

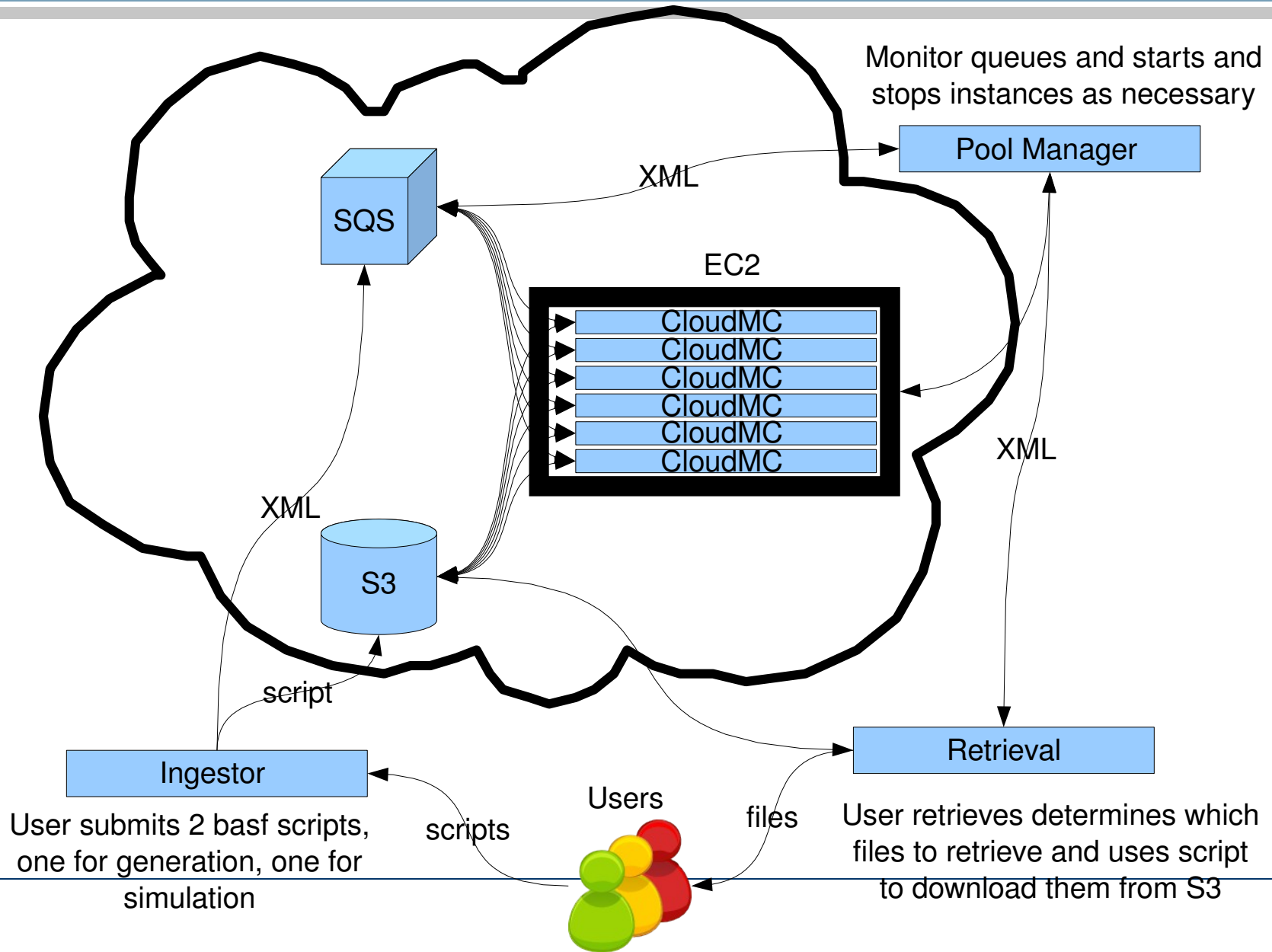


Tom Fifield
fifieldt@unimelb.edu.au

Production MC events on the Amazon EC2 Cloud

- Have Belle MC Analysis working on EC2
- “cloud computing ... will provide extra resources on top of the baseline (grid) resources in case we are in urgent need of CPU power.”
- Automating it was the next step.
- Followed by a production MC run

- Automate the use of Amazon Elastic Compute Cloud to do Belle MC Analysis
- Provide a queuing system, where users can submit Belle MC Analysis jobs to be run on the cloud
- Allow greatly simplified security configuration for access to postgres database from the cloud



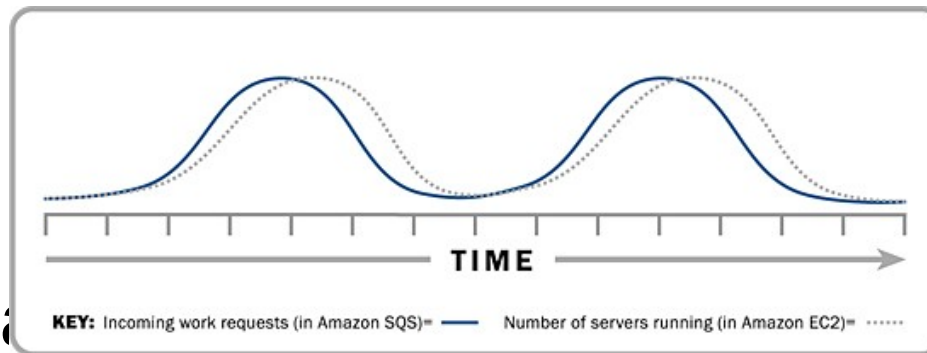
- One-line submit:
 - `./cloud_sub generation.scr simulation.scr`
`[project] [batch#] [#jobs]`
 - Combines the scripts with a template header to make cloud-runnable code
- Simple retrieve
 - `./retrieve -p project [file1 file2...]`
- Can be made prettier



- Constantly monitors the queue
- Starts and stops AMIs as necessary
- Deals with non-responsive AMIs
- Tracks job status

- Upon starting, AMI:
 - downloads the latest version of our automation & analysis software
 - Initiates a Java service, which retrieves the next waiting job in the queue
 - Runs the user-provided code and uploads results to S3
- Uses SSH tunnel to contact postgres server securely – no firewall changes needed

- Takes about 6 minutes to start an AMI
 - (probably the enormous 7Mb download from Australia!)
 - Generally looks like this:



- You can see:
 - eg how many servers you start when a new job comes in
 - how long to leave them idle

- Ingesting

```
xenon lifeguard #  
xenon lifeguard # ./cloud_sub.sh generation.scr simulation.scr TestCloudSub2 001  
Feb 24, 2009 4:21:04 PM com.directthought.lifeguard.ingestor.FileIngestor main  
FINE: using bucket :autocloudmc  
Feb 24, 2009 4:21:04 PM com.directthought.lifeguard.util.QueueUtil getQueueOrElse  
FINE: trying...  
Feb 24, 2009 4:21:07 PM com.directthought.lifeguard.util.QueueUtil getQueueOrElse  
FINE: trying...  
Feb 24, 2009 4:21:11 PM com.directthought.lifeguard.IngestorBase ingest  
FINE: ingested:input.sh [E9275C64EAAE8996E1B0600F1BF18550]
```

- Pool Manager starts a new instance

```
INFO: queue:0 servers:0 load:0 ii:50 bi:0  
Feb 24, 2009 4:21:08 PM com.directthought.lifeguard.PoolManager run  
INFO: queue:0 servers:0 load:0 ii:54 bi:0  
Feb 24, 2009 4:21:12 PM com.directthought.lifeguard.PoolManager run  
INFO: queue:1 servers:0 load:0 ii:59 bi:0  
>>>>>>>>> instance started : i-d44ed8bd  
Feb 24, 2009 4:21:20 PM com.directthought.lifeguard.PoolManager run  
INFO: queue:1 servers:1 load:0 ii:0 bi:0  
Feb 24, 2009 4:21:25 PM com.directthought.lifeguard.PoolManager run  
INFO: queue:1 servers:1 load:0 ii:4 bi:0  
Feb 24, 2009 4:21:29 PM com.directthought.lifeguard.PoolManager run  
INFO: queue:1 servers:1 load:0 ii:9 bi:0
```



- 10 jobs in the queue, 9 Instances start

Reservation ID ▲	Instance ID	AMI	State	Pub...	Privat...	Key	...	R...	...	Type	Local Launch...	Availability Z...	Platf...	民
r-b069d0d9	i-6823b501	ami-06fc1b6f	running	ec2-1...	domU-12...	fifieldt	default	0		m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-6720b60e	ami-06fc1b6f	pending			fifieldt	default	0		m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-6620b60f	ami-06fc1b6f	pending			fifieldt	default	1		m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7920b610	ami-06fc1b6f	pending			fifieldt	default	2		m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7820b611	ami-06fc1b6f	pending			fifieldt	default	3		m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7b20b612	ami-06fc1b6f	pending			fifieldt	default	4		m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7a20b613	ami-06fc1b6f	pending			fifieldt	default	5		m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7d20b614	ami-06fc1b6f	pending			fifieldt	default	6		m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7c20b615	ami-06fc1b6f	pending			fifieldt	default	7		m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7f20b616	ami-06fc1b6f	pending			fifieldt	default	8		m1.small	2009-02-25 10:4...	us-east-1b		

- 10 jobs in the queue, 10 instances run

Reservation ID ▲	Instance ID	AMI	State	Pub...	Priva...	Key	...	R...	...	Type	Local Launch...	Availability Z...	Platf...	民
r-b069d0d9	i-6823b501	ami-06fc1b6f	running	ec2-17...	domU-1...	fifieldt	default	0		m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-6720b60e	ami-06fc1b6f	running	ec2-72...	domU-1...	fifieldt	default	0		m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-6620b60f	ami-06fc1b6f	running	ec2-17...	domU-1...	fifieldt	default	1		m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7920b610	ami-06fc1b6f	running	ec2-72...	domU-1...	fifieldt	default	2		m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7820b611	ami-06fc1b6f	running	ec2-17...	domU-1...	fifieldt	default	3		m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7b20b612	ami-06fc1b6f	running	ec2-72...	domU-1...	fifieldt	default	4		m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7a20b613	ami-06fc1b6f	running	ec2-17...	domU-1...	fifieldt	default	5		m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7d20b614	ami-06fc1b6f	running	ec2-17...	domU-1...	fifieldt	default	6		m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7c20b615	ami-06fc1b6f	running	ec2-17...	domU-1...	fifieldt	default	7		m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7f20b616	ami-06fc1b6f	running	ec2-75...	domU-1...	fifieldt	default	8		m1.small	2009-02-25 10:4...	us-east-1b		

- Jobs finish, idle timeout expires, shut down

Reservation ID ▲	Instance ID	AMI	State	Pub...	Priva...	Key	...	Re...	...	Type	Local Launch...	Availability Z...	Platf...	民
r-b069d0d9	i-6823b501	ami-06fc1b6f	shuttin...	ec2-17...	domU-1...	fifieldt	default	User in...	0	m1.small	2009-02-25 10:4...	us-east-1b		
r-9468d1fd	i-9c20b6f5	ami-06fc1b6f	shuttin...	ec2-17...	domU-1...	fifieldt	default	User in...	0	m1.small	2009-02-25 10:5...	us-east-1b		
r-3e68d157	i-6620b60f	ami-06fc1b6f	shuttin...	ec2-17...	domU-1...	fifieldt	default	User in...	1	m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7920b610	ami-06fc1b6f	shuttin...	ec2-72...	domU-1...	fifieldt	default	User in...	2	m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7820b611	ami-06fc1b6f	shuttin...	ec2-17...	domU-1...	fifieldt	default	User in...	3	m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7b20b612	ami-06fc1b6f	shuttin...	ec2-72...	domU-1...	fifieldt	default	User in...	4	m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7a20b613	ami-06fc1b6f	shuttin...	ec2-17...	domU-1...	fifieldt	default	User in...	5	m1.small	2009-02-25 10:4...	us-east-1b		

- Using the project name from ingest, list & retrieve files produced

```
xenon lifeguard # ./retrieve.py -p charged-01-03 -l
charged-01-03 302 charged-01-03-302-service.log
charged-01-03 302 charged-01-03-302-service.log
charged-01-03 302 charged-01-03-302-service.log
charged-01-03 335 charged-01-03-335-service.log
charged-01-03 335 charged-01-03-335-service.log
charged-01-03 302 charged-01-03-302-service.log
charged-01-03 397 charged-01-03-397-service.log
charged-01-03 370 charged-01-03-370-service.log
charged-01-03 386 charged-01-03-386-service.log
charged-01-03 347 charged-01-03-347-service.log
charged-01-03 335 charged-01-03-335-service.log
charged-01-03 336 charged-01-03-336-service.log
charged-01-03 302 charged-01-03-302-service.log
charged-01-03 359 charged-01-03-359-service.log
charged-01-03 355 charged-01-03-355-service.log
```

```
xenon lifeguard # ./retrieve.py -p charged-01-03 charged-01-03-365-service.log
```

• Replacing an unresponsive instance

```
>>>>>>>> instance unresponsive : i-6720b60e
>>>>>>>> instance idle : i-7d20b614 load : 77
eb 25, 2009 10:56:20 AM com.directthought.lifeguard.PoolManager run
NFO: queue:4 servers:10 load:69 ii:50 bi:0
>>>>>>>> instance unresponsive : i-6720b60e
>>>>>>>> instance idle : i-7a20b613 load : 96
>>>>>>>> instance idle : i-6823b501 load : 63
eb 25, 2009 10:56:25 AM com.directthought.lifeguard.PoolManager run
NFO: queue:4 servers:10 load:69 ii:55 bi:0
>>>>>>>> instance unresponsive : i-6720b60e
eb 25, 2009 10:56:25 AM com.directthought.lifeguard.PoolManager run
EVERE: Instance i-6720b60e is being replaced
>>>>>>>> instance terminated : i-6720b60e
>>>>>>>> instance started : i-9c20b6f5
```

Reservation ID ▲	Instance ID	AMI		State	Pub...	Priva...	Key	...	R...	...	Type	Local Launch...	Availability Z...	Platf...	⚙
r-b069d0d9	i-6823b501	ami-06fc1b6f		running	ec2-17...	domU-1...	fifieldt	default		0	m1.small	2009-02-25 10:4...	us-east-1b		
r-9468d1fd	i-9c20b6f5	ami-06fc1b6f		pending			fifieldt	default		0	m1.small	2009-02-25 10:5...	us-east-1b		
r-3e68d157	i-6620b60f	ami-06fc1b6f		running	ec2-17...	domU-1...	fifieldt	default		1	m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7920b610	ami-06fc1b6f		running	ec2-72...	domU-1...	fifieldt	default		2	m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7820b611	ami-06fc1b6f		running	ec2-17...	domU-1...	fifieldt	default		3	m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7b20b612	ami-06fc1b6f		running	ec2-72...	domU-1...	fifieldt	default		4	m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7a20b613	ami-06fc1b6f		running	ec2-17...	domU-1...	fifieldt	default		5	m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7d20b614	ami-06fc1b6f		running	ec2-17...	domU-1...	fifieldt	default		6	m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7c20b615	ami-06fc1b6f		running	ec2-17...	domU-1...	fifieldt	default		7	m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-7f20b616	ami-06fc1b6f		running	ec2-75...	domU-1...	fifieldt	default		8	m1.small	2009-02-25 10:4...	us-east-1b		
r-3e68d157	i-6720b60e	ami-06fc1b6f		termina...			fifieldt	default	User ...	0	m1.small	2009-02-25 10:4...	us-east-1b		

- Amazon Simple Queue Service is cheap.
 - \$0.01 per 10,000 Amazon SQS Requests (\$0.000001 per Request)
 - If you run server polling queue at 5sec interval, that's **\$6.30/year**
 - Standard AMI charges

Machine	cost/ 10^4 events	cost/ 10^9 events
Small EC2 Instance	\$2.065	\$206,541.575
Large EC2 Instance	\$1.175	\$117,504.489
Extra Large EC2 Instance	\$1.176	\$117,637.111
HighCPU Med EC2 Instance	\$1.029	\$102,913.583
HighCPU XL EC2 Instance	\$0.475	\$47,548.933



- We can modify any way the service acts
 - e.g. Upload to SRB instead of S3
 - Web-page job status query
 - Hook-in to other automated submission systems

- We need to use addbg
 - Store the addbg data on S3
 - Can then use S3FS (fuse module for S3) to retrieve it as if it were local
 - Cost – depends on how much you need to store
 - Exp 61, Run-Range 3 was 3.1Gb
 - @ \$0.150 per GB per month: \$0.465/month
 - @ \$0.100 per GB transfer cost inbound: \$0.31



- Exp 61 Charged RunRange 3 Stream 1 has 1000764 events
- Run 20 HighCPU-XL instances (8 cores, 17Gb RAM)
- Retrieve addbg data from S3
- Store results in S3 before transfer to KEK
- A way to look at real cost of cloud



Reservation ID	Instance ID	AMI	State	Public DNS	Private DNS	Key	Type	Local Launch Time	Tag	🚩
r-fd46c194	i-ae71efc7	ami-7735d...	terminated			fifieldt	c1.xlarge	2009-03-15 14:22:32		
r-fd46c194	i-a171efc8	ami-7735d...	terminated			fifieldt	c1.xlarge	2009-03-15 14:22:32		
r-e940c780	i-1169f778	ami-7735d...	terminated			fifieldt	c1.xlarge	2009-03-15 16:16:03		
r-aa40c7c3	i-d669f7bf	ami-7735d...	running	ec2-174-129-116-137.compu...	ip-10-250-163-223.ec2.internal	fifieldt	c1.xlarge	2009-03-15 16:19:23		
r-aa40c7c3	i-a969f7c0	ami-7735d...	running	ec2-75-101-243-21.compute-...	ip-10-250-43-159.ec2.internal	fifieldt	c1.xlarge	2009-03-15 16:19:23		
r-aa40c7c3	i-a869f7c1	ami-7735d...	running	ec2-75-101-226-223.compute-...	ip-10-250-94-207.ec2.internal	fifieldt	c1.xlarge	2009-03-15 16:19:23		
r-aa40c7c3	i-ab69f7c2	ami-7735d...	running	ec2-67-202-38-36.compute-1...	ip-10-250-222-239.ec2.internal	fifieldt	c1.xlarge	2009-03-15 16:19:23		
r-aa40c7c3	i-aa69f7c3	ami-7735d...	running	ec2-174-129-114-89.comput...	ip-10-251-63-239.ec2.internal	fifieldt	c1.xlarge	2009-03-15 16:19:23		
r-aa40c7c3	i-ac69f7c5	ami-7735d...	running	ec2-72-44-33-190.compute-1...	ip-10-250-230-255.ec2.internal	fifieldt	c1.xlarge	2009-03-15 16:19:23		
r-aa40c7c3	i-af69f7c6	ami-7735d...	running	ec2-174-129-153-242.compu...	ip-10-250-71-63.ec2.internal	fifieldt	c1.xlarge	2009-03-15 16:19:23		
r-aa40c7c3	i-ae69f7c7	ami-7735d...	running	ec2-75-101-205-44.compute-...	ip-10-250-198-239.ec2.internal	fifieldt	c1.xlarge	2009-03-15 16:19:23		
r-aa40c7c3	i-a169f7c8	ami-7735d...	running	ec2-67-202-1-207.compute-1...	ip-10-250-146-223.ec2.internal	fifieldt	c1.xlarge	2009-03-15 16:19:23		
r-aa40c7c3	i-ad69f7c4	ami-7735d...	terminated			fifieldt	c1.xlarge	2009-03-15 16:19:23		
r-574fc83e	i-2766f84e	ami-7735d...	running	ec2-174-129-156-115.compu...	ip-10-250-211-79.ec2.internal	fifieldt	c1.xlarge	2009-03-15 16:29:30		
r-164fc87f	i-dc66f8b5	ami-7735d...	pending			fifieldt	c1.xlarge	2009-03-15 16:35:35		
r-164fc87f	i-c566f8ac	ami-7735d...	running	ec2-75-101-216-251.comput...	ip-10-250-210-175.ec2.internal	fifieldt	c1.xlarge	2009-03-15 16:35:35		
r-164fc87f	i-c466f8ad	ami-7735d...	running	ec2-75-101-217-173.comput...	ip-10-250-35-175.ec2.internal	fifieldt	c1.xlarge	2009-03-15 16:35:35		
r-164fc87f	i-c766f8ae	ami-7735d...	running	ec2-72-44-36-91.compute-1...	ip-10-250-34-47.ec2.internal	fifieldt	c1.xlarge	2009-03-15 16:35:35		
r-164fc87f	i-d966f8b0	ami-7735d...	running	ec2-174-129-182-249.compu...	ip-10-250-187-207.ec2.internal	fifieldt	c1.xlarge	2009-03-15 16:35:35		
r-164fc87f	i-d866f8b1	ami-7735d...	running	ec2-75-101-243-99.compute-...	ip-10-250-30-159.ec2.internal	fifieldt	c1.xlarge	2009-03-15 16:35:35		
r-164fc87f	i-db66f8b2	ami-7735d...	running	ec2-75-101-181-136.comput...	ip-10-250-127-143.ec2.internal	fifieldt	c1.xlarge	2009-03-15 16:35:35		
r-164fc87f	i-da66f8b3	ami-7735d...	running	ec2-174-129-178-75.comput...	ip-10-250-63-239.ec2.internal	fifieldt	c1.xlarge	2009-03-15 16:35:35		
r-164fc87f	i-dd66f8b4	ami-7735d...	running	ec2-67-202-16-241.compute-...	ip-10-250-218-95.ec2.internal	fifieldt	c1.xlarge	2009-03-15 16:35:35		
r-164fc87f	i-c666f8af	ami-7735d...	terminated			fifieldt	c1.xlarge	2009-03-15 16:35:35		

- Sorry, we only managed to do 752,233 events in time for the presentation
- CPU cost: \$80
 - 20 Instances, 4 hours 57minutes
- Storage cost: \$0.20
 - Storage on S3: Addbg 3.1Gb, pgen 0.5Gb, results 37Gb, \$6.08/month or \$0.20/day
- Transfer cost: \$6.65
 - Addbg, pgen in: \$0.36, mdst out: \$6.29
- **Total Cost: \$86.85**

- BASF doesn't like S3FS
 - Needed to copy addbg data to a 'local' disk before running job (only done once per server, not per job)
- Takes ~30minutes to get full 20 servers running production due to downloads/copying
 - This is about 10% wasted time
 - Can be minimized if running 24x7 (we were looking to save money!)
- Postgres still has a large overhead



- Run postgres server on the cloud?
 - ~\$876.60/year



THE UNIVERSITY OF

MELBOURNE