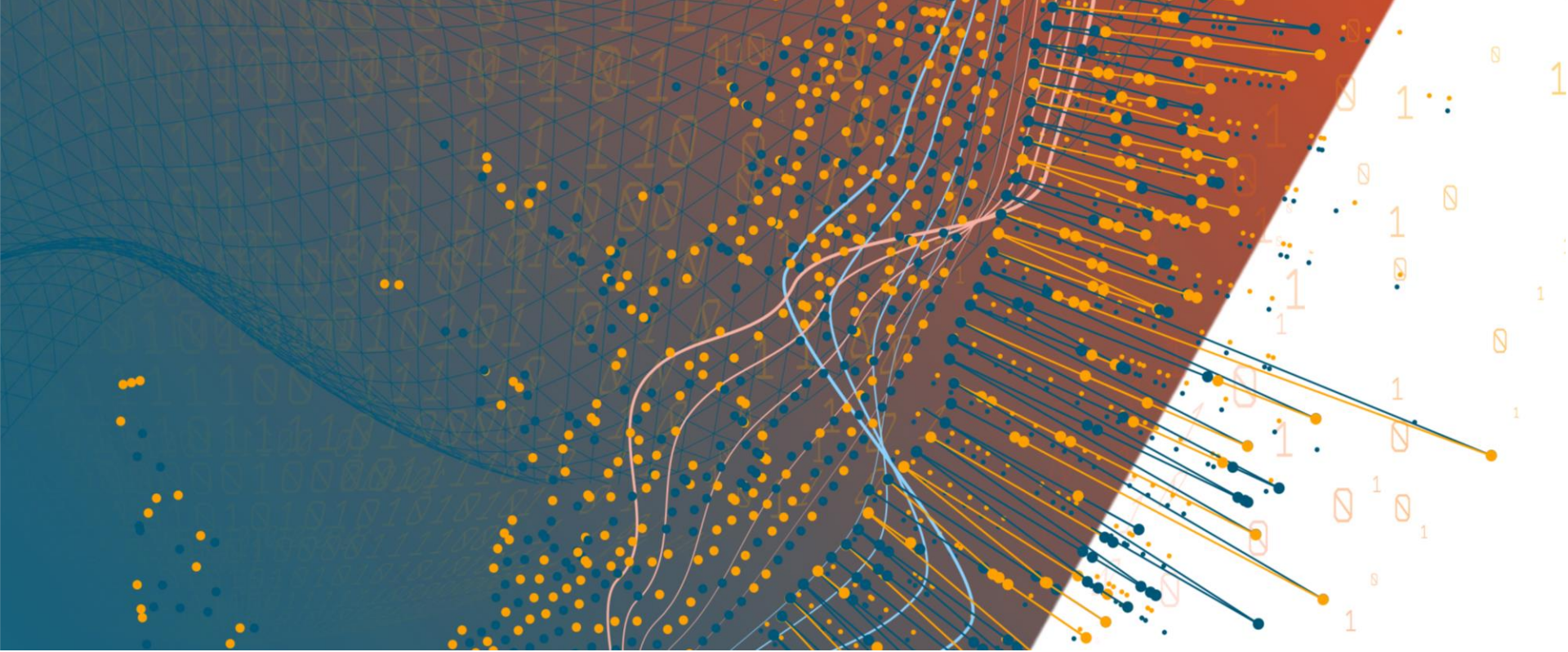


ALTAIR

DELIVERING POWERFUL HPC WITH ALTAIR AND NEC ON THE SX-AURORA TSUBASA SYSTEM

Sujata Patnaik / Senior Software Engineer

Altair PBS Professional™ Capabilities

Altair PBS Professional is a fast, powerful workload manager designed to improve productivity, optimize utilization and efficiency, and simplify administration for HPC clusters, clouds, and supercomputers.

Policy

Meet enterprise-wide scheduling and use policies with resource-based scheduling to ensure the right jobs run at the right times on the right resources

Utilization

Maximize ROI and minimize waste, ensuring the best use of hardware, software licenses, energy, and cloud resources

Performance

Fast and scalable execution to meet the needs of even the most demanding enterprises with thousands of users and millions of cores

Resilience

Secure and hardened to minimize downtime; health checks and automatic failover ensures jobs get done despite faults and failures

Open

Extensible plugin framework to meet each site's unique needs

Simulate

Run a simulation to perform what-if analysis and determine the most productive way to scale HPC resources

Allocate

Allocate resources per compute, storage, budget and limitation requirements; support chargebacks

ALTAIR

Altair PBS Professional™ Features

Automates job scheduling, management, monitoring, and reporting. The trusted solution for Top500 systems and S-M clusters alike.

Highlights

- **Performant:** million-core scalability, fast throughput
- **Secure:** EAL3+ security certification and MLS (SELinux) support
- **Intelligent:** policy-driven and topology-aware scheduling
- **Scalable:** proven to run millions of jobs per day
- **Green Provisioning™:** power management and control
- **Robust:** known in the industry for stability and support
- **Open Architecture:** implement virtually any new policy
- **Modern:** containers, cloud bursting, parallel scheduling
- **Full-spectrum Scheduling**: including both HPC and HTC scheduling
- **Budgeting:** allocation management services to control and optimize budgets
- **Workload Simulating:** Simulate scheduler performance under hypothetical configurations
- **Cloud Bursting:** automatic cloud bursting during peak demand

NASA's workload manager of choice
for all NAS HPC resources
250k+ cores scheduled by one Altair PBS Professional™

ALTAIR

30+ Years of Industry-leading HPC

Automotive	Aerospace	Government + Research	Life Sciences
Ford, Toyota, FCA, Mercedes-Benz, Volkswagen, Honda	BAE Systems, Boeing, Airbus Group, Northrop Grumman, Lockheed Martin, Honeywell	NCAR, NASA, U.S. Naval Research Laboratory, National Institute for Environmental Studies, Federal Aviation Administration, Australian Government Bureau of Meteorology	Vertex, Merck, Biogen, Bayer, Novartis
Oil & Gas	**Semiconductor**	**Education**	**Financial Services**
ConocoPhillips, Chevron, ExxonMobil, Schlumberger, Saudi Aramco	Samsung, tsmc, intel, Qualcomm	Massachusetts Institute of Technology, UNLV, The University of Mississippi, McNeese State University, Johns Hopkins University	American Express, MSCI, Danske Bank, Fifth Third Bank, TransUnion, BMO

ALTAIR

NEC SX-Aurora TSUBASA Vector Engine Integration Overview

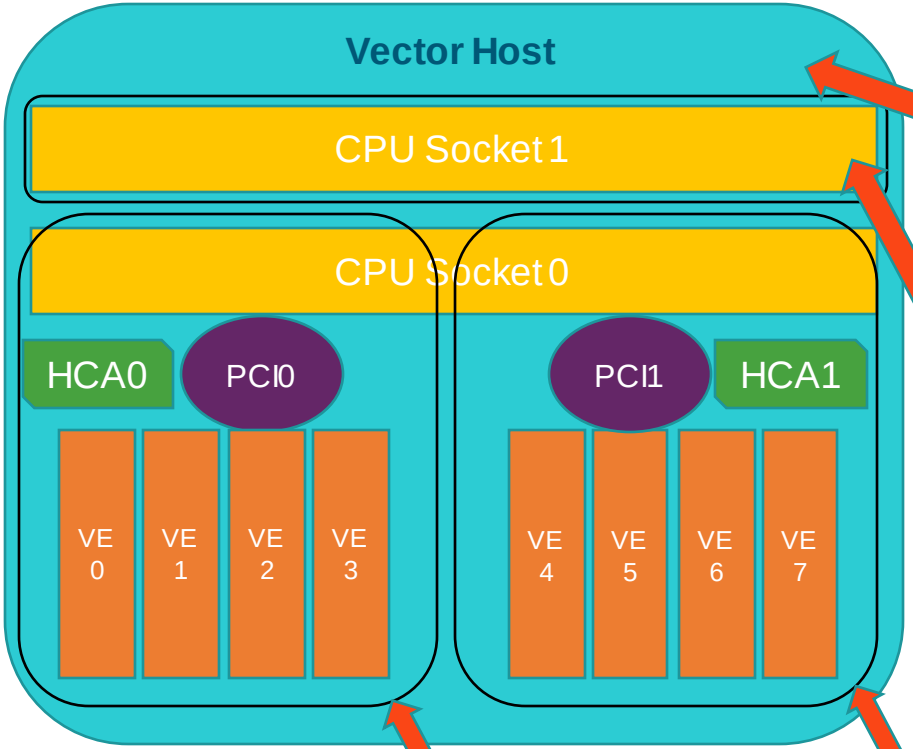
- **Auto detection and configuration of VE's**
- **Topology-aware scheduling**
- **PBS resources - nves**
- **Deep integration with NEC MPI**
  - **Supports all application execution models**: Hybrid, OS offload, VH call, VE offload
- **Integration with NEC accounting**

Adding SX-Aurora TSUBASA Hosts to a PBS Pro Cluster

- **Adding SX-Aurora equipped execution hosts to an existing PBS Pro cluster is simple!**
  - Run a few provided qmgr commands to enable the included SX-Aurora plug-in and create resources
  - Configure scheduler to honor "nves" resource
  - Adjust node sorting to accommodate new class of execution hosts
  - Configure environment variables for NEC MPI integration

PBS Concepts: "vnodes"

- What is a vnode?
  - An abstract object representing a set of resources which form a usable part of a machine
  - Usually represents either a complete host or a NUMA node
  - A single host can be represented as multiple vnodes
  - Each vnode can be managed and scheduled independently
- PBS reflects vector host topology by grouping each PCIe with its associated vector host processors, memory, and vector engines together into one vnode. A NUMA node without its own PCIe is in its own vnode.



pbsnodes –av output extract:

vh1
resources_available.mem = 0b
resources_available.ncpus = 0
resources_available.nves = 0

vh1[1]
resources_available.host = vh1
resources_available.mem = 98304mb
resources_available.ncpus = 40
resources_available.nves = 0

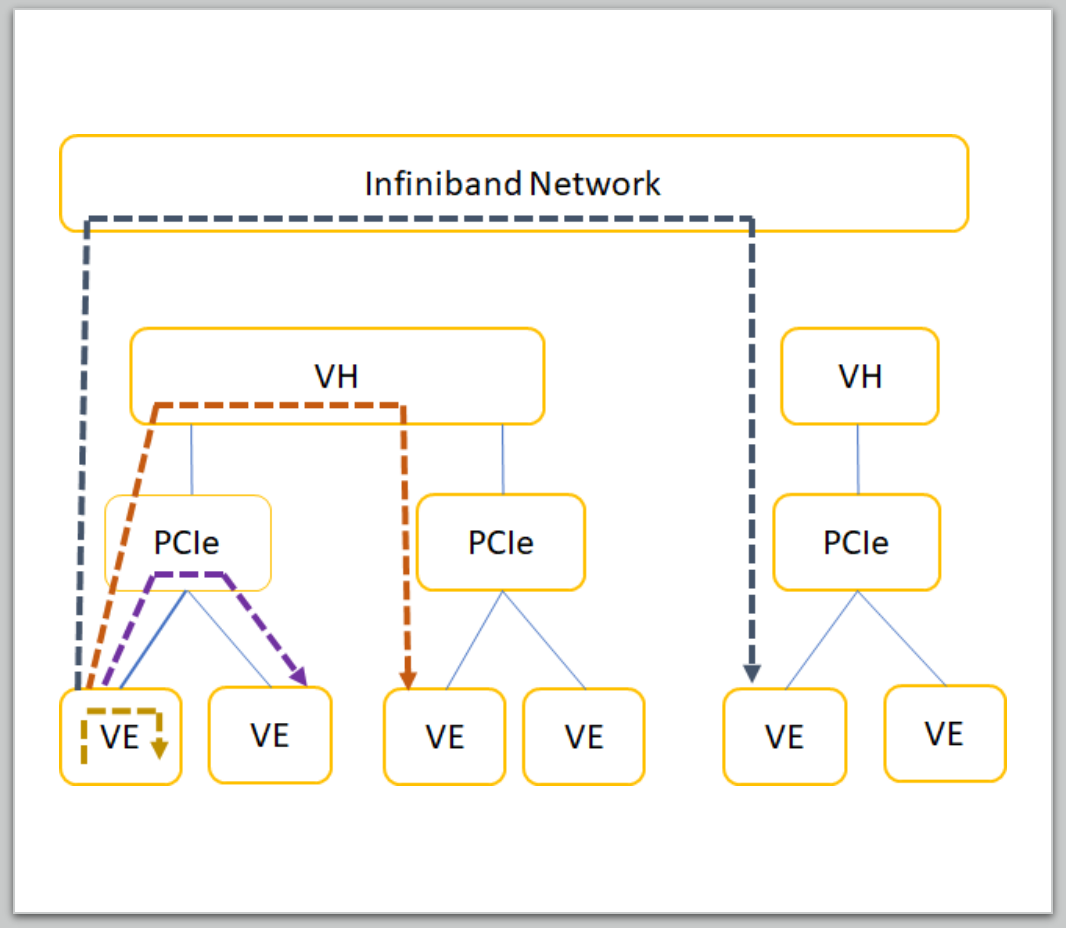
vh1[0]_pci0
resources_available.host = vh1
resources_available.mem = 48470mb
resources_available.ncpus = 16
resources_available.nves = 4

vh1[0]_pci1
resources_available.host = vh1
resources_available.mem = 48470mb
resources_available.ncpus = 16
resources_available.nves = 4

ALTAIR

TSUBASA Interconnect

PBS does topology-aware scheduling by grouping job processes on vector engines in a way that produces the lowest communication overhead



PBS Concepts: Resource Requests in "chunks"

- Job resources are requested in "chunks" via "qsub -lselect="
  - E.g., requesting a total of 32 CPUs and 64GB memory in 4 equal chunks
    - **`qsub -l select=4:ncpus=8:mem=16GB`**
  - Host-based resources are requested in chunks, **including SX-Aurora Vector Engines (nves)**

- Chunks are allocated to compute resources based on "qsub -lplace=" request
  - E.g., to guarantee the above job will run across 4 different hosts, add "-lplace=scatter"
    - `qsub -l select=4:ncpus=8:mem=16GB` **`-lplace=scatter`**
  - Other "place" options include "free", "pack", "vscatter"
  - Sharing/exclusivity/grouping options are available as well

ALTAIR

Execution Model Example: OS Offload

NEC MPI launches processes onto VE processors

```
qsub -l select=ncpus=2:mpiprocs=8:nves=3
```

In job script:

```
mpirun -np 8 ve.out
```

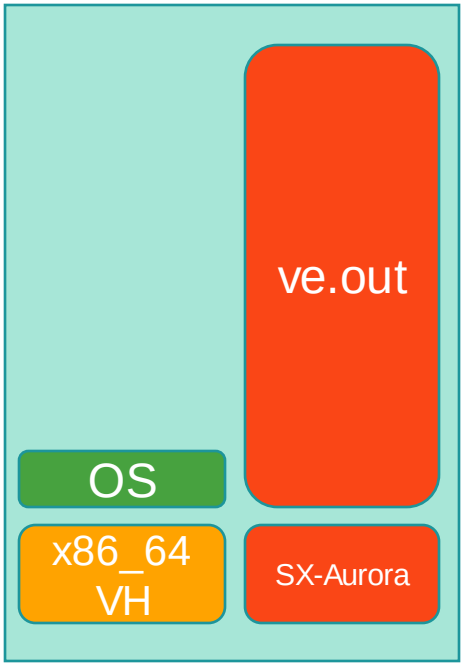
VE 1 gets 3 VE processes, VE 2 gets 3 VE processes, VE 3 gets 2 VE processes

```
qsub -lselect=1:nves=3:mpiprocs=6 -vNEC_PROCESS_DIST=1:3:2
```

In job script:

```
mpirun -np 6 ve.out
```

1 process will launch on the first VE, 3 on the second, and 2 on the third



ALTAIR

Execution Model Example: Hybrid

NEC MPI launches process onto both VH(x86_64) processors and VE processors

```
qsub -
lselect=1:ncpus=2:nves=1:mpiprocs=5+1:ncpus=3:mpiprocs=3+1:ncpus=2:nves=1
:mpiprocs=4 -v NEC_PROCESS_DIST=S2:3+S3+4
```

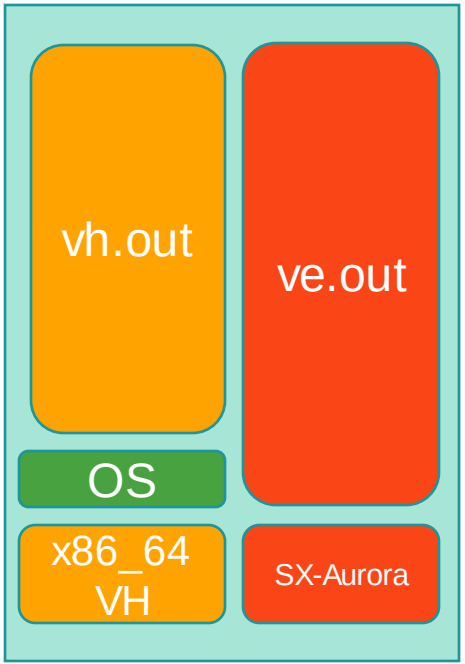
In job script:

```
mpirun -vh -np 2 vh.out : -np 3 ve.out : -vh -np
3 vh.out : -np 4 ve.out
```

In the first chunk, NEC MPI will launch the first 2 processes on the VH, and the last 3 will run on the VE.

In the second chunk, NEC MPI will launch all 3 processes on the VH.

In the third chunk, NEC MPI will launch all 4 processes on a VE



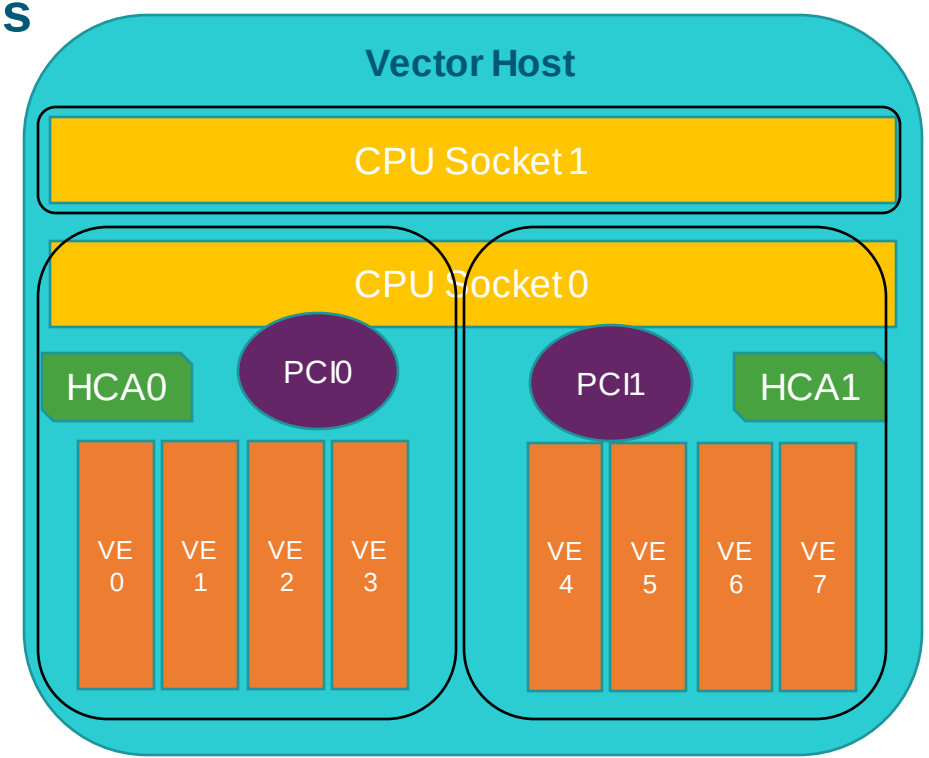
ALTAIR

Grouping Job Processes on Nodes

Chunk -

```
 1:ncpus=2:nves=1:mpiprocs=5+1
:ncpus=3:mpiprocs=3+1:ncpus=2
:nves=1:mpiprocs=4 -v
NEC_PROCESS_DIST=S2:3+S3+4
```

(vh1[0]_pci0:ncpus=2:nves=1)+(vh1[0]_pci0:ncpus=3)+(vh1[0]_pci0:ncpus=2:nves=1)



ALTAIR

Execution Model Example: VH Call

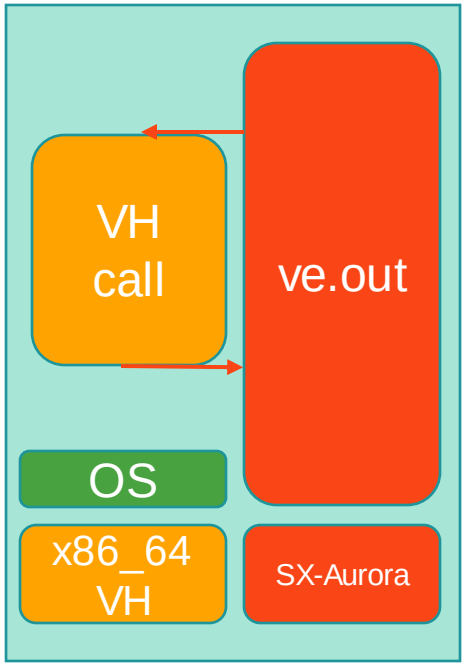
NEC MPI launches processes onto VE processors, which offload some work to VH(x86_64) processors

```
qsub -lselect=1:ncpus=2:nves=3:mpiprocs=6
```

In job script:

```
mpirun -np 6 ve.out
```

2 processes will launch on each VE, and ve.out starts processes on the associated VH processors



ALTAIR

Execution Model Example: VE Offload

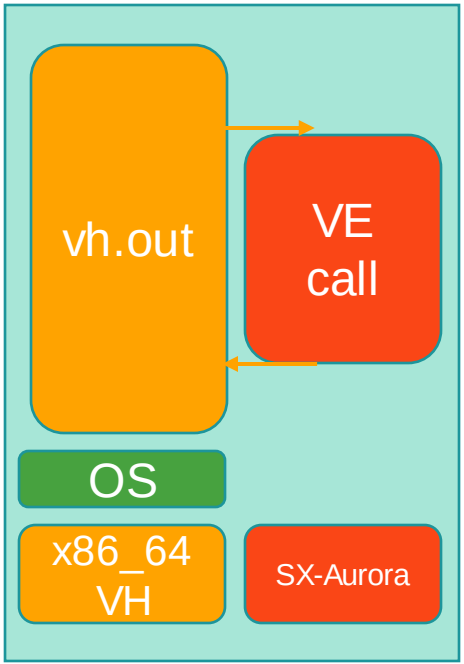
NEC MPI launches processes onto VH(x86_64) processors, which offload some work to VEs processors

```
qsub -l select=1:ncpus=2:mpiprocs=2:nves=2 -
vNEC_PROCESS_DIST=S2:0
```

In job script:

```
mpirun -vh -np 2 vh.out
```

2 processes will launch on the VH, and vh.out starts processes on associated VEs



ALTAIR

SX-Aurora Specific Accounting Metrics

- **ve_mem:** total memory consumption of a job on assigned vector engines
- **ve_cput:** total VE processor time consumption of a job on assigned vector engines

Example end of job record:

06/04/2021 10:49:08;E;1022.pbsserver;user=sujata group=sujata project=_pbs_project_default jobname=STDIN queue=workq ctime=1597933702 qtime=1597933702 etime=1597933702 start=1597933702 exec_host=pbsserver/0*8 exec_vnode=(vnode1:ncpus=8:nves=2) Resource_List.ncpus=8 Resource_List.nodect=1 Resource_List.place=free Resource_List.select=ncpus=8:nves=2 session=20340 end=1597987148 Exit_status=0 resources_used.cpupercent=0 resources_used.cput=00:00:00 resources_used.mem=0kb resources_used.ncpus=8 resources_used.nves=2 resources_used.vmem=0kb **resources_used.ve_mem=1934190kb resources_used.ve_cput=7380** resources_used.walltime=00:61:57 run_count=1

Suspend / Resume

- When a job is suspended, PBS suspends the execution of VE processes and VH processes belonging to the job and swaps out the VE processes.
- PBS runs the higher priority job.
- Once the higher priority job is finished, the VE processes of the suspended jobs are swapped in to the respective VE's before resume.

THANK YOU

altair.com

#ONLYFORWARD

ALTAIR