

# SLURM

Берсенёв Александр, МГКН1  
mail: [bay@hackerdom.ru](mailto:bay@hackerdom.ru)  
icq: 1862222

# SLURM

---

- The Simple Linux Utility for Resource Management
- Основные функции:
  - Выделение ресурсов задаче
  - Старт и наблюдение за процессами на узлах
  - Ведение очереди

# Background

---

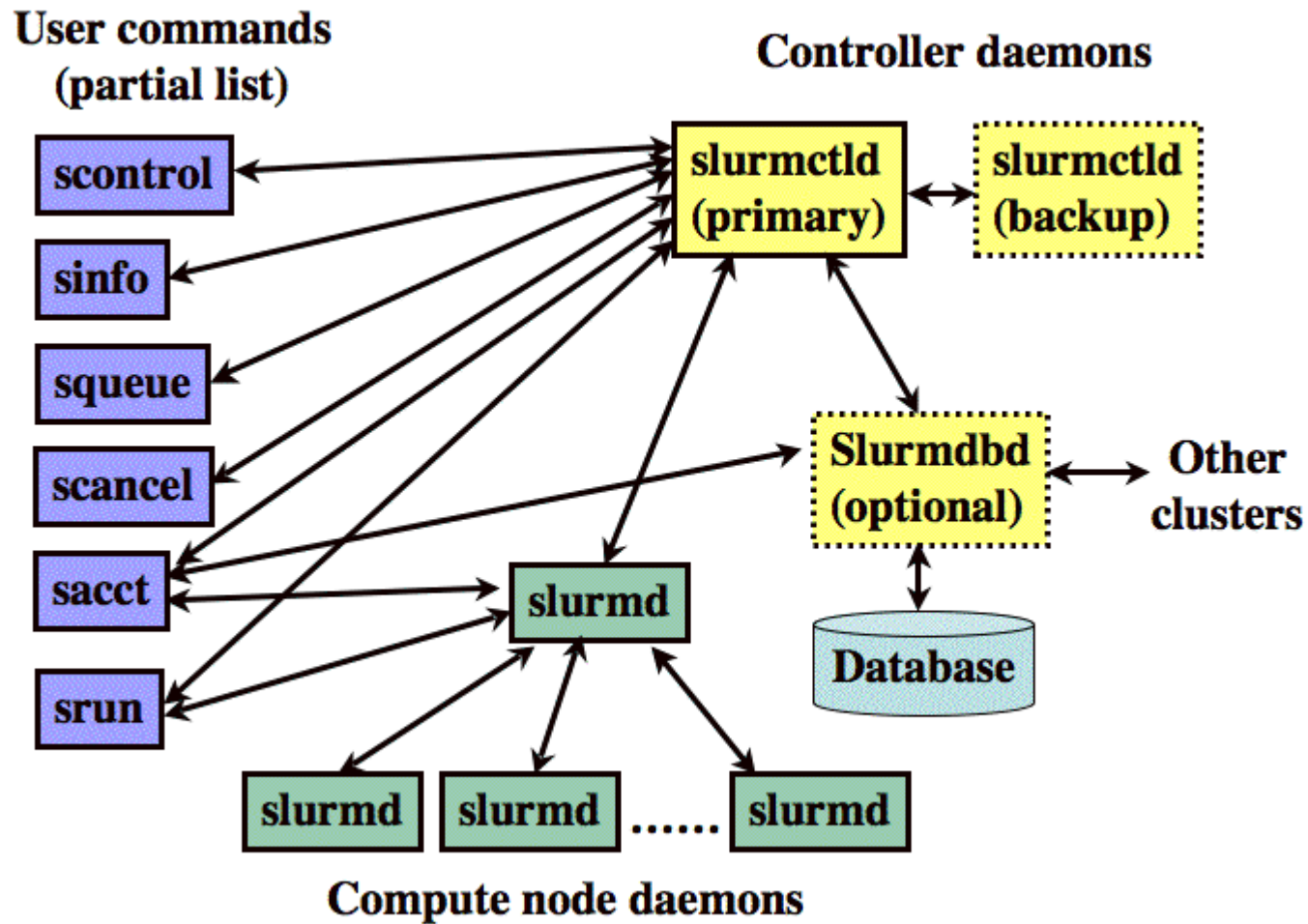
- Разработка началась в 2002 году
  - разработчик: **Lawrence Livermore National Laboratory**
  - позднее присоединились: Hewlett-Packard, SchedMD, Bull, Linux NetworX и др.
- Активно развивается в настоящее время(2011)
- По оценкам, в 2010 году SLURM используется на ~40% систем из Top500
  - Tianhe-1A(14,336 Intel CPUs and 7,168 NVIDIA Tesla M2050 GPUs)
  - Tera 100(140,000 Intel Xeon 7500 processing cores)
  - BlueGene/P(147,456 PowerPC 450 cores)

# Обзор

---

- Open source(GPL)
- Написан на C
- Модульная архитектура
  - модули отвечают за различные виды соединения узлов, механизмы аутентификации, планировщики и.т.д.

# КОМПОНЕНТЫ



# Возможности

---

- Поддержка MPI
- Поддержка Checkpoint/Restart
- Поддержка GPU
- Поддержка резервирования ресурсов
- Одновременное обслуживание нескольких кластеров, нескольких очередей, поддержка неоднородных кластеров
- Продвинутое ведение статистики

# Конфигурирование

---

- Пример описания кластера

`/etc/slurm/slurm.conf:`

`ControlMachine=um64`

`NodeName=umu[2-24] Procs=4 Sockets=2 CoresPerSocket=2 ThreadsPerCore=1`

`State=UNKNOWN`

`PartitionName=main Nodes=umu[2-24] Default=YES MaxTime=INFINITE`

`State=UP`

# Работа с SLURM

- Все команды SLURM начинаются с буквы s:

**srun**

sbatch

salloc

**scancel**

sinfo

**squeue**

smap

sattach

scontrol

sbcast

sview

strigger

sreport

sacct

sacctmgr



# Работа с SLURM

---

- **srun** — запуск задачи на счет
  - аналог mpirun

# Работа с SLURM

---

- **srun — запуск задачи на счет**

- аналог mpirun

- интересные ключи:

- n – число задач для запуска(аналог -np)

- t – максимальное время работы задачи(аналог -maxtime)

- пример: `srun -n 16 -t 2:00:00 hostname`

# Работа с SLURM

## ▪ **srun** — запуск задачи на счет

– аналог mpirun

– интересные ключи:

-n – число задач для запуска(аналог -np)

-t – максимальное время работы задачи(аналог -maxtime)

– пример: `srun -n 16 -t 2:00:00 hostname`

– еще ключи: `--begin=<time>` – *отложить запуск задачи*

`--comment=<string>` – *комментарий*

`--exclusive` и `--share` – *могут ли 2 задачи быть на 1 узле*

`--mail-type=<type>` и `--mail-user=<user>` – *оповещение*

`--mpi=<mpi_type>` – *тип MPI*

`--N <int>` – *количество узлов для запуска*

`--time-min=<time>` – *минимальное допустимое время*

`--odelist=<hosts>` и `--exclude=<hosts>`

# Работа с SLURM

## ▪ **squeue** — просмотр очереди

- аналог mqinfo
- пример вывода:

JOBID	PARTITION	NAME	USER	ST	TIME	NODES	NODELIST (REASON)
340	main	sleep	u1333	PD	0:00	1	(Priority)
341	main	sleep	u1333	PD	0:00	1	(Priority)
339	main	sleep	u1333	PD	0:00	1	(Priority)
338	main	sleep	u1333	PD	0:00	1	(Resources)
337	main	sleep	u1333	R	7:22	1	umu24
336	main	sleep	u1333	R	7:35	22	umu[2-23]

- Ключи: --start, -l

## ▪ **scancel** — отмена выполнения

- аналог mqdel и mkill
- пример: scancel 336 337

## ▪ **smap**

# Работа с SLURM

---

- **sattach** – присоединение к `stdin` и `stdout` процесса
- **salloc** – выделение ресурсов
  - ключи – как у `srun`
  - пример: `salloc -n 5`
- **sbatch** – запуск пакетной задачи

# Администрирование SLURM

## ▪ **scontrol** – управление кластером

– если запустить без параметров, попадем в интерфейс утилиты

– Команды:

show и update

*аналог lockproc и unlockproc:*

```
update NodeName=umu24 State=POWER_DOWN Reason="Changing a memory"
```

```
update NodeName=umu24 State=DOWN Reason="Cloud experiments"
```

```
update NodeName=umu24 State=IDLE
```

hold и release

create

```
create reservation starttime=2009-02-06T16:00:00 \
```

```
duration=120 user=root flags=maint,ignore_jobs nodes=ALL
```

```
create reservation user=alan,brenda \
```

```
starttime=noon duration=60 flags=daily nodecnt=10
```

# Статистика

## ▪ **sacct** — информация о законченных задачах

JobID	JobName	Partition	Account	AllocCPUS	State	ExitCode
348.0	hostname		oso	2	COMPLETED	0:0
348.1	hostname		oso	2	COMPLETED	0:0
348.2	hostname		oso	2	COMPLETED	0:0
349	bash	main	oso	5	RUNNING	0:0
349.0	hostname		oso	5	COMPLETED	0:0
349.1	hostname		oso	5	COMPLETED	0:0

## ▪ **sreport** — фабрика отчетов

- Какие пользователи больше всего считали в отделе, какие отделы больше считали
- Размер задач по отделам
- Использование резервов
- По пользователям

# Безопасность

---

- Для аутентификации используется munge
- Для управления доступом на узел – pam\_slurm
- Два уровня администраторов: operator и admin



Спасибо!