

# Intro to Docker

PYCON ■ 2014

@amjithr

Hi



@amjithr



# Why?

# Why?



@amjithr

# Why?



# Why?



# Why?





# Why?



# Why?



**Celery**



# Why?



WHAT'S

Docker?

WHAT'S

Docker?

Not Pants

# History

DotCloud -> Docker Inc

Jan 2013

Written in Go

Active community

“Docker is an open-source engine that automates the deployment of any application as a lightweight, portable, self-sufficient container that will run virtually anywhere\*.”

# What?

CHROOT  
BSD JAILS  
SOLARIS ZONES



# What?

Linux Containers

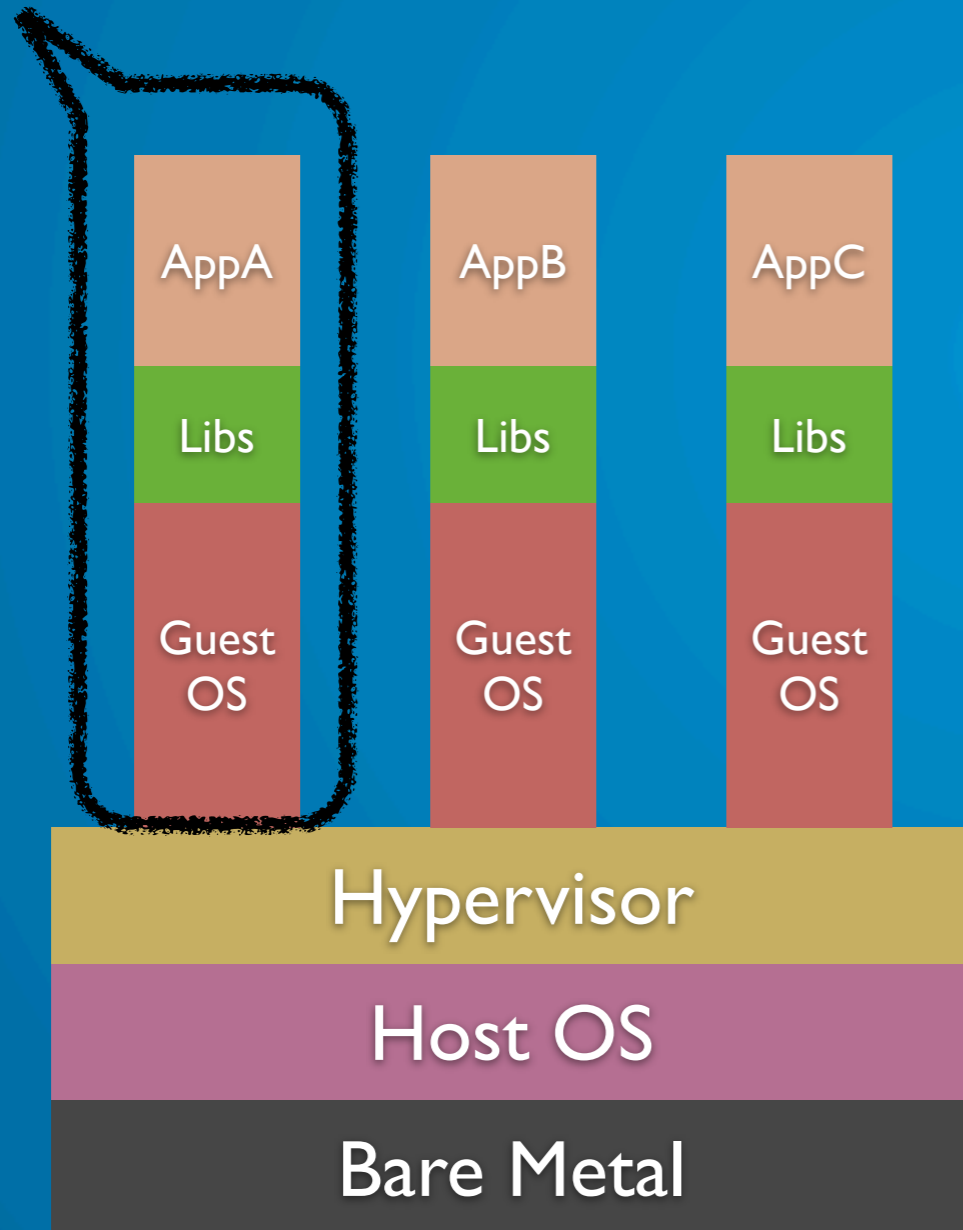
AUFS

Git like versioning

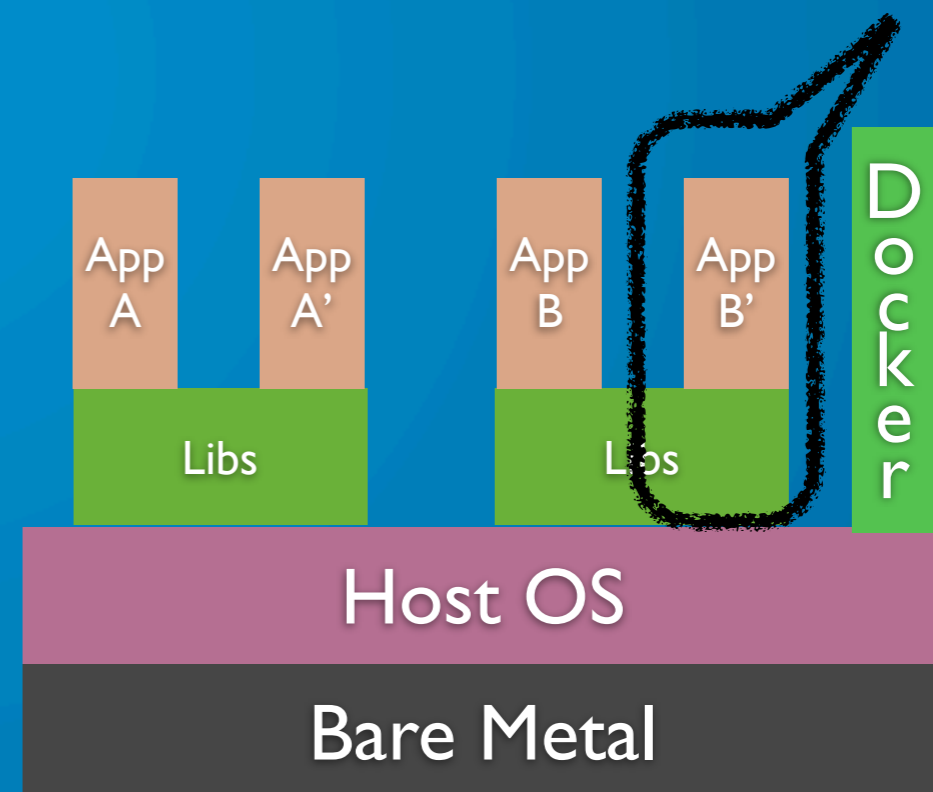
REST API

# What?

VM



Container



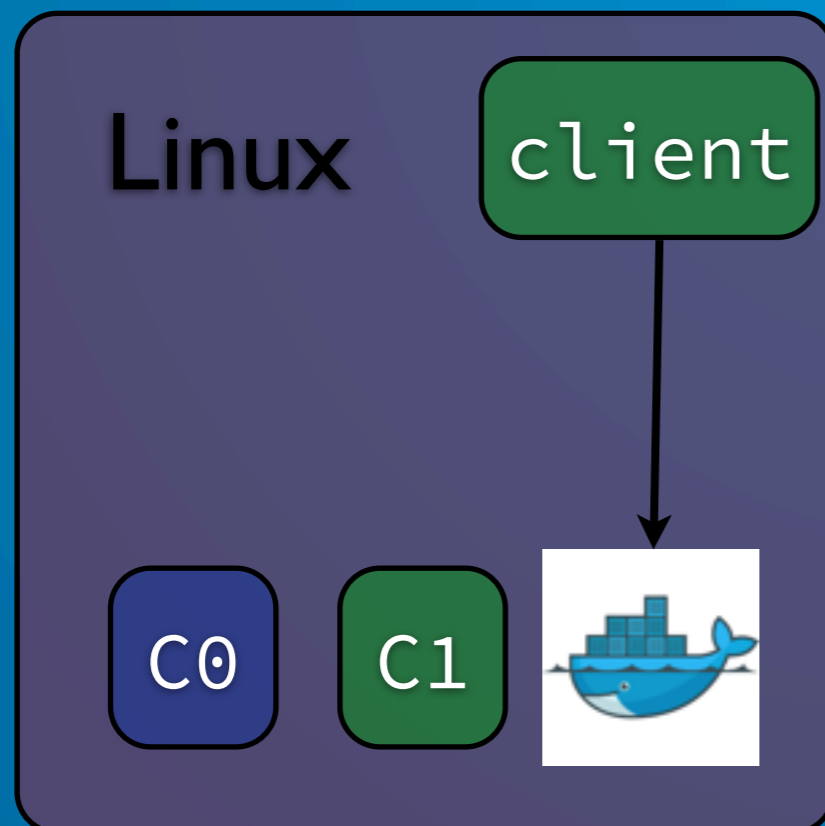
# Why?

-  Lightweight
-  Isolated Instances
-  Faster than VMs

# Setup

[docs.docker.io](https://docs.docker.io) - Installation instructions

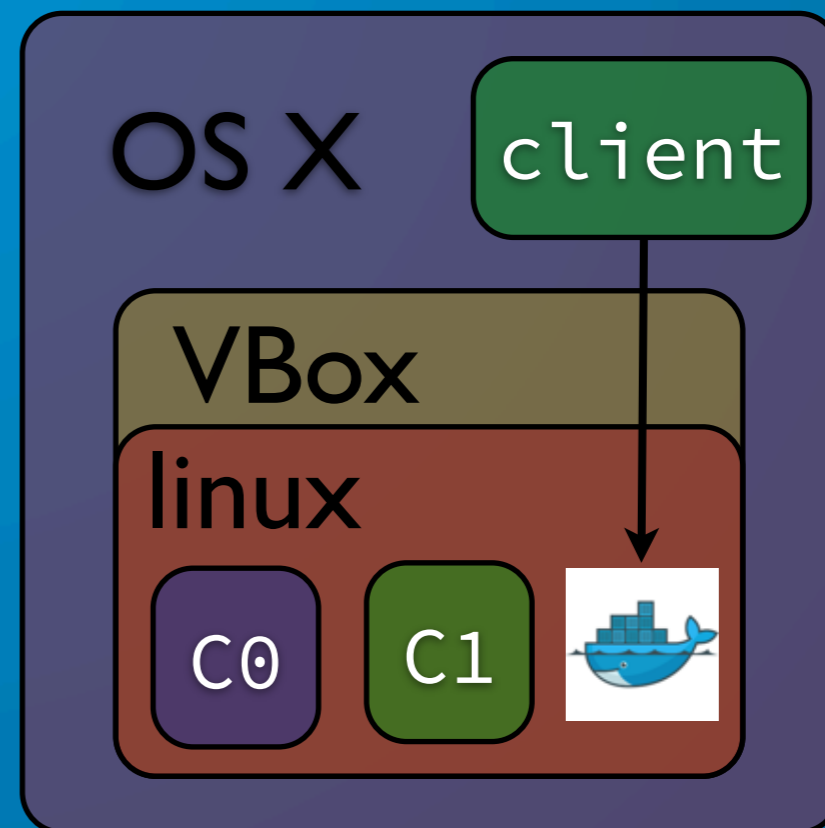
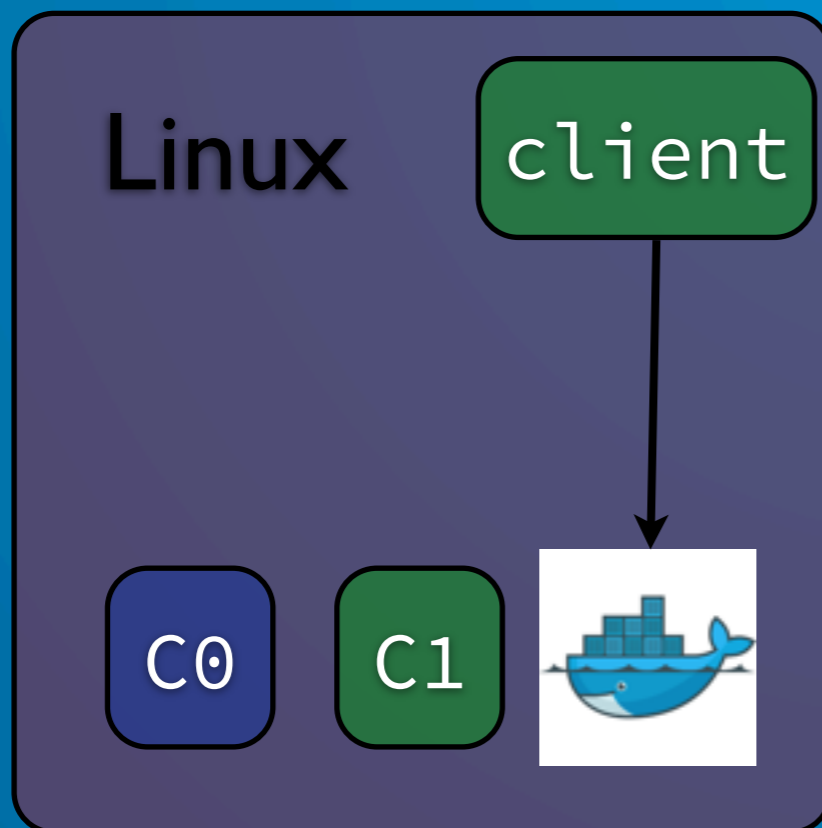
OS X - boot2docker (VM) + docker client



# Setup

[docs.docker.io](https://docs.docker.io) - Installation instructions

OS X - boot2docker (VM) + docker client

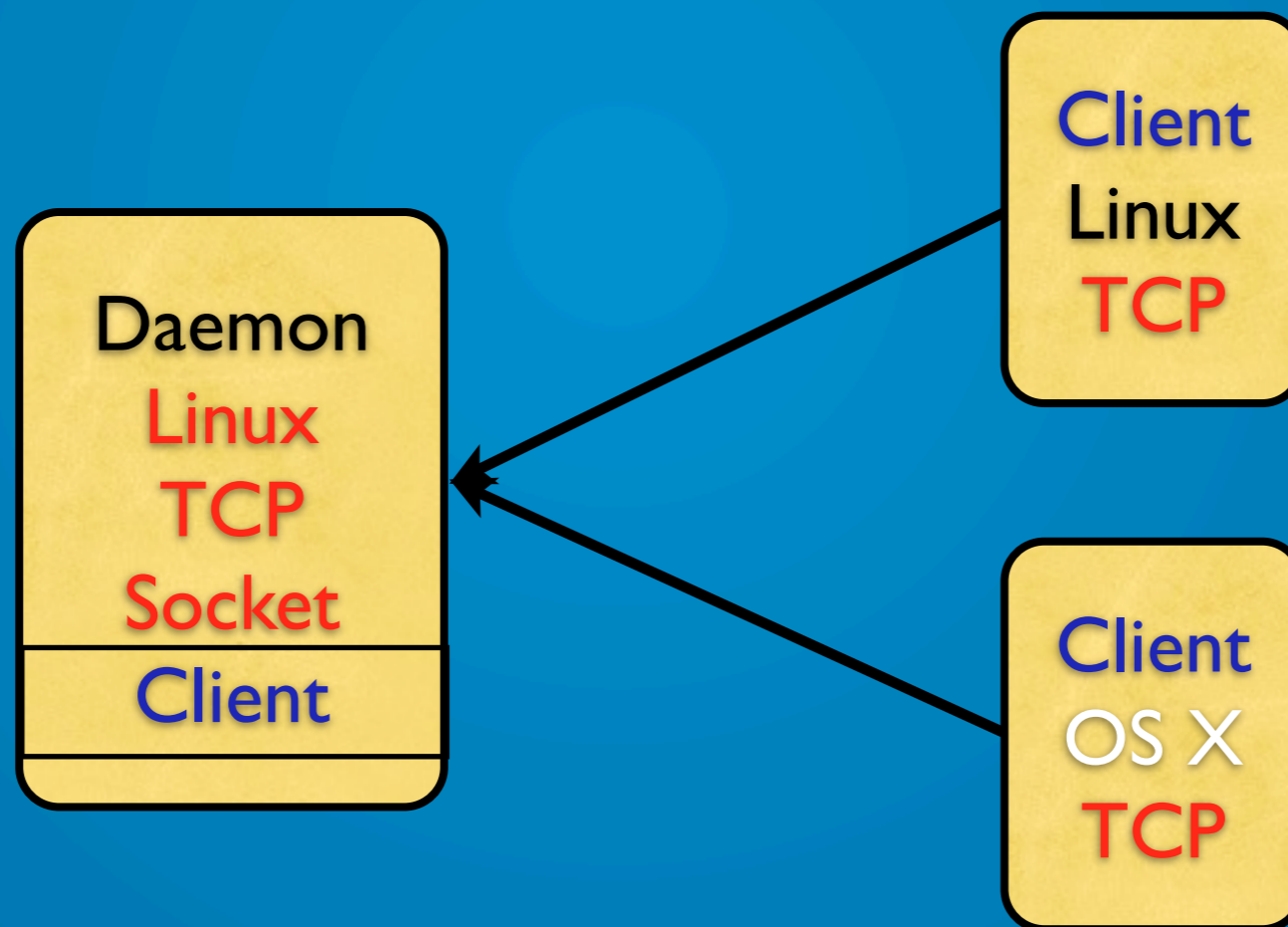


# Terminology

Docker Daemon - Server

Docker Client - Client

# Terminology



# DEMO



@amjithr



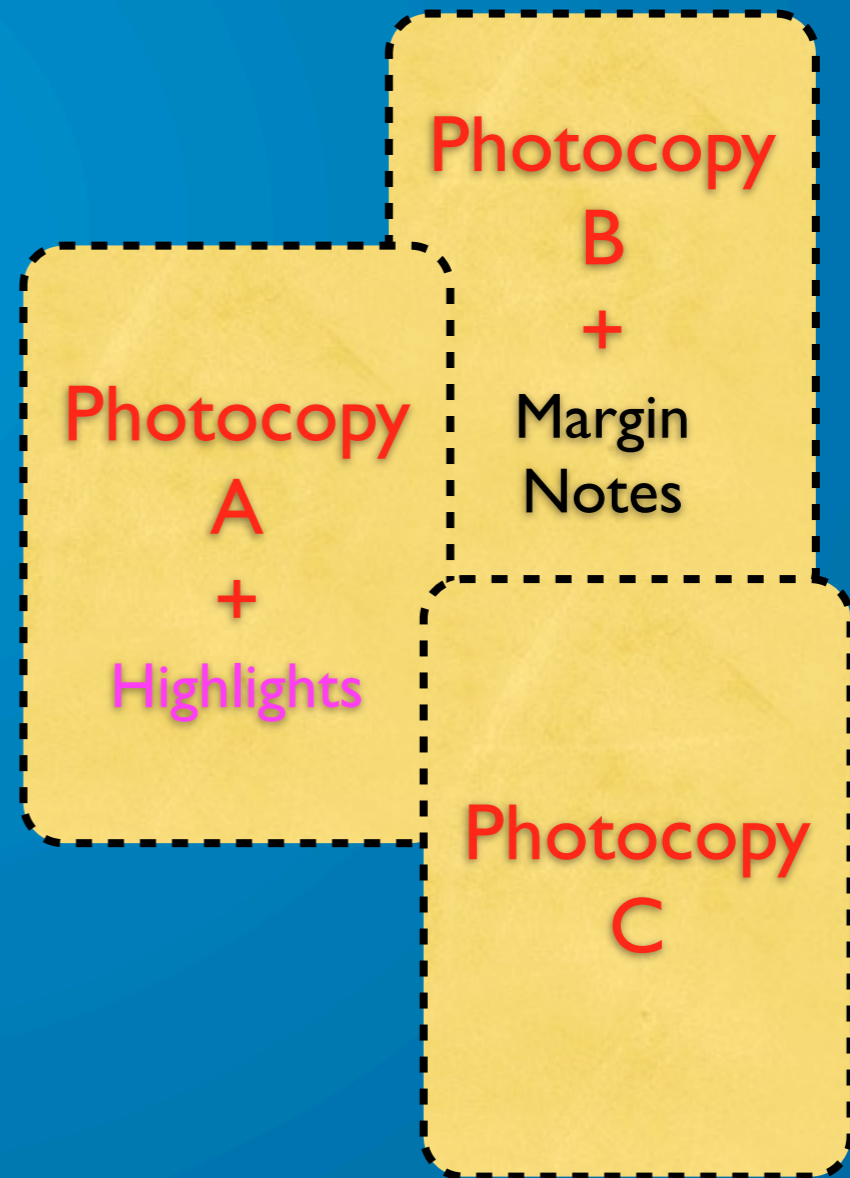
# Terminology

**Images** - Read-only snapshot

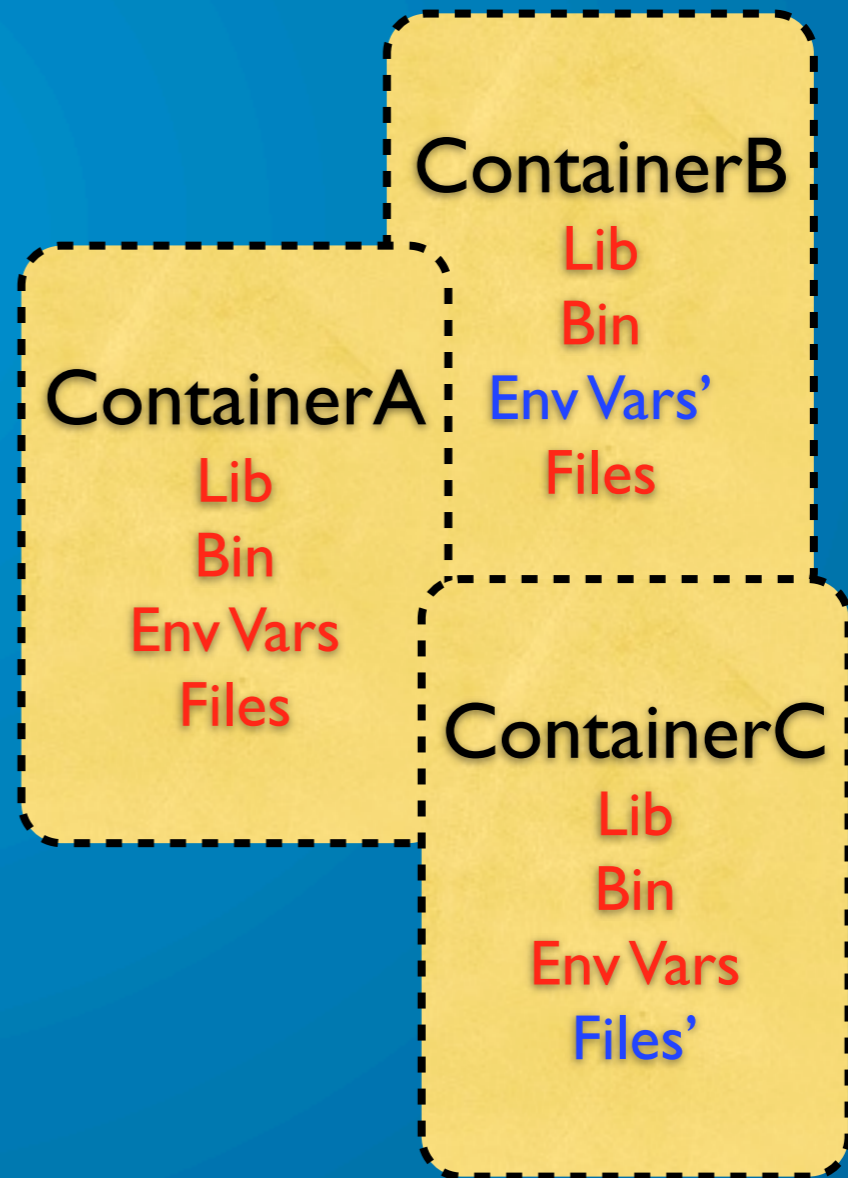
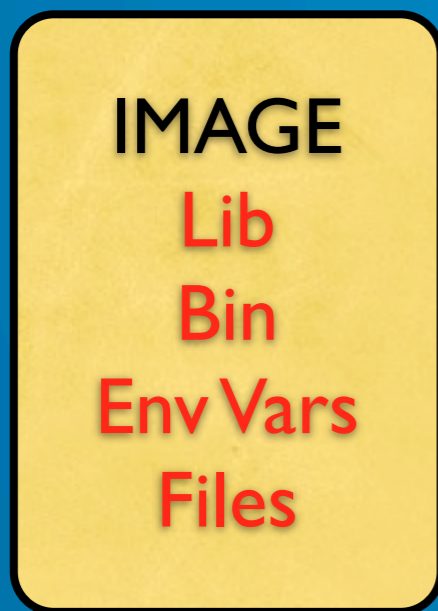
**Containers** - Instantiation of an Image

# Terminology

Master  
Copy



# Terminology



# Terminology

**Registry** - PyPI for docker images

**Repository** - Projects in the Registry

# Basics

```
docker pull <image-name>
```

```
docker images
```

```
docker run <image-name> <command>
```

```
docker ps
```

```
docker logs
```

# Automation

Dockerfile - Series of commands

# Dockerfile

```
FROM base
ENV TEST_ENV true
ADD test_file /
RUN apt-get update
RUN apt-get install -y fortune cowsay
CMD fortune | cowsay
```

# Network

Django

Port Forwarding

```
docker run -d django-docker
```



# Network

Django

Port Forwarding

```
docker run -d -p host:container django-docker
```

# Misc

**Volumes** - Mount folders - host/container

```
docker run -v host_path:container_path django
```

**Links** - Service discovery through env vars

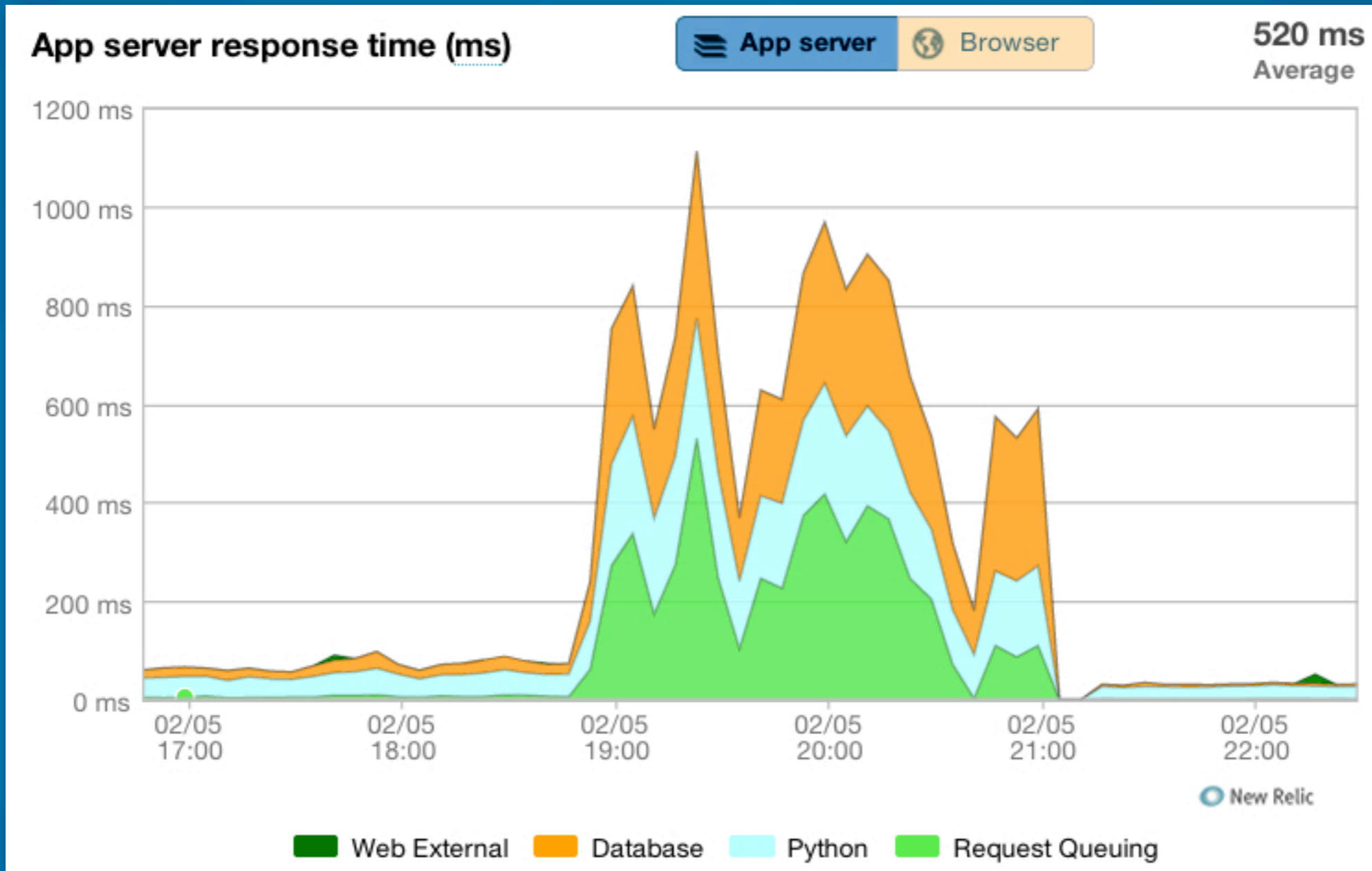
```
docker run --link mysql:db --name webapp django
```

# REST API

Docker daemon is also a server  
REST API

- List of images/containers
- Create/Start/Stop/Delete containers
- Websocket attach
- Shipyard





# PERFORMANCE!

# New Relic

adapter\_cherrypy.py  
adapter\_flup.py  
adapter\_gevent.py  
adapter\_gunicorn.py  
adapter\_meinheld.py  
adapter\_paste.py  
adapter\_waitress.py  
adapter\_wsgiref.py  
application\_celery.py  
application\_gearman.py  
component\_cornice.py  
component\_piston.py  
component\_tastypie.py  
coroutines\_gevent.py  
database\_dbapi2.py  
database\_mysql.py  
database\_mysqlldb.py

database\_psycopg2.py  
database\_sqlite.py  
datastore\_pymongo.py  
external\_dropbox.py  
external\_facepy.py  
external\_feedparser.py  
external\_httplib.py  
external\_httplib2.py  
external\_pywapi.py  
external\_requests.py  
external\_thrift.py  
external\_urllib.py  
external\_urllib2.py  
external\_urllib3.py  
external\_xmlrpclib.py  
framework\_bottle.py  
framework\_cherrypy.py

framework\_django.py  
framework\_flask.py  
framework\_pylons.py  
framework\_pyramid.py  
framework\_tornado.py  
framework\_twisted.py  
framework\_web2py.py  
framework\_webpy.py  
memcache\_memcache.py  
memcache\_pylibmc.py  
memcache\_umemcache.py  
middleware\_weberror.py  
solr\_pysolr.py  
solr\_solrpy.py  
template\_genshi.py  
template\_jinja2.py  
template\_mako.py

# New Relic

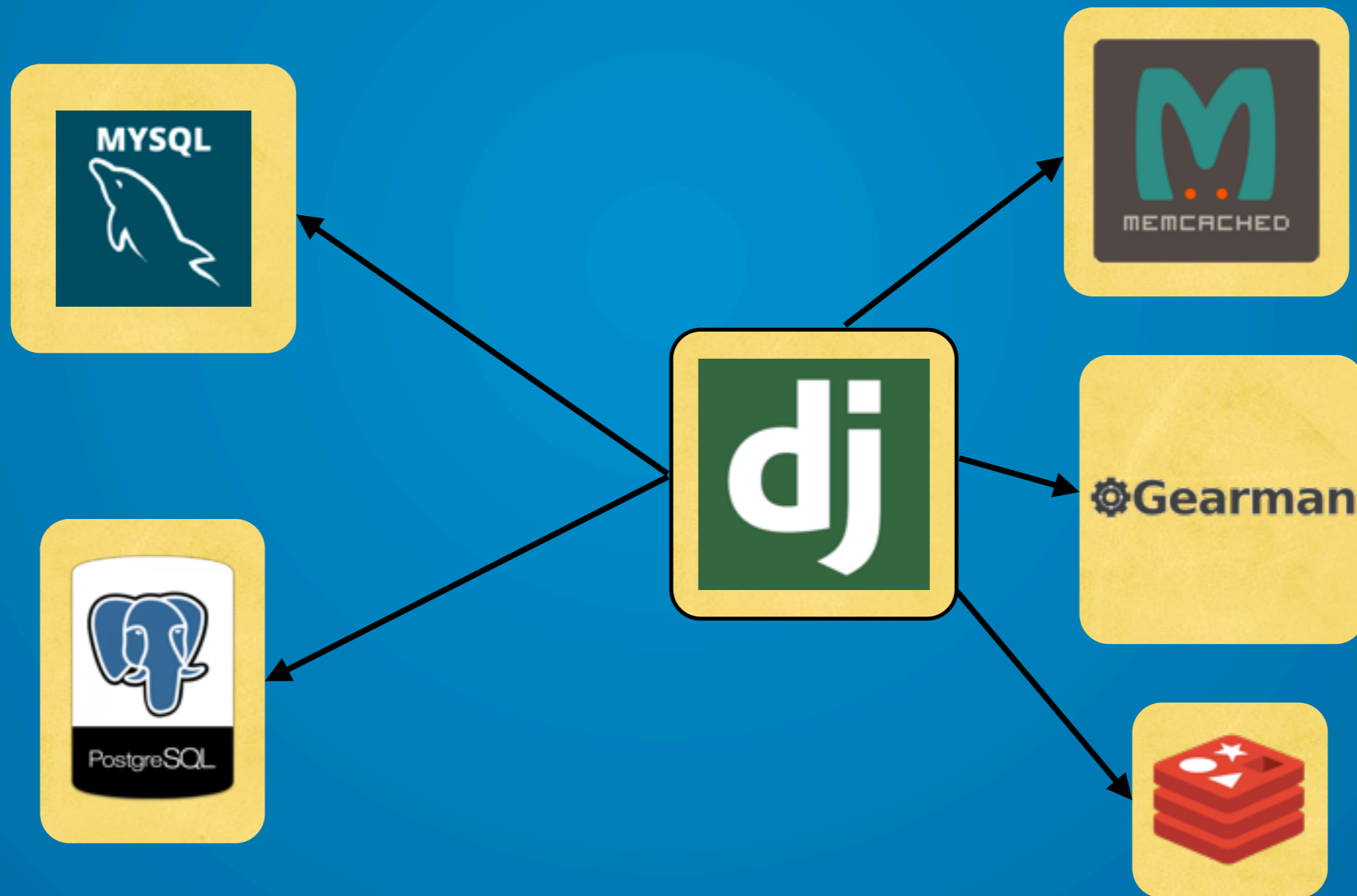


# New Relic

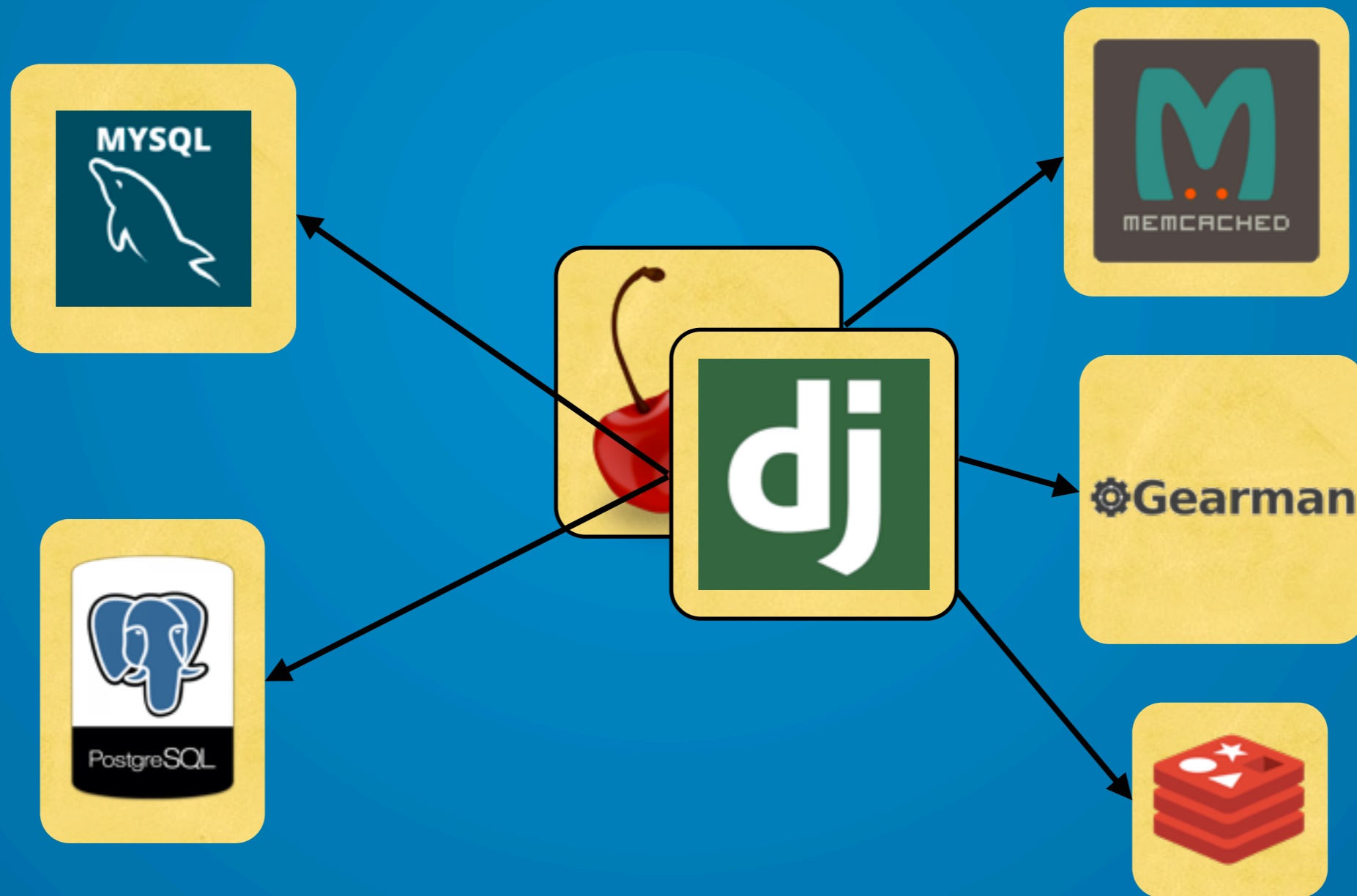




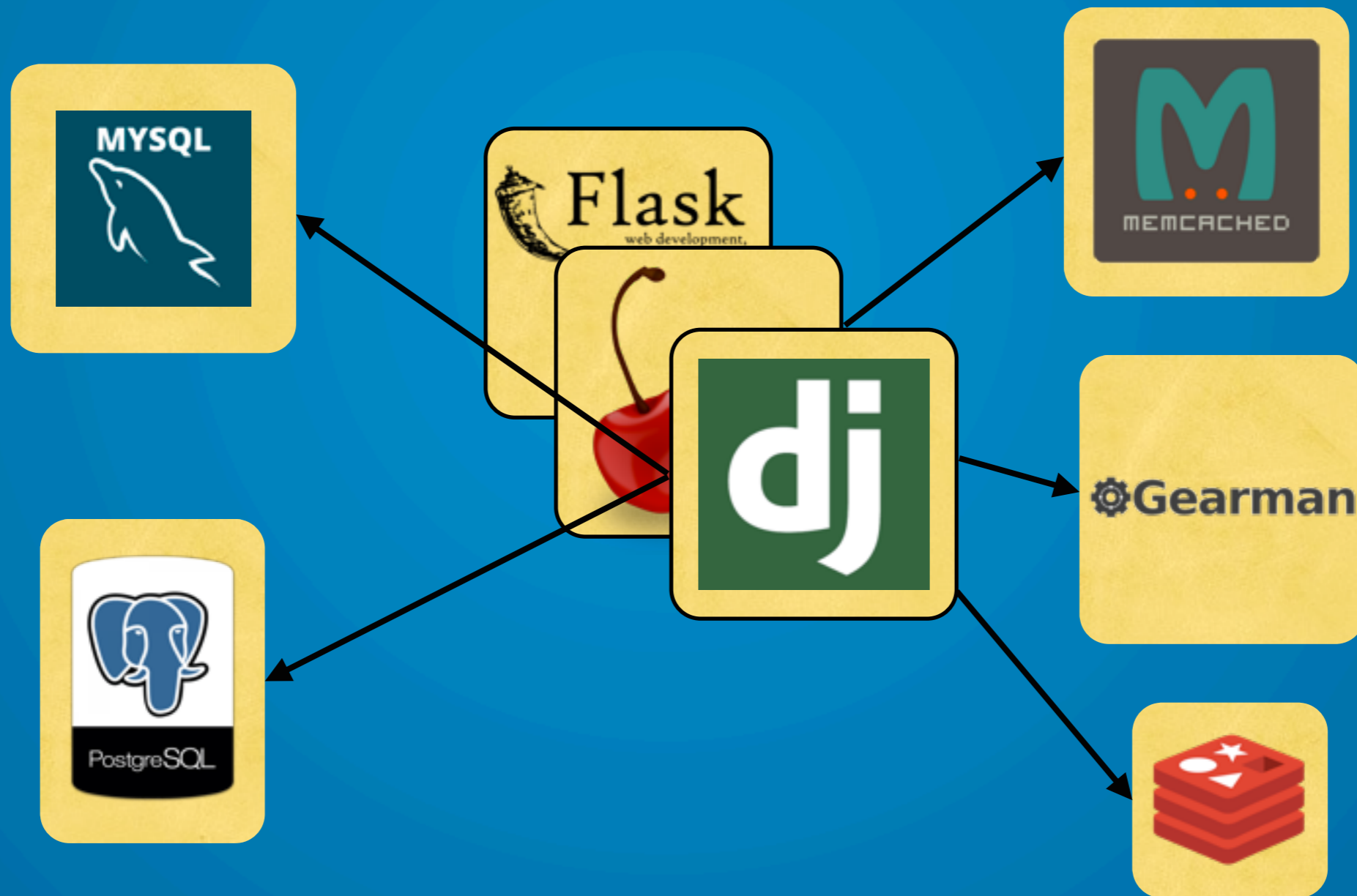
# New Relic



# New Relic



# New Relic



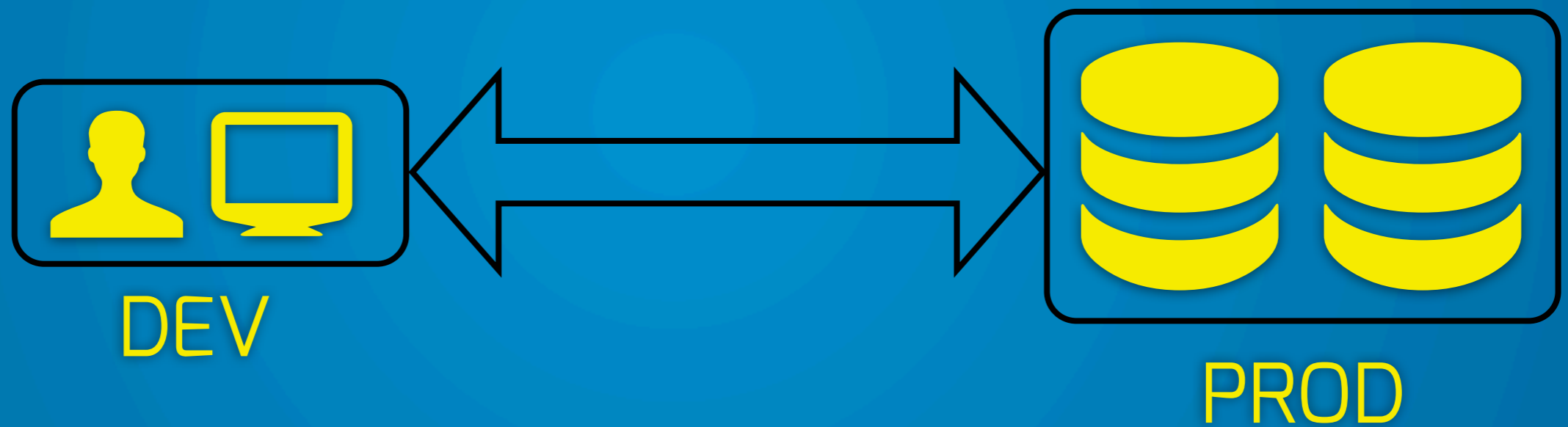
# Problem



# Solution

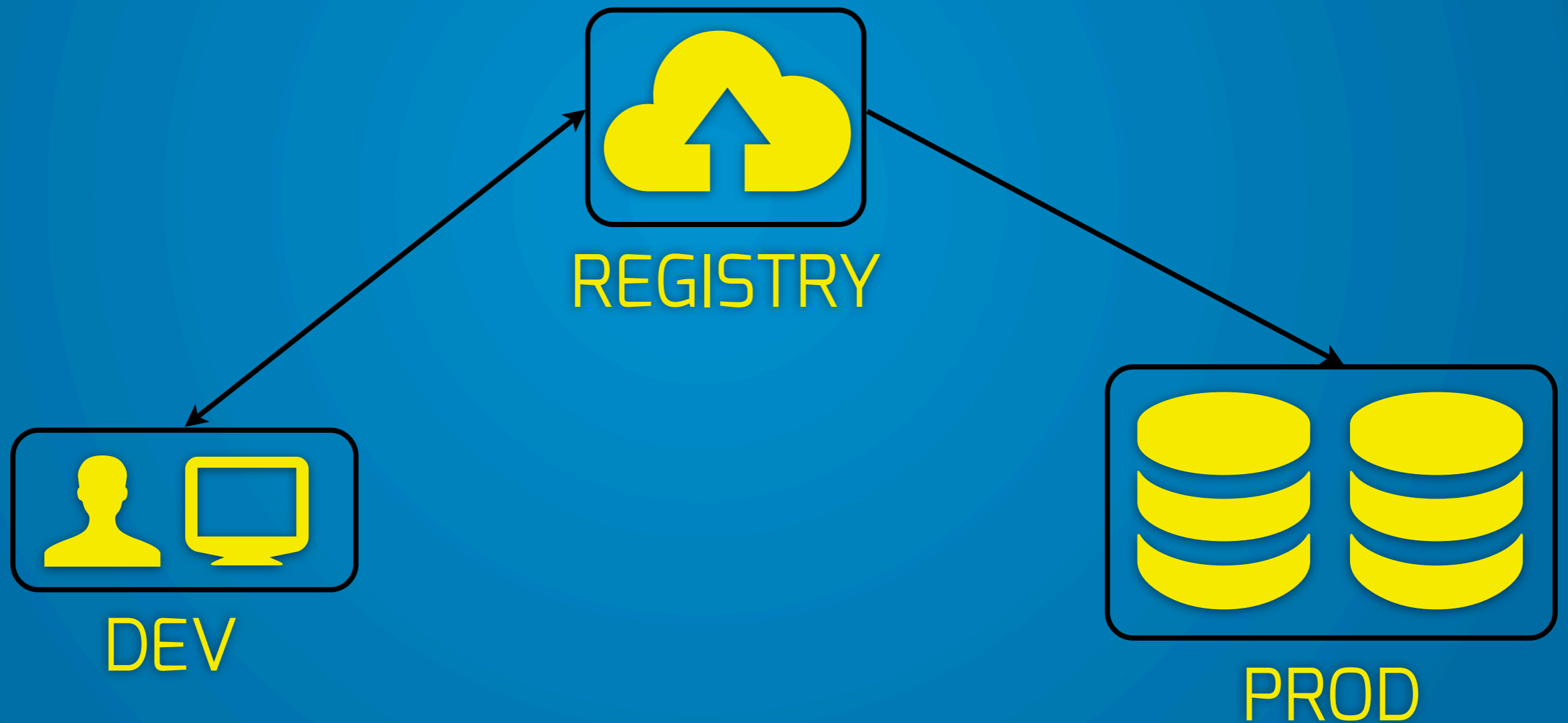


# Why?



REBUILDING the server is trivial.  
Cloning a new server is effortless.

# Solution



# Community

Website: [www.docker.io](http://www.docker.io)

IRC: freenode #docker

google groups: docker-user



**THANK YOU!**

@amjithr

@amjithr

# Questions?

 New Relic.

- Promo: <http://newrelic.com/pycon>
- 30 days of Pro.
- Free T-shirt.
- In-depth data.

