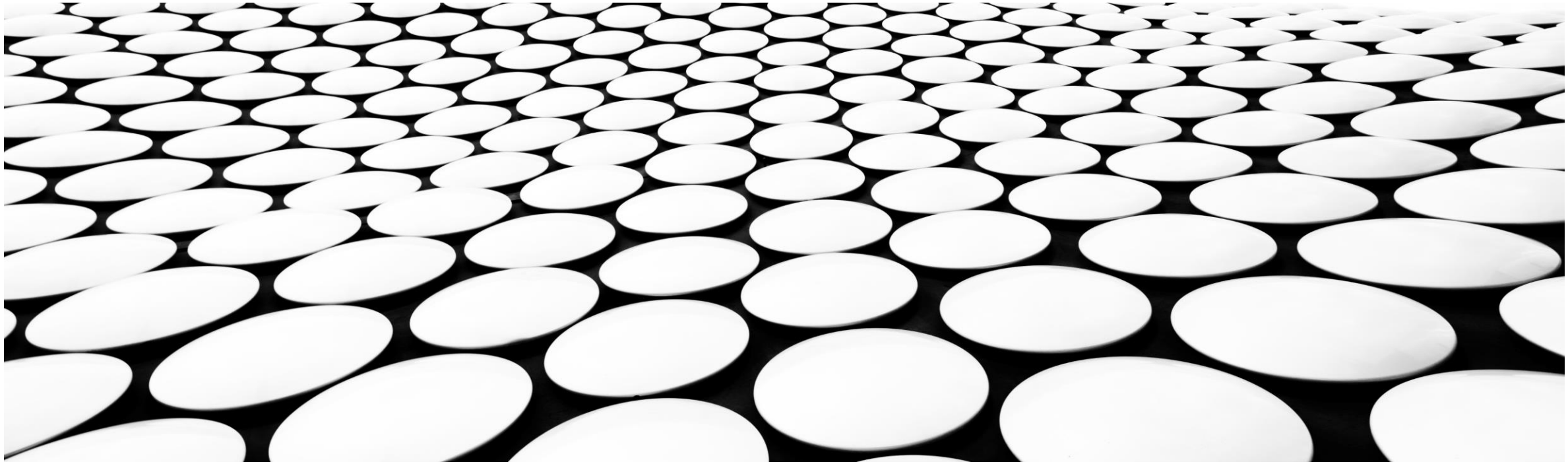
DOCKER AND KUBERNETES

DHANANJAYAN

20TH JULY – 24TH JULY 2020



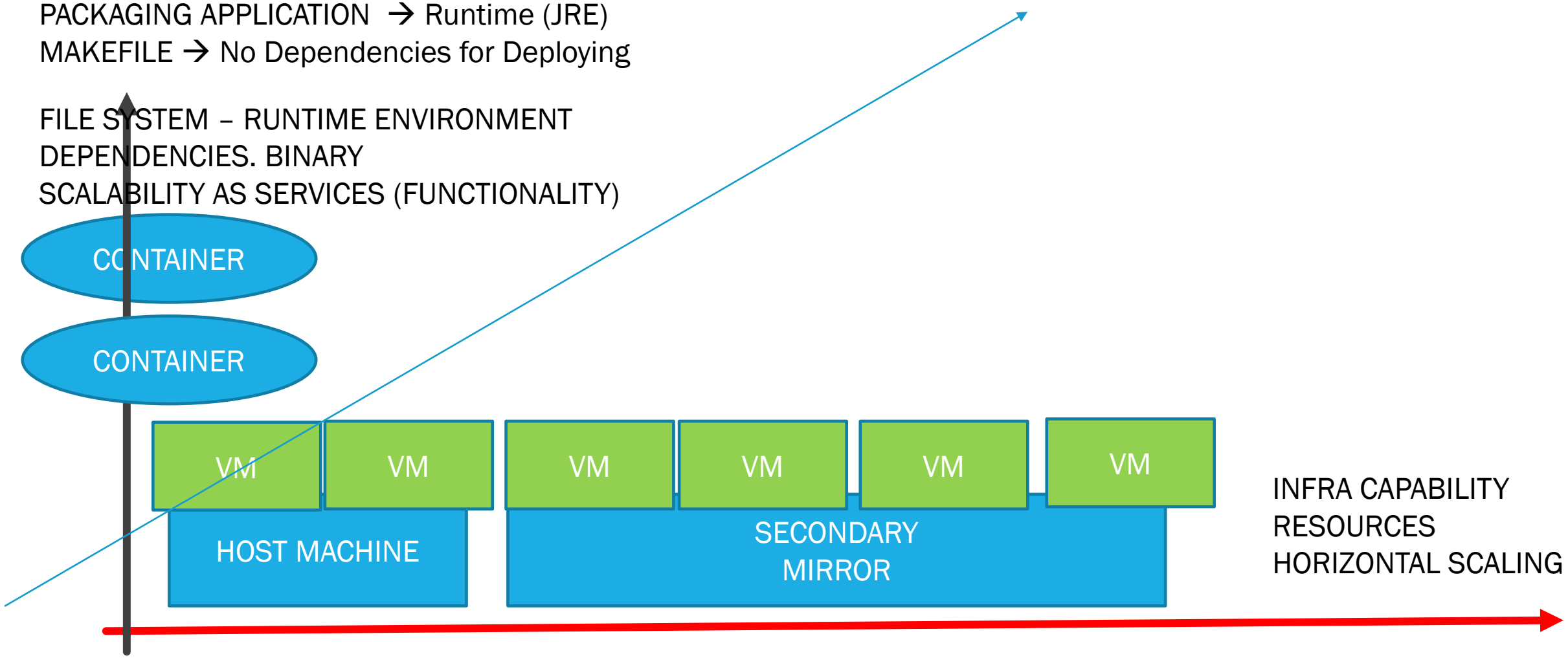
INFRA PRE-REQUISITE

- NO VPN
- ORACLE VIRTUALBOX (www.virtualbox.org) → 6.0 or above
- 16 GB RAM , 20 GB FREE HDD
- WIN 10 → VIRTUALIZATION → ENABLED
- Uninstall Docker Desktop for Windows (Restart) (.docker) → Desktop.docker
- Windows – Turn Windows Features ON /OFF – HYPERV (Unchecked)
- Hub.docker.com
- Github.com
- App.slack.com

SCALING CUBE

PACKAGING APPLICATION → Runtime (JRE)
MAKEFILE → No Dependencies for Deploying

FILE SYSTEM – RUNTIME ENVIRONMENT
DEPENDENCIES. BINARY
SCALABILITY AS SERVICES (FUNCTIONALITY)

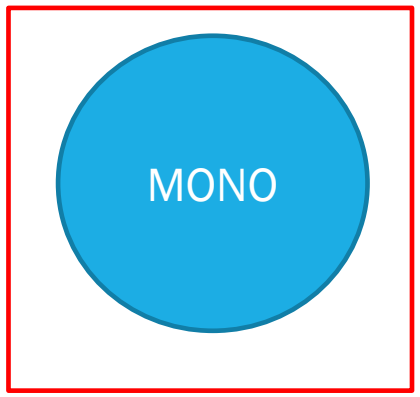


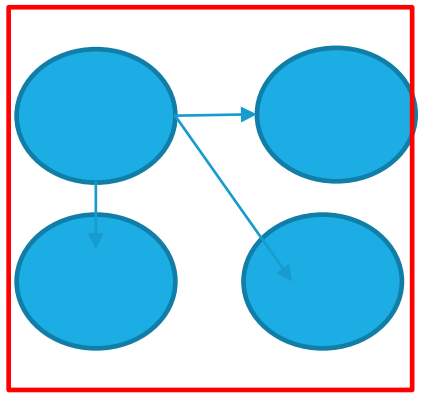
CLOUD NATIVE APPLICATIONS (CNCF)

- AS MANY ROLLOUTS POSSIBLE
  - AUTOMATIONS ON CHANGE, MONITORING , SUPPORT
- AGILE (DEV+OPS)
- MINIMAL DOWNTIME – VERY HIGH AVAILABILITY (SERVICES, INFRASTRUCTURE)
  - SMALL FUNCIONALITY BLOCKS → MICROSERVICES
  - DEPLOY SERVICES AS CONTAINERS
- NO COMPROMISE ON PERFORMANCE OF THE APPLICATION
- AT AN OPTIMUM COST (VIRTUALIZATION , CLOUD AND VMS)

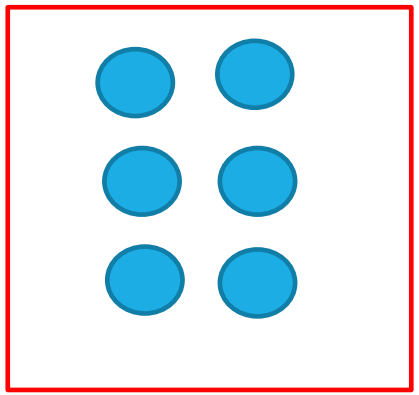
EVOLUTION

APP DELIVERY	APP DESIGN	APP DEPLOYMENT	APP INTEGRATION
Iterative	Monolothic	Host (Bare Metal)	Data Center
Agile (Dev, QA)	N tier Layer (SOA)	VM	On Premises
DevOps (Dev+Ops) Adapting to Changes Many Rollouts	Microservices (MSA)	Containers → Serverless Computing	Cloud (OCI)





Coarse Grained - SOA

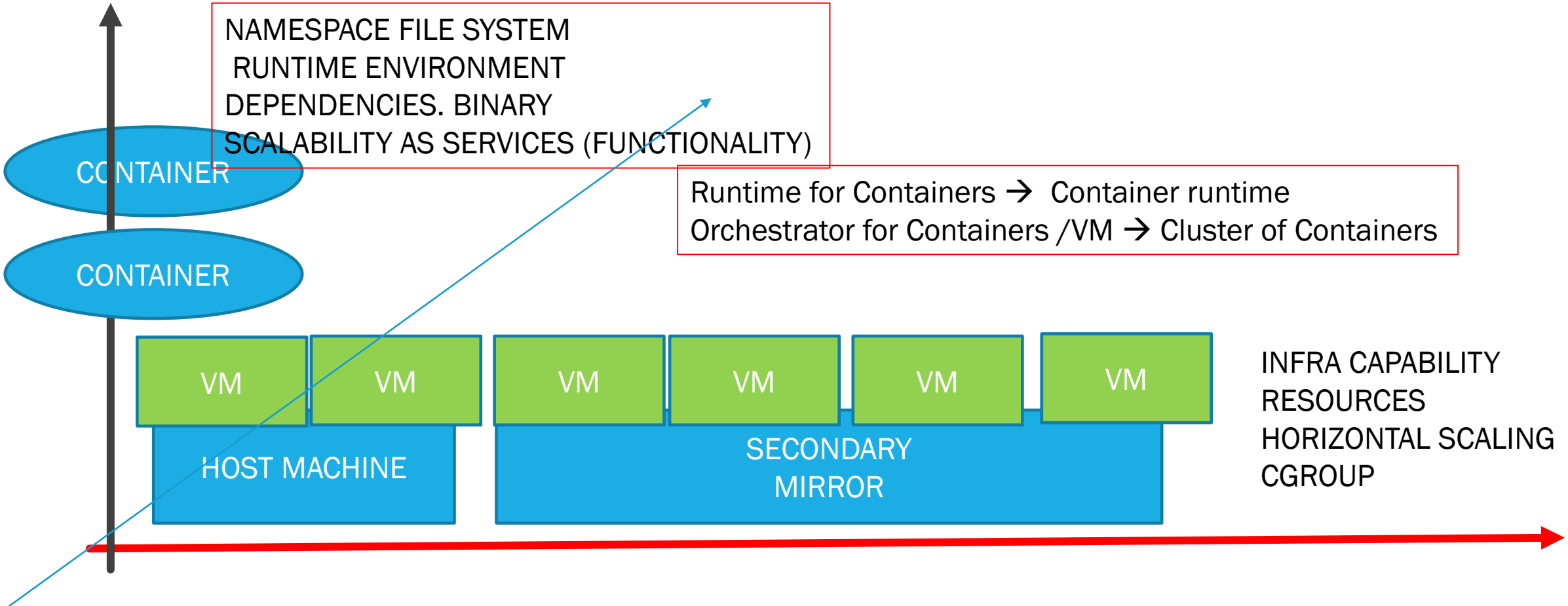


FineGrained - MSA

Containers/Machines
Are monitored 24 x7
Log, Monitor, Diagonose , Repaired

SCALING CUBE

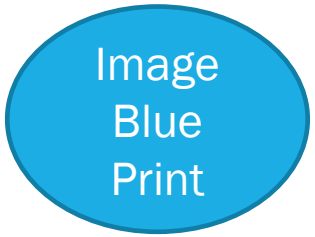
PACKAGING APPLICATION → Runtime (JRE)
MAKEFILE → No Dependencies for Deploying



VM VS CONTAINER

VM	Container
Machine (Limited to Infra) – Horizontal Scaling	Processes as Services (Functional Capability) – Vertical Scaling
Image associated with VM – File (ISO)	Blue print – Image (Filesystem, Dependencies..BootStrapper) Develop once Deploy Anywhere Portable OBJECT
Resources – Fixed (Memory, Disk) I/O (Read/write)→CPU, Networks (Static Resource management)	Resources are dynamic - Fluctuate
Boot time to start up the machine	Customize my Boot Strapper (main ()) Lazy Loading

PACKAGING APPLICATION → Runtime (JRE)
MAKEFILE → No Dependencies for Deploying

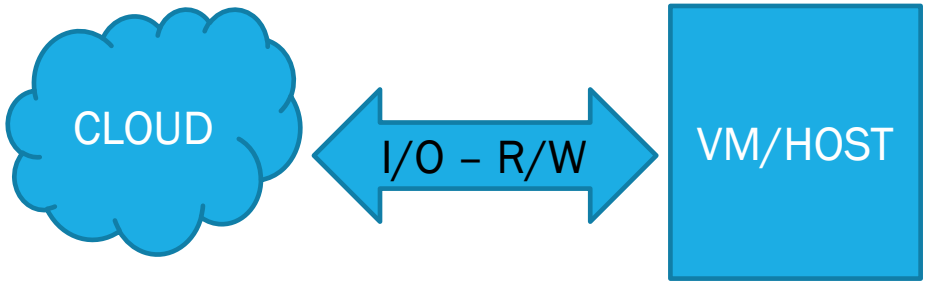


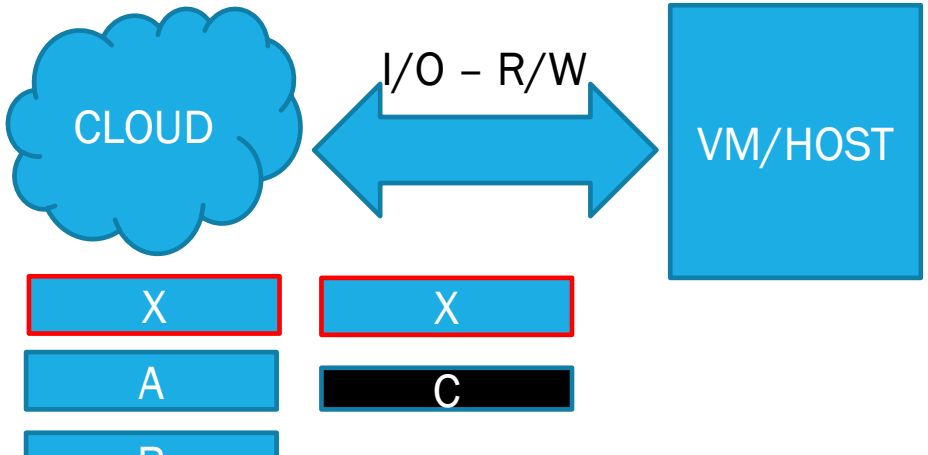
IMAGE

Design Architecture – MSA or MONO or SOA
Packaging Application

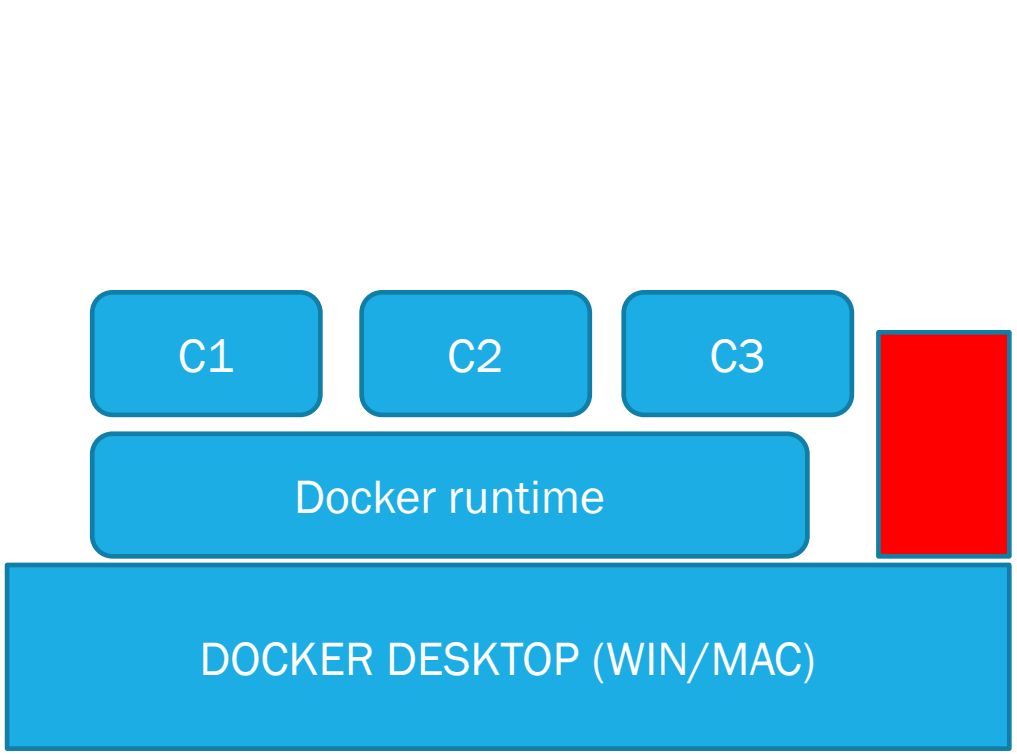
VM IMAGE	CONTAINER IMAGE
FILE	OBJECT
REPOSITORY (SCM) Source code →	IMAGE REPOSITORY hub.docker.com /Cloud Image Repository OCIR – Docker images in OCI Cloud Container Image – SHA256
OEL 7.5 → 4.3 GB OEL 7.6 → 4.45 GB (FULL OPERATION)	Image is made up of one or more layers Facilitate Incremental Update

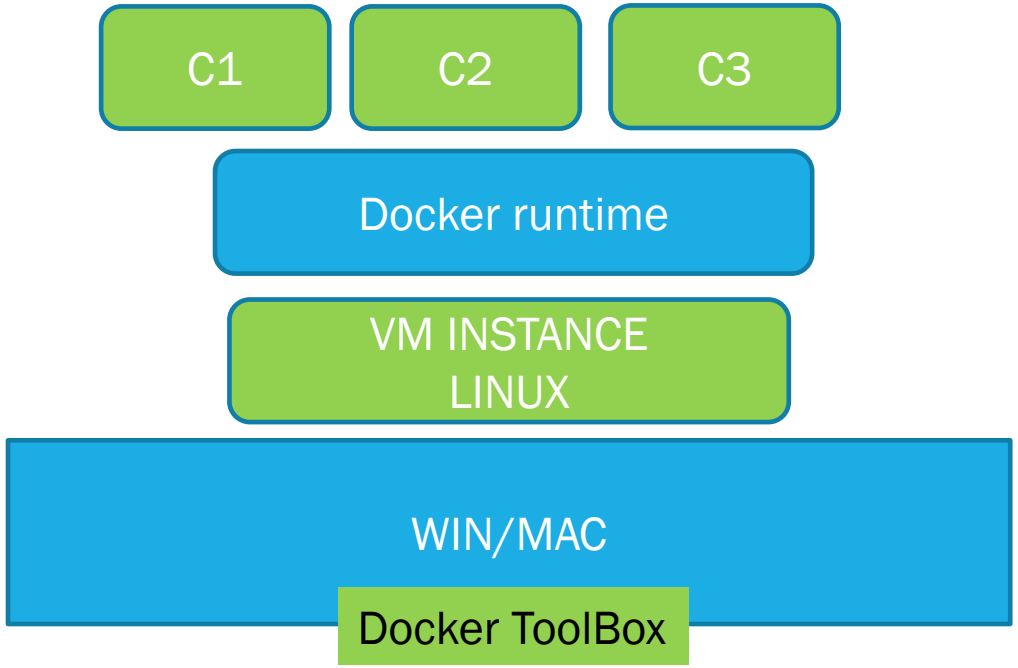
Latency, Resources , Performance



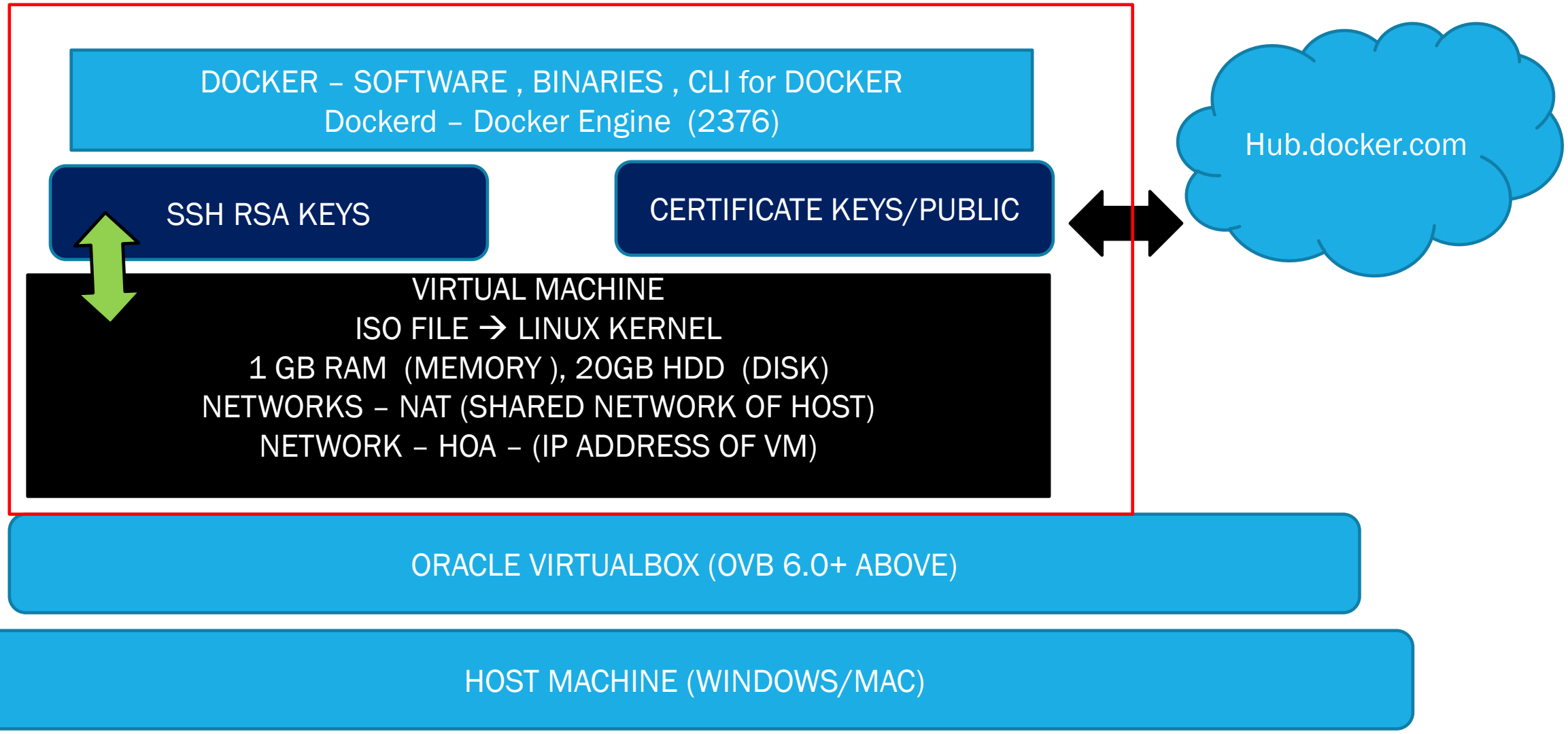


PREFERRED INSTALLATION – DOCKER TOOLBOX



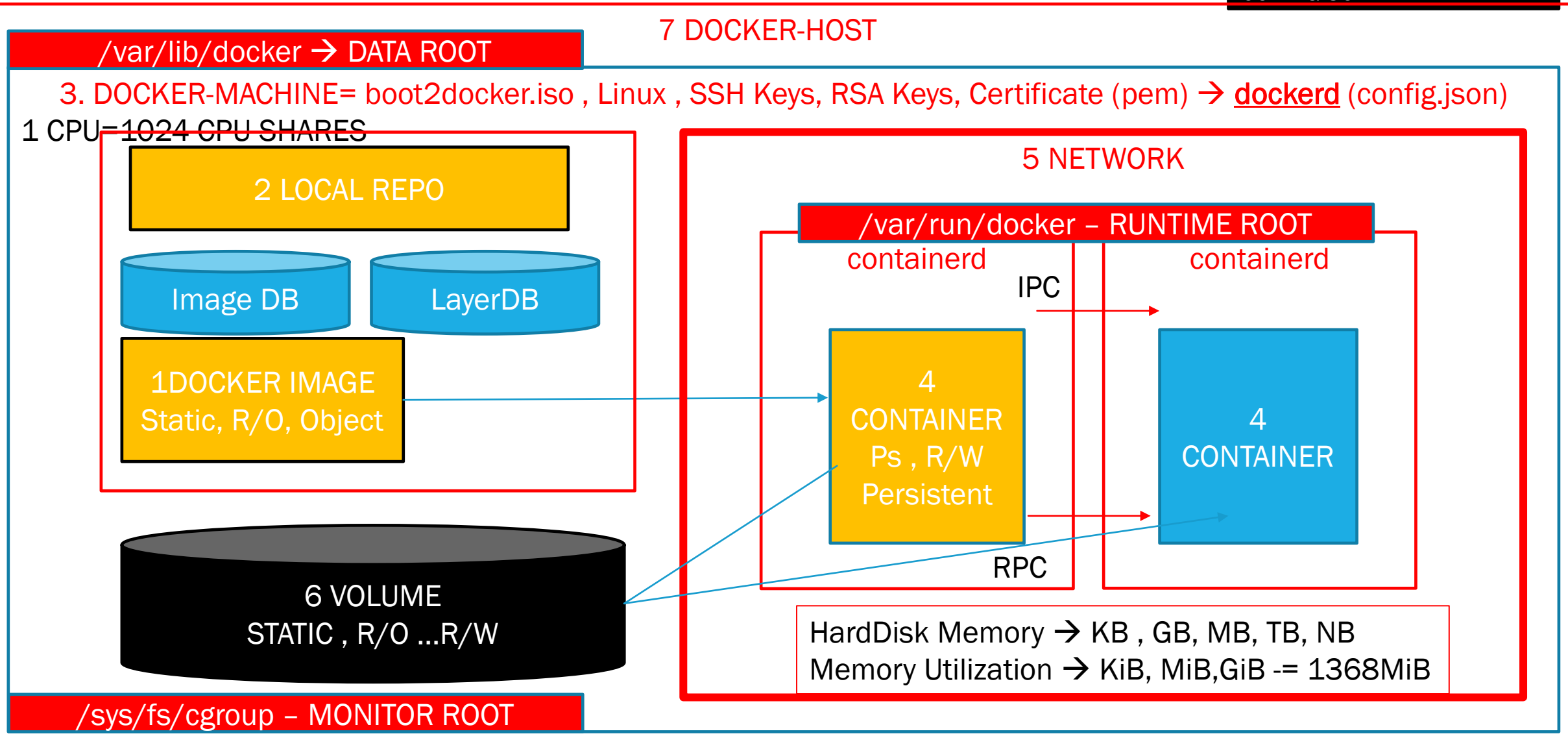


DOCKER IMPLEMENTATION



DOCKER ARCHITECTURE
7 OBJECTS , 8 FILES , 9 IMPLEMENTATION KERNELS

INSTALL DOCKER:
STORAGE DRIVER: BTRFS++
OBJECTS PROPERTY: JSON
LAYERS IMAGE : SHA256
CONFIG SCRIPT : YAML

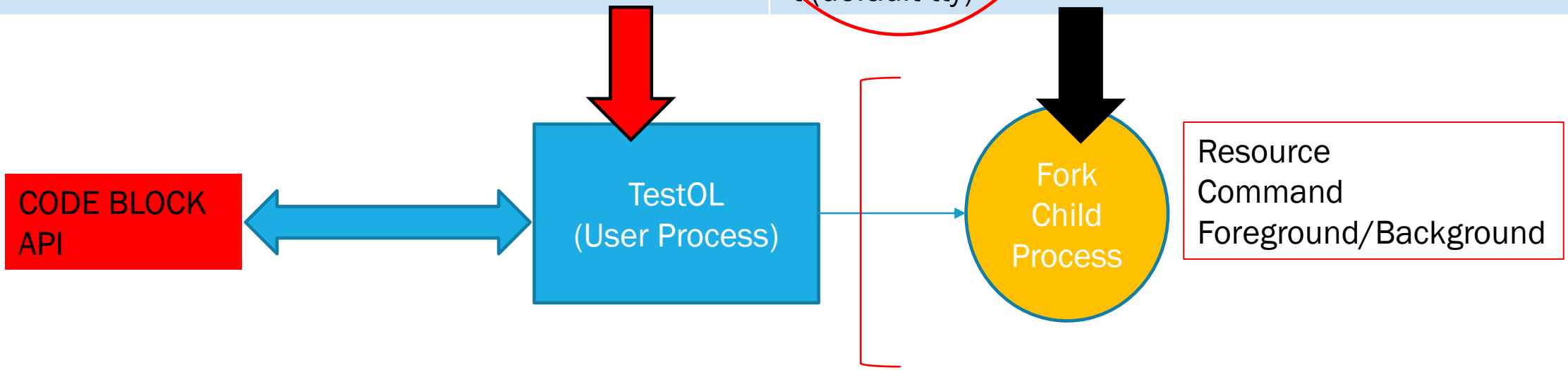


WHAT TO LOOK FOR IN A DOCKER IMAGE ?

- Digest /Created timestamp
- Parent
- ContainerConfig
  Config
- Exposed Ports – Service API
- Environmental (Env)
- Cmd → Default Command when container starts (main ())
  Infinite Loop , Daemon Thread ,
- OS /Architecture
- Layers
- Meta Data

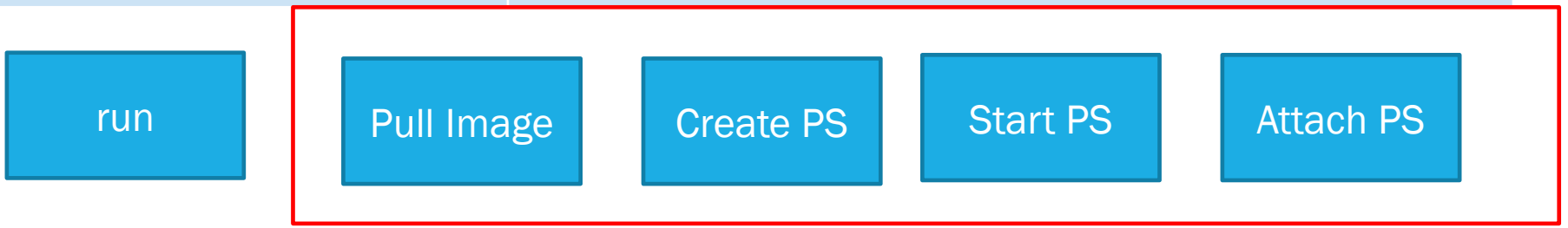
PROCESS IN CMD (IMAGE)

SERVER PROCESS	USER Process
Listening Process, Daemon Process	Interactive Process , Stdin , Stdout , Stderror
Communicate to Server Process → Connection (Listening for a Port)	Input , Devices , File System Shell – Interactive process
Background Process -d (detached)	Foreground Process → Terminal - tty -I (interactive) -t (default tty)

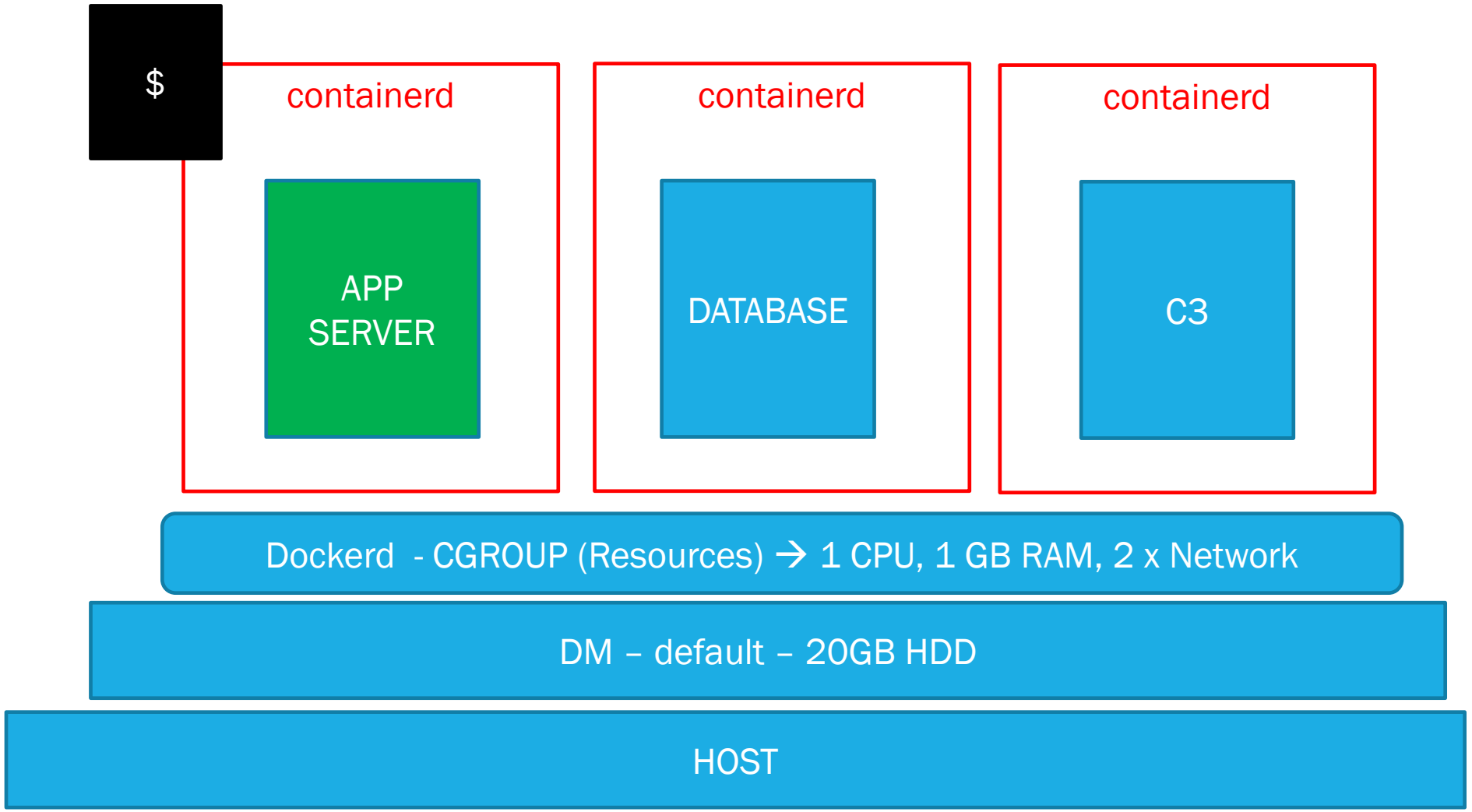


DIMENSION – WHEN PARENT/CHILD ?

PARENT PROCESS	CHILD PROCESS
Use Case: (AMC) Administer Resources Confgure Environments Manager Services /Process	Use Case: (LMR) Read Logs Monitor Resources Read a File Service/Report Service
Server Process → Connection (PORT)/API User Process → attach	Exec Shares resources of parent # docker exec <container> <Command> Specify whether process in Interactive mode or background.
# docker attach	# docker exec



ARCHITECTURE 2



SKETCH WORKS