
Job Packs – A New Slurm Feature For Enhanced Support of Heterogeneous Resources

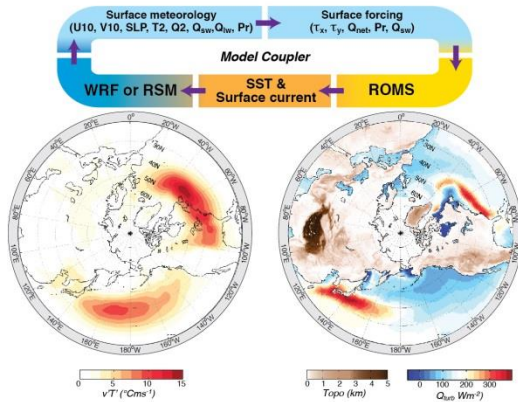
Slurm User Group 2016 : Andry Razafinjatovo, Atos
Martin Perry, Atos
Yiannis Georgiou, Atos
Matthieu Hautreux, CEA



Multiple Program Multiple Data usage

- ▶ Application lifecycle is more and more complex, usage of MPMD programming will allow:

Scripps Coupled Ocean Atmosphere Regional (SCOAR) Model
Seo et al. (2007; 2014, J. Climate); <http://hseo.whoi.edu/scoar/>



to be more **precise**
by using coupled applications

to be more **accurate**
by validating results
through vizualisation

to be more **performant**
by using software/hardware
accelerators

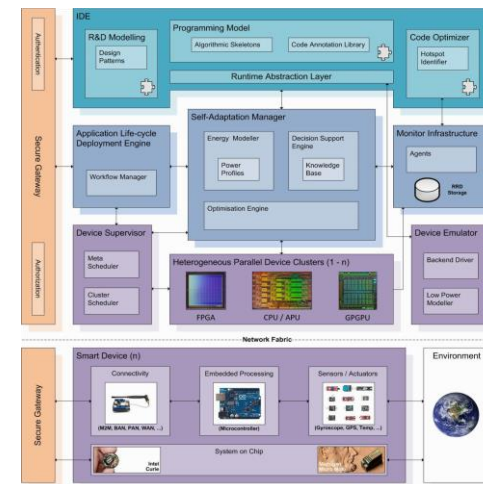
Considering energy efficiency from code design to execution for heterogeneous architectures



- ▶ Atos/Bull leading the Tango project (Transparent Heterogeneous hardware Architecture deployment for eEnergy Gain in Operation)
- ▶ Extension of currently available programming models and resource and job management systems to support complex heterogeneous architectures
- ▶ Code optimizer engine with the aim of optimizing code mapping. to reduce power consumption by the application.
- ▶ Power-awareness integrated in the whole software development optimization and execution process



<http://www.tango-project.eu/>



Slurm SPMD Support

- ▶ Conventional Slurm jobs/steps support the SPMD (Single Program Multiple Data) programming model.
- ▶ The job/step may be allocated a diverse collection of resources (nodes with different characteristics), but the allocated resources are managed as a single group. A single type of executable (program) is run on the allocated resources. Example:
 - ▶ `srun -w bignode1,smallnode1 -n2 ./myprog`

Limited MPMD Support

- ▶ Slurm also supports a limited form of MPMD (Multiple Program Multiple Data), with the `srun --multi-prog` option.
- ▶ This option allows multiple programs to be executed by a single step, but the allocated resources are still managed as a single group and the user has limited control over which program is run on which node. Example:

```
srun -w bignode1,smallnode1 -n4 --multi-prog prog.conf
```

Need For Enhanced MPMD Support

- ▶ For some multi-program applications, it is desirable to target a particular program to a particular subset of the allocated resources. For example:
 - A node with lots of memory for the serial startup/wrapup phase.
 - Lots of nodes with GPU for the parallel phase.
 - Nodes with fast I/O to store the results.

For MPI applications, the programs may need to belong to a common `MPI_COMM_WORLD`.

Introducing Job Packs

- ▶ Job Packs is a new Slurm feature providing enhanced support for the MPMD programming model.
- ▶ A Job Pack is a group of **co-scheduled** jobs.
- ▶ Each job has its own group of resources, but no job is scheduled until the resources required for all of the jobs in the pack are available.
- ▶ Multiple steps may be launched concurrently for some or all of the jobs in the same pack. The steps may be combined into a single MPI application.
- ▶ This allows the user to allocate multiple, heterogeneous groups of resources and launch multiple program types concurrently, targeting each program to a group of resources tailored to its requirements.

Job Packs User Interface

- ▶ A job pack is created with a single sbatch, salloc or (standalone) srun command.
- ▶ Multiple steps may be launched concurrently for one or more jobs in the pack with a single srun command.
- ▶ The sbatch, salloc and srun command syntax has been expanded to support this new functionality. The new syntax is fully backwards-compatible with the current syntax. The man pages have been updated with the new syntax.

Job Packs User Interface Detail

- ▶ A job pack is defined by a **job pack description**.
- ▶ A job pack description consists of a series of colon-separated **job descriptions**.
- ▶ Each job description defines a single job in the pack.
- ▶ A job pack may have an arbitrary number of jobs.
- ▶ Each job is assigned an index called a **pack group** based on the order of its job description in the job pack description.
- ▶ The first job has a pack group of zero and is the **pack leader**. The remaining jobs in the pack are **pack members**.

- ▶ srun is used to launch one or more concurrent steps for a job pack created by salloc or sbatch.
- ▶ The steps to be launched are defined by a series of colon-separated **step descriptions**. A new srun option, **--pack-group**, is used to identify each job for which a step is to be launched from a step description.

Job Packs User Interface Example

The following example uses `salloc` to create a job pack containing three jobs, the pack leader plus two pack members:

```
salloc -p ctlnodes -N1 : -p computenodes -N10 : -p ionodes -N4
```

Job Pack Description		
Job Description	Job Description	Job Description
Pack Group 0	Pack Group 1	Pack Group 2
Pack Leader	Pack Member	Pack Member

The following `srun` command launches one step for each job in the pack. Each step runs a different program:

```
srun --pack-group=0 ./controller : --pack-group=1 -n 20 ./worker :  
--pack-group=2 -n 6 ./storer
```

Job Packs sstat & sacct Changes

- ▶ Two new fields have been added to the step accounting record and in-memory step record. sacct and sstat have been enhanced to report these new fields.

PackJobID

Jobid of first step in multi-step srun

PackStepID

Stepid of first step in multi-step srun

- ▶ This information is saved for every step in a multi-step srun, providing a unique identifier that ties together all of the steps launched by the same srun command.

Job Packs Additional Command Changes

► Additional command changes to support Job Packs:

squeue new option **--dependency** to display dependencies between the pack leader and pack members.

scontrol *show job* output enhanced to display dependencies between the pack leader and pack members.

scancel new option **--pack-member** required to cancel pack members.

srun **--label** option displays utaskids by default. utaskid is a new type of taskid that is unique for each task launched by the same srun command. For conventional (single-step) sruns, utaskid is the same as gtaskid.

Job Packs Configuration

- ▶ No new or changed Slurm configuration options are required to use Job Packs.
- ▶ A new debug flag, JobPack, has been defined to provide detailed log entries about Job Pack scheduling, allocation, and task management. The new log entries are prefaced with the string **JPCK**.

Job Packs Environment Variables

- ▶ Many new environment variables have been created to facilitate communication of Job Pack-related information between salloc/sbatch and srun, and between srun and slurmstepd. The new environment variables may also be of value for user software (monitoring, etc.)
- ▶ The Slurm documentation has been updated to describe these new variables.

Job Packs MPI MPMD Support

- ▶ Tight integration of srun with MPI MPMD
- ▶ MPI support allows multiple steps launched concurrently by a single srun command, for one or more jobs in a job pack, to be combined into a single MPI application; that is, a single MPI_COMM_WORLD communicator with a single set of MPI ranks.
- ▶ Users may also choose to run each step launched by a multi-step srun as a separate MPI application, using the new srun option **--mpi-combine=no**.
- ▶ Job Packs support tight integration for srun through PMI-1 and PMI-2.
- ▶ IntelMPI and OpenMPI implementations have been validated but others may also work if they support PMI-1 or PMI-2
- ▶ We plan to add support for OpenMPIv2/PMIx.

Job Packs MPI MPMD Support - Example

```
[trek2] (slurm) mpihello> salloc -w trek7 --exclusive : -w trek8 --exclusive  
: -w trek9 --exclusive  
salloc: Pending job allocation 10236  
salloc: Pending job allocation 10237  
salloc: Granted job allocation 10238  
salloc: Nodes trek7 are ready for job 10238  
salloc: Nodes trek8 are ready for job 10237  
salloc: Nodes trek9 are ready for job 10236
```

```
[trek2] (slurm) mpihello> srun --pack-group=0 -n3 ./mpiexec1 : --pack-group=1  
-n3 ./mpiexec2 : --pack-group=2 -n3 ./mpiexec3 | sort  
Hello world, I am mpiexec1, rank 0 of 9 - running on trek7  
Hello world, I am mpiexec1, rank 1 of 9 - running on trek7  
Hello world, I am mpiexec1, rank 2 of 9 - running on trek7  
Hello world, I am mpiexec2, rank 3 of 9 - running on trek8  
Hello world, I am mpiexec2, rank 4 of 9 - running on trek8  
Hello world, I am mpiexec2, rank 5 of 9 - running on trek8  
Hello world, I am mpiexec3, rank 6 of 9 - running on trek9  
Hello world, I am mpiexec3, rank 7 of 9 - running on trek9  
Hello world, I am mpiexec3, rank 8 of 9 - running on trek9
```


Job Packs Example

The following example illustrates the use of Job Packs with MPI. The (standalone) `srun` command creates a Job Pack with three jobs and launches one step for each job. Each job is allocated a unique set of nodes. The pack leader jobid is 61663 and the pack member jobids are 61661 and 61662. The first step, for the pack leader, runs a single copy of program *controller*. The second step, for the first pack member, runs 4 copies of program *worker*. The third step, for the second pack member, runs two copies of program *storer*. The three steps are combined into a single `MPI_COMM_WORLD` with a MPI size of 7 and MPI ranks of 0-6.

```
$ srun -JLdr -w trek7 ./controller : -JMr1 --gres=gpu -N2 --tasks-per-node=2 ./worker  
      : -JMr2 -pt96iopx -N2 ./storer &
```

```
$ squeue
```

JOBID	PARTITION	NAME	USER	ST	TIME	NODES	NODELIST (REASON)
61661	t96iopx	Mr2	slurm	R	0:03	2	trek[8-9]
61662	trekall	Mr1	slurm	R	0:03	2	trek[4-5]
61663	trekall	Ldr	slurm	R	0:03	1	trek7

```
This is the controller, Name=Ldr Id=61663 MPI Rank 0 of 7, Running on host trek7
```

```
This is a worker, Name=Mr1 Id=61662 MPI Rank 1 of 7, Running on host trek4
```

```
This is a worker, Name=Mr1 Id=61662 MPI Rank 2 of 7, Running on host trek4
```

```
This is a worker, Name=Mr1 Id=61662 MPI Rank 3 of 7, Running on host trek5
```

```
This is a worker, Name=Mr1 Id=61662 MPI Rank 4 of 7, Running on host trek5
```

```
This is a storer, Name=Mr2 Id=61661 MPI Rank 5 of 7, Running on host trek8
```

```
This is a storer, Name=Mr2 Id=61661 MPI Rank 6 of 7, Running on host trek9
```

Limitations

▶ **Array jobs**

- Job Packs **don't support array jobs** "yet". An error message is displayed in case array jobs are part of a job pack

▶ **Preemption**

- **Job Packs** (pack leader and pack members) **cannot be preempted** by other higher priority jobs. This is an exception for now and it will be treated in an upcoming version.
- The pack leader can preempt legacy jobs, but only the resource requirements of the leader will be used to find lower priority. (When the pack leader's request is evaluated, the members have already been allocated.)
- **Pack members cannot preempt other jobs.** This is because, when a Job Pack is to be allocated, each member is allocated. When any member (or the leader) can't be allocated, the previous members are deallocated. If such a deallocated job had preempt a normal job, that normal job should still be running.

▶ **PMI-x**

- PMI-x is not yet supported through the srun MPMD tight integration

Status & Documentation

- ▶ Job Packs will be available in the next Slurm Version (17.02).
- ▶ A Job Packs Guide is provided in the Slurm html documentation.
- ▶ Man pages for all affected commands have been updated.
- ▶ Next steps:
 - PMI-X support
 - Support of array jobs

Design & Development

Rod Schultz, Atos

Martin Perry, Atos

Bill Brophy, Atos

Doug Parisek, Atos

Nancy Kritkausky, Atos

Yiannis Georgiou, Atos

Matthieu Hautreux, CEA

Job Packs development is part of the functionalities developed for the European-funded H2020 Project Tango.

Thanks

For more information please contact:

T+ 33 1 98765432

F+ 33 1 88888888

M+ 33 6 44445678

martin.perry@atos.net, yiannis.georgiou@atos.net

Atos, the Atos logo, Atos Consulting, Atos Worldgrid, Worldline, BlueKiwi, Bull, Canopy the Open Cloud Company, Yunano, Zero Email, Zero Email Certified and The Zero Email Company are registered trademarks of the Atos group. August 2015. © 2015 Atos.

Confidential information owned by Atos, to be used by the recipient only. This document, or any part of it, may not be reproduced, copied, circulated and/or distributed nor quoted without prior written approval from Atos.
