



11 - Deploying Docker Containers

Containers in Development vs Containers in Production

Containers Are Always Great!

In Development

Isolated, standalone environment

Reproducible environment, easy to share and use

In Production

Isolated, standalone environment

Reproducible environment, easy to share and use

No surprises!

What works on your machine (in a container) will also work after deployment

What to watch out for during Production

Development to Production: Things To Watch Out For

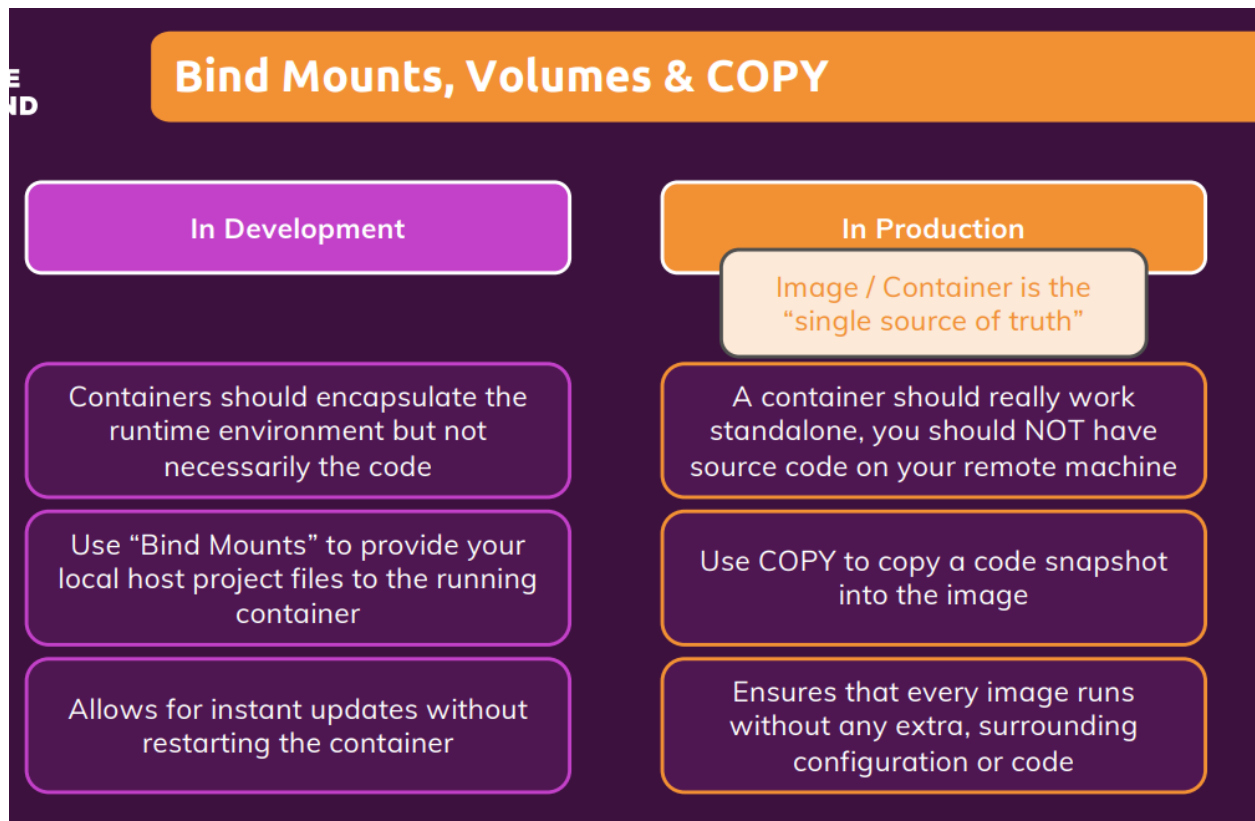
Bind Mounts shouldn't be used in Production!

Containerized apps **might need a build step** (e.g. React apps)

Multi-Container projects might need to be **split** (or should be split) across multiple hosts / remote machines

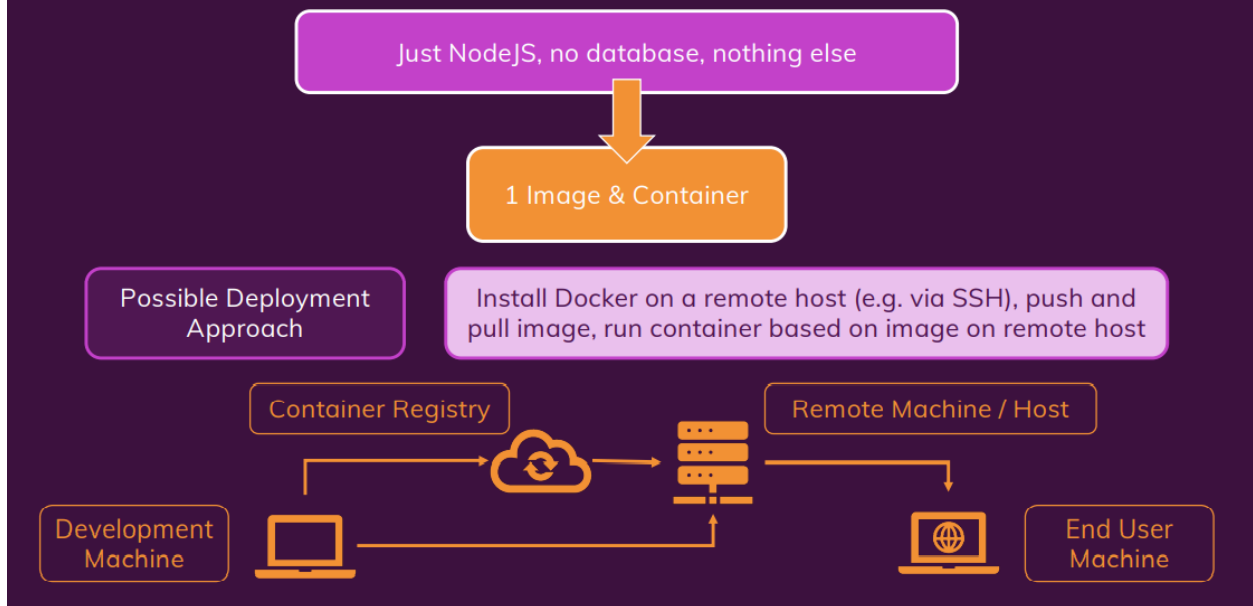
Trade-offs between **control** and **responsibility** might be worth it!

Bind Mounts, Volumes and COPY in Production vs in Development



Example of a stand-alone app deployment

A Basic First Example: Standalone NodeJS App



Most famous network providers

Cloud Computing Services - Amazon Web Services (AWS)

Whether you're looking for compute power, database storage, content delivery, or other functionality, AWS has the services to help you build sophisticated applications with increased flexibility,

 <https://aws.amazon.com>



Cloud Computing Services | Microsoft Azure

Invent with purpose, realize cost savings, and make your organization more efficient with Microsoft Azure's open and flexible cloud computing platform.

 <https://azure.microsoft.com/en-us/>



Cloud Computing, Hosting Services, and APIs | Google Cloud

Start building on Google Cloud with \$300 in free credits and 20+ always free products. Tell us what you're solving for. A Google Cloud expert will help you find the best solution.

 <http://shorturl.at/bmDTU>



Steps of Deploying to AWS EC2

Example: Deploy to AWS EC2

AWS EC2 is a service that allows you to spin up and manage your own remote machines

1

Create and launch EC2 instance, VPC and security group

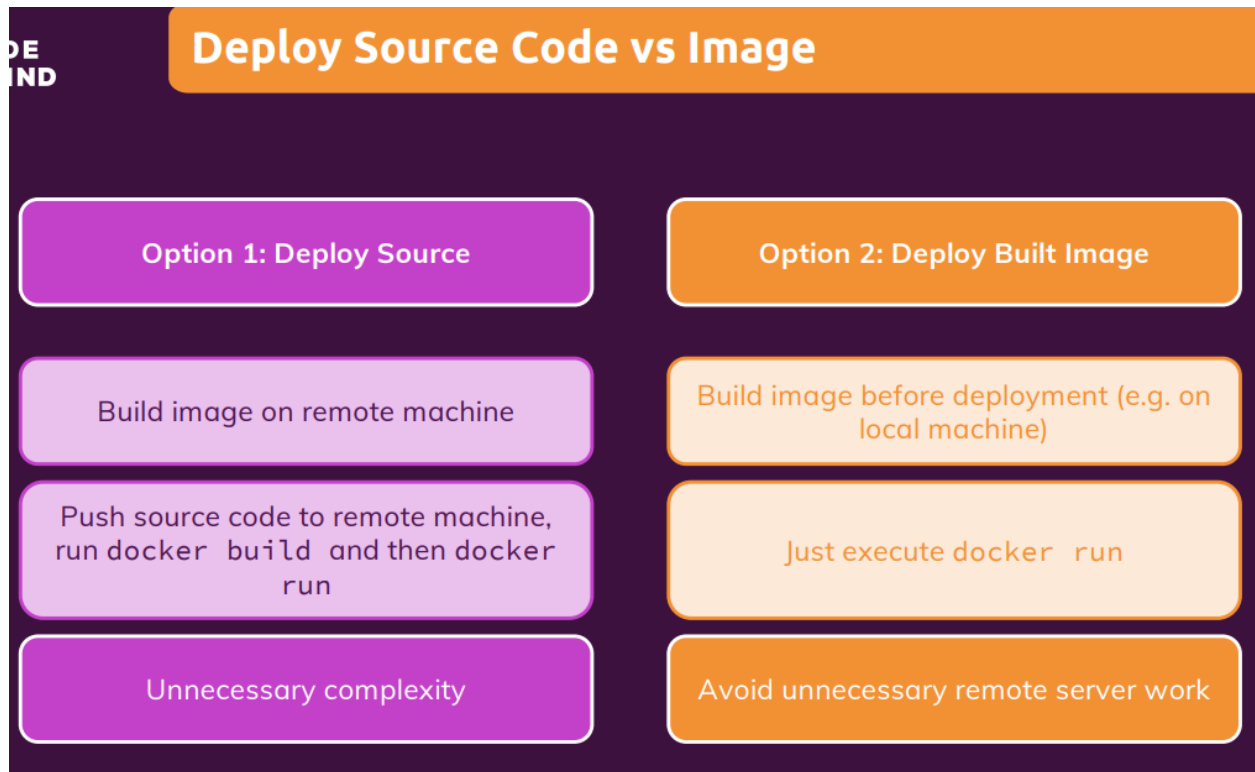
2

Configure security group to expose all required ports to WWW

3

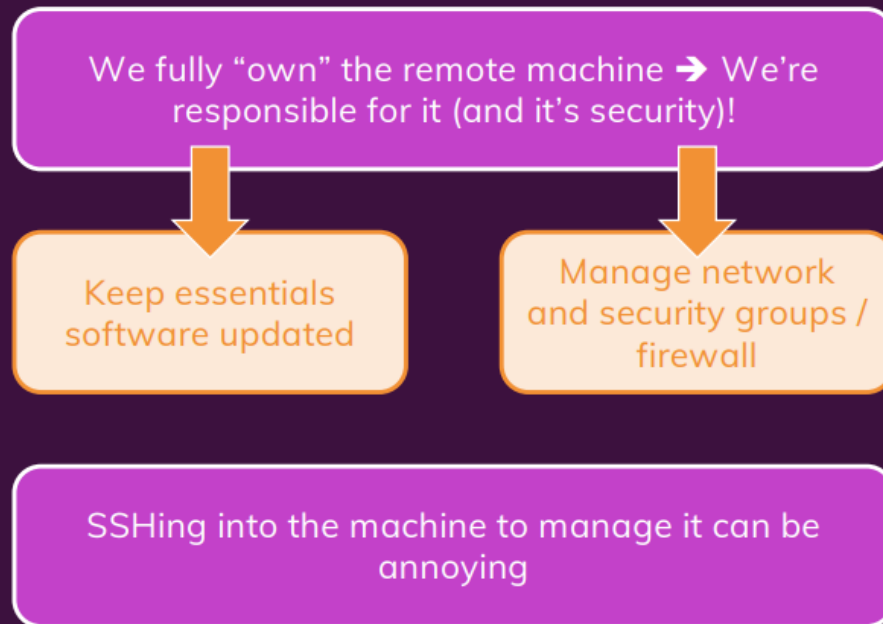
Connect to instance (SSH), install Docker and run container

Deploying a Container vs Deploying Source Code



Having Full Control Setbacks

“Do-it-yourself” Approach – Disadvantages



Automated Approach

A Managed / Automated Approach



Your Own Remote Machines
e.g. AWS EC2

You need to create them, manage them, keep them updated, monitor them, scale them etc.

Great if you're an experienced admin / cloud expert



Managed Remote Machines
e.g. AWS ECS

Creation, management, updating is handled automatically, monitoring and scaling is simplified

Great if you simply want to deploy your app / containers

A Note about Databases

You can absolutely manage your own Database containers

but ...



Scaling & managing availability can be challenging



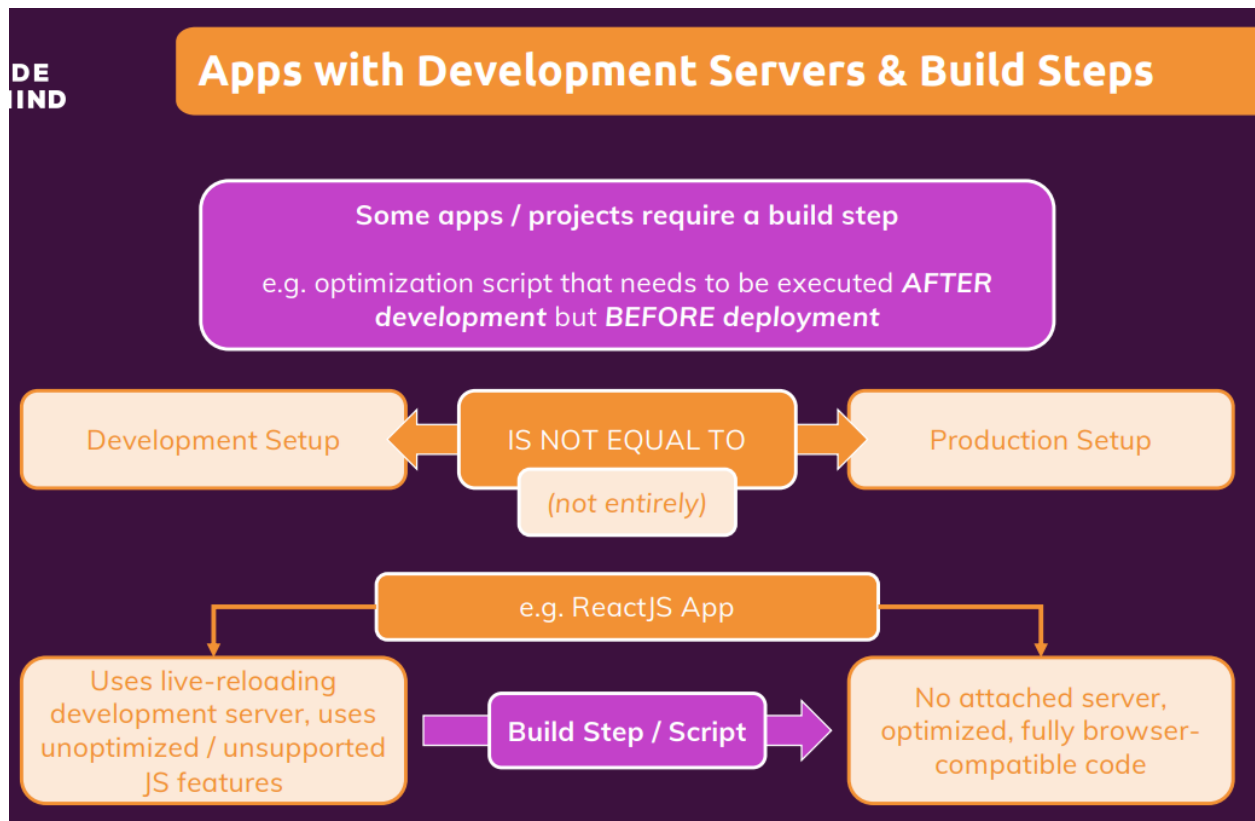
Performance (also during traffic spikes) could be bad



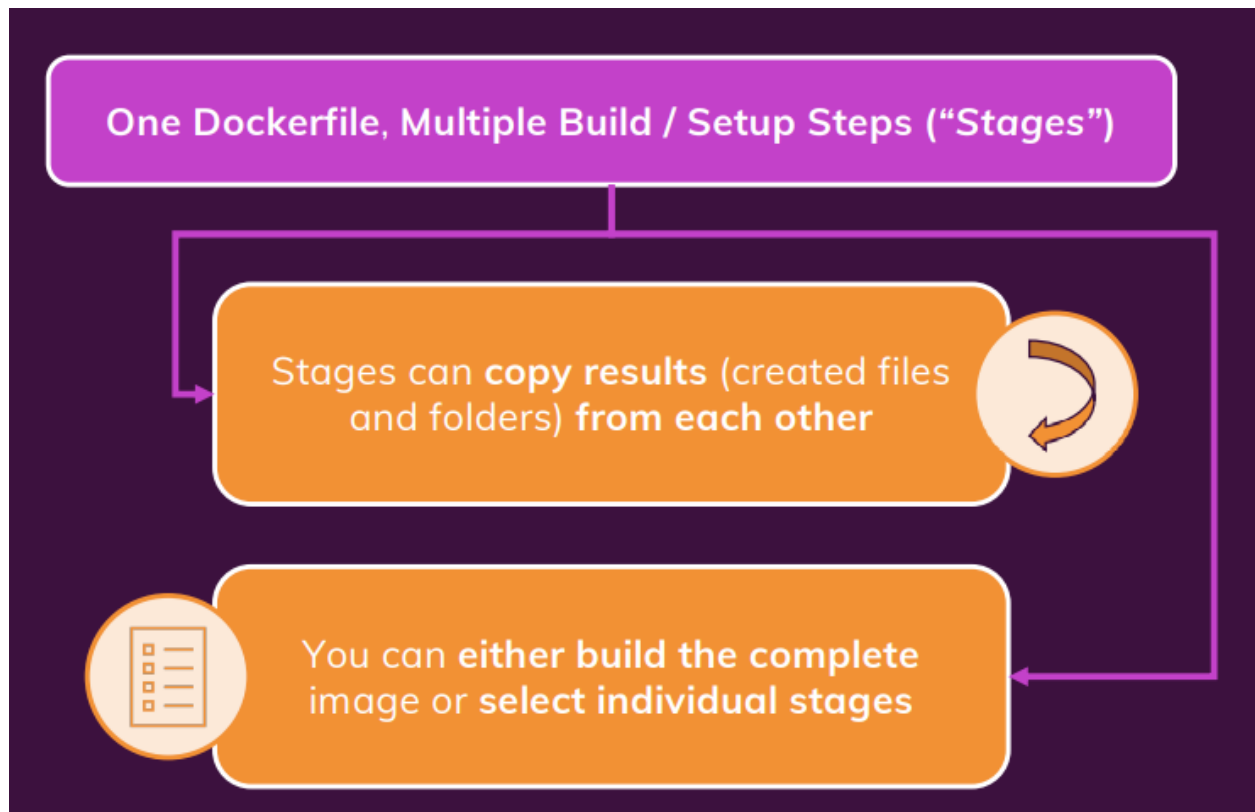
Taking care about **backups and security** can be challenging

Consider using a **managed Database service** (e.g. AWS RDS, MongoDB Atlas, ...)

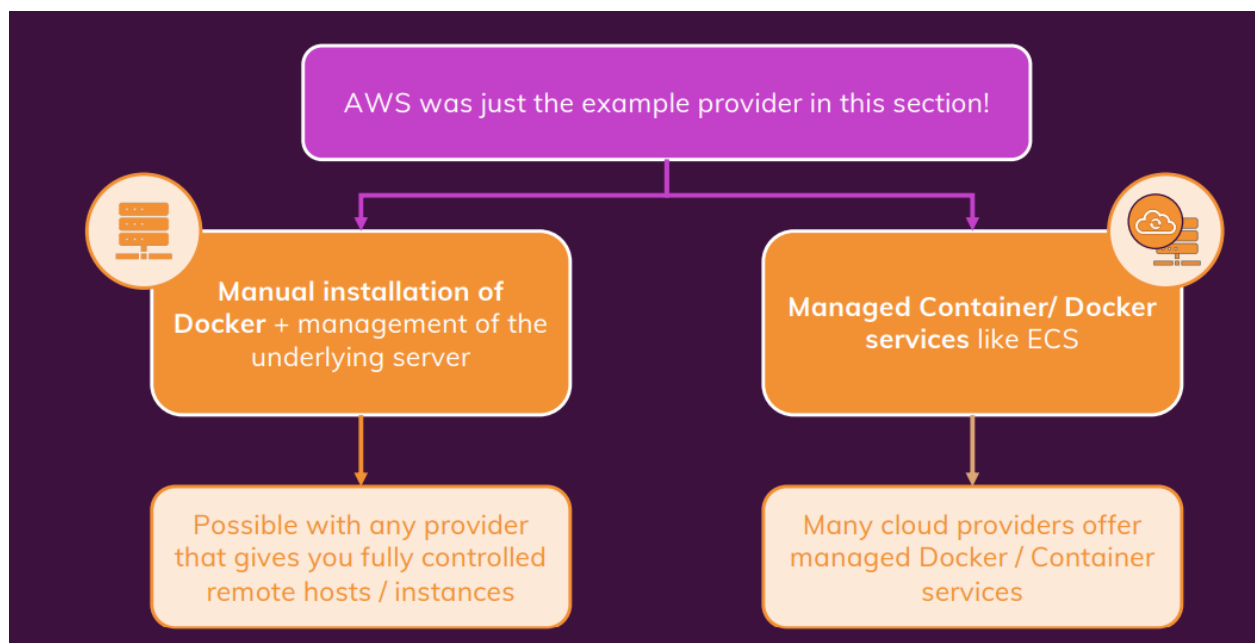
Apps with development Servers



Multi-Stage Builds



From AWS to other providers



Future Improvements

