جامعة الملك عبدالله للعلوم والتقنية
King Abdullah University of Science and Technology

Core Labs and
Research Infrastructure

Shaheen III HPC 101 - Storage

Bilel Hadri, KAUST Supercomputing Lab

October, 28th 2025

Shaheen Storage

"A supercomputer is a device for converting a CPU-bound problem into an I/O bound problem." [Ken Batcher]

- HPC systems consist of three main components:
  - Compute nodes
  - High-speed interconnect
  - I/O infrastructure

- Most optimization work on HPC applications is carried out on
  - Single node performance across many cores
  - Network performance (communication)
  - I/O only when it becomes a real problem (or when the sys-admin contact us…)

Shaheen Storage

- I/O is commonly used by scientific applications to achieve goals like:
  - storing numerical output for later analysis
  - loading initial conditions or datasets for processing
  - checkpointing to files that save the state of an application in case of system failure

- On Supercomputer, you will need parallel file system

Shaheen Storage Why do we need parallel I/O?

- Imagine a 24 hours simulation on 16 cores.
  - 1% of run time is serial I/O.
  - You get the compute part of your code to scale to 6 nodes, ie 1152 cores.
  - Up to 72x speedup in compute: I/O is 39% of run time (22’16” in computation and 14’24’’ in I/O).

- Parallel I/O is needed to
  - Spend more time doing science
  - Not waste resources
  - Prevent affecting other users
  - Easier to use and get the performance without major change in the workflow.

Shaheen III Hardware Specifications: Storage

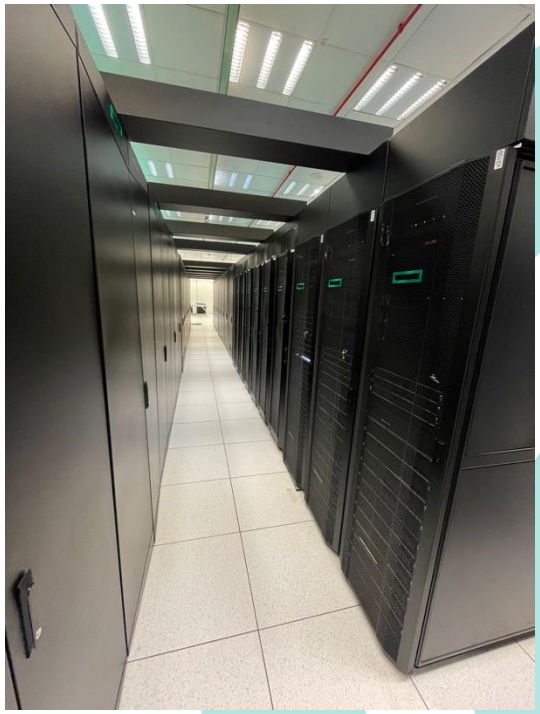
Characteristics Tiered storage	Shaheen III /scratch storage /scratch/username
Total Capacity (usable)	32 PB
Capacity tier (HDD)	25 PB
Capacity tier perf Read/Write	330/260 GB/s
BW tier capacity	6.8PB
BW Perf. tier Read/Write	3750/2500 GB/s
IOPS tier capacity	338 TB
IOPS tier IOPS (Read/Write)	10+M IOPS

Characteristics	Shaheen III /projects /project/kxxxxx
Total Capacity (usable)	57 PB
Capacity perf Read/Write	~300/220 GB/s

Shaheen Storage

- /scratch
  - Read + Write on login and Compute nodes
  - Fast file system
- /project
  - Read only on Compute nodes
  - Read + Write on PPN and DTN
  - Keeping large file for the lifetime of the project
- /home
  - Only available on login nodes
- https://docs.hpc.kaust.edu.sa/policy/shaheen3.html

Tiered Storage /scratch



Shaheen 3 uses tiered storage for Scratch (80TB quota per PI)

- Capacity Tier: default
  - /scratch/username or cd $SCRATCH
  - 10 TB quota per user
- Bandwidth Tier
  - /scratch/username/bandwidth cd $SCRATCH_BW
  - 1 TB quota per user
  - For files which do require high bandwidth, large files…
- IOPs Tier
  - /scratch/username/iops or cd $SCRATCH_IOPS
  - 50 GB quota per user
  - For small files with high IOPs pattern requirements

Shaheen Data Transfer

- Data transfer between tiers: just **copy** the data with the **cp** command.
- Data transfer between scratch <-> project: SLURM job script:

```
user@login4:> cat dtn_copy_job.sh
#!/usr/bin/env bash
#SBATCH -N 1
#SBATCH -n 1
#SBATCH -t 20
#SBATCH -p dtn
#SBATCH --output=SLURM_LOGS/copy_out.txt
#SBATCH --error=SLURM_LOGS/copy_err.txt
#SBATCH --job-name=COPY_project_to_scratch
rsync <options> -v --progress /lustre2/project/k????? /scratch/user/
```

- For outside Shaheen, use scp command

```
scp -r myusername@shaheen.hpc.kaust.edu.sa:/path/directory .
```

- For large files, Globus via dtn6. Contact help@hpc.kaust.edu.sa for any guidance.

Storage Best Practices

- /scratch meant for temporary use during the lifetime of the job.
- /project for persistent storage during the lifetime of the project
- Users must delete their unused data from scratch and move necessary files to project filesystem.
- For overall quota on scratch:

```
lfs quota -uh $USER /scratch
```

- For quota on capacity tier on scratch:

```
lfs quota -uh $USER --pool capacity /scratch
```

- For quota on bandwidth tier on scratch:

```
lfs quota -uh $USER --pool bandwidth /scratch
```

- For quota on IOPS tier on scratch:

```
lfs quota -uh $USER --pool iops /scratch
```

Storage Best Practices 2

- My usage of storage quota: command kuq

kuq

```
---------------------------------------------------------------------------------
Filesystem quota limits for user hadrib
Tier          Filesystem   used  quota  limit  grace  files  quota  limit  grace
---------------------------------------------------------------------------------
capacity       /scratch  119.1G     0k    10T      -  240782      0      0      -
bandwidth      /scratch  924.1G     0k     1T      -  240782      0      0      -
iops           /scratch  60.88M     0k    50G      -  240782      0      0      -
project        /project  7.778T     0k     0k      -  209596      0 1000000      -
---------------------------------------------------------------------------------
```

```
lfs quota -uh $USER /scratch
Disk quotas for usr xxx (uid xxx):
    Filesystem   used  quota  limit  grace  files  quota  limit  grace
     /scratch  1.043T     0k    11T      -  240782      0 1024000      -
```

```
lfs quota -uh $USER /project
Disk quotas for usr hadrib (uid 129285):
    Filesystem   used  quota  limit  grace  files  quota  limit  grace
     /project  7.778T     0k     0k      -  209596      0 1000000      -
```

Storage Best Practices 3

- Quota for /project, kpq project-id . Quota 80 TB per PI.

```
kpq k10005
---------------------------------
PI quota for : Principal Investigator
---------------------------------
Filesystem  used   quota   limit   grace   files   quota   limit
grace
/project     16k      0k     80T       -
       4       0       0       -
```

- Don’t be shy, ask for help. help@hpc.kaust.edu.sa
- Limit the number of files per directory
- Clean up after running a job. (remove temporary, slurm output…)
- Important files shall be copied to your personal workstation.

جامعة الملك عبدالله للعلوم والتقنية
King Abdullah University of Science and Technology
Core Labs and Research Infrastructure

KAUST SUPERCOMPUTING CORE LAB

جامعة الملك عبدالله للعلوم والتقنية
King Abdullah University of Science and Technology

Shaheen III: Best Practices summary

Who can use Shaheen III?

- Every Shaheen user must be an official member of at least one project, and every project must originate from an approved organization.
- PI needs to be a faculty/manager to endorse the project.
- Access to Shaheen is available to a select group of academic and industry partners.
- Following the terms and conditions
- Submitting all necessary documents

Charges of core hours

- Unit of measure is core hours for the CPU nodes
  - Each node has 192 cores.
  - For exclusive usage, full node (ie 192 cores) will be charged

- Check the usage of the project: sb k1xxxx

```
Title: KSL computational scientists
PI : Saber Feki
---------------------------------
Project k01     expiry: 2030-12-31
---------------------------------
Allocations:
2024-02-08                100000000
---------------------------------
Total allocations:        100000000
Core hours used:            1300532
Remaining balance:         98699468
---------------------------------
```

Queue

- FIFO policy with backfilling as long as
  - core hours available
  - Maximum of number of jobs/nodes not reached

- 5 QoS
  - workq (entire system up to 4608 nodes)
    - Limit per user 2048 nodes max, 500 job running, 1000 max job queued
  - shared (16 nodes in total)
  - debug (4 nodes in total)
    - 1 node per user
  - 72hours: up 128 nodes partition shared by all users
    - 16 nodes maximum per user, 3 job running, maximum 32 job queued
  - ppn nodes: post processing nodes: 15 nodes

Storage quotas for each tiers,

- /scratch
  - Total aggregated PI 80 TB default
  - Per user 1M files, 10 TB in /scratch/username and 1 TB in bandwidth and 50 GB in IOPS
- /project
  - Total aggregated PI 80 TB default
  - Per user 1M files
- Check:
  - Check the quota with kpq, kuq on project and overall
  - `lfs quota -uh $USER /scratch`
  - `Disk quotas for user username (uid 123456):`
  - `Filesystem used quota limit grace files quota limit grace`
  - `/scratch 402.7G 0k 11T - 123339 0 1024000 -`

Applications available

- Over 500 version of libraries, tool and applications available
- Module avail ; module avail –S nameofsoftware

```
------------------ /opt/cray/pe/perftools/23.09.0/modulefiles ------------------
perftools
perftools-lite
perftools-lite-events
perftools-lite-gpu
perftools-lite-hbm
perftools-lite-loops
perftools-preload

------------------ /opt/cray/pe/modulefiles ------------------
PrgEnv-aocc/8.4.0(default)
PrgEnv-aocc/8.5.0
PrgEnv-cray/8.4.0(default)
PrgEnv-cray/8.5.0
PrgEnv-gnu/8.4.0(default)
PrgEnv-gnu/8.5.0
PrgEnv-intel/8.4.0(default)
PrgEnv-intel/8.5.0
PrgEnv-nvhpc/8.4.0(default)
PrgEnv-nvhpc/8.5.0
PrgEnv-nvidia/8.4.0(default)
PrgEnv-nvidia/8.5.0
atp/3.15.0
atp/3.15.1(default)
atp/3.15.2
cce/16.0.0
cce/16.0.1(default)
cce/17.0.0
cpe/23.05
cpe/23.09(default)
cpe/23.12
cpe-cuda/23.05
cpe-cuda/23.09(default)
cpe-cuda/23.12
cray-R/4.2.1.2(default)
cray-R/4.3.1
cray-ccdb/5.0.0
cray-ccdb/5.0.1(default)
cray-ccdb/5.0.2
cray-cti/2.18.0
cray-cti/2.18.1(default)
cray-cti/2.18.2
cray-dsmml/0.2.2(default)
cray-dyninst/12.2.0
cray-dyninst/12.3.0(default)
cray-dyninst/12.3.1
cray-fftw/3.3.10.4
cray-fftw/3.3.10.5(default)
cray-fftw/3.3.10.6
cray-hdf5/1.12.2.3
cray-hdf5/1.12.2.7(default)
cray-hdf5/1.12.2.9
cray-hdf5-parallel/1.12.2.3
cray-hdf5-parallel/1.12.2.7(default)
cray-hdf5-parallel/1.12.2.9
cray-libpals/1.2.12(default)
cray-libsci/23.05.1.4
cray-libsci/23.09.1.1(default)
cray-libsci/23.12.5
cray-libsci_acc/23.12.0(default)
cray-mpich/8.1.26
cray-mpich/8.1.27(default)
cray-mpich/8.1.28
cray-mpich-abi/8.1.26
cray-mpich-abi/8.1.27(default)
cray-mpich-abi/8.1.28
cray-mpich-abi-pre-intel-5.0/8.1.26
cray-mpich-abi-pre-intel-5.0/8.1.27(default)
cray-mpich-abi-pre-intel-5.0/8.1.28
cray-mpich-ucx/8.1.26
cray-mpich-ucx/8.1.27(default)
cray-mpich-ucx/8.1.28
cray-mpixlate/1.0.1.11
cray-mpixlate/1.0.2(default)
cray-mpixlate/1.0.3
cray-mrnet/5.1.0
cray-mrnet/5.1.1(default)
cray-mrnet/5.1.2
cray-netcdf/4.9.0.3
cray-netcdf/4.9.0.7(default)
cray-netcdf/4.9.0.9
cray-netcdf-hdf5parallel/4.9.0.3
cray-netcdf-hdf5parallel/4.9.0.7(default)
cray-netcdf-hdf5parallel/4.9.0.9
cray-openshmemx/11.6.0
cray-openshmemx/11.6.1(default)
cray-openshmemx/11.7.0
cray-pals/1.2.12(default)
cray-parallel-netcdf/1.12.3.3
cray-parallel-netcdf/1.12.3.7(default)
cray-parallel-netcdf/1.12.3.9
cray-pmi/6.1.11
cray-pmi/6.1.12(default)
cray-pmi/6.1.13
cray-python/3.10.10(default)
cray-python/3.11.5
cray-stat/4.12.0
cray-stat/4.12.1(default)
cray-stat/4.12.2
cray-ucx/1.14.0(default)
cray-ucx/2.7.0-1
cray-ucx/default(default)
craype/2.7.21
craype/2.7.23
craype/2.7.30(default)
craype-dl-plugin-ftr/22.06.1.2(default)
craype-dl-plugin-py3/21.02.1.3
craype-dl-plugin-py3/21.04.1
craype-dl-plugin-py3/22.06.1.2
craype-dl-plugin-py3/22.08.1
craype-dl-plugin-py3/22.09.1
craype-dl-plugin-py3/22.12.1(default)
craypkg-gen/1.3.28
craypkg-gen/1.3.30(default)
craypkg-gen/1.3.31
gcc/10.3.0
gcc/11.2.0
gcc/12.2.0(default)
gcc-native/12.3(default)
gdb4hpc/4.15.0
gdb4hpc/4.15.1(default)
gdb4hpc/4.16.0
iobuf/2.0.10(default)
papi/7.0.0.2
papi/7.0.1.1(default)
papi/7.0.1.2
perftools-base/23.05.0
perftools-base/23.09.0(default)
perftools-base/23.12.0
sanitizers4hpc/1.1.0
sanitizers4hpc/1.1.1(default)
sanitizers4hpc/1.1.2
valgrind4hpc/2.13.0
valgrind4hpc/2.13.1(default)
valgrind4hpc/2.13.2

------------------ /opt/cray/modulefiles ------------------
chapel/1.30.0
cray-lustre-client-ofed/2.15.0.7_rc2_cray_25_ga33b7d9-2.5_5.24__ga33b7d9745.shasta(default)
cudatoolkit/23.3_11.0
cudatoolkit/23.3_11.8
cudatoolkit/23.3_12.0(default)
libfabric/1.15.2.0(default)
xpmem/2.6.2-2.5_2.27__gd067c3f.shasta(default)

------------------ /opt/modulefiles ------------------
amduprof/4.2.850
aocc/4.2.0(default)
intel/19.0.5.281
intel/2023.1.0(default)
intel-classic/2023.1.0(default)
intel-oneapi/2023.1.0(default)

------------------ /opt/cray/pe/craype-targets/default/modulefiles ------------------
craype-accel-amd-gfx90a
craype-accel-amd-gfx908
craype-accel-amd-gfx940
craype-accel-host
craype-accel-nvidia70
craype-accel-nvidia80
craype-arm-grace
craype-hugepages1G
craype-hugepages2G
craype-hugepages2M
craype-hugepages4M
craype-hugepages8M
craype-hugepages16M
craype-hugepages32M
craype-hugepages64M
craype-hugepages128M
craype-hugepages256M
craype-hugepages512M
craype-network-none
craype-network-ofi
craype-network-ucx
craype-x86-genoa
craype-x86-milan
craype-x86-milan-x
craype-x86-rome
craype-x86-spr
craype-x86-spr-hbm
craype-x86-trento

------------------ /sw/ex109genoa/modulefiles ------------------
abinit/9.6.2
abinit/9.10.3
adf/2019.301
airss/0.9.4
alamode/1.3.0
alamode/1.4.2
almabte/1.3.2
amber/14
amber/14_mpi
amber/14_mpi_plumed
amber/18
amber/18_mpi
amber/23_mpi
ams/2023.103
amset/0.4.18
amset/0.4.18_parallel
amset/0.4.20
ansys/v22R1-fluids
ansys/v23R1-fluids
ansys/v23R1-structures
ansys/v23R2-electronics
ansys/v23R2-fluids
aocl/4.2.0(default)
arm-forge/23.1.2
ase/3.19.0
ase/3.22.1
atk/2019.03sp1
atompaw/4.2.0.3
autodockvina/1.2.3
bader/1.04
bader/1.05
bands4vasp/0.4
bazel/6.1.0
bazel/6.5.0(default)
bazel/7.1.0
berkeleygw/2.1
berkeleygw/3.1.0
bio/blast+/2.15.0
bio/bwa/0.7.17
bio/edta/2.2.0
bio/gatk/4.1.6
bio/java/8u401
bio/java/21.0.2
bio/plink/1.9b
bio/plink/2.0a
bio/samtools/1.8
bio/samtools/1.18
bio/tabix/0.2.6
blitz/1.0.2
boltztrap/1.2.5
boltztrap2/24.1.1
boost/1.85
castep/21.11
chemshell/21.0.2
chemshell/23.0.1
chimera/1.16
cif2cell/1.2.10
cmake/3.28.3
codee/2024.1.4
columbus/7.2
columbus/7.2.2
comsol/6.2
comsol/6.2_tmp
converge/3.0.13
converge/3.0.13_udf
converge/3.0.25
converge/3.0.25_udf
converge/3.0.26
converge/3.0.26_udf
converge/3.0.27
converge/3.0.27_udf
converge/3.1.4
converge/3.1.4_udf
converge/3.1.5
converge/3.1.5_udf
converge/3.1.6
converge/3.1.6_udf
converge/3.1.7
converge/3.1.7_udf
converge/3.1.8
converge/3.1.8_udf
converge/3.1.11
converge/3.1.11_udf
converge/3.2.1
converge/3.2.1_udf
converge/3.2.4
converge/4.0.0
converge/4.0.0_udf
converge/4.0.2
cp2k/2023.2
critic2/1.1dev
critic2/1.1stable
crystal14/1.0.3
cuby4/4
deepspeed/0.14.0
dftbplus/21.2
dftd4/2.5.0
dlpoly/4.09
dlpoly/4.09_plumed
dlpoly/5.1.0
dlpoly/5.1.0_plumed
dssp/2.3.0
eddp/0.2
edmftf/Apr2021
edmftf/Jan2019
egsnrc/2020
egsnrc/2023
eigen/3.3.7
elk/6.3.2
elk/9.2.12
elpa/2023.05.001
espresso/6.4.1
espresso/6.4.1_environ
espresso/6.8
espresso/6.8_elpa
espresso/6.8_libxc
espresso/7.2
espresso/7.2_elpa
espresso/7.2_environ
espresso/7.2_libxc
espresso/7.3
excimontec/1.0.0
exciting/neon21
exciting/nitrogen14
fermisurfer/2.2.1
fermisurfer/2.4.0
ffmpeg/6.1.1(default)
fhiaims/210716_3
fhiaims/221103
fhiaims/231212_1
flex/2.6.4
fourphonon/1.1
fourphonon/20211001
gamess/30Sept2022R2
gamess/30Sept2023R2
gaussian09/d.01
gaussian16/c.02
go/1.21.3
gollum2/2.0
gpaw/22.1.0
gpaw/24.1.0
grace/5.1.25
gromacs/2023.1
gsl/2.6-cce16.0.1
gsl/2.6-gcc12.2.0
gulp/6.0
gulp/6.2
horovod/0.28.1-tf212-torch1131
horovod/0.28.1-torch221(default)
ifermi/0.3.3
java/19.0.1
jdftx/1.7.0
jmol/14.31.44
koopmans/1.0.1
ktf/0.1.5
ktf/0.7
kwant/1.4.3
lammps/6Aug2023
lev00/4.01
libxc/4.2.3
libxc/4.3.4
libxc/5.1.7
libxc/5.2.3
libxc/6.2.2
likwid/5.3.0
lobster/5.0.0
materstudio/2023
milo/1.0.3
mkl/2024.0.0
mohid/19.10
mohid/23.10
molden/6.9
molden/7.2.1
moleculargsm/20240115
molgw/3.2
molpro/2012.1p16
moltemplate/2.20.20
mopac/22.1.0
motif/2.3.8
mpifileutils/0.11.1
mrcc/2020-02-22_mpi
mrcc/2020-02-22_omp
mrcc/2023-08-28_mpi
mrcc/2023-08-28_omp
multiwfn/3.6
multiwfn/3.8dev
music/4.0
mysw/0.0.1
mytools/1.01
namd/2.14_gcc
namd/2.14_smp_gcc
nbo6/6.0_g09
nbo6/6.0_g16
nwchem/6.8.1
nwchem/7.2.2
octopus/11.3
octopus/13.0
oommf/2.0alpha3
oommf/2.1alpha0
openbabel/3.1.1
openfoam/2212
openmolcas/23.10
openmx/3.9.9
orca/5.0.4
orca/6.0.0
ovito/2.9.0
ovito/3.10.1
p4vasp/0.3.30
pacchem/20200322
packmol/18.169
packmol/20.14.3
paraview/5.11.2
paraview/5.12.0-egl
paraview/5.12.0-mesa(default)
paraview/5.12.1
periodic_nbo/20191008
perturbo/2.2.0
phono3py/1.18.2
phonopy/2.21.0
polyrate/17-C_rp
polyrate/17-C_vrc
psi4/1.8.0
psi4/1.9
py4vasp/0.7.4
pyprocar/6.1.7
pyrx/0.9.9
python/3.10.13(default)
pytorch/1.13.1(default)
pytorch/2.2.1
qchem/5.4_mpi
qchem/5.4_smp
qchem/6.1
qchem/6.2
quantumkite/1.0
quantumkite/1.1
qvasp/2.23
raspa2/2.0.3
seissol/1.1.4_gnu
shengbte/1.5.0
siesta/4.1.5
siesta/psml204
simmate/0.13.2
simmate/0.16.1
simmate/0.17.0
singularity/4.0.1
smina/20220122
sod/0.47
sod/0.52
spack/0.20.0
sumo/2.3.8
sumo/2.3.8_saber
tbmodels/1.4.3
tdep/20231024
tecplot/2023r2
tensorflow/2.12(default)
thirdorder/1.1.1
timemory/3.3(default)
totalview/2023.3.10
towhee/8.2.3
turbomole/7.1
turbomole/7.1_mpi
turbomole/7.1_smp
uspex/10.5
vampire/6.0
vasp/5.4.4
vasp/5.4.4.pl2
vasp/5.4.4.pl2_dftd4
vasp/5.4.4.pl2_libbeef
vasp/5.4.4.pl2_nbo
vasp/5.4.4.pl2_occmat
vasp/5.4.4.pl2_optaxis
vasp/5.4.4.pl2_transopt
vasp/5.4.4.pl2_vaspsol
vasp/5.4.4.pl2_vtst178
vasp/5.4.4.pl2_z2pack
vasp/6.4.2
vasp/6.4.2_optaxis
vasp/6.4.2_scpc
vasp/6.4.2_vtst198
vaspkit/1.4.1
vaspkit/1.5.1
vasputil/6.1
vesta/3.4.3
virtualflow/15.7
visit/3.3.3
vmd/1.9.3
vmd/1.9.4
wannier90/1.2
wannier90/2.1.0
wannier90/3.1.0
wannierberri/0.15.0
wanniertools/2.5.1
wanniertools/2.7.0
wham/2.0.11
wien2k/21.1_elpa
wien2k/21.1_scalapack
wien2k/23.2_elpa
wien2k/23.2_scalapack
wrf/4.5.2_cray
wrf/4.5.2_intel
xcrysden/1.5.60
xcrysden/1.6.2
xtb/6.4.1
xtb/6.5.0
xtb/6.7.0
xthi/0.1
yambo/5.0.4
yambo/5.0.4_slepc
yambo/5.2.1
yambo/5.2.1_slepc
z2pack/2.2.0
zendnn/4.1
```

Acknowledging

Terms and Conditions regarding Research Publications

Whenever the results of research conducted on the HPC systems at KAUST are published, or the research involved personnel from KAUST Supercomputing Laboratory (KSL), Principal Investigators (PIs) are required to acknowledge the usage of the HPC systems at KAUST and/or the involvement of KSL personnel in their research in their publications. For example, the following statement could be used: **“For computer time, this research used Shaheen III managed by the Supercomputing Core Laboratory at King Abdullah University of Science & Technology (KAUST) in Thuwal, Saudi Arabia**.

Best Practices / Getting help

- When submitting a ticket to help@hpc.kaust.edu.sa requesting help, you will likely get faster resolution by supporting a few best practices:
  - Where possible, provide helpful details that can help speed the process. For example: Project ID, relevant directories, job scripts, jobIDs, modules at compile/runtime, login name, etc.

- One issue per ticket. Do not add unrelated questions on existing tickets
- Do not open multiple tickets on the same unresolved topic.
- Let us know if your issue is fixed or you solved it(and let us know what worked!)

- MOST IMPORTANT: do not hesitate to contact us help@hpc.kaust.edu.sa;

Best Practices

- Check the documentations: https://docs.hpc.kaust.edu.sa/policy/shaheen3.html

- Read the announcements/newsletter sent by emails and available in the website:
  - hpc.kaust.edu.sa/

Thanks !

Agenda

- 8:30am Welcome
- 8:35am Shaheen III Overview
- 8:55am How to apply on Shaheen III
- 9:05am Getting Started on Shaheen III
- 9:15am Software Environment
- 9:35am Job Scheduling
- 10:00am Coffee Break
- 10:15am Storage overview & Best practices
- **10:30am Applications software example: VASP workflow**
- 10:50 am Applications software example: CFD applications
- 11:10 am Applications software example: Bio informatics workflow
- 11:20-11.30am Q&A and Open Discussion

Shaheen III Survey