4th GRP Workshop@Limassol/Cyprus

국가와 국민을 위한
데이터 생태계 중심기관 KISTI

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

KISTI/KREONET
Jeonghoon Moon
9th Oct 2023

KiSTi 한국과학기술정보연구원
Korea Institute of Science and Technology Information
www.kisti.re.kr

Contents

Overview of the project

[1] Overview & Background

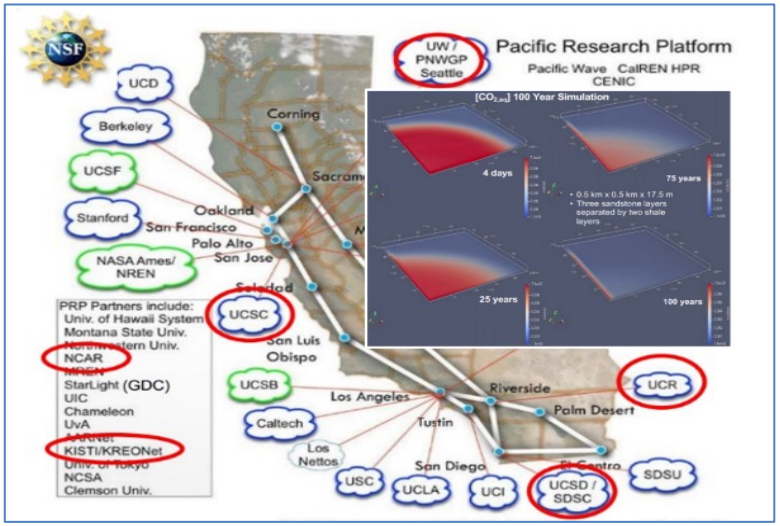
Overview of APRP Project

- **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

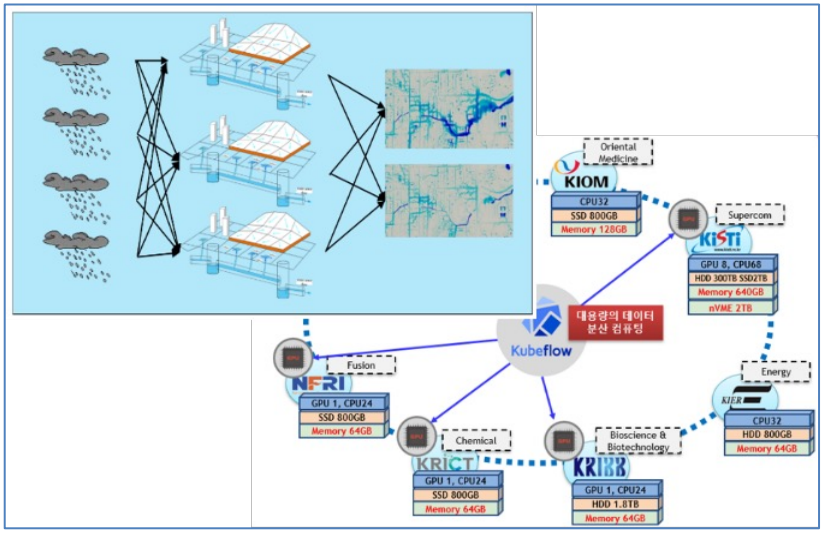
Asi@Connect
TEIN CC COOPERATION CENTER
European Union

Project Background

- **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

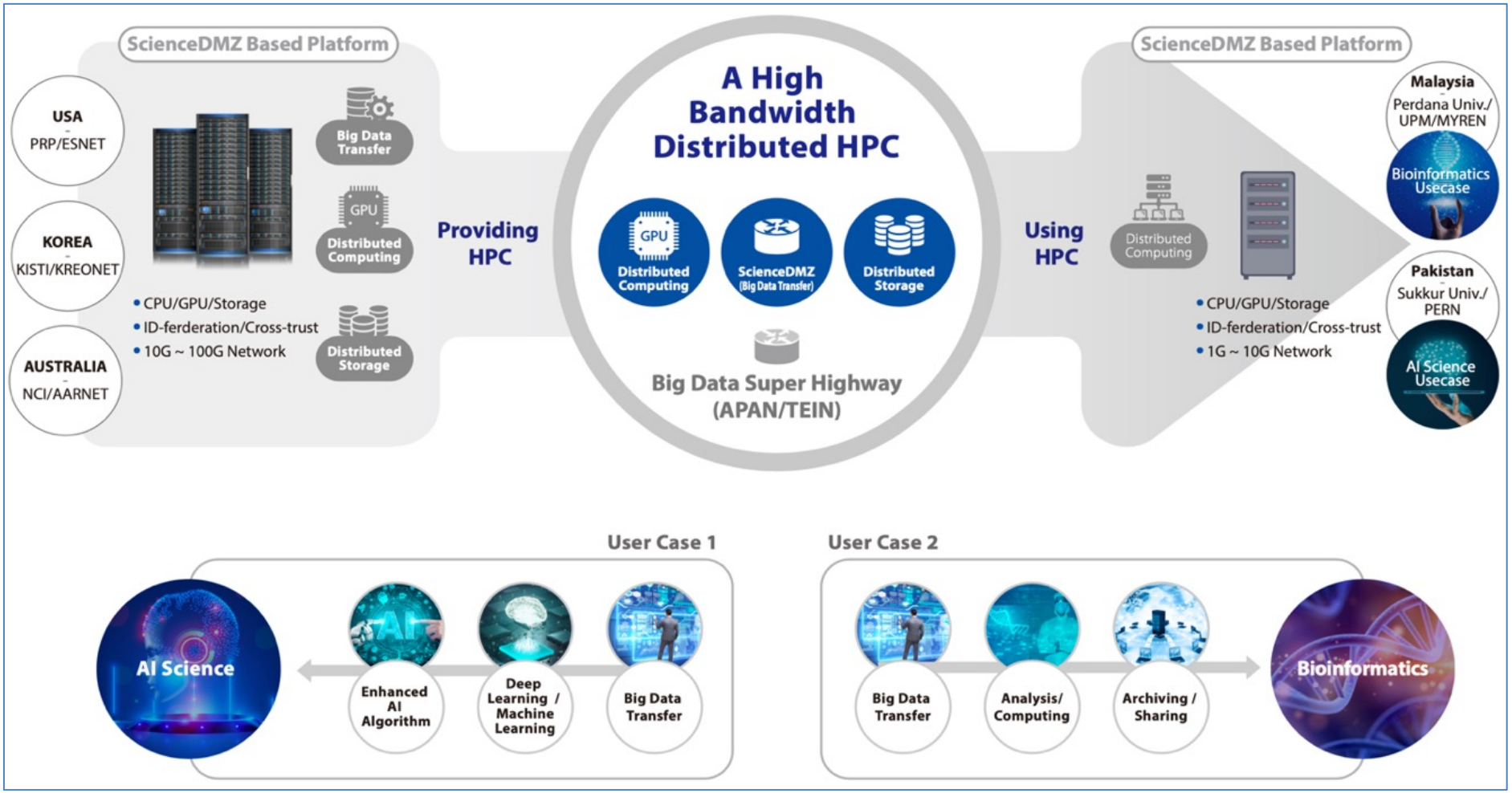
PRP→NRP→GRP→APRP→Asi@Connect project

Activities of the Project

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

Architecture of a High Bandwidth Distributed HPC

- 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

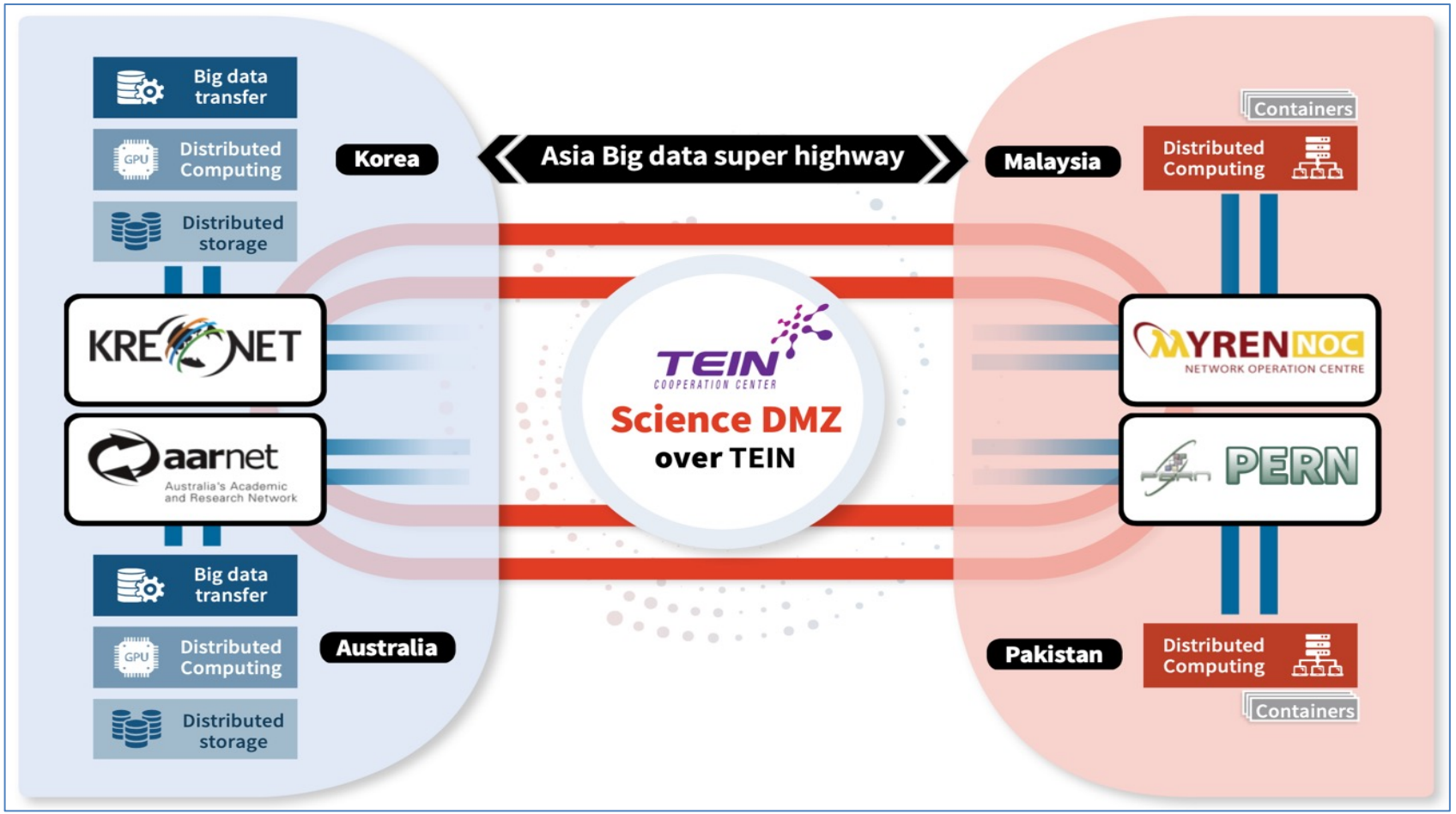


Building Strategy

- **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)

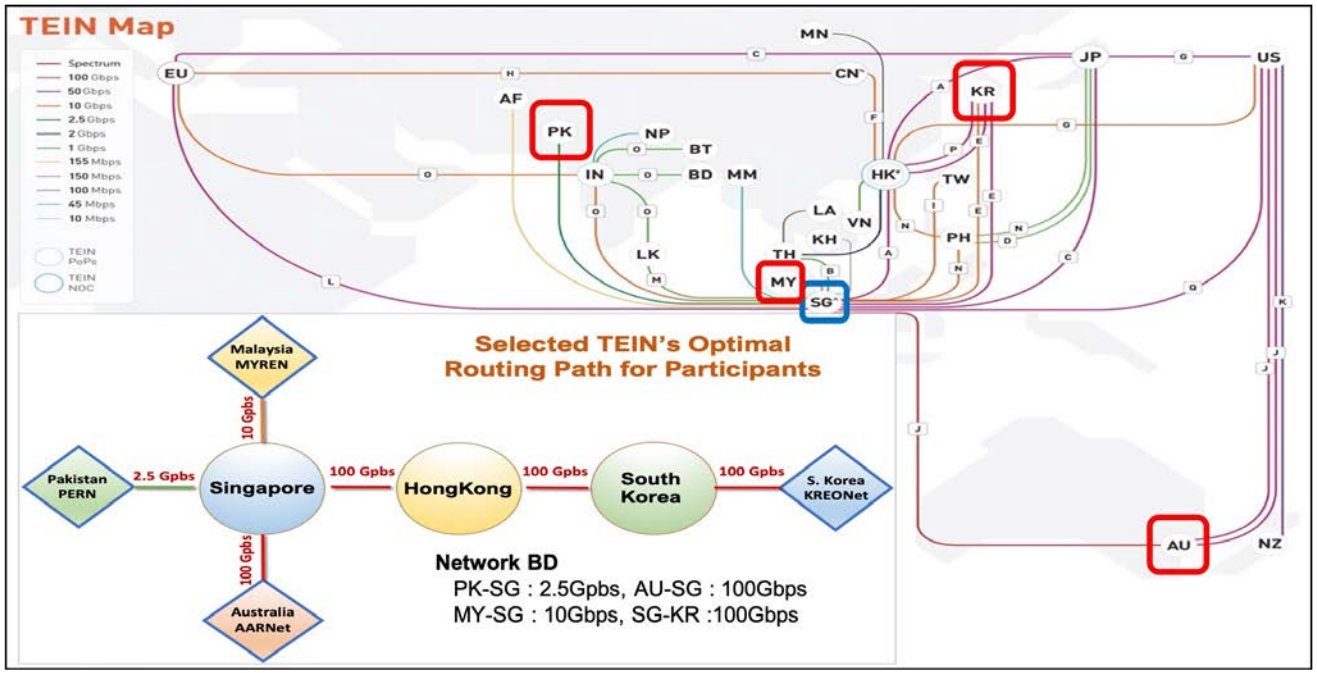
Activity 1

- **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)

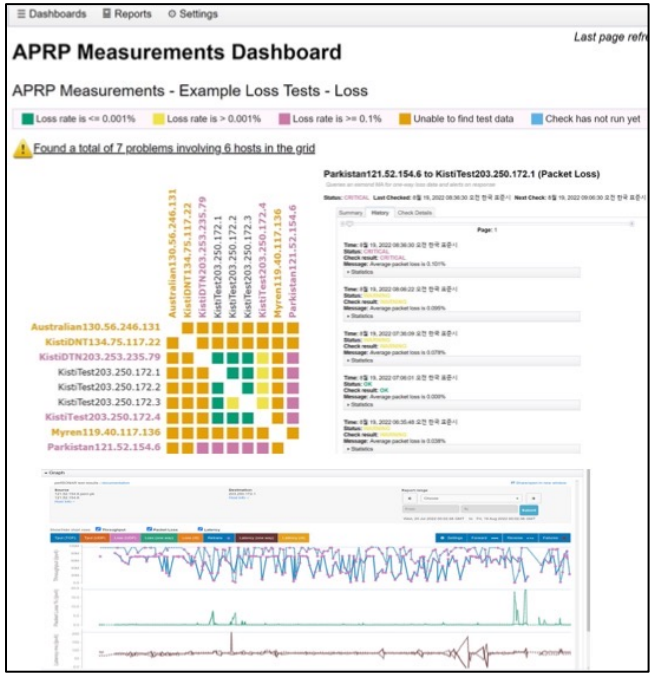


Detail of Activity 1 – 1 Interconnecting participant NRENs via TEIN

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



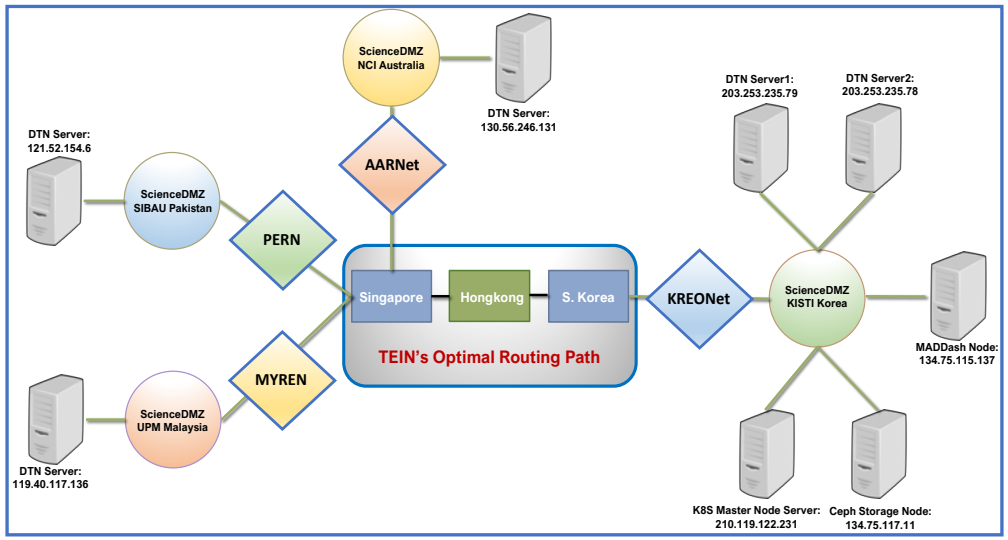
NREN topology & Bandwidth



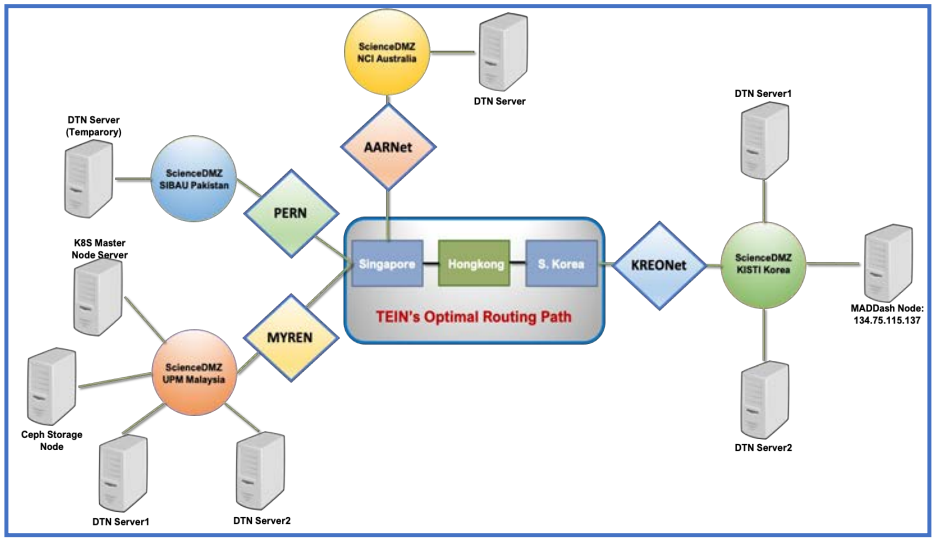
PerfSONAR

Detail of Activity 1 – 2 Installing and adding local DTN

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



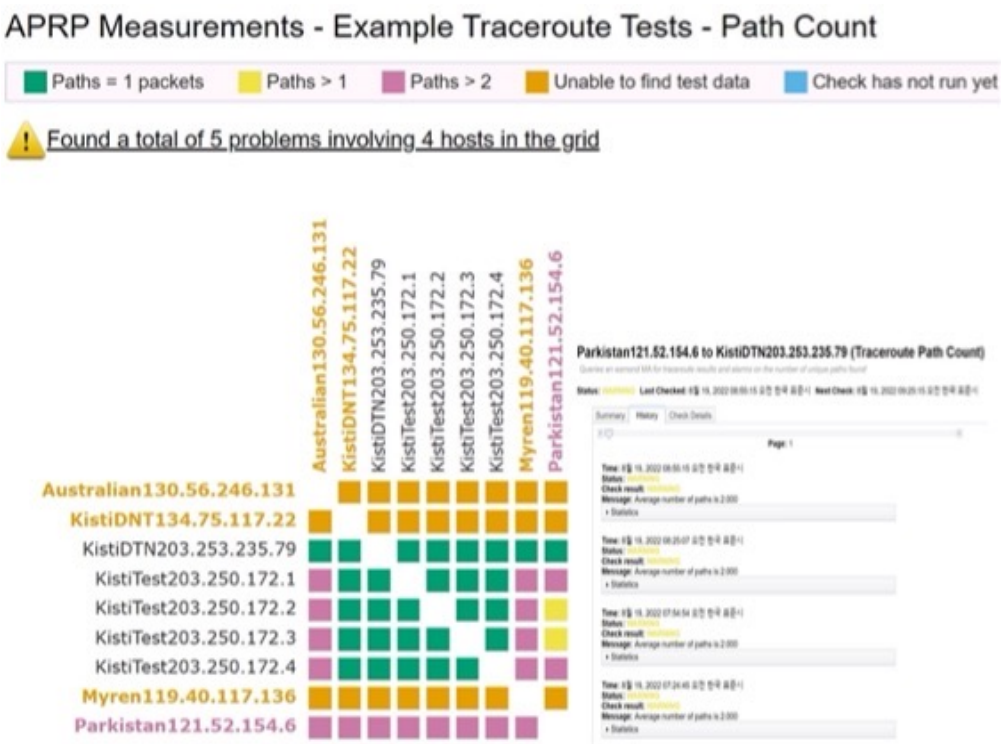
The previous end-to-end network topology of temporary DTN servers



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

Detail of Activity 1 – 3 Measuring the ScienceDMZ

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

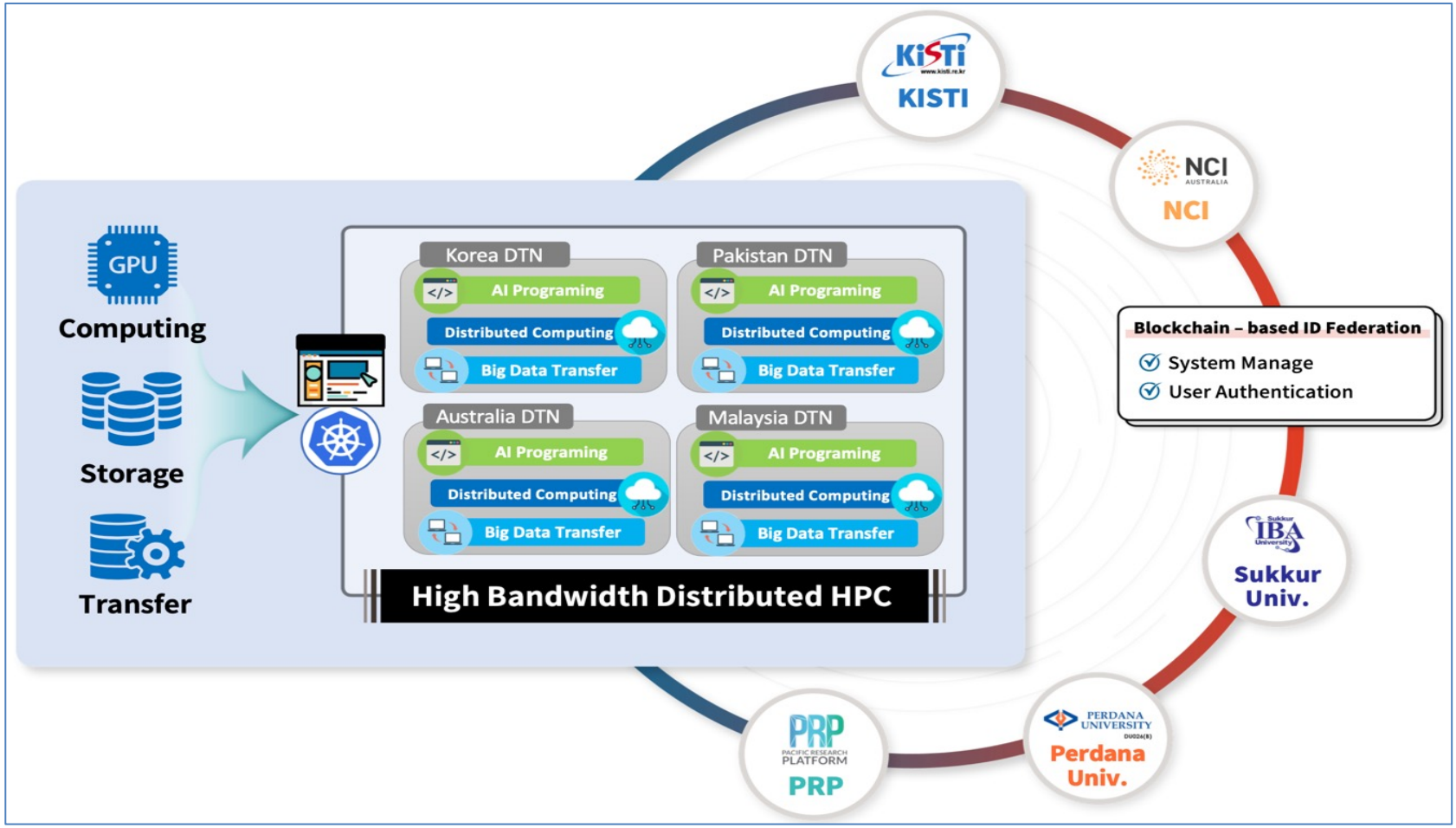


MaDDash Dashboard: Example of Traceroute Tests among DTN servers

Docker-based PerfSONAR test-points running at participating institutions

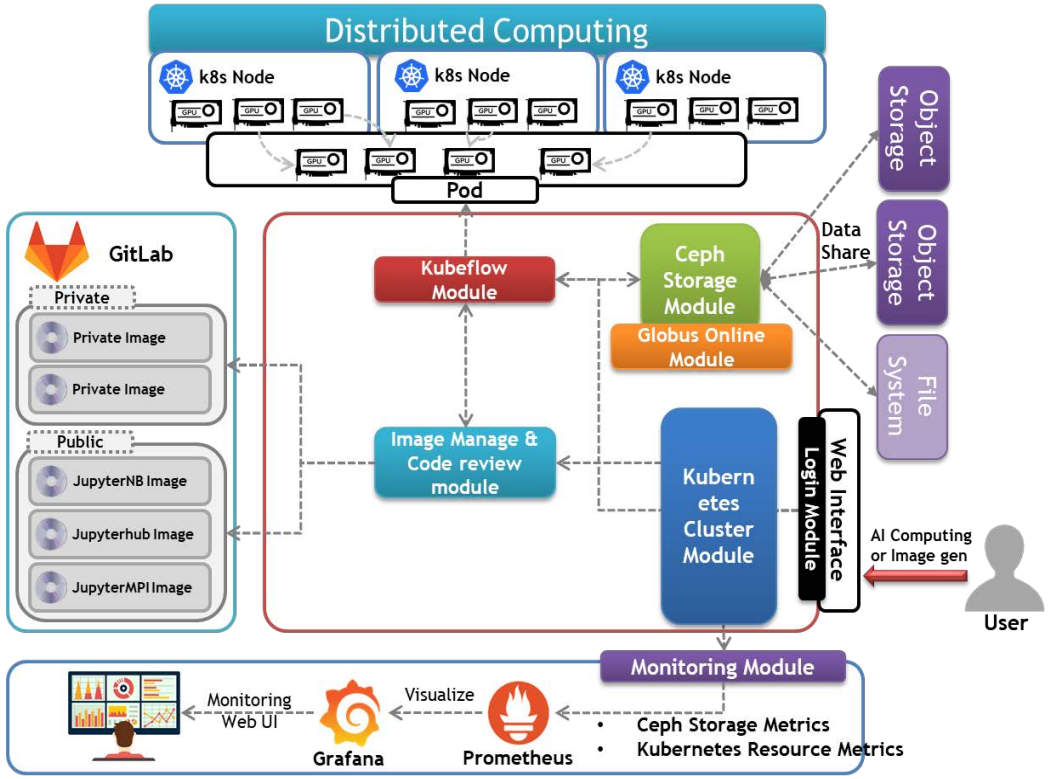
Activity 2

- **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

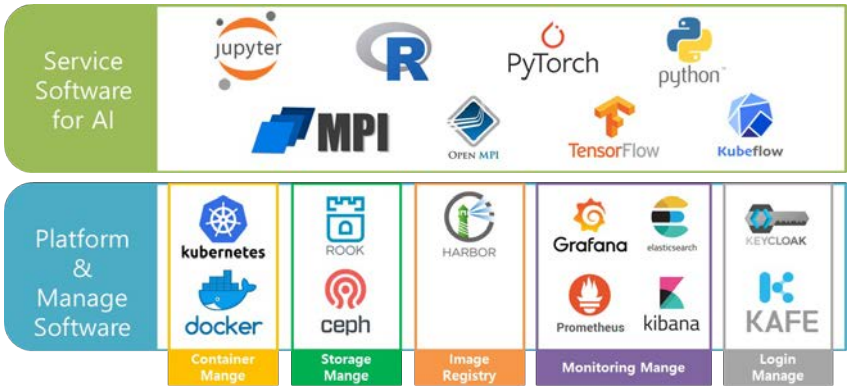


Detail of Activity 2-1 Building a platform to manage CPU/GPU

- **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	Memory	Storage	Network
Master Node	No GPU	Intel(R) Xeon(R) Gold 6226R CPU @ 2.90GHz 32-Core Processor	DDR4 251GB	HDD 480GB	1GB
Pakistan Node	NVIDIA A40 1GPU	AMD EPYC 7502 32-Core Processor	DDR4 188GB	SSD 960GB NVMe 960GB	1GB
Malaysia Node	NVIDIA A40 1GPU	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

Detail of Activity 2-1 Building a platform to manage CPU/GPU

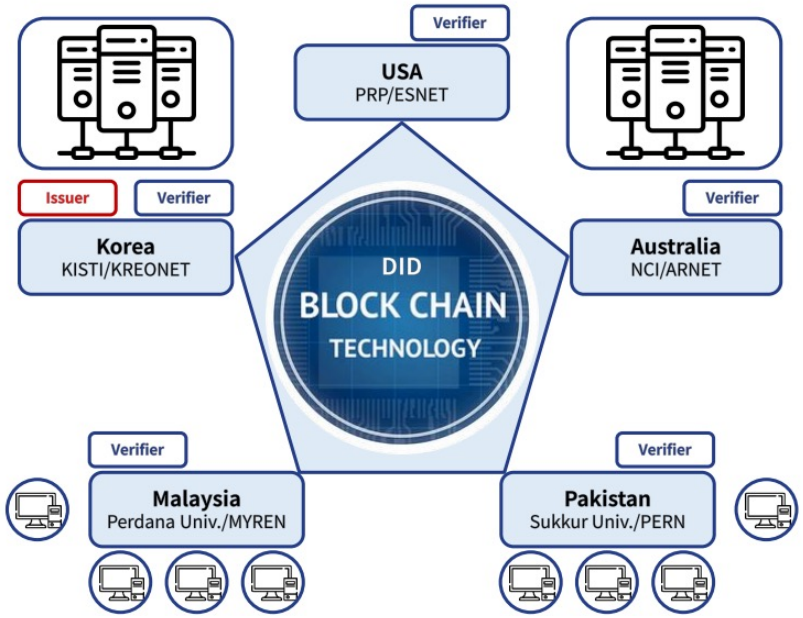
Opening ceremony

MoU

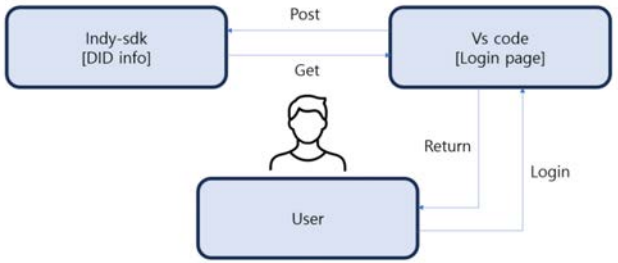
Built Platform at UPM

Opening ceremony
& Workshop

Detail of Activity 2 – 2 User management base on Bock-Chain



Asi@Connect Platform Block chin Diagram



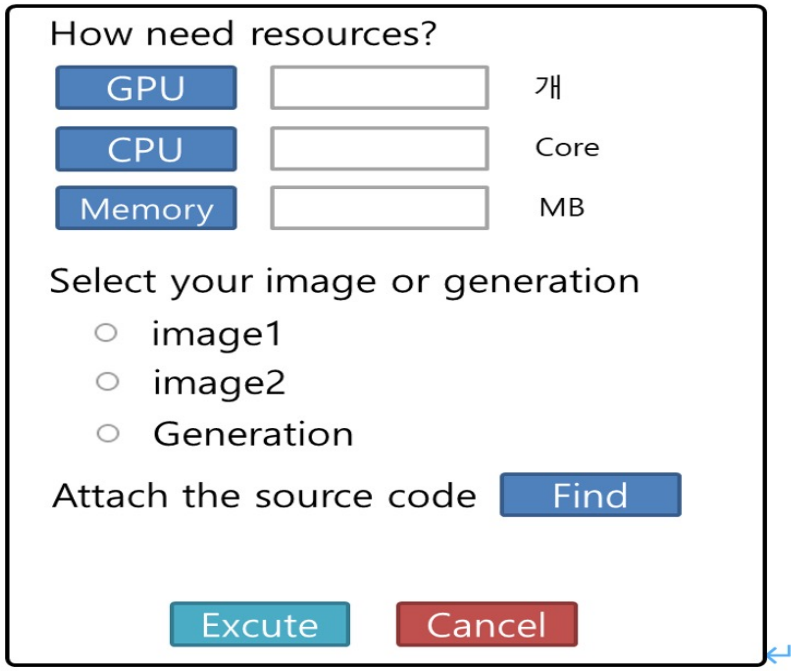
A simple process

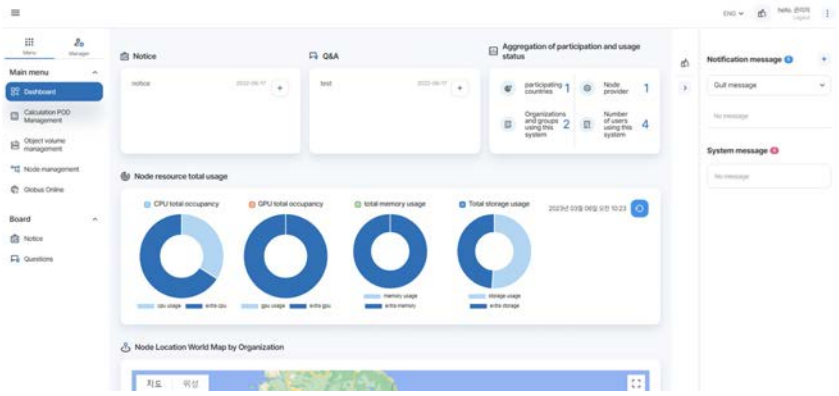
Function	Description
open_wallet	Call the wallet.open_wallet function provided by the Indy SDK to open the wallet. If the wallet doesn't exist, create it and open it.
get_did_info	Call the Indy SDK's did.list_my_dids_with_meta function to retrieve information about the user's Distributed Identifier (DID) 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

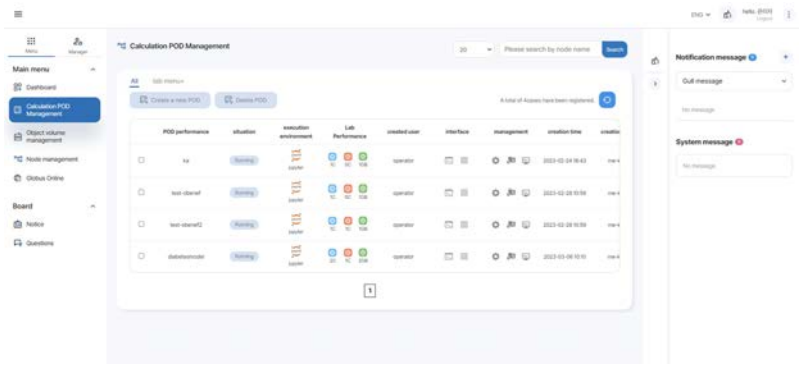
Detail of Activity 2-3 Easy user interface

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**





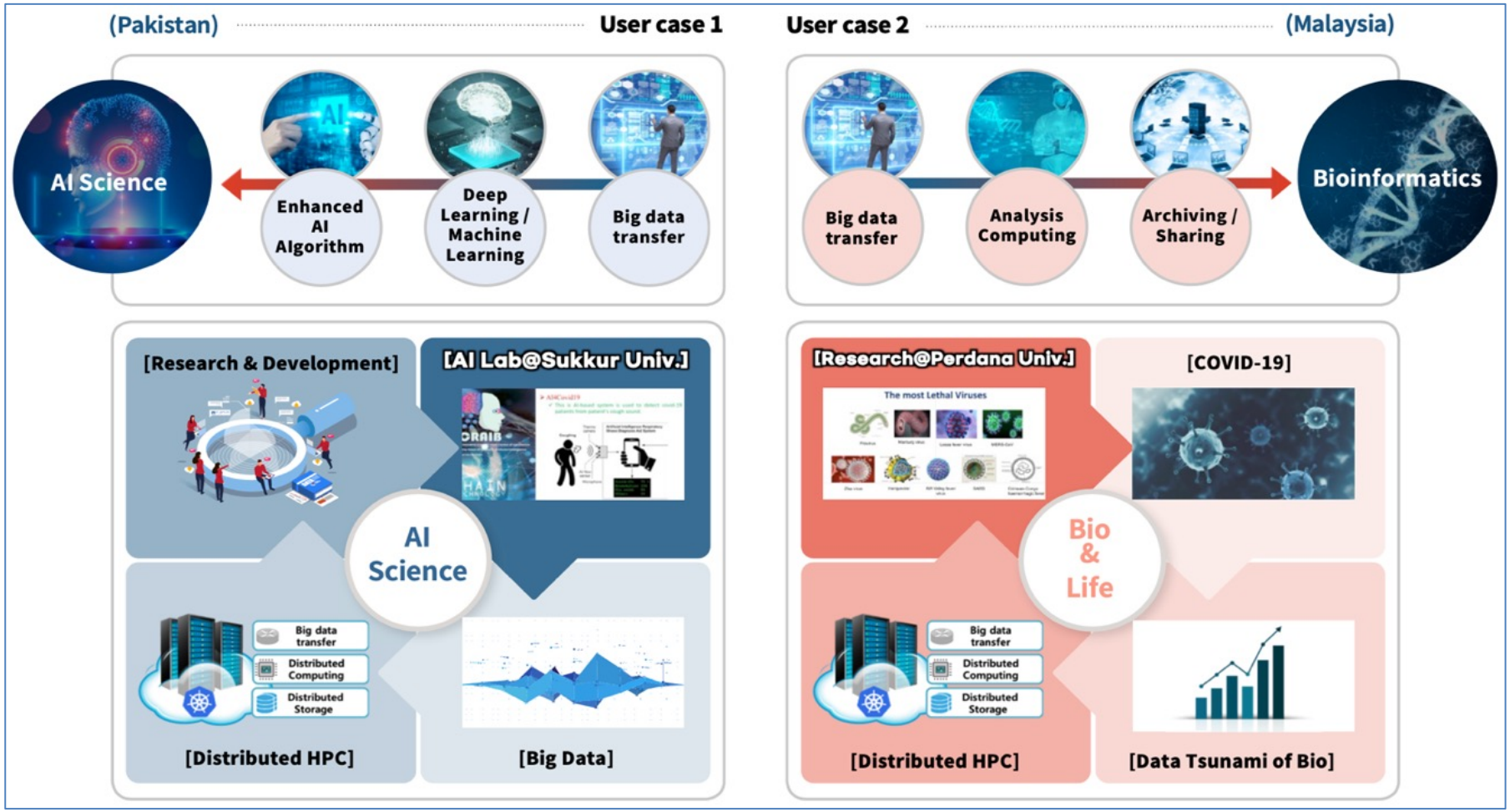
Check the resources



Status of the selected pod

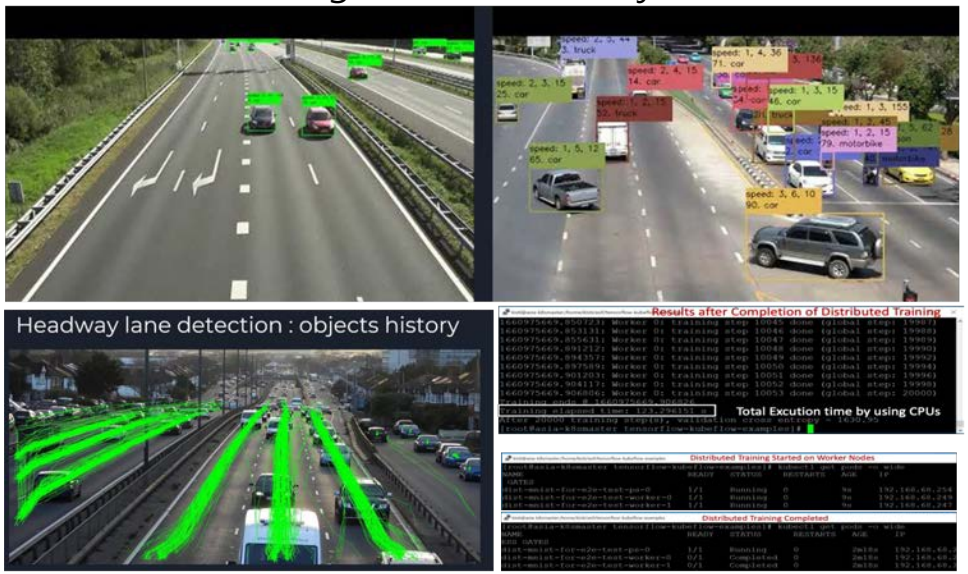
Activity 3

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

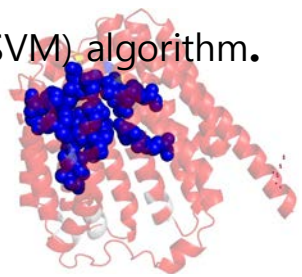


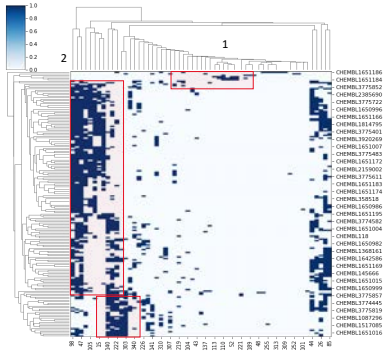
Detail of Activity 3 – Case Studies of AI and Bio Science

- **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 RefAligne

Detail of Activity 3 – Examples of AI-Science

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

```
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
          nodeSelector:
            zone: kisti
```

One Parameter Server (PS)

Two Worker Nodes

Experimenting with 2xCPUs per Worker

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

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
```

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   192.168.68.2
dist-mnist-for-e2e-test-worker-0    0/1     Completed   0          2m18s   192.168.68.2
dist-mnist-for-e2e-test-worker-1    0/1     Completed   0          2m18s   192.168.68.2
```

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

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 Excution 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 2xCPUs on per worker.

Detail of Activity 3 – Examples of AI-Science cont.

```
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:
                  nvidia.com/gpu: 1
          nodeSelector:
            zone: kisti
```

One Parameter Server (PS)

Two Worker Nodes

Experimenting with 1xGPU 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.

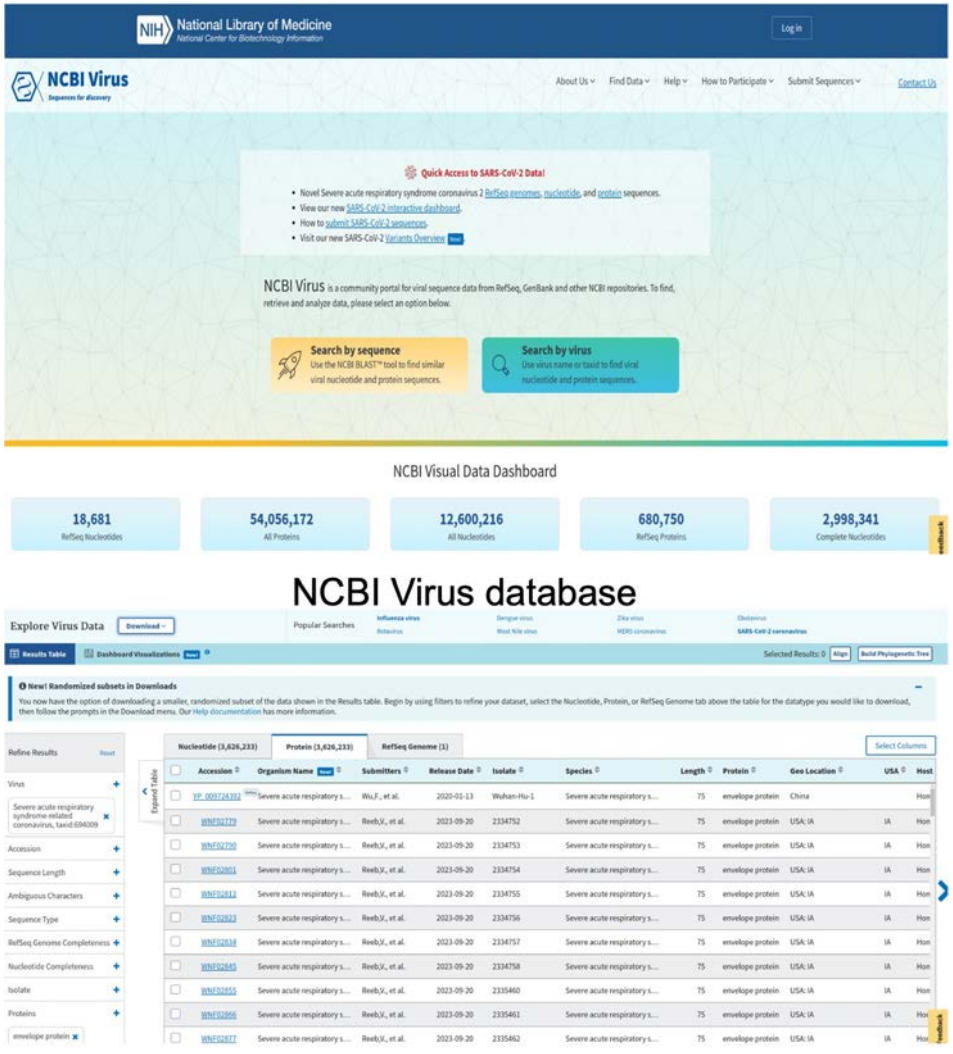
```
kisti@asia-k8smaster:/home/kisti/asif/tensorflow-kubeflow-examples
1660976095.789878: Worker 0: training step 16079 done (global step: 19986)
1660976095.792231: Worker 0: training step 16080 done (global step: 19987)
1660976095.794775: Worker 0: training step 16081 done (global step: 19989)
1660976095.796886: 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 Excution 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.

Detail of Activity 3 – Examples of Bio-Science #1

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

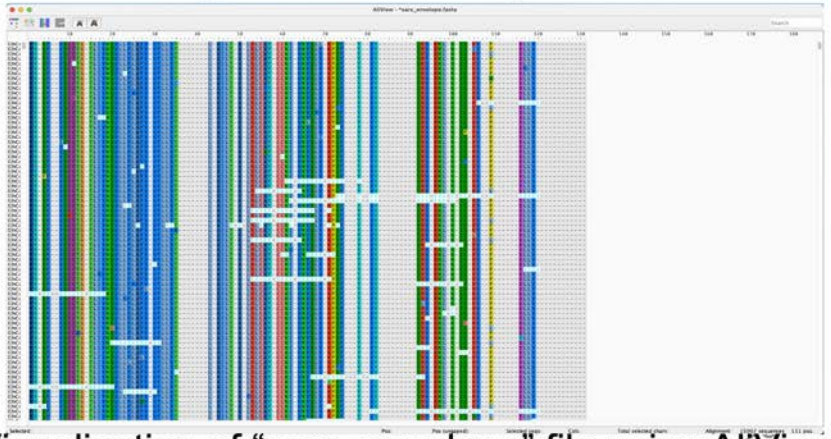


Filtering options to get SARS-CoV-2 Envelope protein

```
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)
    -diags             Find diagonals (faster for similar sequences)
    -maxiters <n>      Maximum number of iterations (integer, default 16)
    -maxhours <h>      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 -diags -sv -distance1 kbit20_3
Fastest possible (nucleotides): -maxiters 1 -diags
```

```
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
```

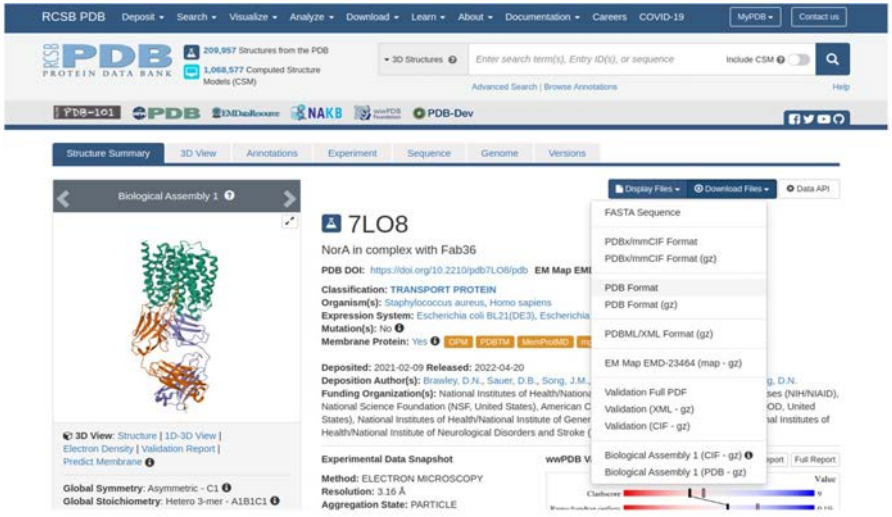
Basic usage of MUSCLE algorithm for MSA.



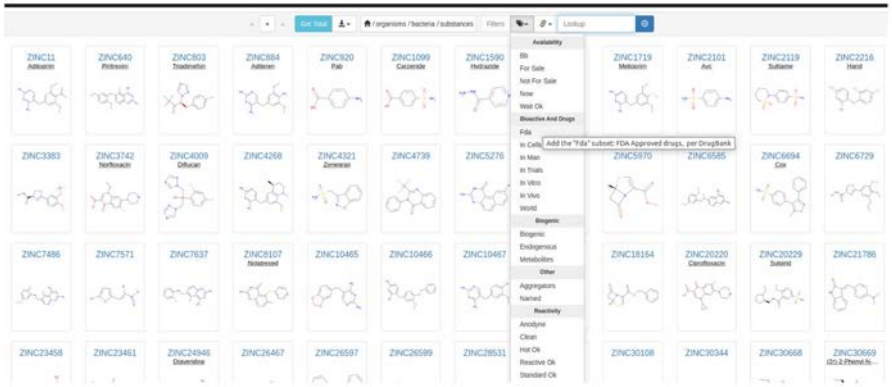
Visualisation of “sars_envelope” file using AliView

Detail of Activity 3 – Examples of Bio-Science #2

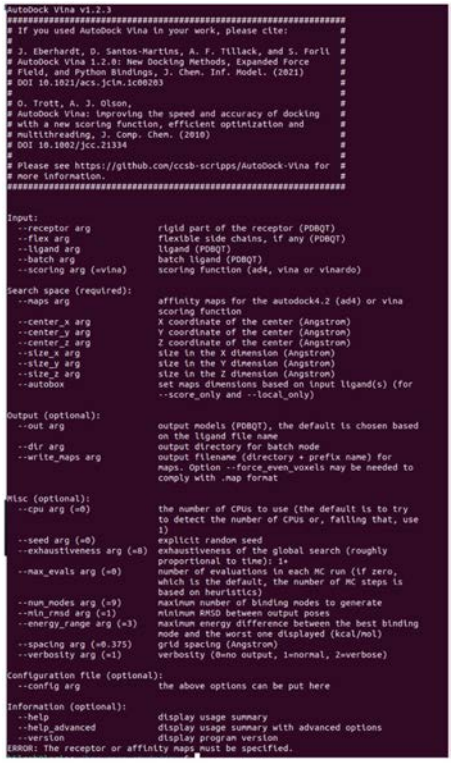
- **Applying distributed High-Performance Computing in Drug Discovery and Molecular Simulation**



3D structure of NorA efflux pump stored in PDB



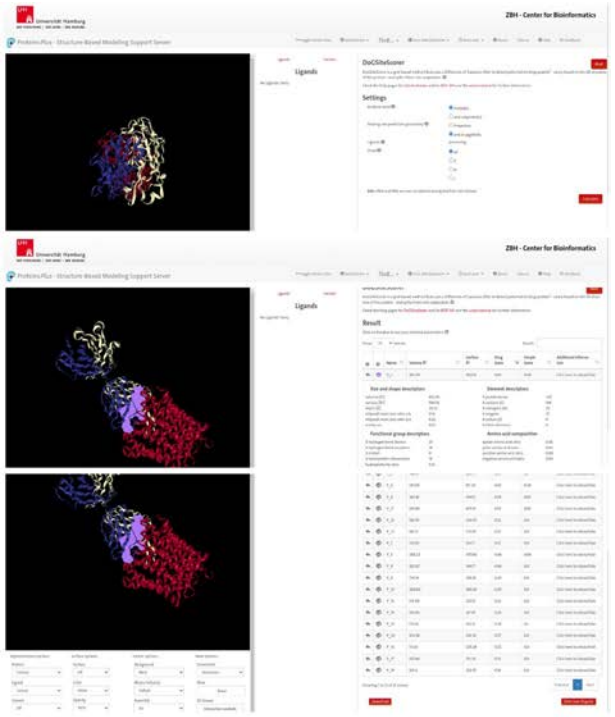
Selecting organisms from the ZINC database and downloading data



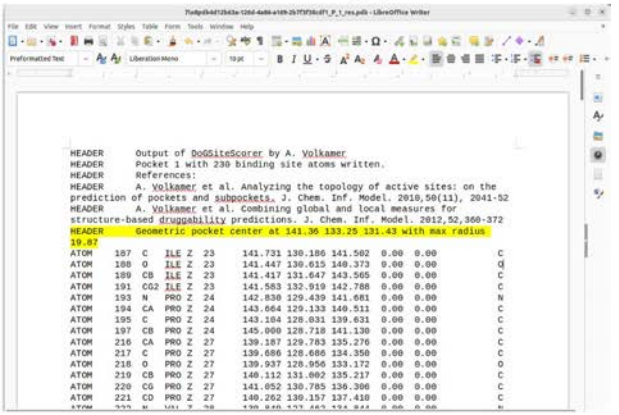
Autodock Vina arguments

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



Usage of Dogsitescorer to find binding pockets



Dogsitescorer results of the P1

Other Achievement

- **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

APAN APRP(Asia-Pacific Research Platform) 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 Meeting & APRP WG session

- **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)

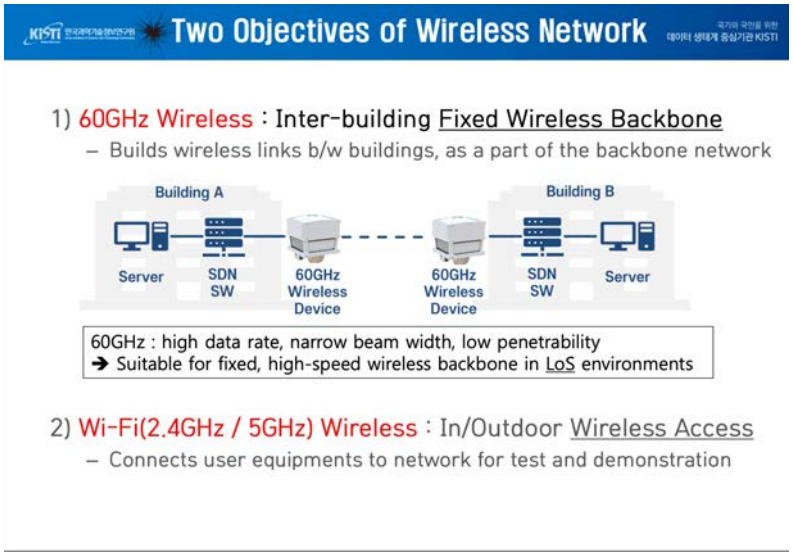
Session 1

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)

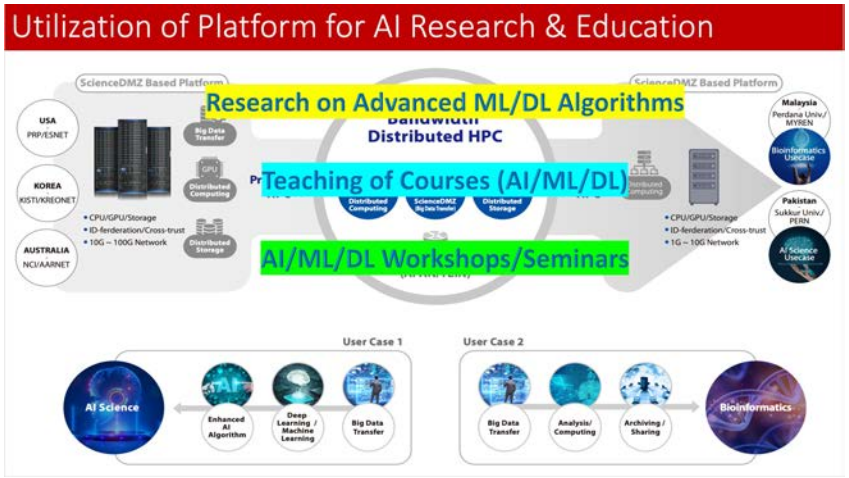
Session 2

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 ScienceIoT 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)

Summary of Sessions



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



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

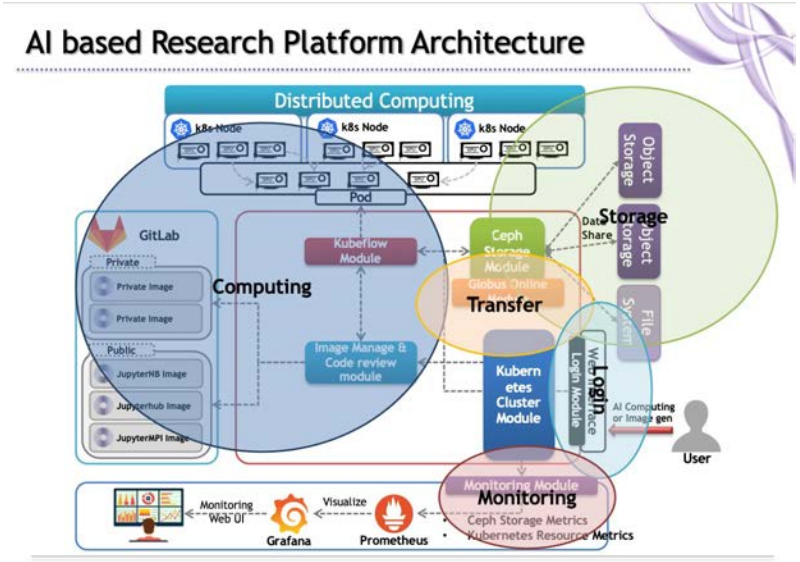
- The LHCONE Multidomain Service (Enzo Capone)

Summary of Sessions cont.

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

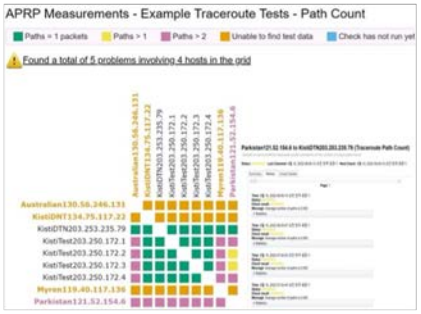
- Distributed High Bandwidth HPC at Malaysia (Nor Asilah)

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



7

- **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

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

Future plan & Conclusion

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

THANK YOU