        spec:
          containers:
            - name: tensorflow
              image: asifraza/tf-dist-mnist-testing:latest
              resources:
                limits:
                  cpu: 2
                requests:
                  cpu: 2
              nodeSelector:
                zone: kisti
    Worker:
      replicas: 2
      restartPolicy: Never
      template:
        spec:
          containers:
            - name: tensorflow
              image: asifraza/tf-dist-mnist-testing:latest
              resources:
                limits:
                  cpu: 2
                requests:
                  cpu: 2
              nodeSelector:
                zone: kisti
  
```

One Parameter Server (PS)

Two Worker Nodes

Experimenting with 2xCPU per Worker

Configuration of YAML file in which we configured one Parameter Server (PS) and Two Worker node. However, worker nodes will use the 2xCPU to train the model in distributed way.

```

kisti@asia-k8smaster/home/kisti/asif/tensorflow-kubeflow-examples Distributed Training Started on Worker Nodes
[root@asia-k8smaster tensorflow-kubeflow-examples]# kubectl get pods -o wide
NAME                                READY   STATUS    RESTARTS   AGE   IP
GATES
dist-mnist-for-e2e-test-ps-0        1/1     Running  0           9s    192.168.68.254
dist-mnist-for-e2e-test-worker-0    1/1     Running  0           9s    192.168.68.249
dist-mnist-for-e2e-test-worker-1    1/1     Running  0           9s    192.168.68.247
  
```

```

kisti@asia-k8smaster/home/kisti/asif/tensorflow-kubeflow-examples Distributed Training Completed
[root@asia-k8smaster tensorflow-kubeflow-examples]# kubectl get pods -o wide
NAME                                READY   STATUS    RESTARTS   AGE   IP
ESS GATES
dist-mnist-for-e2e-test-ps-0        1/1     Running  0           2m18s  192.168.68.254
dist-mnist-for-e2e-test-worker-0    0/1     Completed 0           2m18s  192.168.68.249
dist-mnist-for-e2e-test-worker-1    0/1     Completed 0           2m18s  192.168.68.247
  
```

Execution of distributed training on worker nodes created as pods by using Kubeflow's TensorFlow operator. Once distributed training completed then it shows the status as Completed

```

kisti@asia-k8smaster/home/kisti/asif/tensorflow-kubeflow-examples Results after Completion of Distributed Training
1660975669.850723: Worker 0: training step 10045 done (global step: 19987)
1660975669.853131: Worker 0: training step 10046 done (global step: 19988)
1660975669.855631: Worker 0: training step 10047 done (global step: 19989)
1660975669.891212: Worker 0: training step 10048 done (global step: 19990)
1660975669.894357: Worker 0: training step 10049 done (global step: 19992)
1660975669.897589: Worker 0: training step 10050 done (global step: 19994)
1660975669.901203: Worker 0: training step 10051 done (global step: 19996)
1660975669.904117: Worker 0: training step 10052 done (global step: 19998)
1660975669.906806: Worker 0: training step 10053 done (global step: 20000)
Training ends @ 1660975669.906826
Training elapsed time: 123.296151 s
After 20000 training step(s), validation cross entropy = 1630.95
[root@asia-k8smaster tensorflow-kubeflow-examples]#
  
```

Total Execution time by using CPUs

Results of distributed training of Dist-MNIST example. The total elapsed time of distributed training is 123.29 seconds by using 2xCPU on per worker.

```

kisti@asia-k8smaster/home/kisti/asif/tensorflow-kubeflow-examples
apiVersion: "kubeflow.org/v1"
kind: "TFJob"
metadata:
  name: "dist-mnist-for-e2e-test"
spec:
  tfReplicaSpecs:
    PS:
      replicas: 1
      restartPolicy: Never
      template:
        spec:
          containers:
            - name: tensorflow
              image: asifraza/tf-dist-mnist-testing:latest
              resources:
                limits:
                  cpu: 2
          nodeSelector:
            zone: kisti

    Worker:
      replicas: 2
      restartPolicy: Never
      template:
        spec:
          containers:
            - name: tensorflow
              image: asifraza/tf-dist-mnist-testing:latest
              resources:
                limits:
                  cpu: 2
                  nvidia.com/gpu: 1
          nodeSelector:
            zone: kisti
  
```

One Parameter Server (PS)

Two Worker Nodes

Experimenting with 1xGPU per Worker

```

kisti@asia-k8smaster/home/kisti/asif/tensorflow-kubeflow-examples
1660976095.789878: Worker 0: training step 16079 done (global step: 19986)
1660976095.792231: Worker 0: training step 16080 done (global step: 19987)
1660976095.794775: Worker 0: training step 16081 done (global step: 19989)
1660976095.796886: Worker 0: training step 16082 done (global step: 19990)
1660976095.799147: Worker 0: training step 16083 done (global step: 19991)
1660976095.801587: Worker 0: training step 16084 done (global step: 19992)
1660976095.803909: Worker 0: training step 16085 done (global step: 19993)
1660976095.806296: Worker 0: training step 16086 done (global step: 19995)
1660976095.808914: Worker 0: training step 16087 done (global step: 19996)
1660976095.811960: Worker 0: training step 16088 done (global step: 19997)
1660976095.814662: Worker 0: training step 16089 done (global step: 19998)
1660976095.817131: Worker 0: training step 16090 done (global step: 20000)
Training ends @ 1660976095.817149
Training elapsed time: 42.016328 s
After 20000 training step(s), validation cross entropy = 1829.45
[root@asia-k8smaster tensorflow-kubeflow-examples]#
  
```

Total Execution time by using GPUs

Results of total elapsed time of distributed training which is 42.01 seconds. We can see that it clearly reduced the total training time because here we are using 1xGPU on per worker.

Same configuration of YAML file that we use above, but the worker nodes are configured with the 1xGPU to train the model in distributed way.

- Multiple sequence alignment (MSA) of large number of sequences on HPC: a case study of SARS-CoV-2 Envelope Protein

Accession	Organism Name	Submitters	Release Date	Isolate	Species	Length	Protein	Geo Location	USA	Host
YP_009724332	Severe acute respiratory s...	Wu, F., et al.	2020-01-13	Wuhan-Hu-1	Severe acute respiratory s...	75	envelope protein	China		Hon
WINE01279	Severe acute respiratory s...	Reeb, V., et al.	2023-09-20	2334752	Severe acute respiratory s...	75	envelope protein	USA:IA	IA	Hon
WINE01280	Severe acute respiratory s...	Reeb, V., et al.	2023-09-20	2334753	Severe acute respiratory s...	75	envelope protein	USA:IA	IA	Hon
WINE01281	Severe acute respiratory s...	Reeb, V., et al.	2023-09-20	2334754	Severe acute respiratory s...	75	envelope protein	USA:IA	IA	Hon
WINE01282	Severe acute respiratory s...	Reeb, V., et al.	2023-09-20	2334755	Severe acute respiratory s...	75	envelope protein	USA:IA	IA	Hon
WINE01283	Severe acute respiratory s...	Reeb, V., et al.	2023-09-20	2334756	Severe acute respiratory s...	75	envelope protein	USA:IA	IA	Hon
WINE01284	Severe acute respiratory s...	Reeb, V., et al.	2023-09-20	2334757	Severe acute respiratory s...	75	envelope protein	USA:IA	IA	Hon
WINE01285	Severe acute respiratory s...	Reeb, V., et al.	2023-09-20	2334758	Severe acute respiratory s...	75	envelope protein	USA:IA	IA	Hon
WINE01286	Severe acute respiratory s...	Reeb, V., et al.	2023-09-20	2335460	Severe acute respiratory s...	75	envelope protein	USA:IA	IA	Hon
WINE01287	Severe acute respiratory s...	Reeb, V., et al.	2023-09-20	2335461	Severe acute respiratory s...	75	envelope protein	USA:IA	IA	Hon
WINE01288	Severe acute respiratory s...	Reeb, V., et al.	2023-09-20	2335462	Severe acute respiratory s...	75	envelope protein	USA:IA	IA	Hon

Filtering options to get SARS-CoV-2 Envelope protein

```
MUSCLE v3.8.1551 by Robert C. Edgar
http://www.drive5.com/muscle
This software is donated to the public domain.
Please cite: Edgar, R.C. Nucleic Acids Res 32(5), 1792-97.

Basic usage
muscle -in <inputfile> -out <outputfile>

Common options (for a complete list please see the User Guide):
-in <inputfile>      Input file in FASTA format (default stdin)
-out <outputfile>   Output alignment in FASTA format (default stdout)
-dlags              Find diagonals (faster for similar sequences)
-maxiters <n>       Maximum number of iterations (integer, default 16)
-maxhours <ch>     Maximum time to iterate in hours (default no limit)
-html              Write output in HTML format (default FASTA)
-msf               Write output in GCG MSF format (default FASTA)
-clw               Write output in CLUSTALW format (default FASTA)
-clwstrict         As -clw, with 'CLUSTAL W (1.81)' header
-log[a] <logfile>  Log to file (append if -loga, overwrite if -log)
-quiet             Do not write progress messages to stderr
-version           Display version information and exit

Without refinement (very fast, avg accuracy similar to T-Coffee): -maxiters 2
Fastest possible (amino acids): -maxiters 1 -dlags -sv -distance1 kbit20_3
Fastest possible (nucleotides): -maxiters 1 -dlags
```

```
MUSCLE v3.8.1551 by Robert C. Edgar
http://www.drive5.com/muscle
This software is donated to the public domain.
Please cite: Edgar, R.C. Nucleic Acids Res 32(5), 1792-97.

*** WARNING *** Assuming Amino (see -seqtype option), invalid letters found: J
sequences 3626233 seqs, lengths min 1, max 76, avg 74
```

Basic usage of MUSCLE algorithm for MSA.



Visualisation of "sars_envelope" file using Aliview

- Applying distributed High-Performance Computing in Drug Discovery and Molecular Simulation

RCSB PDB Protein Data Bank. 298,957 Structures from the PDB, 1,068,577 Computed Structure Models (CSM).

Structure Summary: 7LO8
NorA in complex with Fab36
PDB DOI: <https://doi.org/10.2210/pdb/7LO8/pdb> EM Map EMI

Classification: TRANSPORT PROTEIN
Organism(s): Staphylococcus aureus, Homo sapiens
Expression System: Escherichia coli BL21(DE3), Escherichia coli
Membrane Protein: Yes

Deposited: 2021-02-09 Released: 2022-04-20
Deposition Author(s): Brawley, D.N., Sauer, D.B., Song, J.M., et al.
Funding Organization(s): National Institutes of Health/National Science Foundation (NSF, United States), American Cancer Society, National Institutes of Health/National Institute of General Health Sciences, National Institutes of Health/National Institute of Neurological Disorders and Stroke

Experimental Data Snapshot
Method: ELECTRON MICROSCOPY
Resolution: 3.16 Å
Aggregation State: PARTICLE

3D structure of NorA efflux pump stored in PDB

ZINC database search results for 'bacteria-fda_ligand_001'. The interface shows a grid of ligand structures with their names and IDs, such as ZINC11, ZINC649, ZINC303, ZINC884, ZINC20, ZINC1099, ZINC1590, ZINC1719, ZINC2101, ZINC2119, ZINC2216, ZINC3383, ZINC3742, ZINC4009, ZINC4268, ZINC4301, ZINC4739, ZINC5276, ZINC6894, ZINC6729, ZINC7486, ZINC7571, ZINC7637, ZINC8107, ZINC10465, ZINC10466, ZINC10493, ZINC18154, ZINC20220, ZINC20229, ZINC21786, ZINC23458, ZINC23461, ZINC24946, ZINC26467, ZINC26597, ZINC26599, ZINC28531, ZINC30158, ZINC30344, ZINC30668, ZINC30669.

Selecting organisms from the ZINC database and downloading data

```
Autodock Vina v1.2.3
=====
If you used Autodock Vina in your work, please cite:
=====
J. Eberhardt, D. Santos-Martins, A. F. T. Lillack, and S. Forli
Autodock Vina 1.2.0: New Docking Methods, Expanded Force
Fields, and Python Bindings, J. Chem. Inf. Model. (2021)
DOI: 10.1021/acs.jcim.1c00933
=====
O. Trott, A. J. Olson,
Autodock Vina: Improving the speed and accuracy of docking
with a new scoring function, efficient optimization and
multithreading, J. Comp. Chem. (2010)
DOI: 10.1002/jcc.21334
=====
Please see https://github.com/ccb-rcptps/Autodock-Vina for
more information.
=====
Input:
--receptor arg          rigid part of the receptor (PDBQT)
--flex arg             flexible side chains, if any (PDBQT)
--ligand arg           ligand (PDBQT)
--batch arg            batch ligand (PDBQT)
--scoring arg (vina)  scoring function (add, vina or vinaudo)

Search space (required):
--maps arg             affinity maps for the autodock4.2 (add) or vina
                        scoring function
--center_x arg        X coordinate of the center (Angstrom)
--center_y arg        Y coordinate of the center (Angstrom)
--center_z arg        Z coordinate of the center (Angstrom)
--size_x arg          size in the X dimension (Angstrom)
--size_y arg          size in the Y dimension (Angstrom)
--size_z arg          size in the Z dimension (Angstrom)
--autobox             set maps dimensions based on input ligand(s) (for
                        --score_only and --local_only)

Output (optional):
--out arg             output models (PDBQT), the default is chosen based
                        on the ligand file name
--dir arg            output directory for batch mode
--write_maps arg     output filename (directory + prefix name) for
                        maps, option --force_even_voxels may be needed to
                        comply with --map format

Misc (optional):
--cpu arg (=0)       the number of CPUs to use (the default is to try
                        to detect the number of CPUs or, failing that, use
                        1)
--seed arg (=0)      explicit random seed
--exhaustiveness arg (=8) exhaustiveness of the global search (roughly
                        proportional to time); 1=
                        exhaustive, the number of mc steps is
                        based on heuristics)
--max_evals arg (=0) number of evaluations in each mc run (if zero,
                        which is the default, the number of mc steps is
                        based on heuristics)
--num_modes arg (=9) maximum number of binding modes to generate
--min_rmsd arg (=1)  minimum RMSD between output poses
--energy_range arg (=3) maximum energy difference between the best binding
                        mode and the worst one displayed (kcal/mol)
--spacing arg (=0.375) grid spacing (Angstrom)
--verbosity arg (=1) verbosity (0=no output, 1=normal, 2=verbose)

Configuration file (optional):
--config arg         the above options can be put here

Information (optional):
--help              display usage summary
--help_advanced    display usage summary with advanced options
--version           display program version
ERROR: the receptor or affinity maps must be specified.
```

Autodock Vina arguments

mode	affinity (kcal/mol)	dist from best mode rmsd l.b. rmsd u.b.
1	-5.681	0 0
2	-5.193	5.576 7.674
3	-3.666	3.605 5.768
4	-2.868	6.445 9.138
5	-2.53	4.926 6.404
6	-1.968	3.832 6.278
7	-1.697	5.952 7.778
8	-0.9508	4.796 6.919
9	1.36	3.348 5.095

Vina results for Nora and bacteria-fda_ligand_001

Dogsitescorer interface showing docking results for a protein-ligand complex. The interface includes a 3D visualization of the protein and ligand, and a table of results.

Output of Dogsitescorer by A. Volkamer
 Pocket 1 with 238 binding site atoms written.
 References:
 A. Volkamer et al. Analyzing the topology of active sites: on the prediction of pockets and subpockets, J. Chem. Inf. Model. 2016, 56(11), 2041-02
 A. Volkamer et al. Combining global and local measures for structure-based druggability predictions, J. Chem. Inf. Model. 2012, 52, 360-372
 HEADER: Geometric pocket center at 141.36 133.25 131.43 with max radius 19.97

ATOM	res	chain	resname	x	y	z	occupancy	b-factor	atomtype
187	C	ILE	Z 23	141.721	130.186	141.502	0.00	0.00	C
188	O	ILE	Z 23	141.447	130.615	140.373	0.00	0.00	O
189	CB	ILE	Z 23	141.417	131.647	143.565	0.00	0.00	C
191	CG2	ILE	Z 23	141.583	132.919	142.788	0.00	0.00	C
193	N	PRO	Z 24	142.839	129.439	141.851	0.00	0.00	N
194	CA	PRO	Z 24	143.664	129.133	149.511	0.00	0.00	C
195	C	PRO	Z 24	143.164	128.031	139.831	0.00	0.00	C
197	CB	PRO	Z 24	145.009	128.718	141.138	0.00	0.00	C
210	CA	PRO	Z 27	139.187	129.783	135.276	0.00	0.00	C
217	C	PRO	Z 27	139.686	128.686	134.350	0.00	0.00	C
218	O	PRO	Z 27	139.937	128.956	133.172	0.00	0.00	O
219	CB	PRO	Z 27	149.112	131.692	135.217	0.00	0.00	C
220	C	PRO	Z 27	141.052	130.785	136.386	0.00	0.00	C
221	CD	PRO	Z 27	140.202	130.157	137.410	0.00	0.00	C

Dogsitescorer results of the P1

- **Conference, Workshop & Publications**
 - Workshop: 1st Dec 2022@UPM
 - Conference: 2nd June 2023@UPM)
 - 2 Publications in Interanational Journals
- **User Tutorial**
 - 2nd June 2023@UPM, Target for several application areas
- **Manager Tutorial**
 - 3rd June 2023@UPM, Construction & Operation of Platform, Management issues
- **Training Network & System engineering**
 - 2 for MY, 2 for PK
- **Education Video Clips**
 - How to use platform for application
- **Deploy for University Class**
 - Sukkur-IBA university
- **2 APAN APRP WG Session**
 - APAN 54th & 55th
- **Technical report**
 - High bandwidth distributed Platform technical document
- **R&E Collaborations**
 - 1 MoU with UPM at MY
 - New partners for 3rd party research areas
 - AI – Ghulam Mujtaba
 - Molecular simulation – Tejo Bimo
 - Avian Infection diseases – Mat Isa Nurfiza



APAN & APRP WG

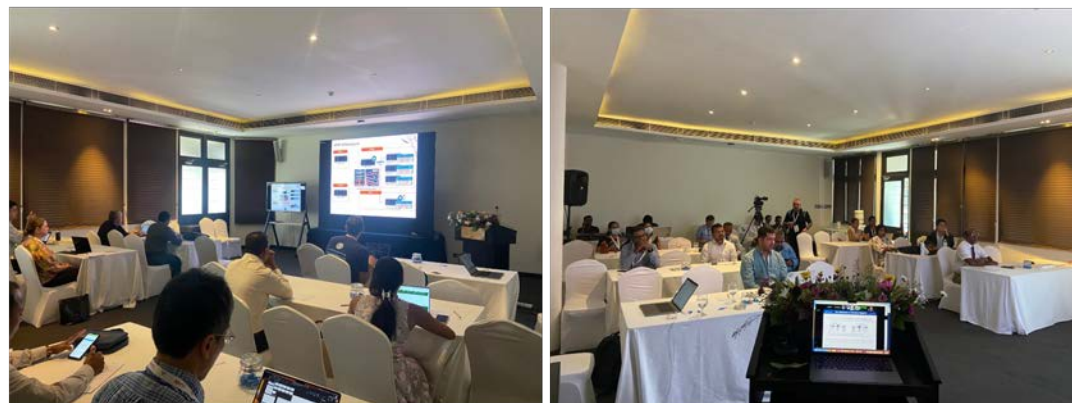
- [1] APAN 56th Meeting
- [2] APRP WG



- **Since 2018 APRP WG initiated at APAN 45th 2018 in Singapore**
APRP – Asia Pacific Research Platform Working Group
APAN meeting held 2 times in a year
- **Objectives**
The goal is to share 'xRP' experience with the members and to propose the establishment of an APRP which will be part of the GRP (Global Research Platform).
 - Promote **HPC ecosystem in the Asia-Pacific.**
 - Engage **APAN members and ASEAN countries**
 - Towards the setting up an **Asia Pacific Research Platform (APRP)** and **become a part of a Global Research Platform**
- **Target**
Academia and Industry
- **Executive member**
Chair : Jeonghoon Moon, KISTI, Korea
Co-Chair : Andrew Howard, NCI, Australia
Secretary : Asif Khan, Perdana Univ. Malaysia
- **Asi@Connect 5th Call project by TEIN*CC**
Title : A High bandwidth distributed HPC (1st April 2022 – 31st July 2023)



- **APAN 56th APRP Meeting**
 - Held at Colombo in Sri Lanka(21st to 25th 2023)
 - Title is **"AI and HPC: Better Together"**
- **APRP WG: 2 Sessions (23rd Aug, 2023)**
 - Infrastructure/Technology & Applications
 - 2 sessions and 14 Presentations from 8 countries
 - Around 40 attendees (on/off-line)



Session 1

Session1. Infra/Technical part (Chair, Jeonghoon Moon)		
1	0900-0905	Jeonghoon Moon – Introduction APRP WG in APAN (KISTI/Korea)
2	0905-0915	Jeonghoon Moon - Building a high bandwidth distributed HPC (KISTI/Korea)
3	0915-0930 Remote	Kihyeon Kim - Asia-Pacific Research Platform update (KISTI/Korea)
4	0930-0945 Remote	Asif Raza - AI Science user cases and achievements using distributed HPC platform (Sukur-IBA Univ./Pakistan & FermiLab in US)
5	0945-1000 Remote	Asif Khan - Update of HPC for Bio-Science: ViVA platform for viral informatics (Perdana Univ./Malaysia)
6	1000-1015	Vincenzo Capone - The LHCONE multidomain service, a global infrastructure to support High Energy Physics (GEANT/EU)
7	1015-1030 Remote	Mat Isa Nurulfiza - Data driven in the fight against avian infectious diseases (UPM/Malaysia)
8	1030-1040	Susumu Date - Update of RED ONION using DTN solution in Osaka Univ. (Osaka Univ./Japan)

Session 2

Session2, Application/Infra part (Chair, Andrew Howard)		
1	1100-1115	Andrew Howard - NCI Research Platform and country update (NCI/Australia)
2	1115-1130 Remote	Kiwook Kim - Update of ScienceloT and Wireless Transmission Scientific data (KISTI/Korea)
3	1130-1145 Remote	Nor Asilah - Distributed HPC at My and country update (UPM/Malaysia)
4	1145-1200	Saranjeet Kaur Bhogal - Navigating the Research Software Engineering Community Landscape in Asia (Research Software Alliance)
5	1200-1215 Remote	Bimo Tejo - Role of Molecular dynamics in drug discovery (UPM/Malaysia)
6	1215-1230	Alex Moura/Jysoo Lee - KAUST & Saudi Arabia update (KAUST/Saudi Arabia)

Two Objectives of Wireless Network

- 1) **60GHz Wireless** : Inter-building Fixed Wireless Backbone
 - Builds wireless links b/w buildings, as a part of the backbone network



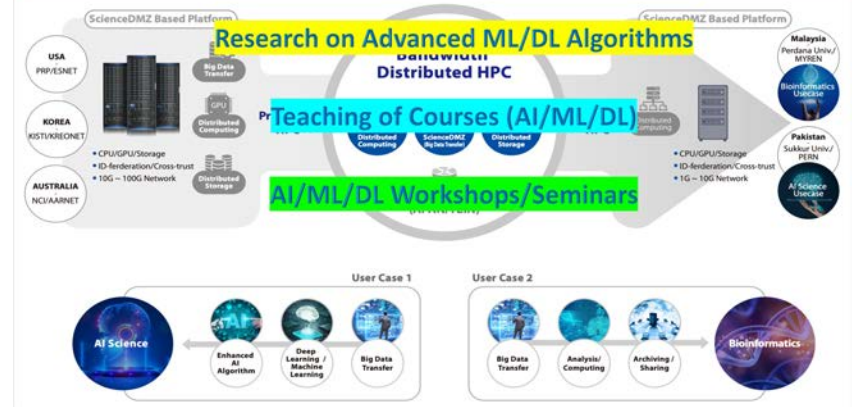
60GHz : high data rate, narrow beam width, low penetrability
 → Suitable for fixed, high-speed wireless backbone in LoS environments

- 2) **Wi-Fi(2.4GHz / 5GHz) Wireless** : In/Outdoor Wireless Access
 - Connects user equipments to network for test and demonstration

- 60GHz Wireless Communication based Inter-building Backbone Network(Kiwook Kim)

- AI Science Use Cases Using Distributed HPC Platform(Asif Raza)

Utilization of Platform for AI Research & Education



Asia Pacific

• CAE-1

- A 10-year collaboration starting in 2019 to connect Europe and Asia-Pacific at 100 Gbps between London and Singapore



• Asia-Pacific Europe Ring (AER)

- A collaboration of ten R&E networks and institutions to provide mutual back-up between Europe and Asia-Pacific across diverse links.



- The LHCONE Multidomain Service (Enzo Capone)



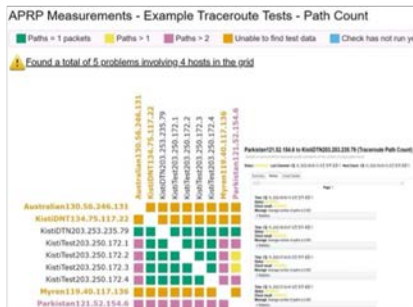
Assoc. Prof. Dr. Nor Asilah Wati Abdul Hamid,
Chairperson, International Conference for High Bandwidth Distributed HPC
Acting Director, Institute for Mathematical Research (INSPEM), UPM.

- Distributed High Bandwidth HPC at Malaysia (Nor Asilah)

- Construction of Asi@connect Project Asia Pacific Research Platform (Kihyeon Kim)



- Measuring the ScienceDMZ using a monitoring system
 - PerfSONAR - Measurement of Network performance and status
 - Docker-based perfSONAR install and adding a MaDDash

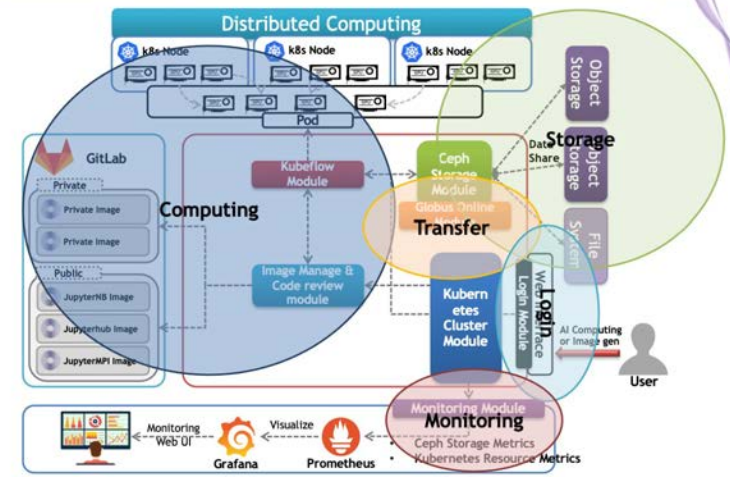


MaDDash Dashboard: Example of Traceroute Tests among DTN servers



Docker-based PerfSONAR test-points running at participating institutions

AI based Research Platform Architecture



- Asi@Connect project: Building a high bandwidth distributed HPC (Jeonghoon Moon)

Future plan & Conclusion



- **Benefits to TEIN and its Community**
 - Networking/computing infrastructure over 4 NRENs based on TEIN
 - Promoting Asian research/education by utilizing TEIN-based distributed HPC resources
 - Several outcomes will be derived
- **Visibility**
 - Promote technical and academic achievement
 - Communicate innovation and practicality achievement
- **Sustainability of the Activity/Program**
 - Using APAN APRP WG for managing and operating
 - Expanding the use cases in Asia & Additional collaboration with Asian countries
 - Motivating further collaborations via TEIN
- **Extension for 3rd party research areas**
 - Smart Agriculture
 - Environmental research on climate change
 - Cloud computing & Wireless communication

