Search

Pinterest

English About

Puppet at Pinterest

PuppetConf 2012 • San Francisco

Ryan Park, Pinterest Operations **rpark@pinterest.com**

Animal

25.media.tumblr.com

A puppet with a puppet :-)

by made by moxie

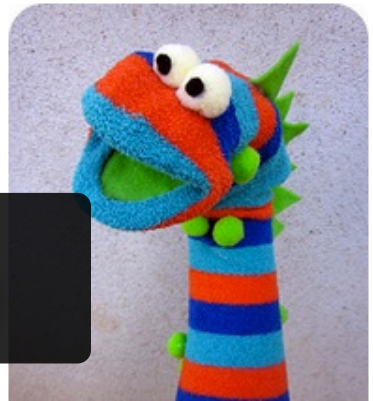
flickr.com

Marionettes

collections.vam.ac.uk

sock puppet!

daniellesplace.com

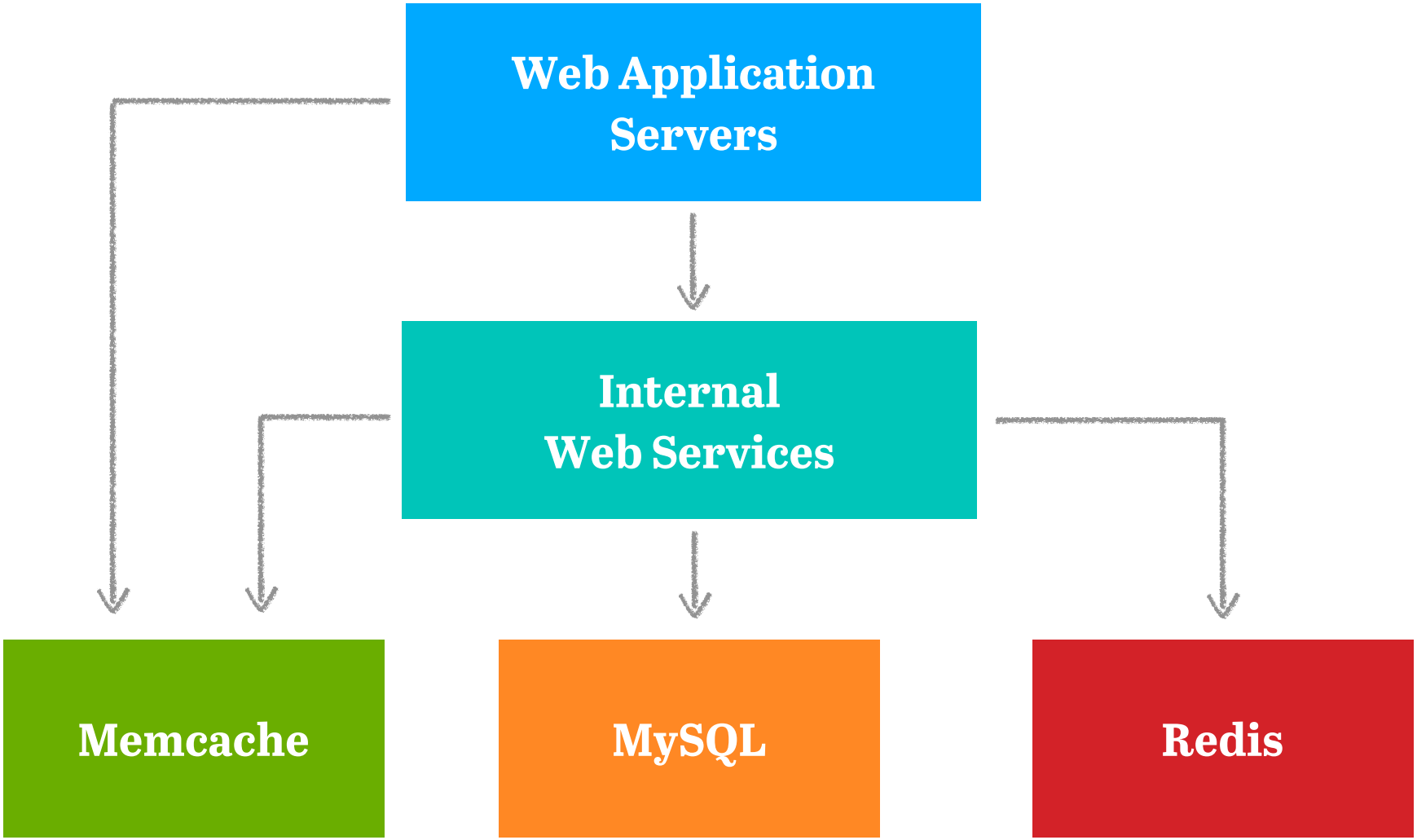


Download slides and code samples at:

https://github.com/pinterest/puppetconf

Ryan Park / rpark@pinterest.com

Ryan Park / rpark@pinterest.com



Ryan Park / rpark@pinterest.com

Before Puppet

- 150 virtual servers: web app, MySQL, Memcache, Membase, Redis, Elastic Search...
- 12 Amazon Machine Images
- `cut -f 1 ~/.ssh/known_hosts`

Ryan Park / rpark@pinterest.com

Puppet Dashboard

- The “source of truth” about what’s running in our infrastructure
- Alternatives we considered
  - Puppet manifests: only useful in Puppet
  - LDAP: difficult to set up
  - Foreman: too much for our needs

Puppet Node Manager

puppet dashboard • 1.2.10 • Home • Nodes • Groups • Classes • Reports • File Search • Enable autorefresh

Background Tasks

All systems go

Nodes

11 Unresponsive
7 Failed
0 Pending
120 Changed
139 Unchanged
8 Unreported
285 All

Add node Radiator View

Group

azkaban	1
azkaban_deploy	0
datalayer	60
datalayer_batch	12
datalayer_canary	1
datalayer_deploy	0
datastore	0
datastore_canary	0
datastore_deploy	0
dev	0

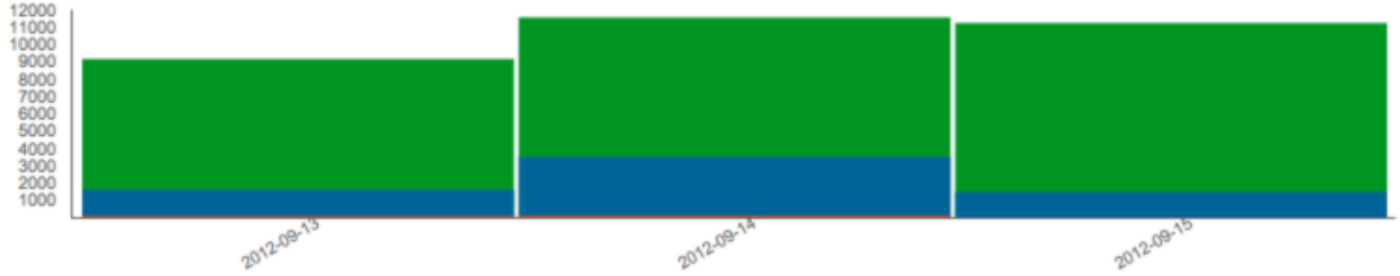
Add group

Class

apache 0

Daily run status

Number and status of runs during the **last 30 days**:



All | Unresponsive | Failed | Pending | Changed | Unchanged

Export nodes as CSV

	Node	↓ Latest report	Resources Total	Failed	Pending	Changed	Unchanged
Total			428832	12	0	239	428591
✓	[illegible]	2012-09-15 22:22 UTC	255	0	0	0	255
✓	[illegible]	2012-09-15 22:22 UTC	364	0	0	0	364
✓	[illegible]	2012-09-15 22:22 UTC	358	0	0	0	358
✓	[illegible]	2012-09-15 22:22 UTC	358	0	0	0	358
✓	[illegible]	2012-09-15 22:22 UTC	257	0	0	0	257
✓	[illegible]	2012-09-15 22:22 UTC	255	0	0	0	255
✓	[illegible]	2012-09-15 22:22 UTC	257	0	0	0	257
✓	[illegible]	2012-09-15 22:22 UTC	318	0	0	0	318
✓	[illegible]	2012-09-15 22:21 UTC	331	0	0	0	331
✓	[illegible]	2012-09-15 22:21 UTC	310	0	0	0	310
✓	[illegible]	2012-09-15 22:21 UTC	358	0	0	0	358
✓	[illegible]	2012-09-15 22:21 UTC	260	0	0	0	260
✓	[illegible]	2012-09-15 22:21 UTC	358	0	0	0	358
✓	[illegible]	2012-09-15 22:21 UTC	358	0	0	0	358
✓	[illegible]	2012-09-15 22:21 UTC	358	0	0	0	358
✓	[illegible]	2012-09-15 22:21 UTC	318	0	0	0	318

Puppet Node Manager

puppet dashboard • 1.2.10 • Home • Nodes • Groups • Classes • Reports • File Search • Enable autorefresh

Background Tasks

2 pending tasks

Nodes

11 Unresponsive
7 Failed
0 Pending
120 Changed
139 Unchanged
8 Unreported
285 All

Add node Radiator View

Group

azkaban	1
azkaban_deploy	0
datalayer	60
datalayer_batch	12
datalayer_canary	1
datalayer_deploy	0
datastore	0
datastore_canary	0
datastore_deploy	0
dev	0

Add group

Class

apache 0

Group: datalayer

Edit Delete

Parameters

Key	Value	Source
swapfile_size	1024	ubuntu
git_checkout_regexp	^deploy_datalayer	datalayer_deploy
supervisor_minfds	65535	datalayer
datalayer_process_count	60	datalayer
authorized_keys_file	/etc/ssh/keys/%u.authorized_keys	prod
ntp_server	[illegible]	prod

Groups

Group	Source
datalayer_deploy	datalayer
ec2	prod
prod	datalayer
ubuntu	prod

Classes

Class	Source
puppet	ec2
users::prod	ubuntu
syslog::node::prod	prod
ec2	ec2
pinterest_code::datalayer	datalayer
ganglia::node	prod
ubuntu	ubuntu
ubuntu::rc_local::prod	prod
dhclient::prod	prod
postfix	ubuntu
swap	ubuntu
ganglia::node::datalayer_process_count	datalayer
process_list::datalayer	datalayer

Derived groups

Group	Source
datalayer_batch	datalayer_batch

Ryan Park / rpark@pinterest.com

Puppet Dashboard

- **Problem:** Some dependencies are configured in Puppet Dashboard, others in Puppet manifests
- **Solution:** Define your dependencies in Puppet manifests when possible

Classes

Class	Source
puppet	ec2
users::prod	ubuntu
syslog::node::prod	prod
ec2	ec2
pinterest_code::datalayer	datalayer
ganglia::node	prod
ubuntu	ubuntu
ubuntu::rc_local::prod	prod
dhclient::prod	prod
postfix	ubuntu
swap	ubuntu
ganglia::node::datalayer_process_count	datalayer
process_list::datalayer	datalayer

Ryan Park / rpark@pinterest.com

Puppet Dashboard

- Node Groups are useful...
- ...but *more* useful when you can use the data to power other systems.
- ...and *even more* useful when you combine Puppet Dashboard data with storedconfigs.

Groups

Name
azkaban
azkaban_deploy
datalayer
datalayer_batch
datalayer_canary
datalayer_deploy
datastore
datastore_canary
datastore_deploy
dev
devapp
dev_sandbox
ec2
featurenags_redis
follower
follower_canary
follower_deploy
follower_redis
followmigration
friend_redis
haproxy
imgmetacache
imgresize
imgsig
kafka
kafka_consumer
ldap
mapred

REST API

```
[ryan@mac:~]$ curl https://puppet-dashboard/api/
{
    "nodes": "https://puppet-dashboard/api/node",
    "node_classes": "https://puppet-dashboard/api/class",
    "node_groups": "https://puppet-dashboard/api/group"
}
```

Self-documenting and nicely formatted

Ryan Park / rpark@pinterest.com

```
[ryan@mac:~]$ curl https://puppet-dashboard/api/group/
[
    {
        "name": "datalayer",
        "url": "https://puppet-dashboard/api/group/datalayer"
    },
    {
        "name": "follower",
        "url": "https://puppet-dashboard/api/group/follower"
    },
    {
        "name": "mysql",
        "url": "https://puppet-dashboard/api/group/mysql"
    },
    ...
]
```

Ryan Park / rpark@pinterest.com

Node Group API

Ryan Park / rpark@pinterest.com

Node Group API

Group: follower_redis

Edit Delete

Parameters

Key	Value	Source
redis_databases	256	follower_redis
redis_appendonly	yes	follower_redis
redis_instances	16	follower_redis

Groups

Group	Source
redis	follower_redis

Classes

Class	Source
redis	redis
redis::backup	follower_redis

Derived groups

— No child groups —

Ryan Park / rpark@pinterest.com

Node Group API

```
[ryan@mac:~]$ curl https://puppet-dashboard/api/group/follower_redis
{
    "nodes": ...,
    "node_classes": ...,
    "parameters": ...,
    "ancestors": ...,
    "descendants": ...
}
```

```
"nodes": [
    {
        "name": "followerredis001a",
        "href": "https://puppet-dashboard/api/node/followerredis001a",
        "source": {
            "type": "node_group",
            "name": "follower_redis",
            "href": "https://puppet-dashboard/api/group/follower_redis"
        }
    },
    {
        "name": "followerredis001b",
        "href": "https://puppet-dashboard/api/node/followerredis001b",
        "source": {
            "type": "node_group",
            "name": "follower_redis",
            "href": "https://puppet-dashboard/api/group/follower_redis"
      }
    },
]
```

```
"node_classes": [
    {
        "name": "redis",
        "href": "https://puppet-dashboard/api/class/redis",
        "source": {
            "type": "node_group",
            "name": "redis",
            "href": "https://puppet-dashboard/api/group/redis"
        }
    },
    {
        "name": "redis::backup",
        "href": "https://puppet-dashboard/api/class/redis::backup",
        "source": {
            "type": "node_group",
            "name": "follower_redis",
            "href": "https://puppet-dashboard/api/group/follower_redis"
        }
    }
]
```

```
"parameters": {
    "swapfile_size": {
        "key": "swapfile_size",
        "value": "10240",
        "source": {
            "type": "node_group",
            "name": "follower_redis",
            "href": "https://puppet-dashboard/api/group/follower_redis"
        }
    }
}
```

Ryan Park / rpark@pinterest.com

Node API

Ryan Park / rpark@pinterest.com

Node API

```
[ryan@mac:~]$ curl https://puppet-dashboard/api/node/followerredis001a
{
    "status": "unchanged",
    "node_groups": ...,
    "node_classes": ...,
    "facts": ...,
    "parameters": ...
}
```

```
"facts": {
    "ipaddress": "10.131.60.134",
    "operatingsystem": "Ubuntu",
    "kernelversion": "2.6.38",
    "ec2_instance_id": "i-17500aaf",
    "ec2_instance_type": "m2.2xlarge",
    "ec2_placement_availability_zone": "us-east-1a"
},
"parameters": {
    "swapfile_size": {
        "key": "swapfile_size",
        "value": "10240",
        "source": {
            "type": "node_group",
            "name": "follower_redis",
            "href": "https://puppet-dashboard/api/group/follower_redis"
        }
    }
}
```

Ryan Park / rpark@pinterest.com

Sample API Client

```
[ryan@mac:~]$ cat puppet_to_hosts.py
import json
import urllib2

def download_and_decode(url):
    request = urllib2.Request(url)
    response = urllib2.urlopen(request)
    return json.loads(response.read())

def main():
    data = download_and_decode("http://puppet-dashboard/api/node/")
    for node in data['nodes']:
        if node.has_key('ipaddress') and node['ipaddress']:
            print node['ipaddress'] + " " + node['name']

if __name__ == "__main__":
    main()
```

Ryan Park / rpark@pinterest.com

Sample API Client

```
[ryan@mac:~]$ python puppet_to_hosts.py
10.150.39.222 azkaban001
10.169.164.132 datalayer001
10.39.63.178 datalayer002
10.97.34.202 datalayer003
10.112.144.31 datalayer004
10.49.10.163 followerredis001a
10.18.185.220 followerredis001b
```

Ryan Park / rpark@pinterest.com

Our API Clients

- Generate /etc/hosts file
- Generate Monit configuration files
- Push hostnames to Amazon Route 53 DNS service
- Remove SSL certificates (puppetca --clean) for nodes that have been deleted from Puppet Dashboard

Ryan Park / rpark@pinterest.com

Our API Clients

- Source code deploy tools
- Monitoring dashboards
- Metrics dashboards

Ryan Park / rpark@pinterest.com

Puppet and Amazon EC2

Ryan Park / rpark@pinterest.com

Bootstrapping EC2

- One custom image for all our instances
  - Start with a basic Ubuntu AMI.
  - Add packages facter, puppet, and ec2-api-tools.
  - Modify /etc/rc.local to run Puppet when the instance launches.

Ryan Park / rpark@pinterest.com

We Cheat

- **Problem:** Using Puppet to install all our dependencies is too slow—it would take 20 minutes to launch an instance.
- **Solution:** We pre-install about 60 Debian packages and 60 Python packages.

Ryan Park / rpark@pinterest.com

EC2 Hostnames

- **Problem:** EC2 instance hostnames look like “ip-10-113-111-43.ec2.internal.”
- **Solution:** Set the hostname when booting the instance.

Ryan Park / rpark@pinterest.com

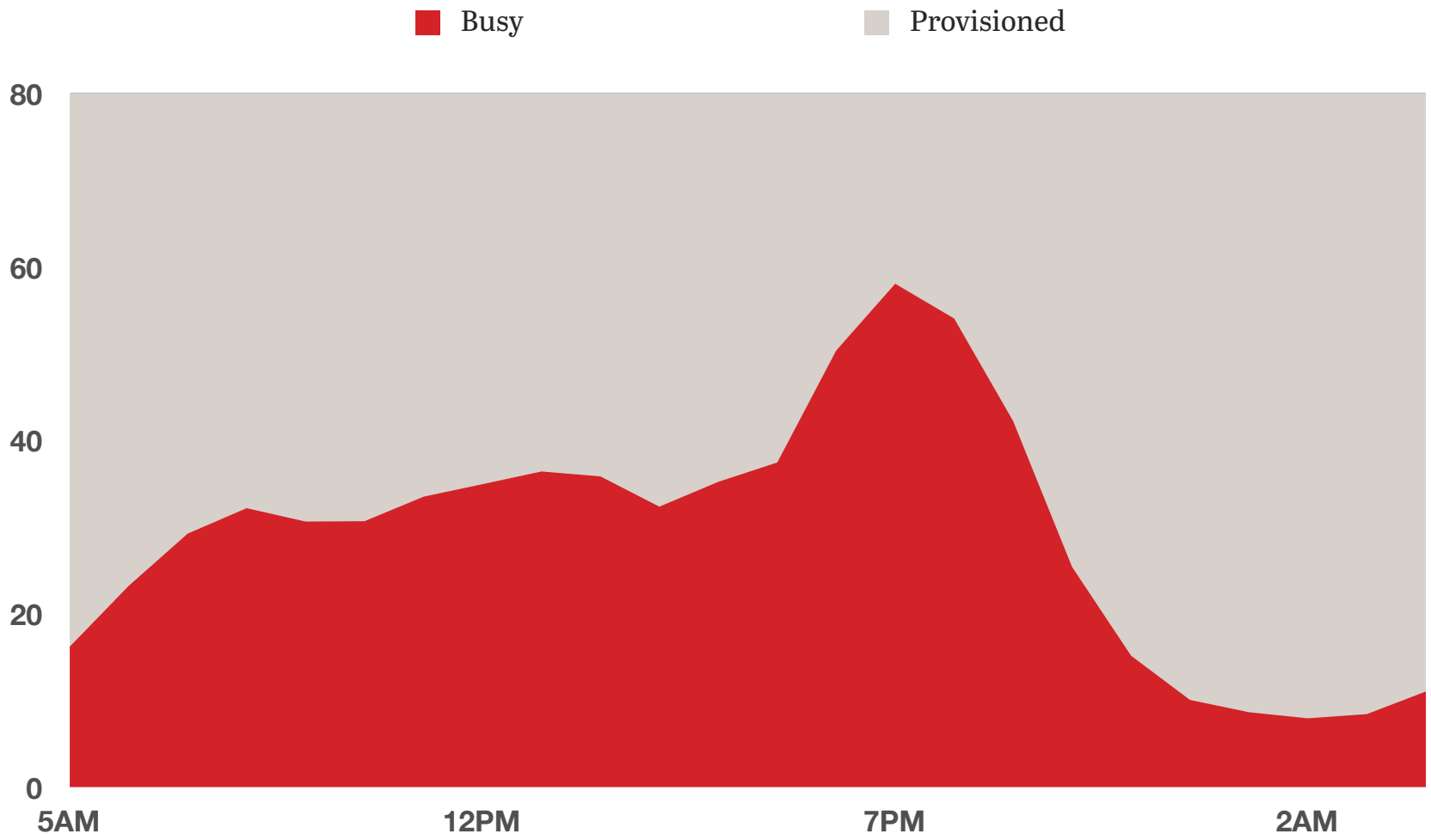
/etc/rc.local

```
[ryan@followerredis001a:~]$ cat /etc/rc.local
#!/bin/bash

# Use ec2-api-tools to determine our instance name.
# /etc/aws/cert.pem and /etc/aws/pk.pem must be present on the AMI,
# along with the Debian packages ec2-api-tools and facter.
export EC2_CERT=/etc/aws/cert.pem
export EC2_PRIVATE_KEY=/etc/aws/pk.pem
INSTANCE_ID=`facter ec2_instance_id`
INSTANCE_NAME=`ec2-describe-tags --filter "key=Name" \
    --filter "resource-type=instance" \
    --filter "resource-id=$INSTANCE_ID" | sed 's/.*\t//g'`
```

```
# Set the hostname to $INSTANCE_NAME.example.com
hostname $INSTANCE_NAME
echo $INSTANCE_NAME > /etc/hostname
sed -i "s/^domain .*$/domain example.com/g" /etc/resolv.conf
sed -i "s/^search .*$/search example.com/g" /etc/resolv.conf
IP_ADDRESS=`facter ipaddress_eth0`
echo "# Additional entries added by bootstrap script" >> /etc/hosts
echo "$IP_ADDRESS $INSTANCE_NAME.example.com $INSTANCE_NAME" \
    >> /etc/hosts

# Puppet will configure this instance based on the classes in the
# Puppet Dashboard.
puppet agent --onetime
```

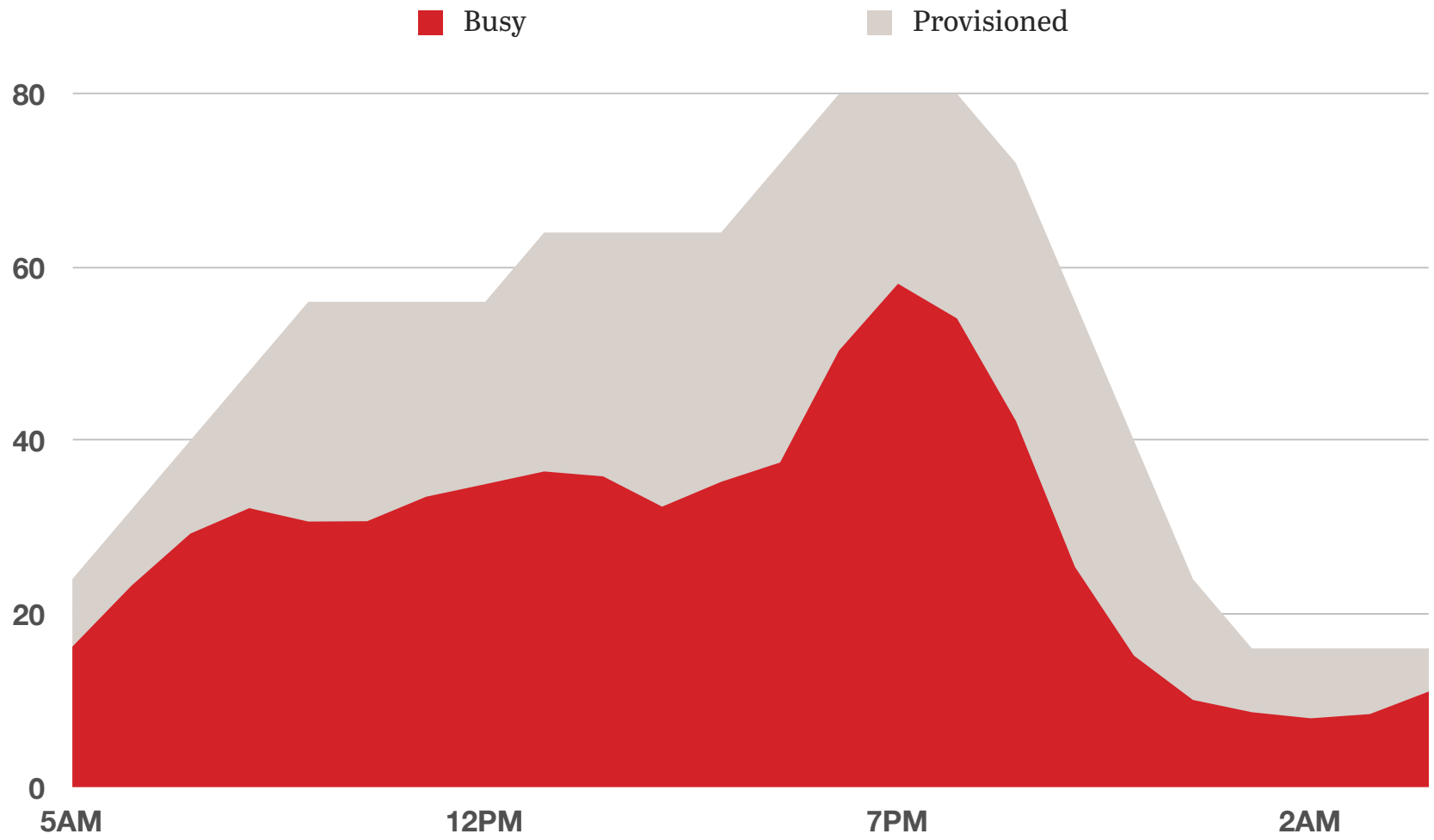
Ryan Park / rpark@pinterest.com

EC2 Auto Scaling



Ryan Park / rpark@pinterest.com

EC2 Auto Scaling



Ryan Park / rpark@pinterest.com

EC2 Auto Scaling

- **Problem:** When using Puppet Dashboard as an external node classifier, every host must be declared explicitly in the Puppet Dashboard database.
- **Solution:** When a new instance starts, have it register itself in the Puppet Dashboard using our REST API.

Ryan Park / rpark@pinterest.com

EC2 Auto Scaling

- A POST to /api/provision/<node_group> adds a node to the Dashboard database and returns the hostname.

```
[root@ip-10-88-155-31:~]# curl -X POST \
    https://puppet-dashboard/api/provision/datalayer

datalayer005
```

- This endpoint returns the hostname as a string, not JSON.

Ryan Park / rpark@pinterest.com

EC2 Auto Scaling: /etc/rc.local

```
# If there's no hostname, there may be a node group name in the
# EC2 user-data string. Use the Puppet Dashboard API to request
# a hostname in that node group.
if [ -z "$INSTANCE_NAME" ]; then
    FILENAME="/var/lib/cloud/instances/$INSTANCE_ID/user-data.txt"
    if [ -f "$FILENAME" ]; then
        NODE_GROUP=`cat $FILENAME`
        if [ ! -z "$NODE_GROUP" ]; then
            INSTANCE_NAME=`curl -X POST \
                https://puppet-dashboard/api/provision/$NODE_GROUP`
        fi
    fi
fi
```

Ryan Park / rpark@pinterest.com

After Puppet

- Hundreds of virtual servers in 60 host groups
- 1 Amazon Machine Image
- Dozens of scripts pull data from Puppet Dashboard's database

Ryan Park / rpark@pinterest.com

Contact

rpark@pinterest.com

ryanpark

@StanfordRyan

Download slides and code samples at:

https://github.com/pinterest/puppetconf