SACINTERACTIVE

DOCKER FOR DEVELOPERS

ABOUT ME

- Application Architect at Clear Capital
- Building web applications since 2000
- Mainly middle tier to backend
- Member of SacInteractive since 2000

AGENDA

- What is Docker?
- Installation/Platforms
- Images & Containers
- Dockerfile
- Docker Compose

AGENDA

- High level view of the Docker ecosystem
- Pull images and create and run containers
- Simple container build script
- Connect multiple containers

WHAT IS DOCKER

- Definition
- Containers vs. virtual machines
- Benefits for developers
- Client and server

WHAT IS DOCKER

- The world's leading software container platform
- Enterprise and Community editions
- Run apps side-by-side in isolated containers
- Developers: simplify collaboration with co-workers
- Enterprises: offers agility, ship new features faster, more securely and with confidence for both Linux and Windows

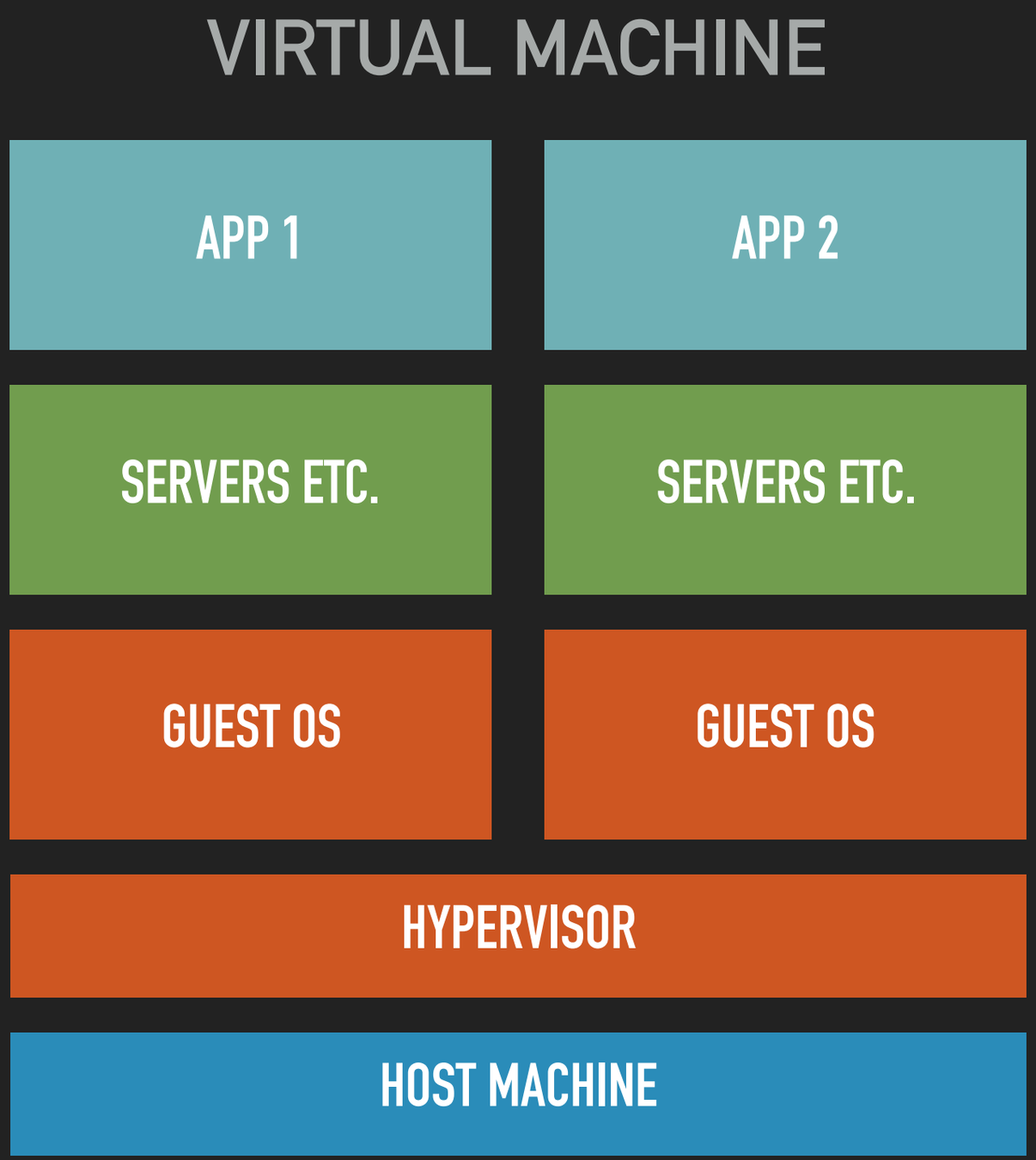
https://www.docker.com/what-docker

WHAT IS A CONTAINER

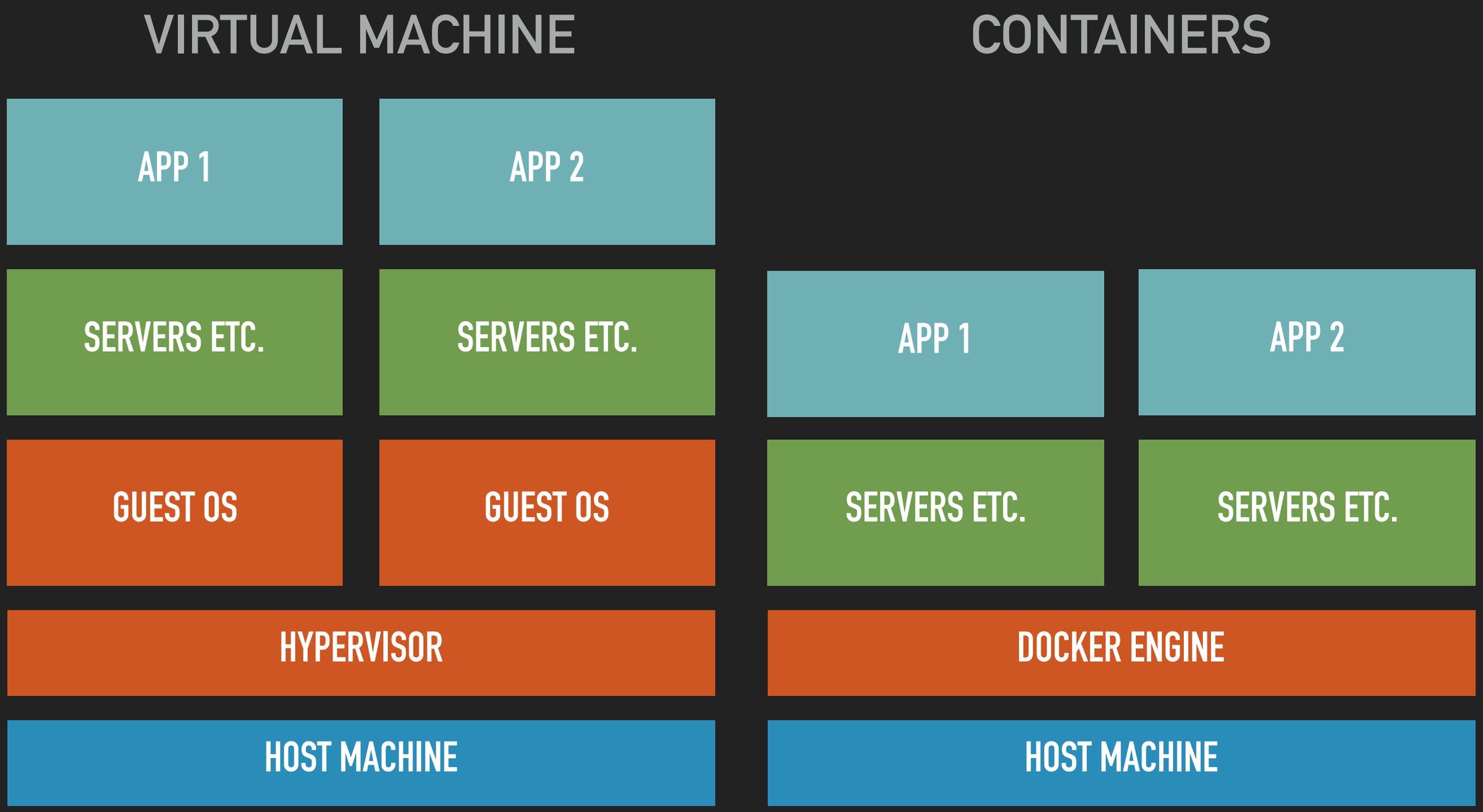
- Everything required to make a piece of software run is packaged into isolated containers
- Unlike VMs, containers do not bundle a full operating system - only libraries and settings required to make the software work are needed
- This makes for efficient, lightweight, self-contained systems and guarantees that software will always run the same, regardless of where it's deployed

https://www.docker.com/what-docker

CONTAINERS VS. VIRTUAL MACHINES



CONTAINERS VS. VIRTUAL MACHINES



BENEFITS FOR DEVELOPERS

- Automates the repetitive tasks of setting up and configuring development environments
- Eliminates the “runs on my machine” syndrome
- Guarantees consistent environments
- Onboarding new developers becomes simple
- Test multiple versions side by side

INSTALLATION / PLATFORMS

- Host Operating Systems
  - Linux and Windows Server 2016, others with VM
- Command Line, IDEs, Kitematic
- Docker Installation
  - Mac and Windows desktops
  - AWS and Azure
  - Windows Server, Various Linux flavors

IMAGES & CONTAINERS

- Images are the blueprint
- Can think of images like a class
- Contain a layered file system
- Make use of other images
- Docker Hub / Docker Store / local images

IMAGES & CONTAINERS

- Containers are the “live” implementation of an image
- Can think of them as objects instantiated from a class
- Adds a read / write layer on top of layered file system
- Create multiple containers from single image
- Are isolated by default

DOCKER HUB / DOCKER STORE

- Public repository of images
- Certified and community images
- Post your own images
- Keep local images

DEMOS: IMAGES & CONTAINERS

- Pull an image
- Run container from image
- A more complex container
- Save customized container
- Compose containers

DEMOS: PULL AND RUN IMAGE

- Hello world image: hello-world
  - `docker pull hello-world`
  - `docker run hello-world`
- Show our images
  - `docker image ls`
  - `docker images`
- Show our containers
  - `docker ps -a`

DEMOS: REMOVE IMAGES AND CONTAINERS

- Remove a container
  - `docker ps -a`
  - `docker rm <first digits of ID>`
- Remove an image
  - `docker image ls`
  - `docker image rm <first digits of container ID>`
  - `docker rmi <first digits of container ID>`
- Can also use full container or image name

DEMOS: KITEMATIC AND IDE

- Tools to manage images and containers
- Kitematic
- IntelliJ IDE
- Eclipse
- Visual Studio / Visual Studio Code

DEMOS: WEB APP CONTAINER

- A more complex container - ColdFusion web app server
- `docker run` with flags
  - Port: -p
  - Detached mode: -d
  - Mapped volumes: -v
  - Named container: —name

DEMOS: WEB APP CONTAINER

- `docker pull ortussolutions/commandbox`
- `docker run -p 8080:8080 ortussolutions/commandbox`
- `docker run -p 8080:8080 -v "$(pwd):/app" ortussolutions/commandbox`
- `docker run -p 8080:8080 -v "$(pwd):/app" -d ortussolutions/commandbox`
- `docker run -p 8080:8080 -v "$(pwd):/app" -d --name sac-interactive-cf ortussolutions/commandbox`

DEMOS: SQL SERVER CONTAINER

- Microsoft SQL Server on Linux
- How long does it take to install SQL Server?

- `docker pull microsoft/mssql-server-linux`
- `docker run -e 'ACCEPT_EULA=Y' -e 'SA_PASSWORD=MyPassword?123' -p 1433:1433 --name sac-interactive-sql microsoft/mssql-server-linux`

DEMOS: DOCKERFILE FOR SQL SERVER IMAGE

- Build a new image based upon an existing image
- Can contain many commands
  - Copy files
  - Environment variables
  - Run commands
  - Mappings
  - Ports

DOCKER COMPOSE

- This is where things get interesting
- Orchestrate multiple containers
- Link containers to their own internal network
- Define an entire environment in one file
- Share with peers, build exactly the same environment

DEMOS: DOCKER-COMPSE

- Create environment with CF and SQL Server
- Docker compose ties containers together
- Build whole environment in one command
  - `docker-compose build`
- Run whole environment in one command
  - `docker-compose up -d`

DEMOS: DOCKER-COMPSE

- View status of running services
  - `docker-compose ps`
- Stream log output
  - `docker-compose logs`
- Stop services
  - `docker-compose down`
  - `docker-compose down —rmi all`

DEMOS: PUSH AN IMAGE TO DOCKER HUB

- Push our ColdFusion image
- `docker run -p 8080:8080 -v "$(pwd):/app" ortussolutions/commandbox`
- `docker login -u username -p password`
- `docker commit <container ID> notthemonkee/sacinteractive-cf`
- `docker push notthemonkee/sacinteractive-cf`

WHERE TO GO FROM HERE

- docker.com
  - https://docs.docker.com/reference/
    - Engine (docker) CLI
- https://hub.docker.com/
- pluralsight.com
- Docker Sacramento Meetup

QUESTIONS?

QUESTIONS

- dave@davidhjones.com
- blog.davidhjones.com
- @notthemonkee