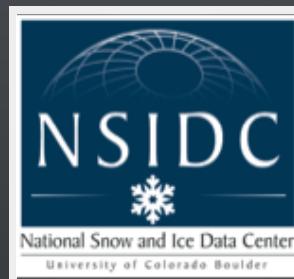
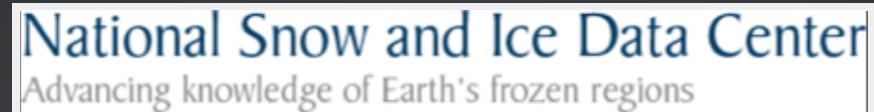


# VIRTUAL GOODNESS WITH VIRTUAL MACHINES

Kevin Beam  
Jeff Braucher  
Matt Savoie  
Hannah Wilcox





- Supports research into Earth's cryosphere
- Manages and distributes scientific data
- Creates tools for data access
- Supports data users
- Performs scientific research
- Performs public education and outreach
- Distributes more than 500 cryospheric data sets

# ABOUT US

- One of Several NSIDC Development Teams
- Dataset Production
- Visualizations
- Dataset Access
- Python
- JavaScript
- A Bit of Ruby
- *We are hiring!*
- *Have we mentioned we're hiring?*

# THE CHALLENGE

**YEAH, WE'RE  
GONNA NEED YOU  
TO CREATE A SITE  
REAL QUICK.**

**THAT'D BE GREAT**

made on imgur

# ARCTIC CHANGE

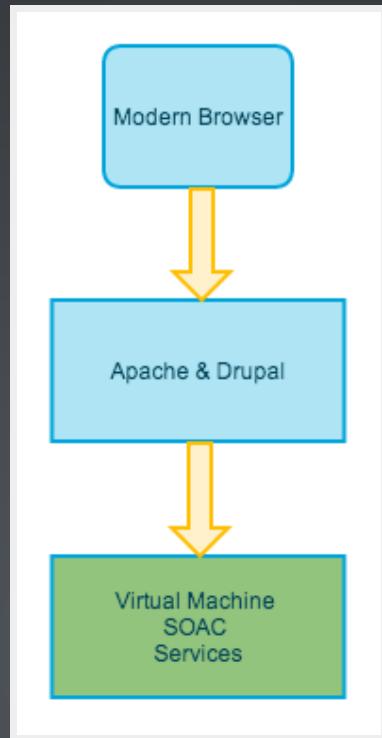
- Showcase Remote Sensing Datasets
- Focus on the Arctic Region
- Long Time-Series Datasets
- Accessible to Non-Specialists

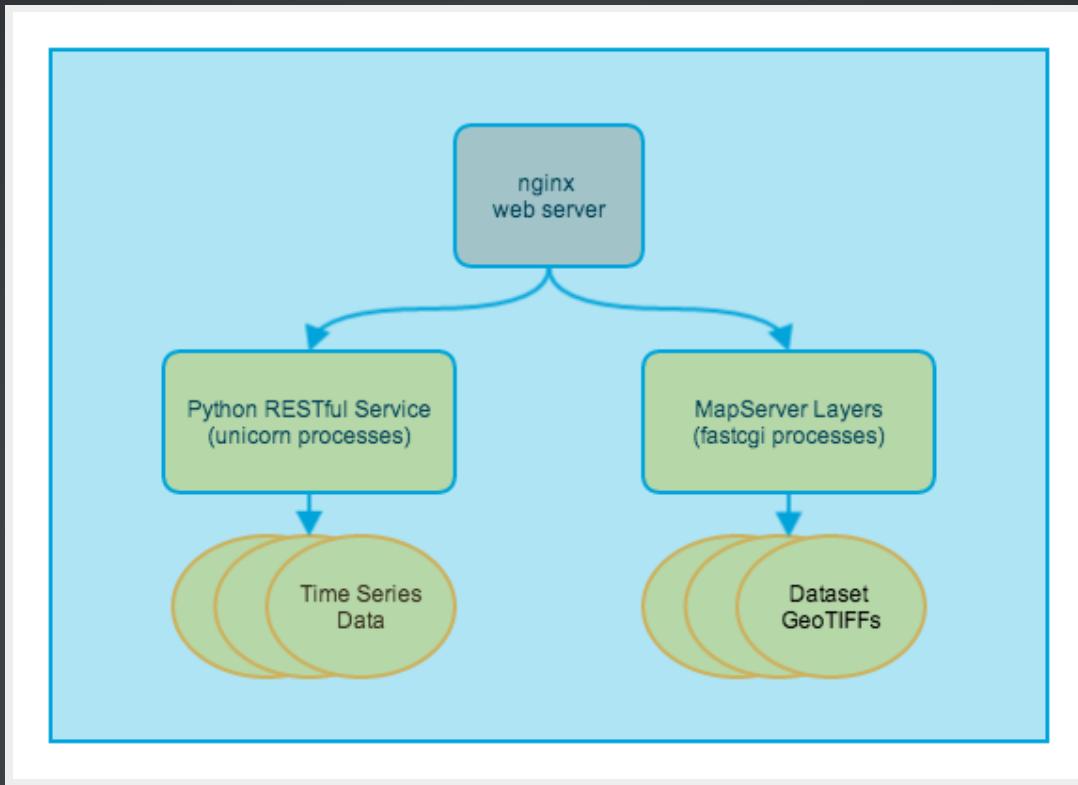
# THE RESULT

Satellite Observations of Arctic Change

**YOUTUBE**

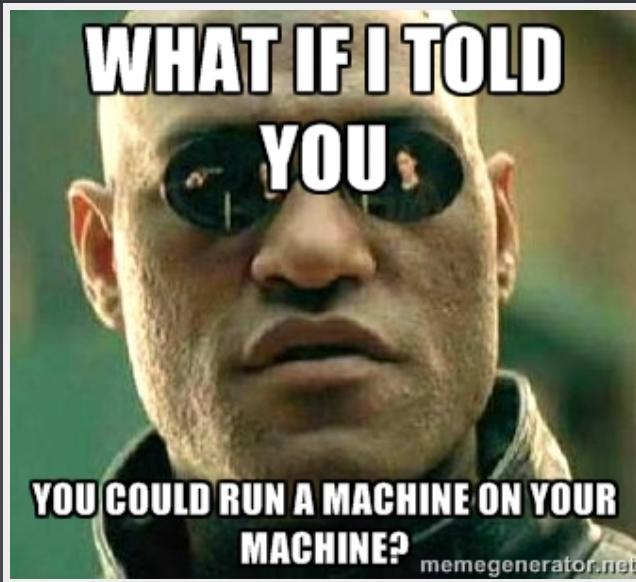






# VIRTUAL MACHINES: OVERVIEW





# ISOLATION OF APPLICATION

- From resource contention
- From server failures
- From maintenance
- ~~From people failures~~

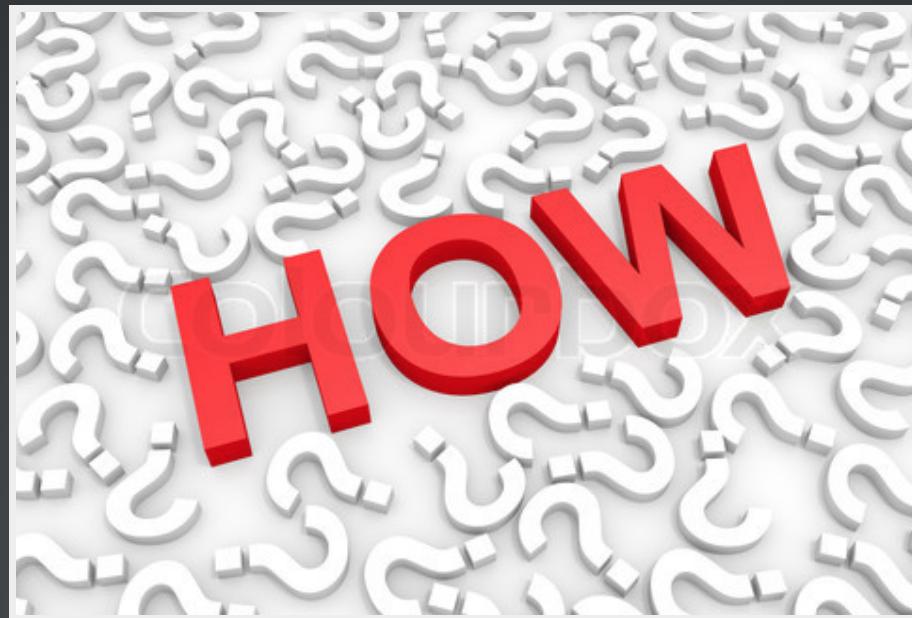
# DEVELOPER CONTROL

- Different version of commonly installed software
- Uncommon/specialty software
- Quick turnaround... as long as devs can figure it out
- Less hassle for sys admins... as long as devs can figure it out

# SAME ENVIRONMENT EVERYWHERE



EVERYWHERE = {LOCAL, INTEGRATION, STAGING, QA, BLUE}



# VAGRANT AND VIRTUALBOX

**VAGRANT USES VIRTUALBOX OUT OF THE BOX**

**VIRTUALBOX PROVIDES VIRTUAL MACHINES**

**VMWARE, AWS, HYPER-V PROVIDE VIRTUAL MACHINES**

**VAGRANT USES THESE AND MORE AS PLUGINS**

# EXAMPLE. LET'S:

- Define a VM
- Create and run a VM
- Login to the VM
- Stop the VM

**YOUTUBE**



# PUPPET



**ONE DOES NOT SIMPLY INSTALL THINGS  
AT THE COMMAND LINE.**

made on imgur

# WHY DO I NEED A PUPPET?

- AUTOMATION!
- Create users, directories, etc
- Install packages (apt-get install)
- Ensure things are running
- Declarative

# BEST OF ALL

*Vagrant can tell Puppet to provision the box after it's been created*

# EXAMPLE. LET'S:

- Install nginx
- Make sure it's running

**YOUTUBE**

# FABRIC

# WHAT'S FABRIC?

*Fabric is a simple, Pythonic tool for remote execution and deployment.*

# WHAT DO WE USE IT FOR?

- Deploy our application to the target machine(s)

```
> fab deploy_application HOSTNAME
```

- Make sure it's restarted / running

```
> fab restart_application HOSTNAME
```

# FABRIC HAS TWO MAIN PARTS:

- Define tasks
- Run remote commands

# EXAMPLE USAGE:

```
@task
def deploy():
    undeploy()

    sudo('mkdir -p /var/www/maps-of-arctic-change')
    sudo('chown -R vagrant:vagrant /var/www/maps-of-arctic-change')

    rsync_project(local_dir=os.path.join(PROJECT_DIR, 'dist/mac'),
                  remote_dir='/var/www/maps-of-arctic-change')

    restart_gunicorn()
```

# OTHER ALTERNATIVES

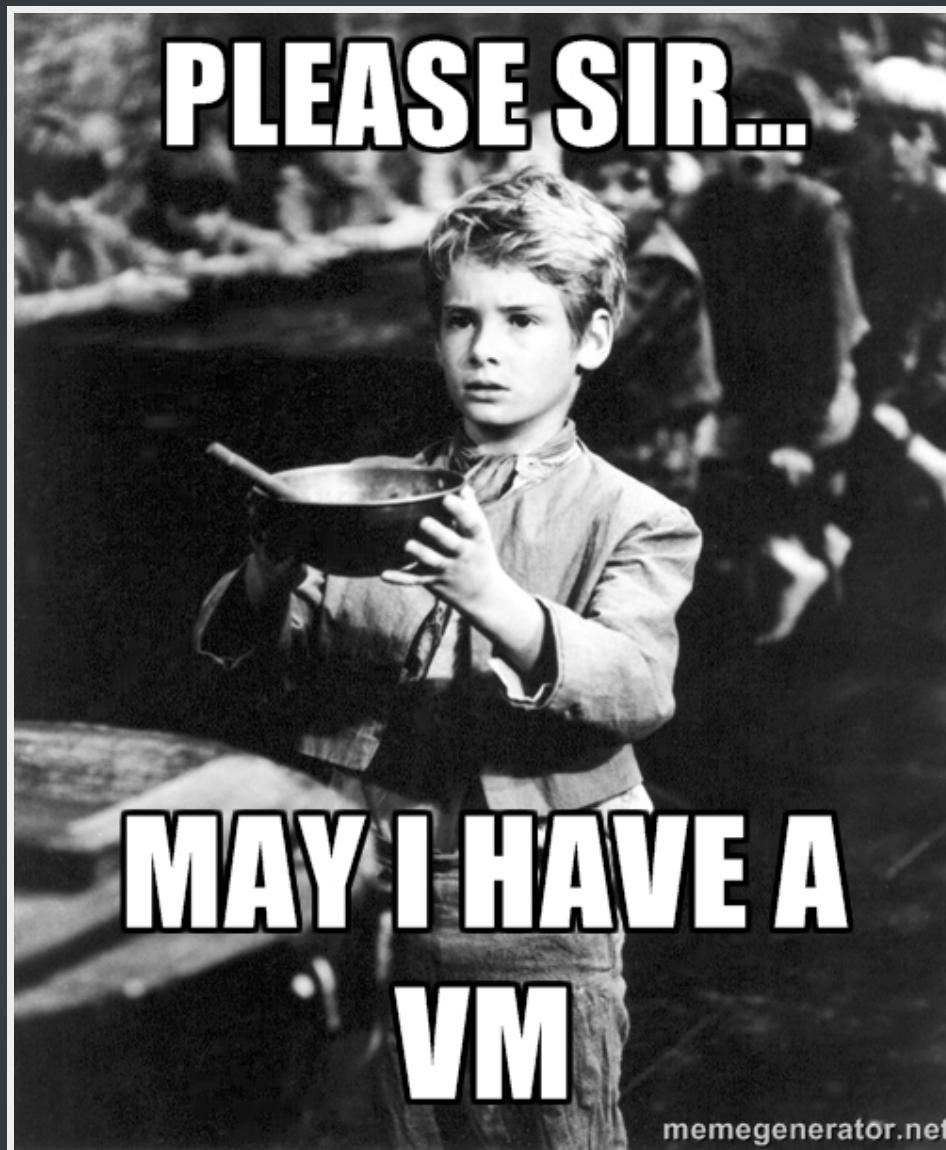
- Puppet!
- Capistrano (Ruby on Rails)
- Language / Platform Specific Tools

# VAGRANT + VSphere

Build your own private cloud.



vSphere is just another vagrant provider



memegenerator.net

# vSphere runs VMs in the datacenter

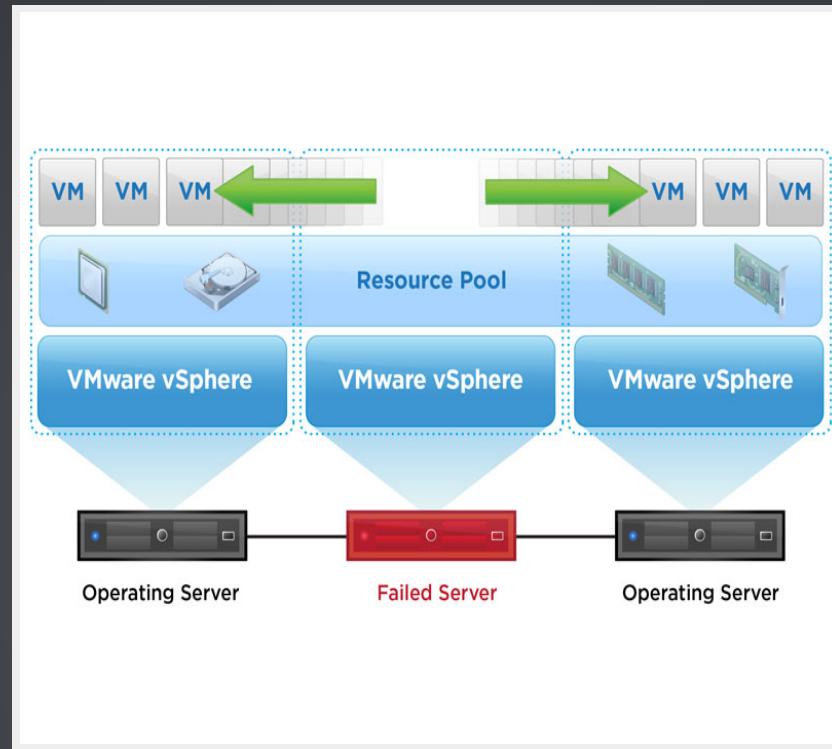


(actual NSIDC datacenter pictured)

# THINGS YOU WILL NEED :

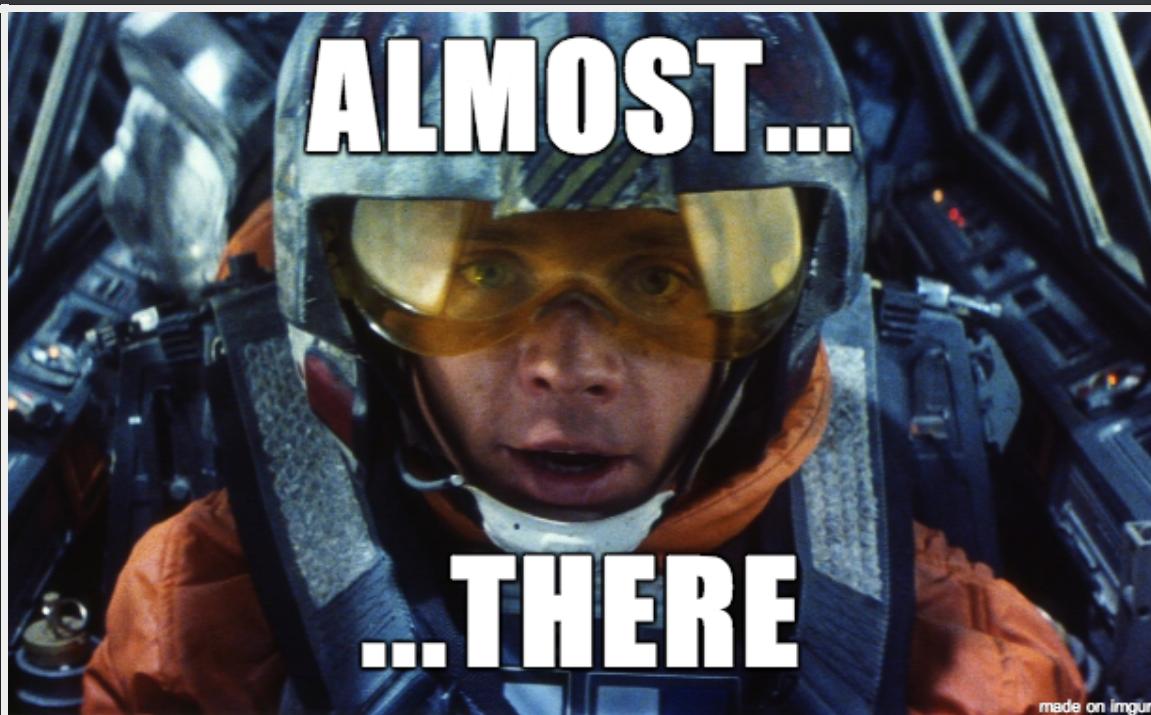
- vagrant-vsphere plugin (available on github)
- server hostname for vSphere API
- username and password for vSphere
- RSA key for vagrant user in vSphere template

# WHAT ELSE DOES VSOPHERE DO?



- provides load balancing and failover
- scales horizontally across physical machines
- provides templates for new VMs
- manages names and IP addresses

ONE LAST THING...

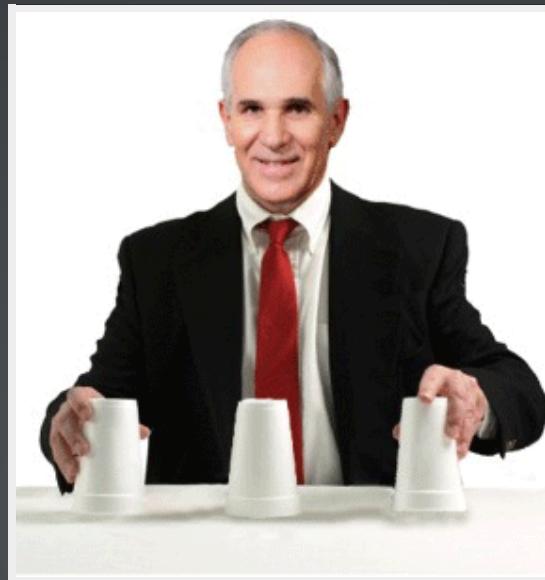


# DYNECT : CLOUD DNS

[apps.nsidc.org](http://apps.nsidc.org)

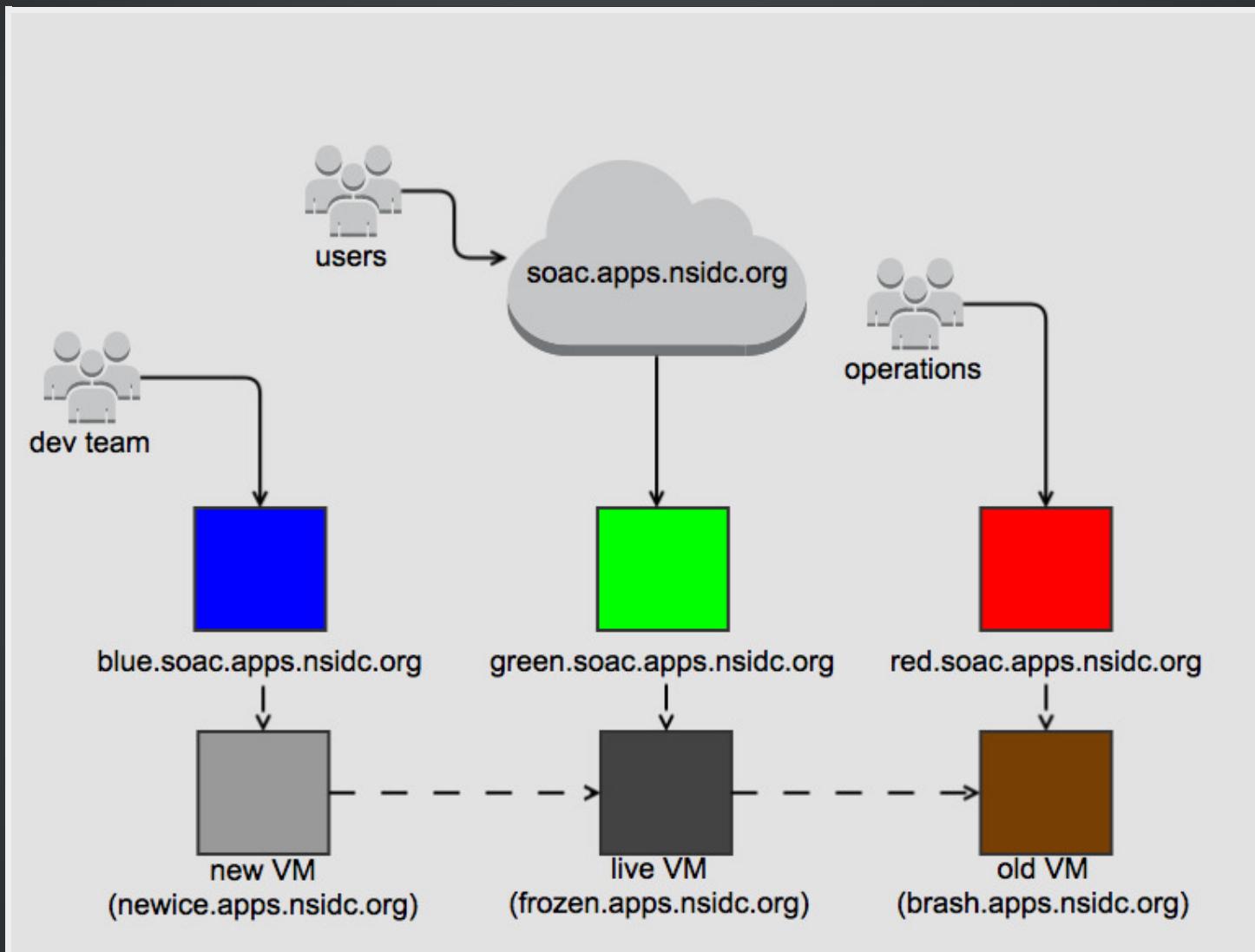
- puts applications in their own sub-domain
- provides a rich DNS management API
- can be integrated with vSphere
- allows teams to create DNS aliases
- enables blue / green deployment

# BLUE / GREEN DEPLOYMENT

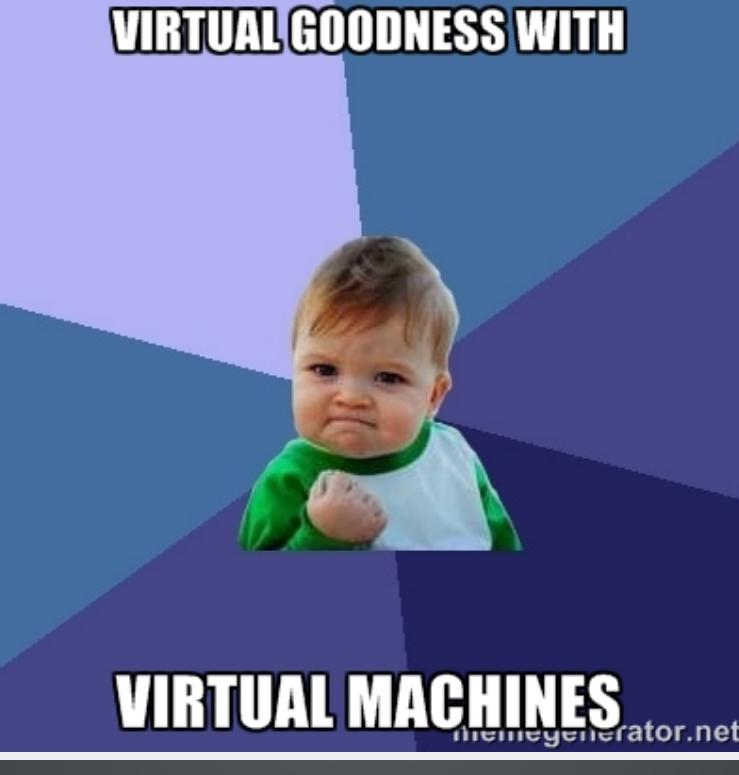


Follow the VM!

# ACTUALLY, WE DO IT LIKE THIS:



**VIRTUAL GOODNESS WITH**



**VIRTUAL MACHINES**

memegenerator.net

# Q & A