

# Обзор планировщиков для SLURM

Jobs which have been queued for more than 15 days will be considered starving and heroic measures will be taken to attempt to run them

Swiss National Computer Centre

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

# План

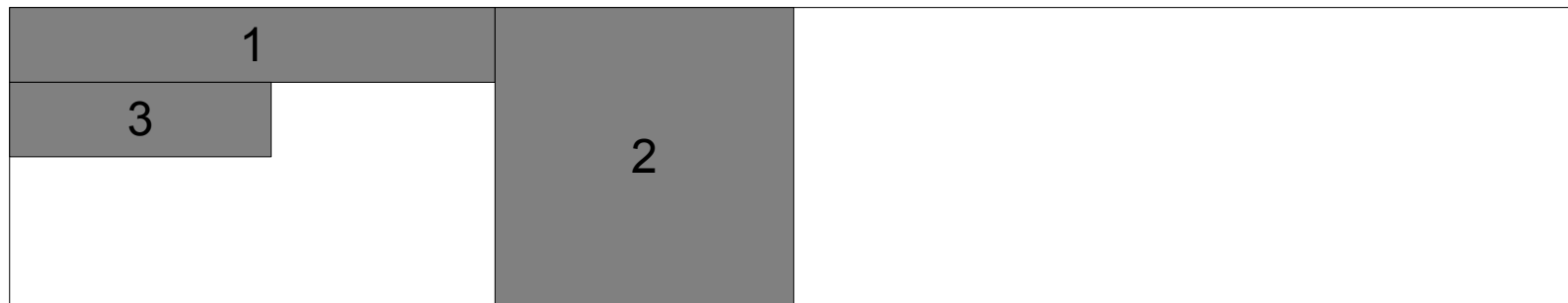
---

- Backfill
- Maui
  
- Вопросы

# Backfill

- Принципы

- ставим задачи в очередь в порядке приоритета
- если имеется менее приоритетная задача и свободные ресурсы, то ставим эту задачу раньше



# Приоритеты

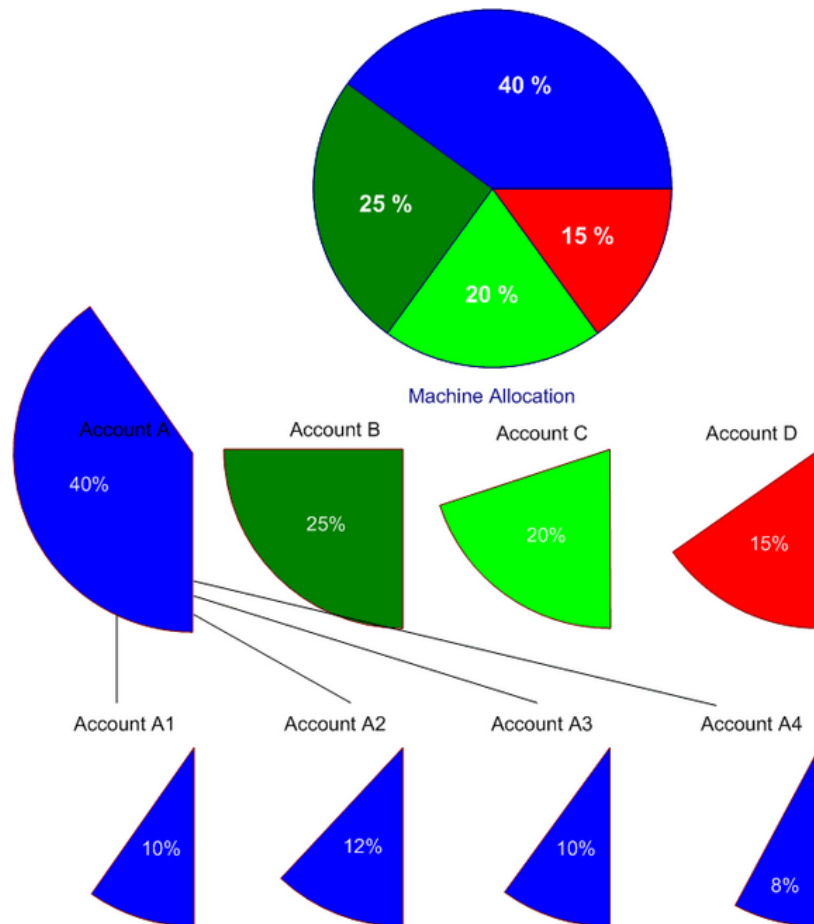
- Факторы, влияющие на приоритет
  - время, проведённое задачей в очереди
  - Fair-share: разница между обещанной долей ресурсов и потребленной долей ресурсов
  - размер задачи
  - коэффициент очереди
  - QOS

Job\_priority =

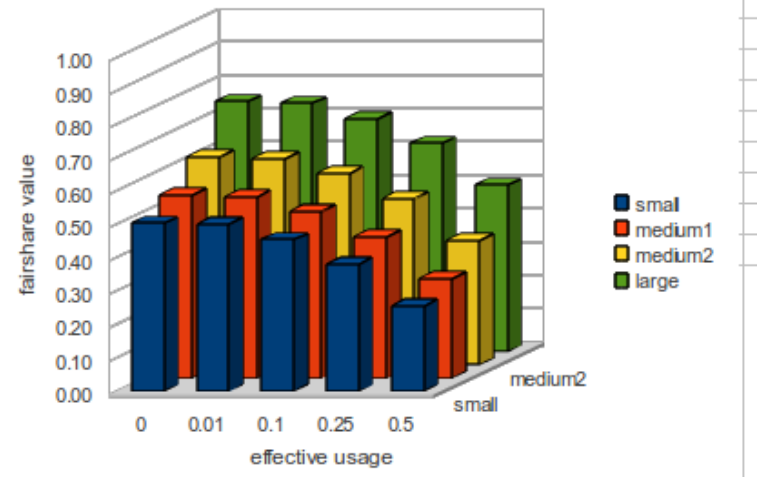
$$\begin{aligned} & (\text{PriorityWeightAge}) * (\text{age\_factor}) + \\ & (\text{PriorityWeightFairshare}) * (\text{fair-share\_factor}) + \\ & (\text{PriorityWeightJobSize}) * (\text{job\_size\_factor}) + \\ & (\text{PriorityWeightPartition}) * (\text{partition\_factor}) + \\ & (\text{PriorityWeightQOS}) * (\text{QOS\_factor}) \end{aligned}$$

# Fair share

- Учитывается  $\langle \text{число процессоров} \rangle * \langle \text{число секунд} \rangle$
- Планируют добавить  $\langle \text{количество используемой памяти} \rangle$ 
  - «Период полураспада»
  - $F = 2^{**}(-U/S)$

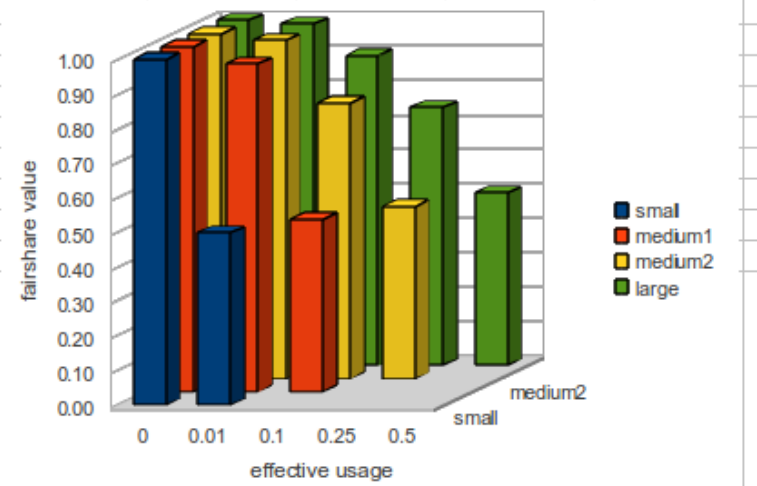


	A	B	C	D	E	F
1	<b>Current FairShare formula</b>					
2			normalized shares			
3			0.01	0.10	0.25	0.50
4			small	medium1	medium2	large
5	effective usage	0.00	0.51	0.55	0.63	0.75
6		0.01	0.50	0.55	0.62	0.75
7		0.10	0.46	0.50	0.58	0.70
8		0.25	0.38	0.43	0.50	0.63
9		0.50	0.26	0.30	0.38	0.50



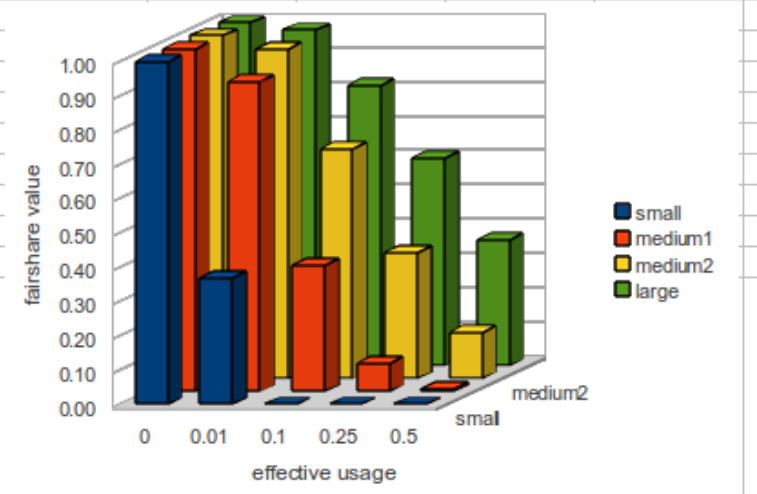
10						
----	--	--	--	--	--	--

11	<b>Linear Relative FairShare with PriorityMinFS = 2</b>					
12			normalized shares			
13			0.01	0.10	0.25	0.50
14			small	medium1	medium2	large
15	effective usage	0.00	1.00	1.00	1.00	1.00
16		0.01	0.50	0.95	0.98	0.99
17		0.10	0.00	0.50	0.80	0.90
18		0.25	0.00	0.00	0.50	0.75
19		0.50	0.00	0.00	0.00	0.50



20						
----	--	--	--	--	--	--

21	<b>Exponential Relative FairShare</b>					
22			normalized shares			
23			0.01	0.10	0.25	0.50
24			small	medium1	medium2	large
25	effective usage	0.00	1.00	1.00	1.00	1.00
26		0.01	0.37	0.90	0.96	0.98
27		0.10	0.00	0.37	0.67	0.82
28		0.25	0.00	0.08	0.37	0.61
29		0.50	0.00	0.01	0.14	0.37



30

# Fair share

```
extern double priority_p_calc_fs_factor(long double usage_efctv,  
                                       long double shares_norm)  
{  
    double priority_fs;  
  
    xassert(usage_efctv != (long double)NO_VAL);  
  
    if (shares_norm > 0.0)  
        priority_fs = pow(2.0, -(usage_efctv / shares_norm));  
    else  
        priority_fs = 0.0;  
  
    return priority_fs;  
}
```

# sprio

- Для объяснения пользователям почему задача имеет такой приоритет используется утилита sprio:

```
[u1333@um64 ~]$ sprio
```

JOBID	PRIORITY	AGE	FAIRSHARE	JOBSIZE	PARTITION	QOS
479	219	0	90	29	100	0
480	219	0	90	29	100	0
481	219	0	90	29	100	0
482	129	0	0	29	100	0
483	129	0	0	29	100	0

```
[u1333@um64 ~]$ sprio -w
```

JOBID	PRIORITY	AGE	FAIRSHARE	JOBSIZE	PARTITION	QOS
Weights		100	100	100	100	100



# Gang scheduling

---

- Идея: останавливать менее приоритетную задачу чтобы запустить более приоритетную.
- Способы остановки:
  - CANCEL
  - CHECKPOINT
  - REQUEUE
  - SUSPEND
- Проблема памяти
  - Есть возможность задавать максимальное число памяти на 1 CPU

# Резервирование ресурсов

- `scontrol create reservation starttime=2009-02-06T16:00:00 \`  
`duration=120 user=root flags=maint,ignore_jobs nodes=ALL`
- `scontrol create reservation user=alan,brenda \`  
`starttime=noon duration=60 flags=daily nodecnt=10`
- !!! Чтобы запустится на зарезервированном ресурсе, пользователь должен указать это явно:

```
sbatch --reservation=alan_6 -N4 my.script
```

# Maui

- Разработка началась в середине 90-х
- Cluster Resources, основной разработчик Maui, переключилась на разработку Moab, коммерческого планировщика в 2005
- Поддерживает несколько менеджеров ресурсов, включая SLURM
- Общается с менеджером ресурсов по протоколу wiki
- Maui does not trust resource manager. All node and job information is reloaded on each iteration(с оф. сайта).
- Предоставляет свой набор утилит для управления очередью
  - использование утилит SLURM так же возможно
- Использует одну очередь

# Maui

- Факторы, влияющие на приоритет:

- Приоритет пользователя/группы/аккаунта/QOS/класса
- Потребление ресурсов пользователем/группой/аккаунтом/qos/классом
- Запрошенное число процессоров/узлов/памяти/swar/диска/времени счета/...
- Время, проведённое задачей в очереди. Количество раз когда backfill

откладывал старт задачи из-за менее приоритетных

- Число секунд прошедшее с момента старта(только для запущенных)
- Примерное время, которая задача должна находиться в очереди

# Maui

- Политики доступа к узлу
  - Одна задача на узел
  - Сколько угодно задач на узел
  - Все задачи на узле – одного пользователя
  - Все задачи на узле – одной группы
  
- Политики доступности узла
  - Узел занят если все его ресурсы заказаны
  - Узел занят если все его ресурсы фактически потреблены
  - Смешанная политика

# Maui

- Политики доступа к узлу
  - Одна задача на узел
  - Сколько угодно задач на узел
  - Все задачи на узле – одного пользователя
  - Все задачи на узле – одной группы
  
- Политики доступности узла
  - Узел занят если все его ресурсы заказаны
  - Узел занят если все его ресурсы фактически потреблены
  - Смешанная политика

# Резервации

## ▪ Отличия от SLURM

- Пользователю не обязательно явно указывать `--reservation`
- Ограничение максимального времени, которое задача может быть в резервации
- Резервация может быть создана пользователем

```
SRCFG[day2] STARTTIME=8:00:00 ENDTIME=19:00:00
```

```
SRCFG[day2] PERIOD=DAY DAYS=MON,TUE,WED,THU,FRI
```

```
SRCFG[day2] TIMELIMIT=1:00:00
```

```
SRCFG[day2] RESOURCES=PROCS:4 TASKCOUNT=10
```

```
SRCFG[day2] FLAGS=SPACEFLEX,PREEMPTEE,BESTEFFORT
```

# Preemption

---

- Очень похоже на Gang scheduling в SLURM

PREEMTPOLICY REQUEUE

QOSCFG[high] QFLAGS=PREEMPTOR

QOSCFG[med]

QOSCFG[low] QFLAGS=PREEMPTEE



# Backfill vs Maui

## Backfill

Активно развивается

Базовые настройки

Интегрирован в SLURM

Использует плагины для выделения узлов и подсчета приоритета

Используются утилиты SLURM

## Maui

В основном исправляются баги

Множество настроек

Не доверяет SLURM'у

Своя система выделения узлов и подсчета приоритета

Используется свой комплект утилит, ведется своя статистика, можно использовать утилиты SLURM

Вопросы?