



# DOCKER FOR DEVELOPERS - INTRODUCTION

@HANRENWEI

---

GET THEM: [ONLINE PRESENTATION](#) / [SOURCE CODE](#) / [DOCKER IMAGE](#)

UNDER [ATTRIBUTION 4.0 INTERNATIONAL](#) LICENSE.

# WHAT IS DOCKER (17.03.0-CE)

*Docker is an open platform for developing, shipping, and running applications.*

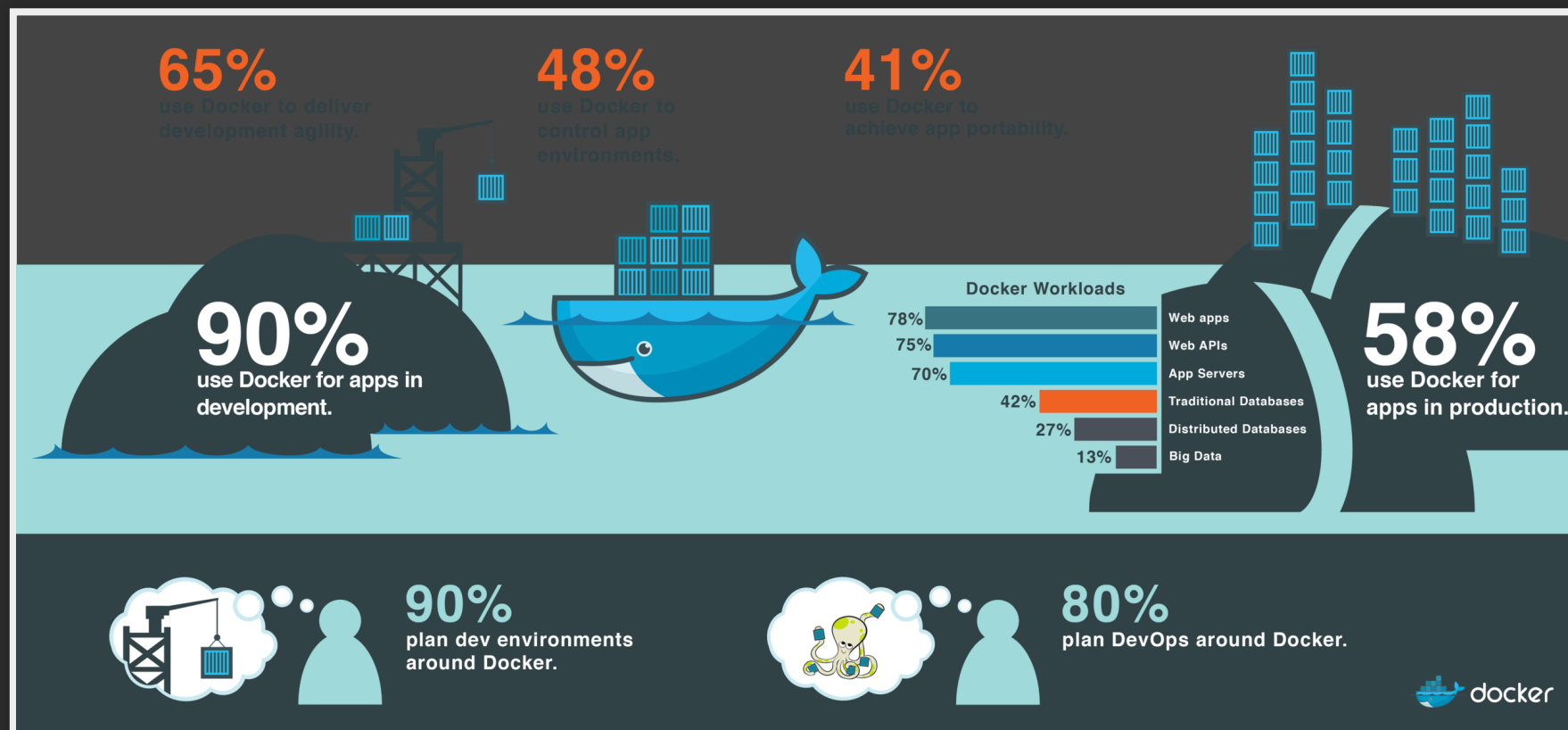
*Docker allows you to package an application with all of its dependencies into a standardized unit for software development.*

# DOCKER BENEFITS

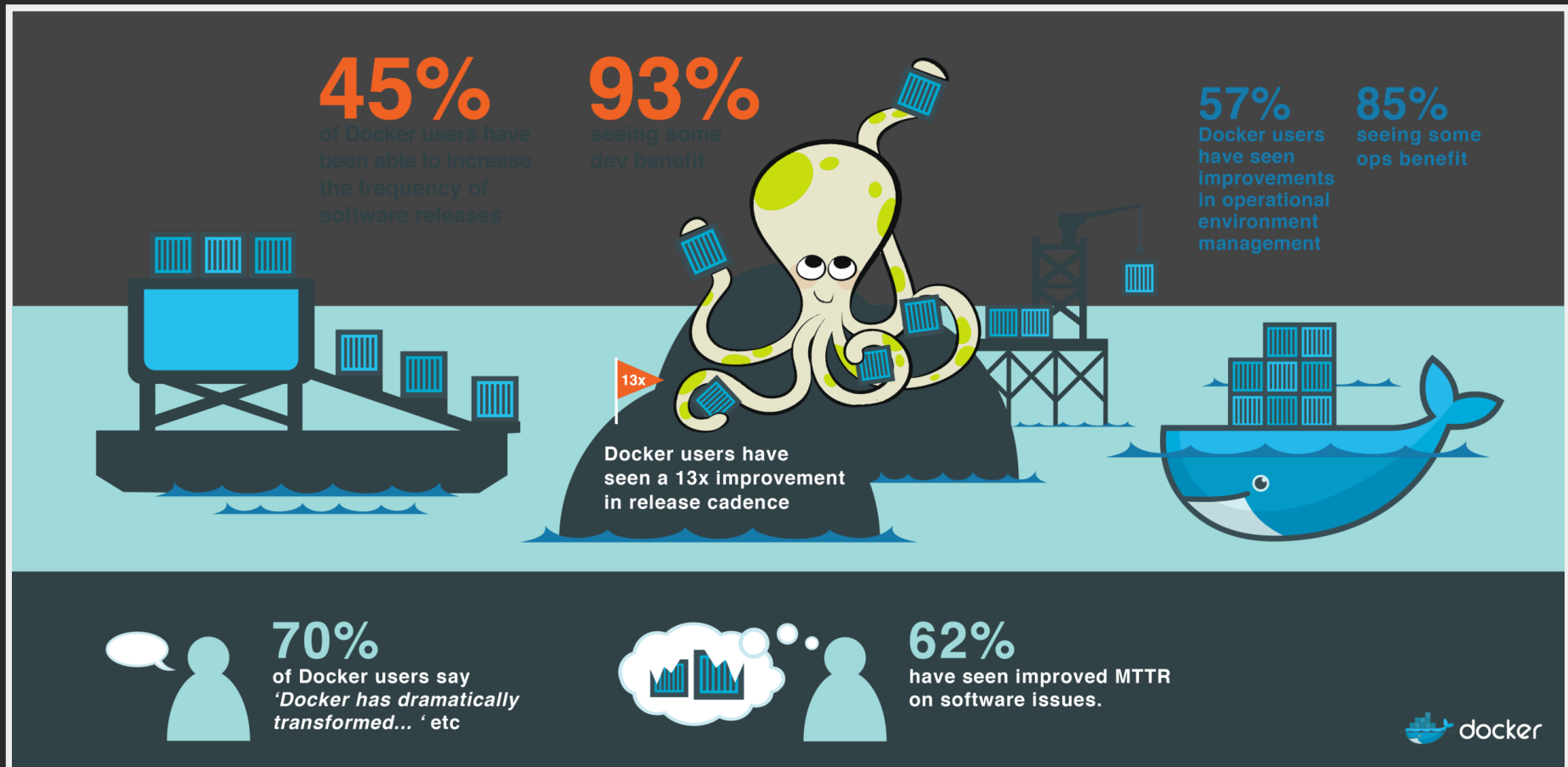
- Fast (deployment, migration, restarts)
- Secure
- Lightweight (save disk & CPU)
- Open Source
- Portable software
- Microservices and integrations (APIs)
- Simplify DevOps
- Version control capabilities

## SEE THE SURVEY RESULTS FOR 2016

- Docker provides the software supply chain with agility, control and portability for app development.



- Docker is delivering quantifiable improvements to application delivery through changing DevOps practices.

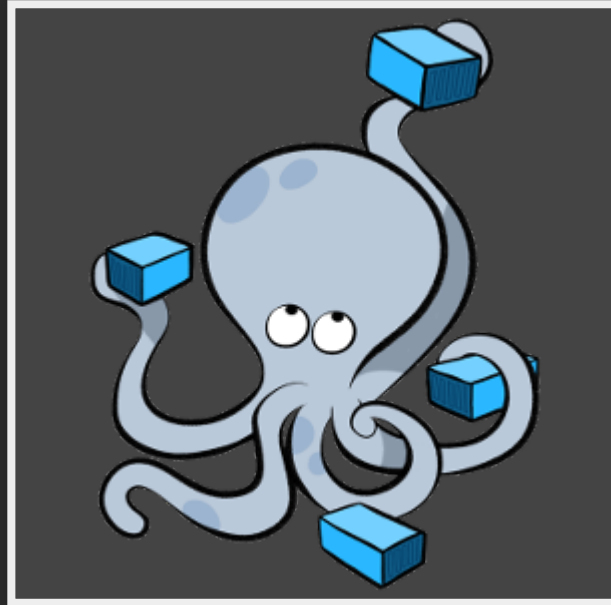




# DOCKER CLIENT

It is the primary user interface to Docker. It accepts commands from the user and communicates back and forth with a Docker daemon.

# DOCKER COMPOSE



A tool for defining and running complex applications with Docker (eg a multi-container application) with a single file.



# STEPS OF A DOCKER WORKFLOW

```
docker run -i -t -d -p 9080:8080 docker.finnplay.net/tomcat:8.5-j
```

- Pulls the tomcat:8.5-jre8-slim **image** from the **registry**
- Creates a new **container**
- Allocates a filesystem and mounts a read-write **layer**
- Allocates a **network/bridge interface**
- Sets up an **IP address**
- Executes a process that you specify

# THE DOCKERFILE

*A Dockerfile is a text document that contains all the commands a user could call on the command line to create an image.*

- [Dockerfile](#)
- [Dockerfile reference](#) on docker docs

# COMMON DOCKER COMMANDS

```
// General info
docker help // docker help run
docker info
docker version
docker network ls

// Images
docker images // docker [IMAGE_NAME]
docker pull [IMAGE] // docker push [IMAGE]

// Containers
docker run
docker ps // docker ps -a, docker ps -l
docker stop/start/restart [CONTAINER]
docker stats [CONTAINER]
docker top [CONTAINER]
```

# DOCKER EXAMPLES

- SSH into a container
- Build an image
- Docker **Volume**
- Using **docker-compose**

# EXAMPLE: SSH INTO A CONTAINER

```
docker pull docker.finnplay.net/tomcat:8.5-jre8-slim
docker run -i -t -d -p 9080:8080 docker.finnplay.net/tomcat:8.5-j
docker ps -a
docker exec -it [container_id] /bin/bash
```

# EXAMPLE: BUILD AN IMAGE

Let's build a **tomcat** image

```
cd ~/docker-presentation
git clone https://github.com/MadeInChina/markdown.git
cd markdown/docker-presentation/introduction-1/examples/dockerfil
docker build -t tomcat:8.5-jre8-slim .

// Test it
docker run -d -p 8099:8080 --name tomcat tomcat:8.5-jre8-slim
// Open http://localhost:8099
```

# EXAMPLE: DOCKER VOLUME

Let's use **tomcat** image

```
cd ~/docker-presentation
mkdir tomcat-example
cd tomcat-example

docker run --name tomcat \
    -p 8099:8080 \
    -v $(pwd):/usr/local/tomcat/webapps/demo \
    -d tomcat:8.5-jre8-slim

// Locally create an index.html file
echo "It works using mount." >> index.html

// Open http://localhost:8099/demo/index.html to view the html fi
```

# EXAMPLE: USING DOCKER COMPOSE

Let's create a tomcat and mysql app with **docker-compose.yml**

```
cd ~/docker-presentation
git clone https://github.com/MadeInChina/markdown.git
cd markdown/docker-presentation/introduction-1/examples/docker-co

// Run docker-compose using the docker-compose.yml
cat docker-compose.yml
docker-compose up -d
```

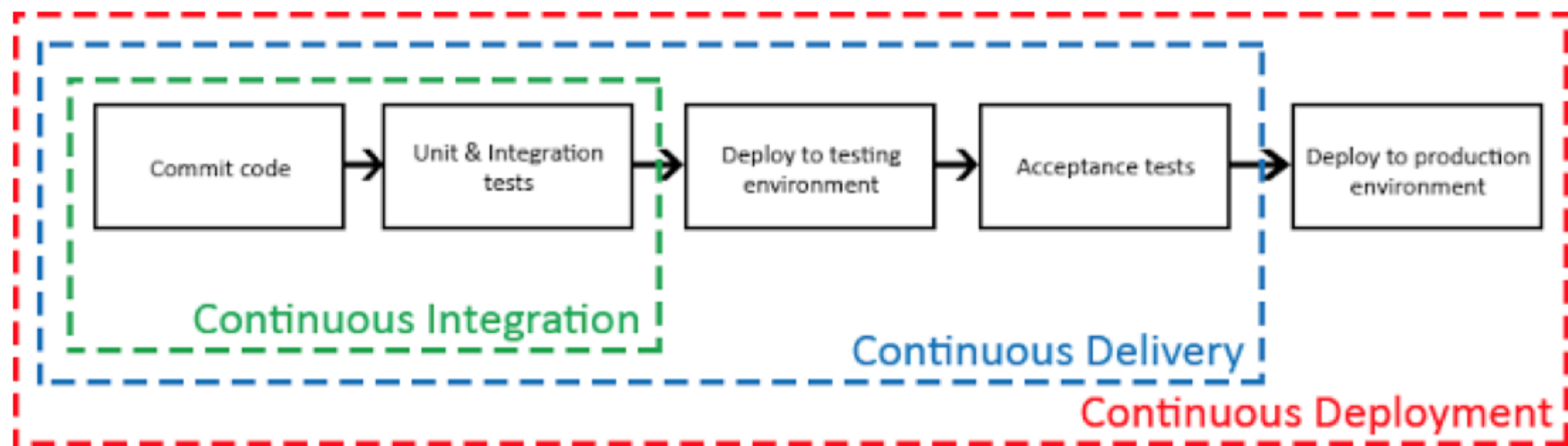
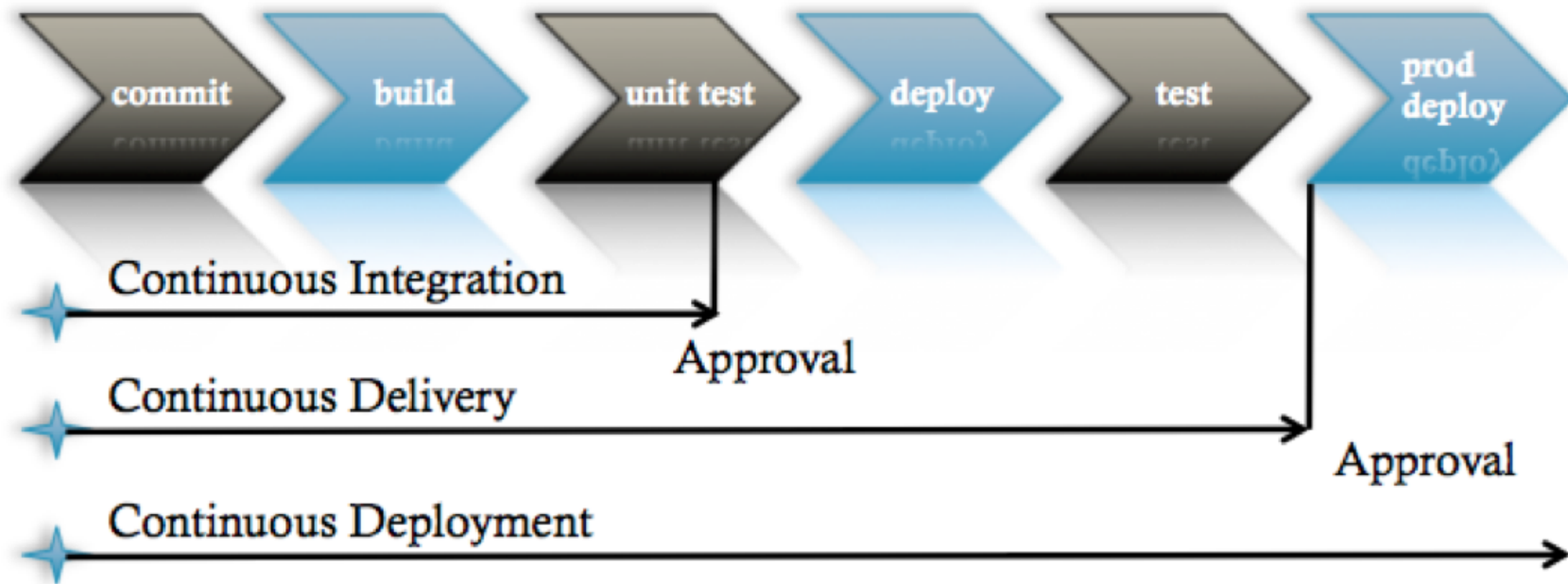


# DOCKER TIPS

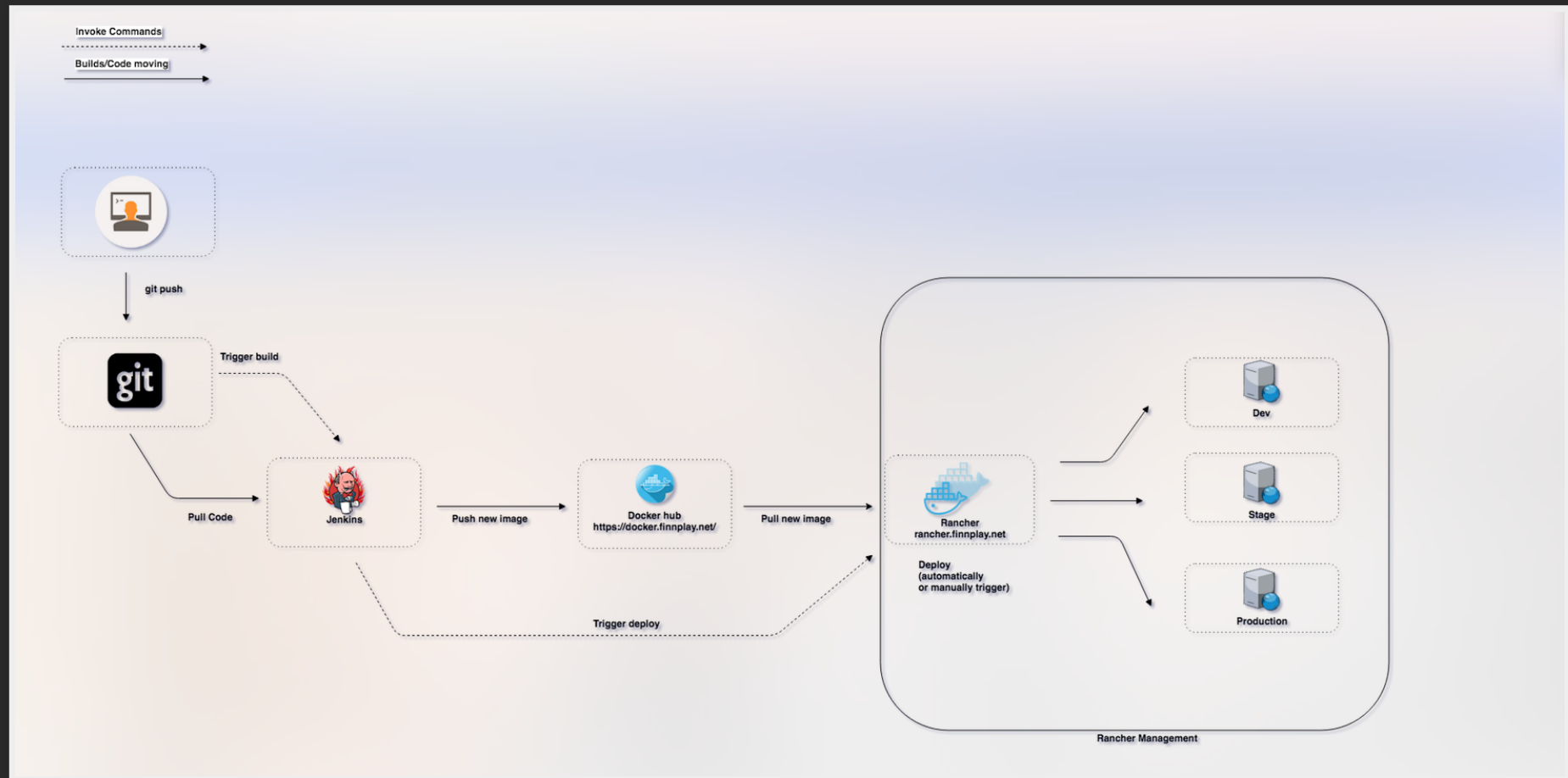
There are known best practices (see a list at [examples/tips](#))

- Optimize containers (check [fromlatest.io](#) and [imagelayers.io](#))
- Create your own tiny base
- Containers are not Virtual Machines
- Full stack Images VS 1 process per Container
- Create your private registry
- Create shortcut commands
- Use docker-compose.yml templates
- Be aware of the [hub.docker.com](#) docker agent version

# CONTINUOUS DELIVERY: JENKINS PIPELINE



# DEMO



# QUESTIONS?

IN THIS PRESENTATION I HAVE USED **OH MY ZSH**, **DOCKER 17.03.0-CE**, **WHARFEE** AND **DRY**.