# Building New Images

**Nigel Poulton**
Author & Trainer

@nigelpoulton | www.nigelpoulton.com

# Overview/ Summary

50K-foot View

Hands-on Workflow

Building Images Deep Dive

Multi-stage Builds

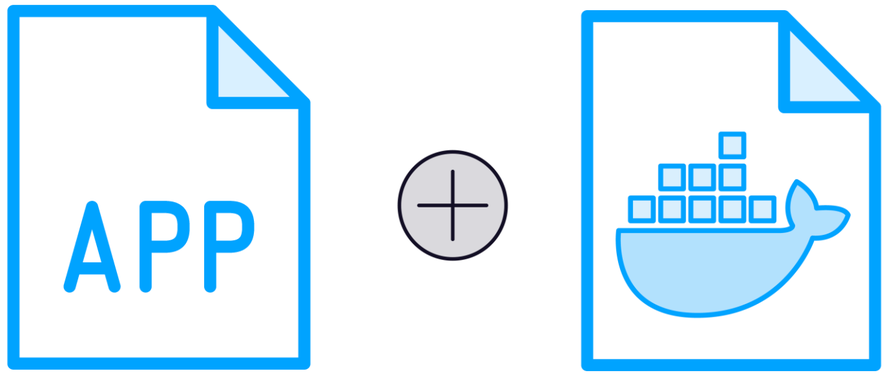
Recap

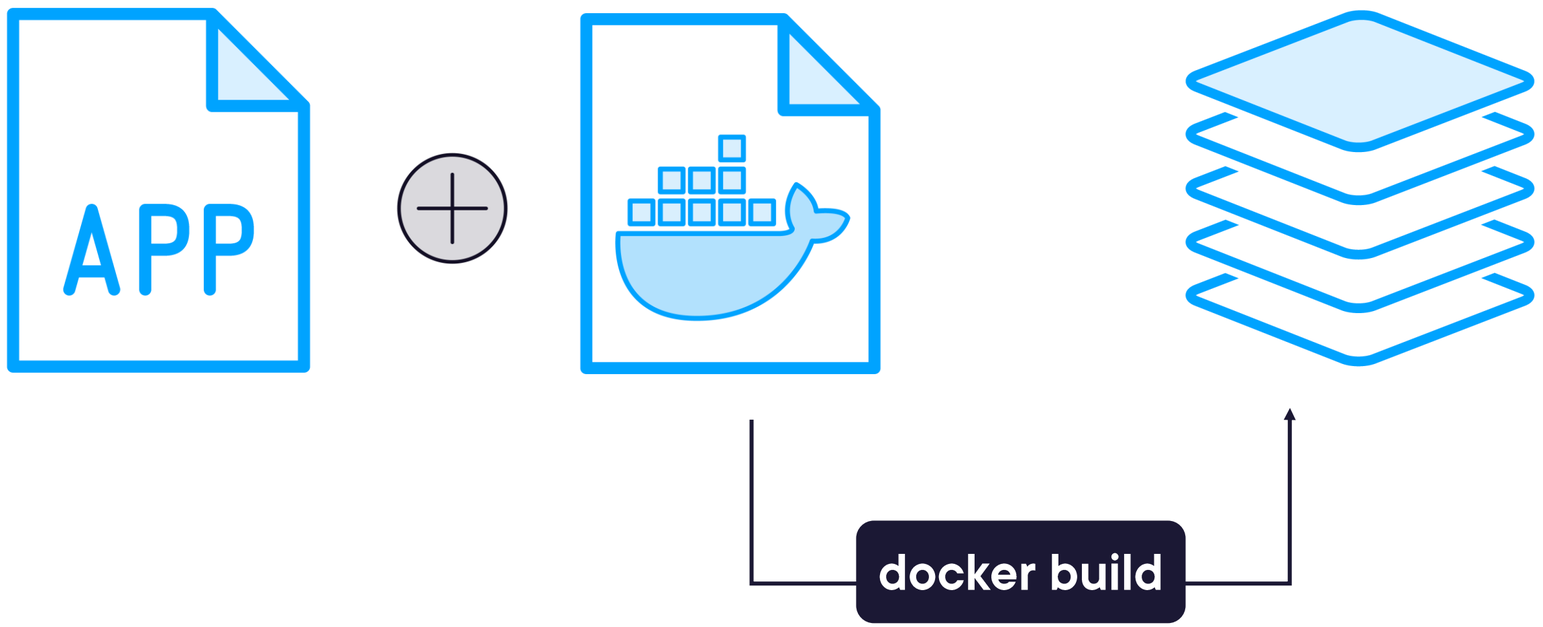
Up Next:

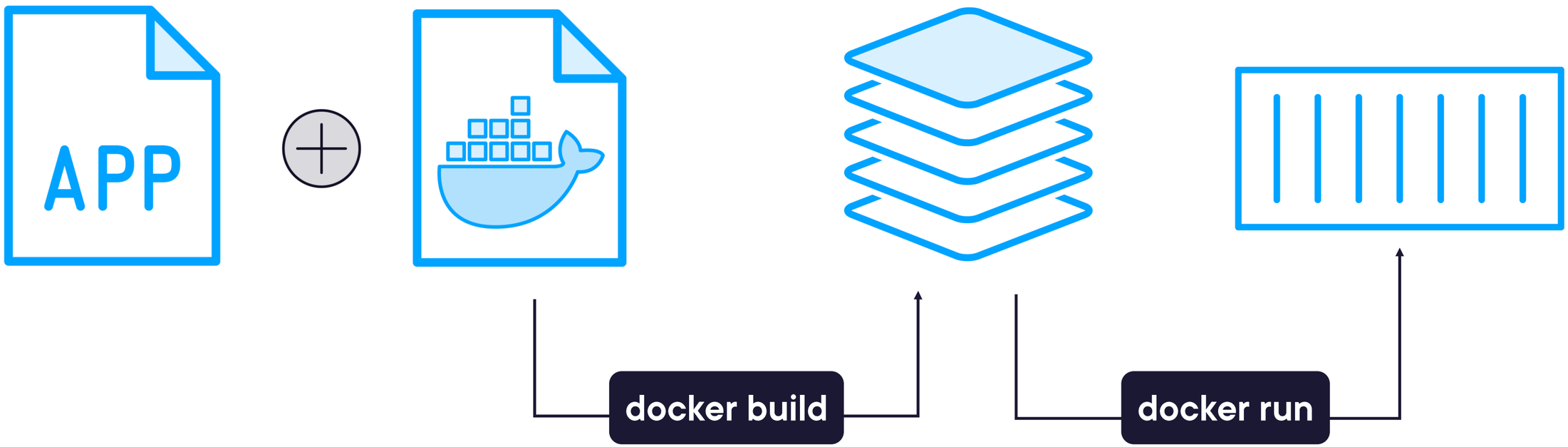
# 50K-foot View

# 50K-foot View

APP


APP


APP
docker build

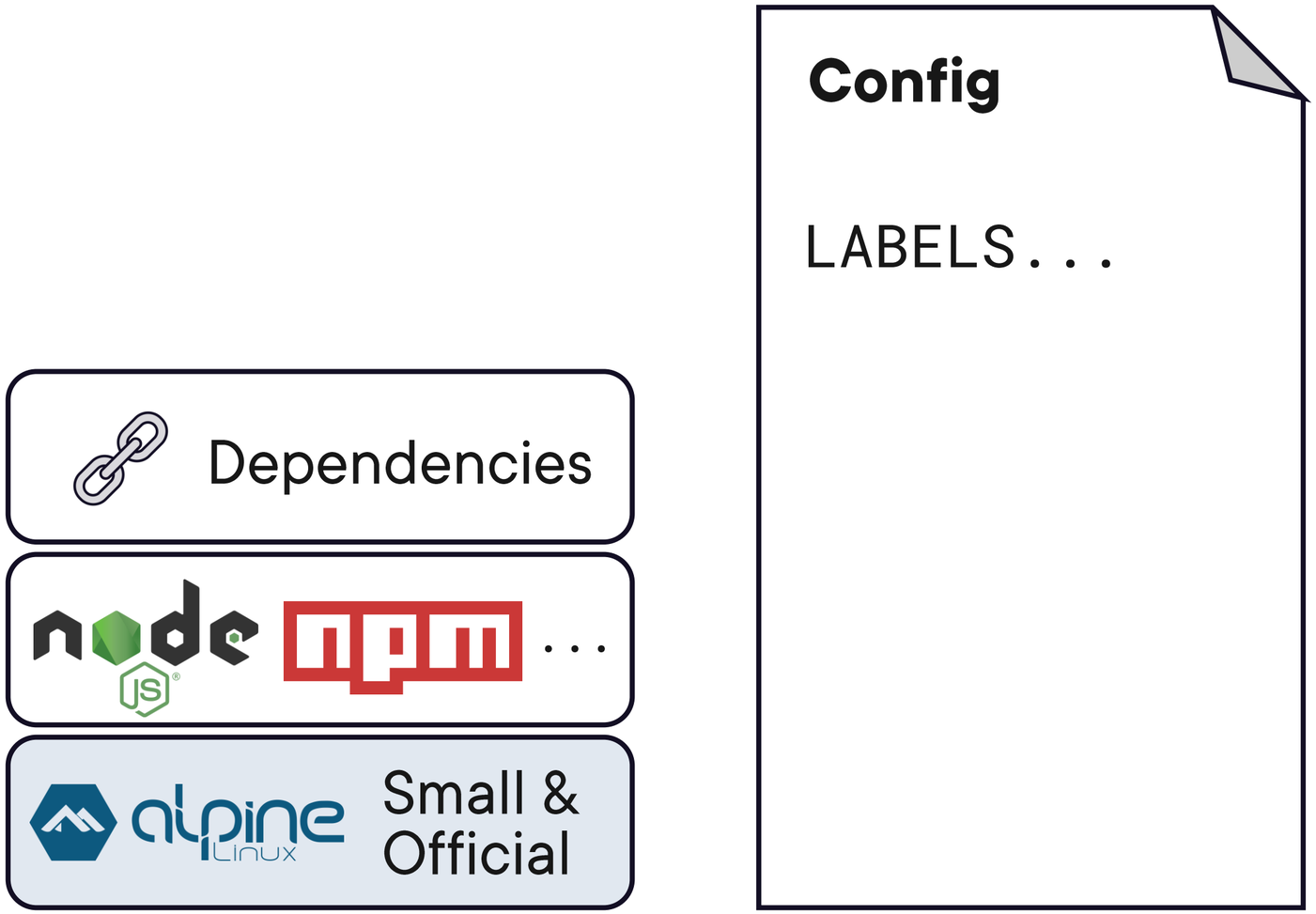

APP
docker build
docker run

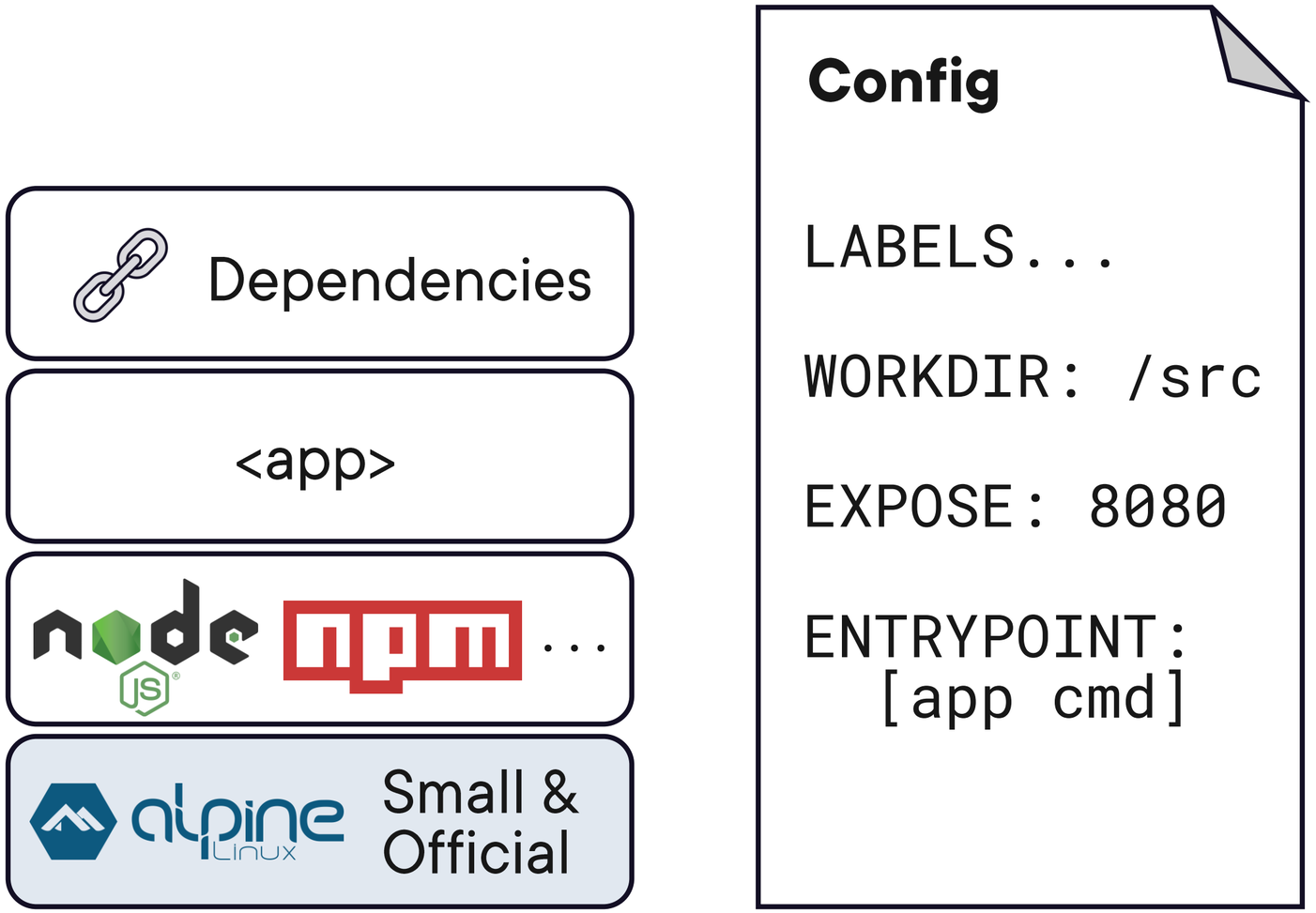
Up Next:

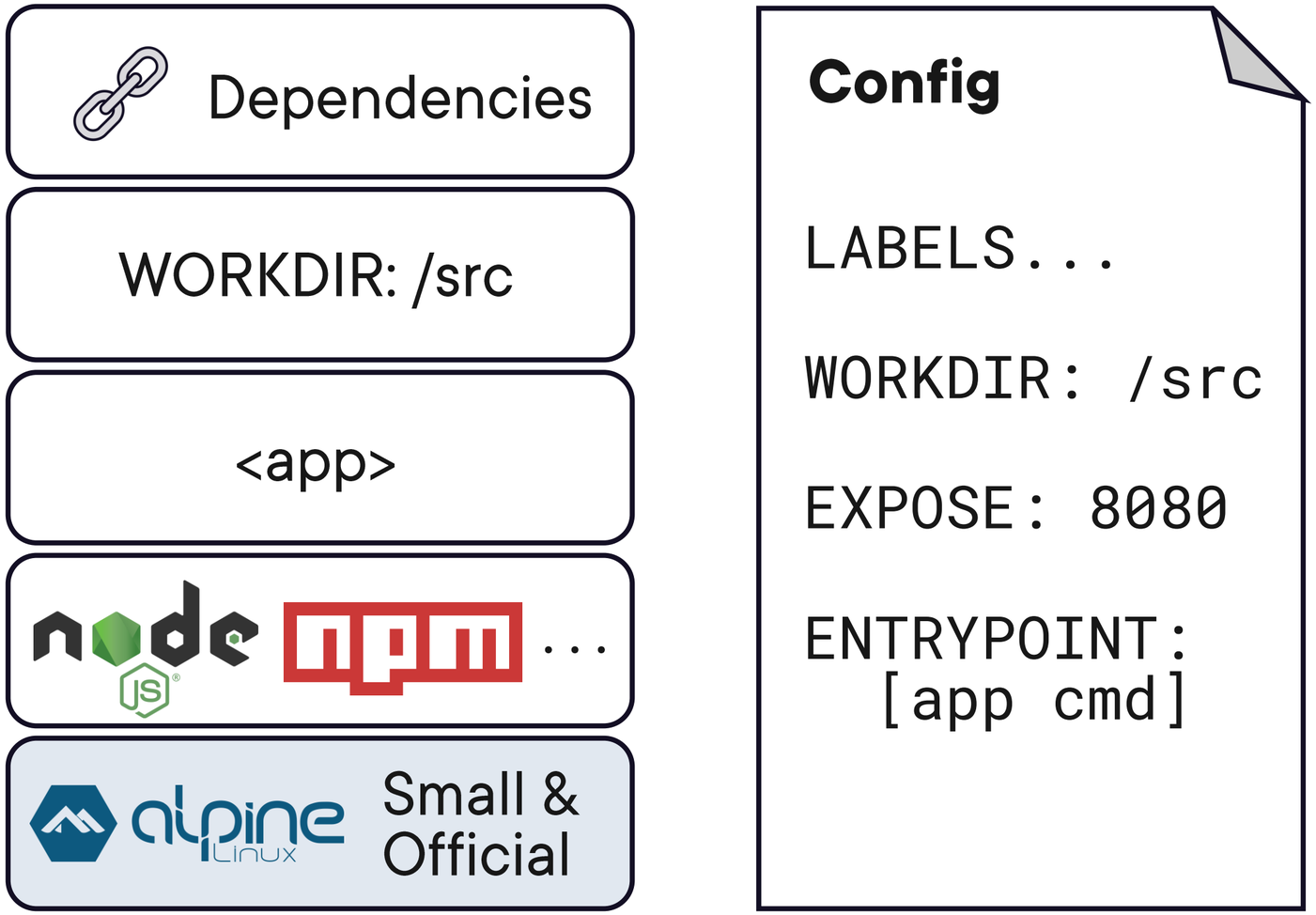
# Hands-on Workflow

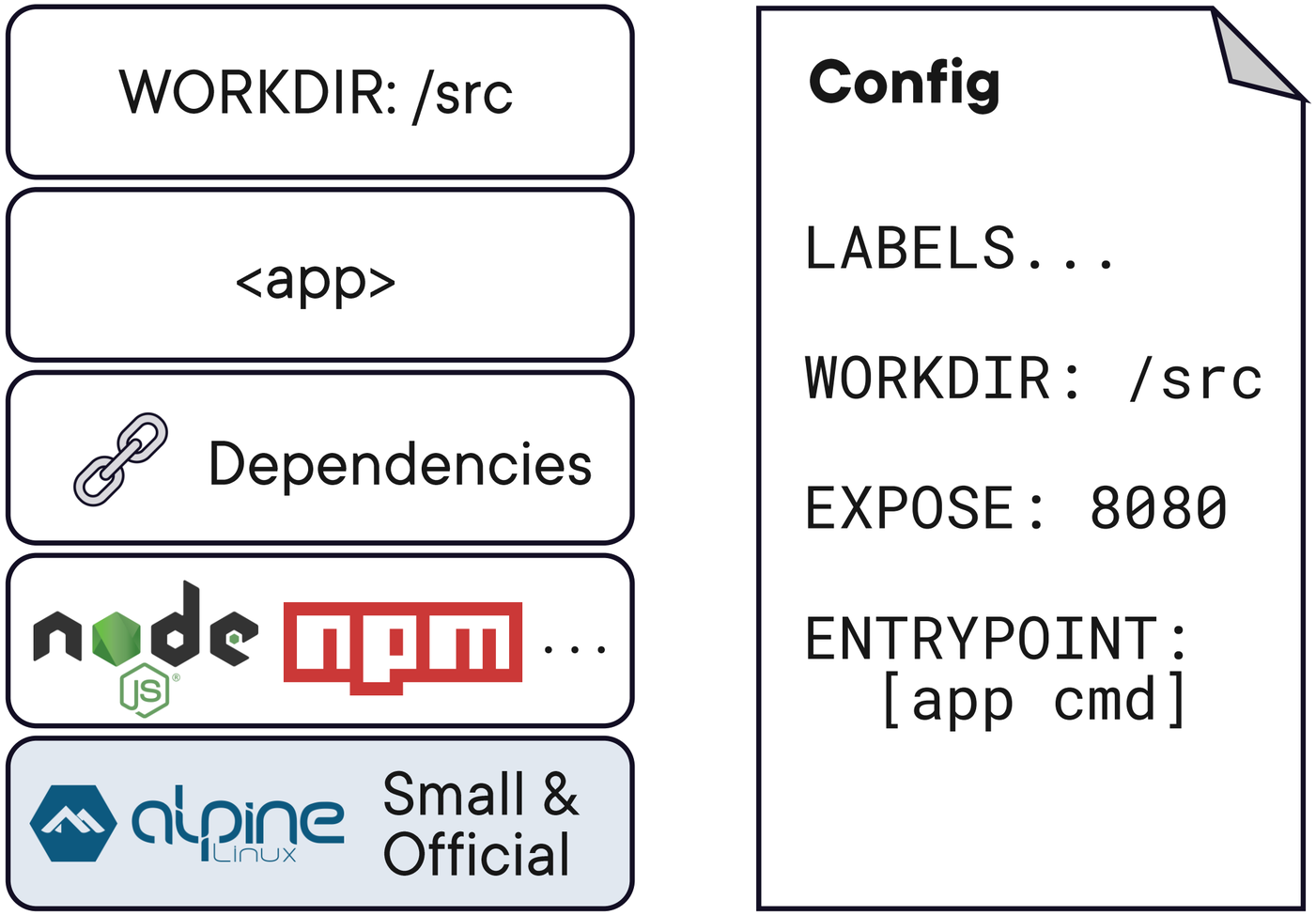
# Hands-on Workflow

APP
docker build


Dependencies
node
npm
...
alpine
Linux
Small &
Official
Config
LABELS...


Dependencies
<app>
node
npm
...
alpine
Linux
Small &
Official
Config
LABELS...
WORKDIR: /src
EXPOSE: 8080
ENTRYPOINT:
[app cmd]


Dependencies
WORKDIR: /src
<app>
node
npm
...
alpine Linux
Small & Official
Config
LABELS...
WORKDIR: /src
EXPOSE: 8080
ENTRYPOINT:
[app cmd]


WORKDIR: /src
<app>
Dependencies
node
npm
...
alpine
Linux
Small &
Official
Config
LABELS...
WORKDIR: /src
EXPOSE: 8080
ENTRYPOINT:
[app cmd]

Up Next:

# Building Images Deep Dive

# Building Images Deep Dive

Up Next:

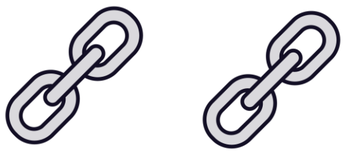
# Multi-stage Builds

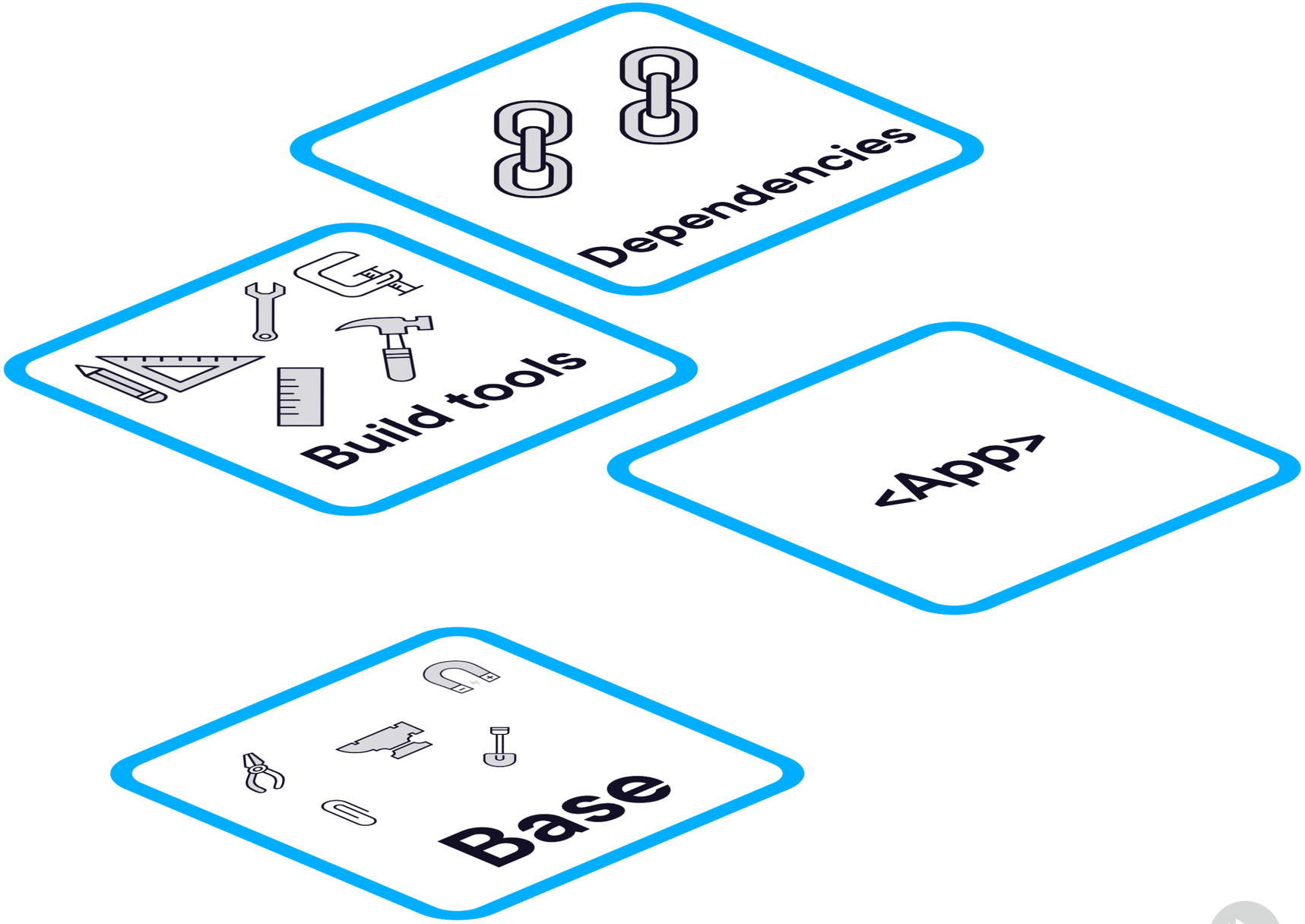
# Multi-stage Builds

# For images… smaller is better!

Tools, apps, packages

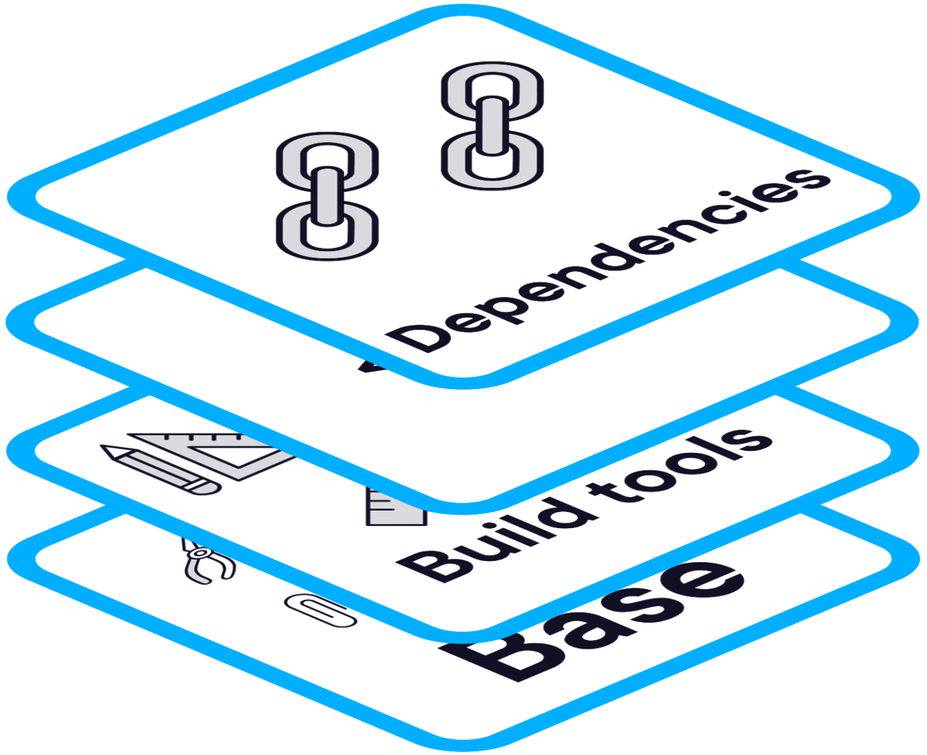
<app>





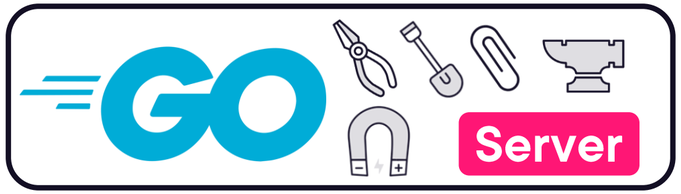
Tools, apps, packages

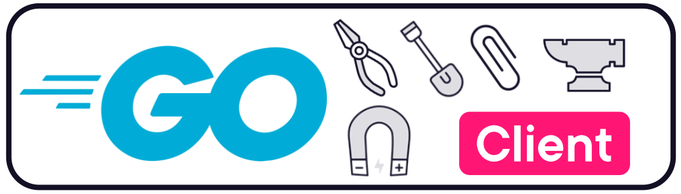
<app>

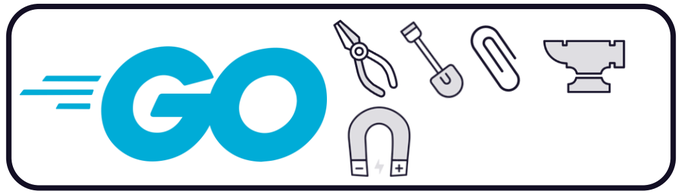



Ship to prod

SCRATCH
Client
Server


GO
Server


GO
Client

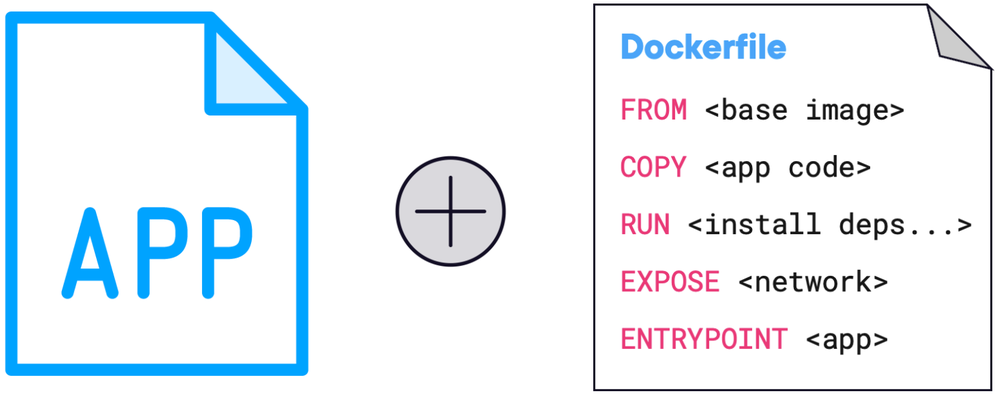

GO

Up Next:

# Recap

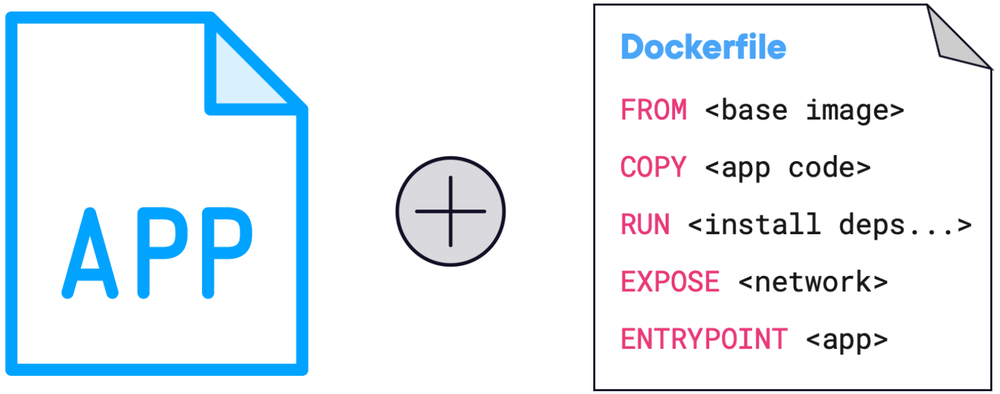
# Recap

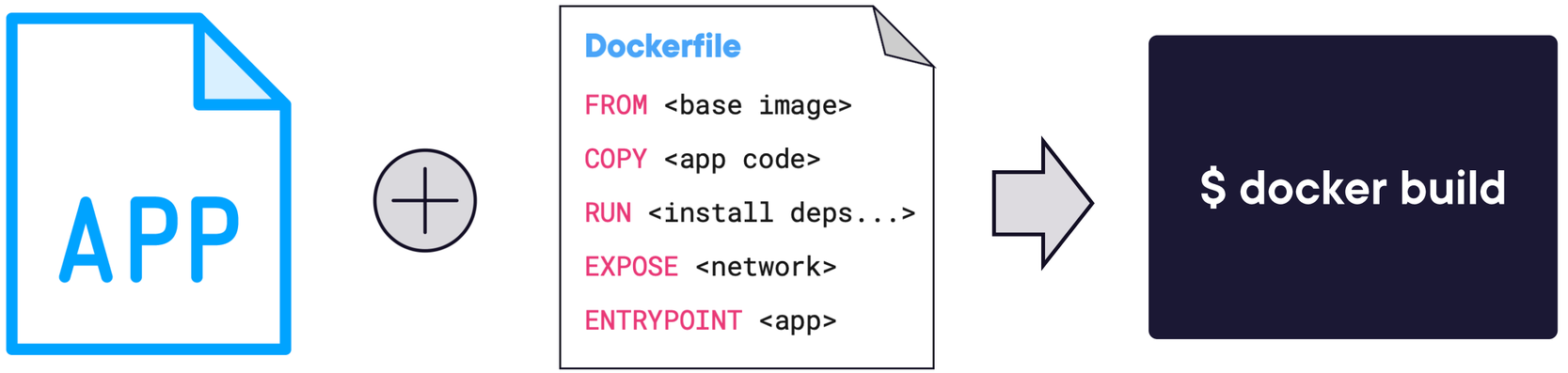
APP

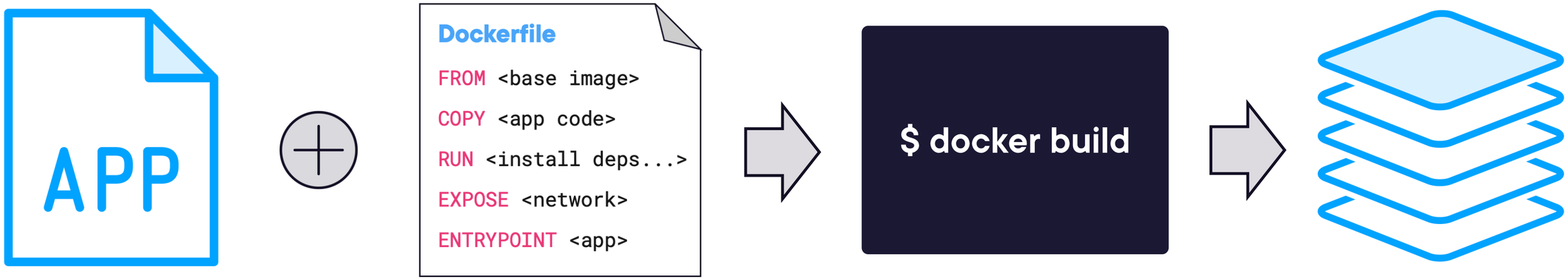

APP
Dockerfile
FROM <base image>
COPY <app code>
RUN <install deps...>
EXPOSE <network>
ENTRYPOINT <app>

**Dockerfile**

```
FROM <base image>

COPY <app code>

RUN <install deps...>

EXPOSE <network>

ENTRYPOINT <app>
```


APP
Dockerfile
FROM <base image>
COPY <app code>
RUN <install deps...>
EXPOSE <network>
ENTRYPOINT <app>


APP
Dockerfile
FROM <base image>
COPY <app code>
RUN <install deps...>
EXPOSE <network>
ENTRYPOINT <app>
$ docker build


APP
Dockerfile
FROM <base image>
COPY <app code>
RUN <install deps...>
EXPOSE <network>
ENTRYPOINT <app>
$ docker build

# Multi-stage builds help us create small images suitable for production.

Up Next:

# Working With Containers