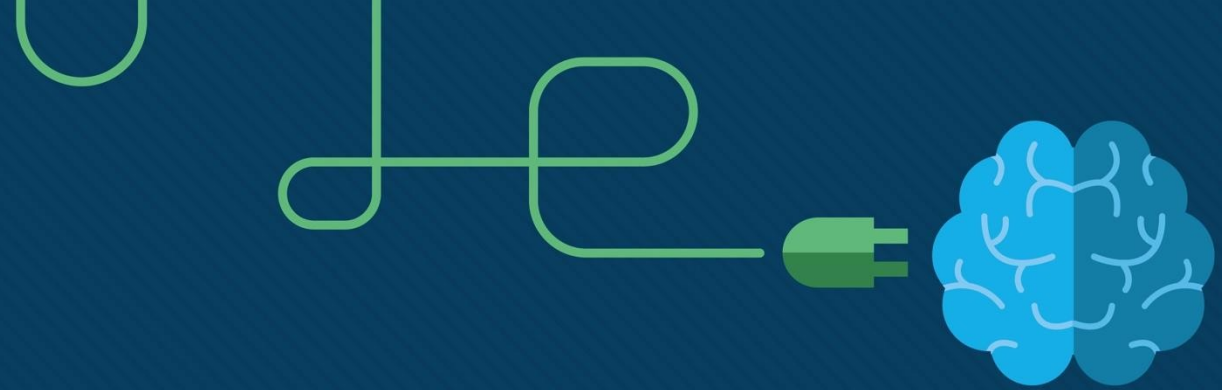


Cloud Computing

Tassadaq Hussain Cheema

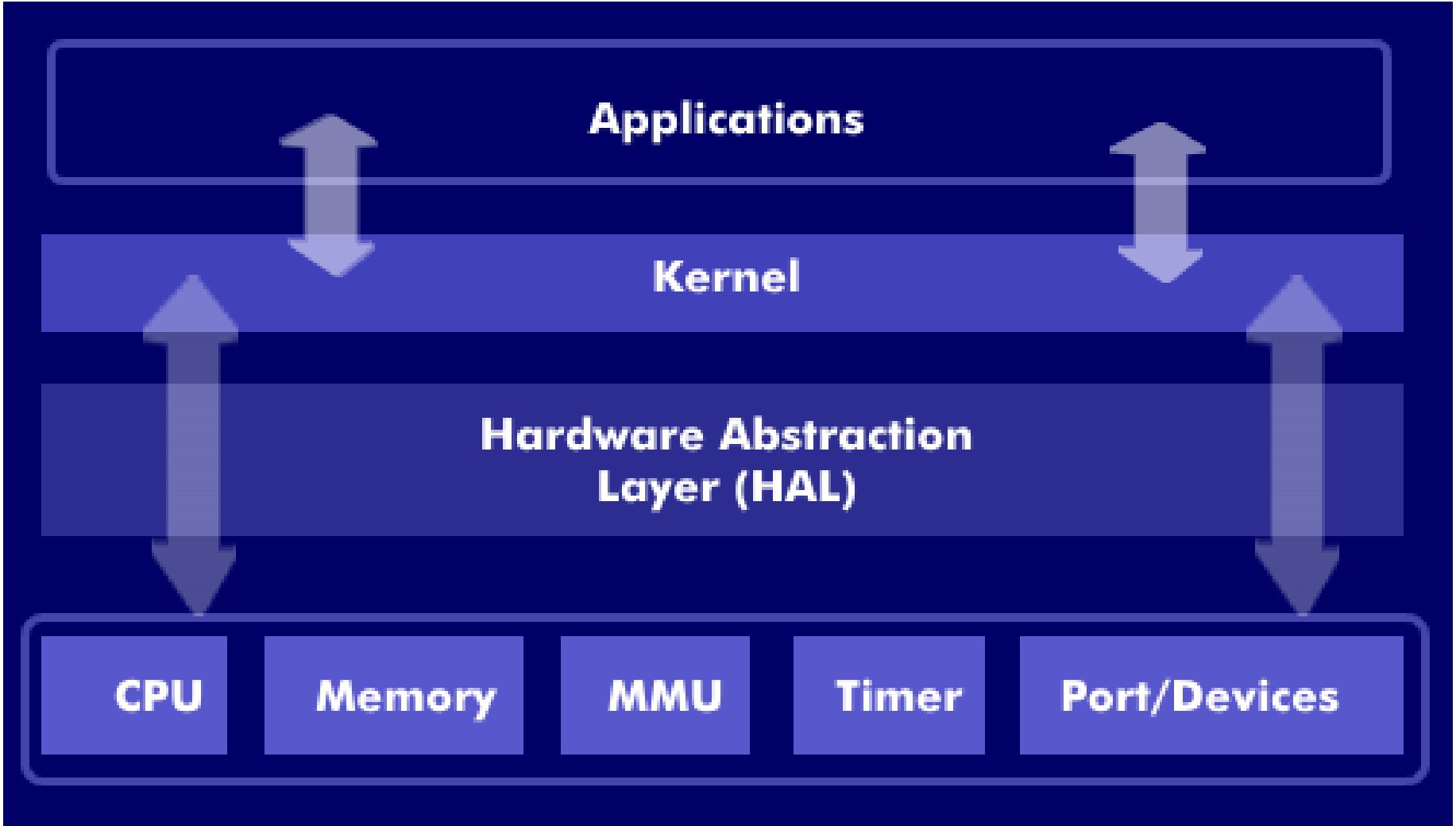
Professor EE Department NAMAL University

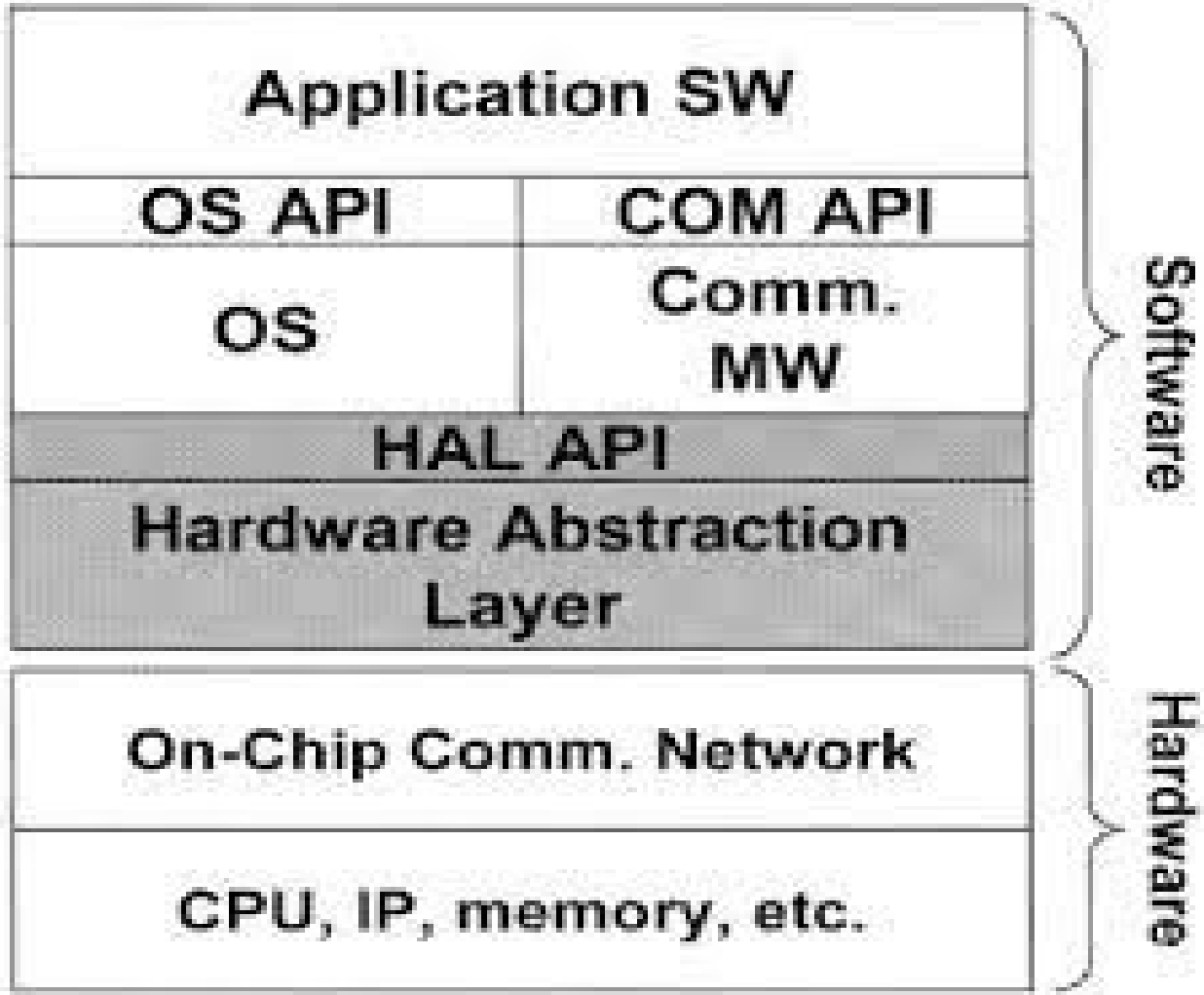
February 27, 2018

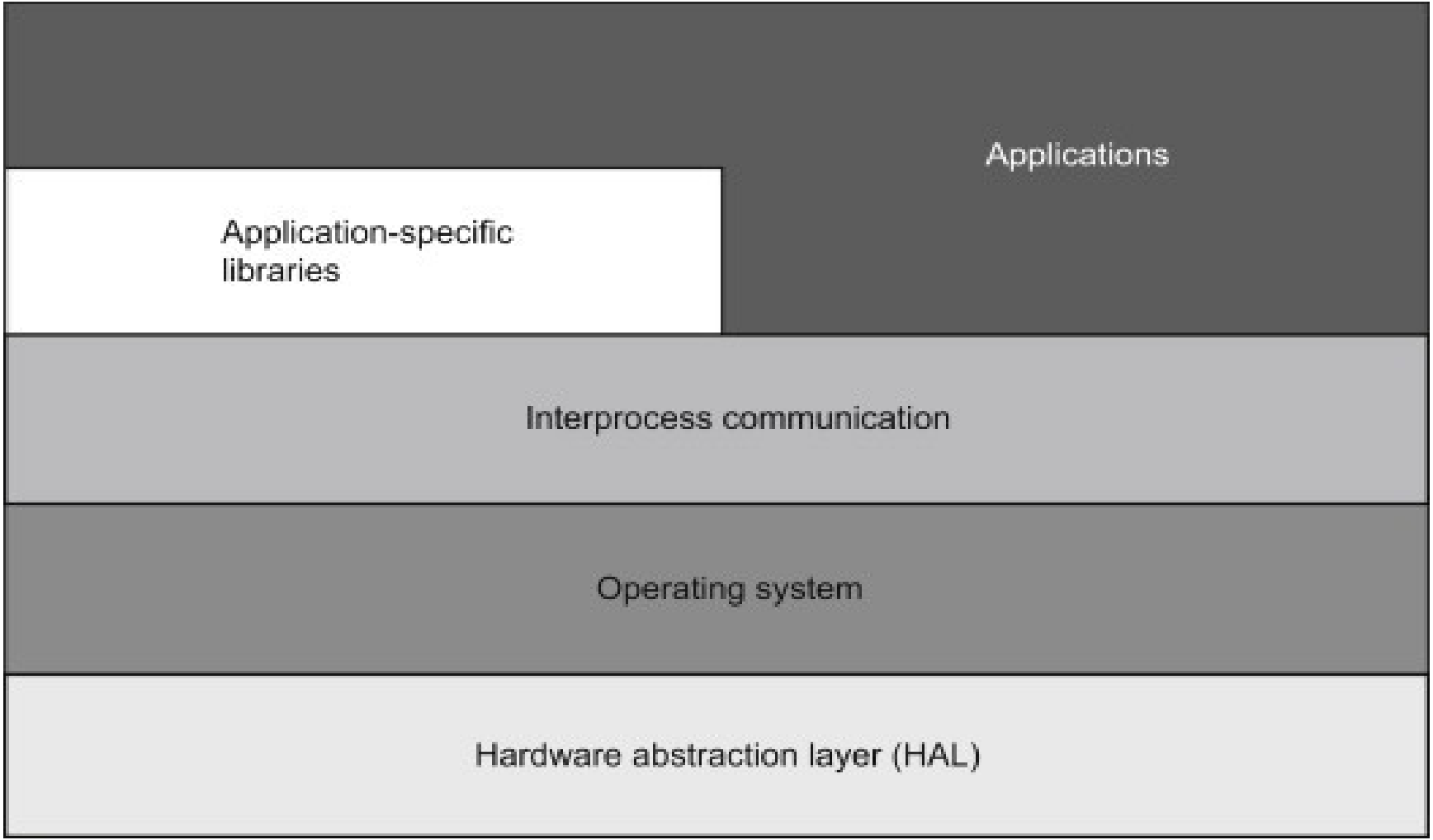
Cloud Computing

- Cloud computing refers to the delivery of computing services over the internet, allowing users to access and utilize a range of resources, including servers, storage, databases, software, and networking, without requiring on-site infrastructure or direct management of physical hardware. Instead of running applications or storing data on local devices or on-premises servers, cloud computing enables users to access and use computing resources remotely through a network connection.

- Infrastructure as a Service (IaaS): Provides virtualized computing resources such as virtual machines, storage, and networking. Users have more control over the underlying infrastructure and can deploy and manage their own software applications.
- Platform as a Service (PaaS): Offers a platform on which users can develop, run, and manage applications without the complexity of infrastructure management. The underlying infrastructure is abstracted, and users can focus on application development.
- Software as a Service (SaaS): Provides fully functional applications accessible over the internet. Users can use software applications hosted on the cloud without worrying about underlying infrastructure or maintenance.

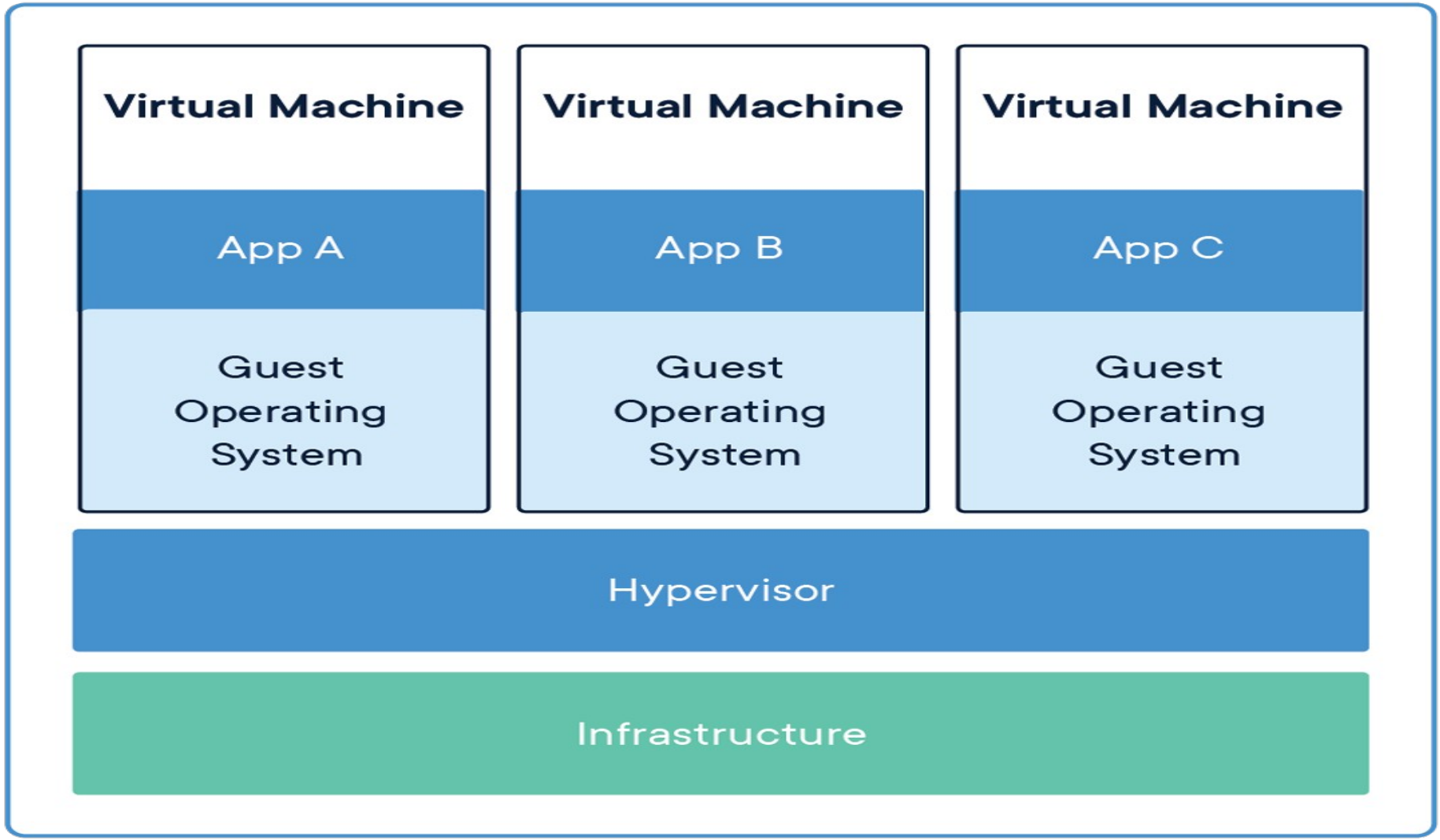

Applications
Kernel
Hardware Abstraction Layer (HAL)
CPU
Memory
MMU
Timer
Port/Devices


Application SW
OS API
COM API
OS
Comm. MW
HAL API
Hardware Abstraction Layer
On-Chip Comm. Network
CPU, IP, memory, etc.
Software
Hardware


Applications
Application-specific libraries
Interprocess communication
Operating system
Hardware abstraction layer (HAL)

VM: Virtual Machine

- The industry standard today is to use Virtual Machines (VMs) to run software applications. VMs run applications inside a guest Operating System, which runs on virtual hardware powered by the server’s host OS.
- VMs are great at providing full process isolation for applications: there are very few ways a problem in the host operating system can affect the software running in the guest operating system, and vice-versa. But this isolation comes at great cost — the computational overhead spent virtualizing hardware for a guest OS to use is substantial.

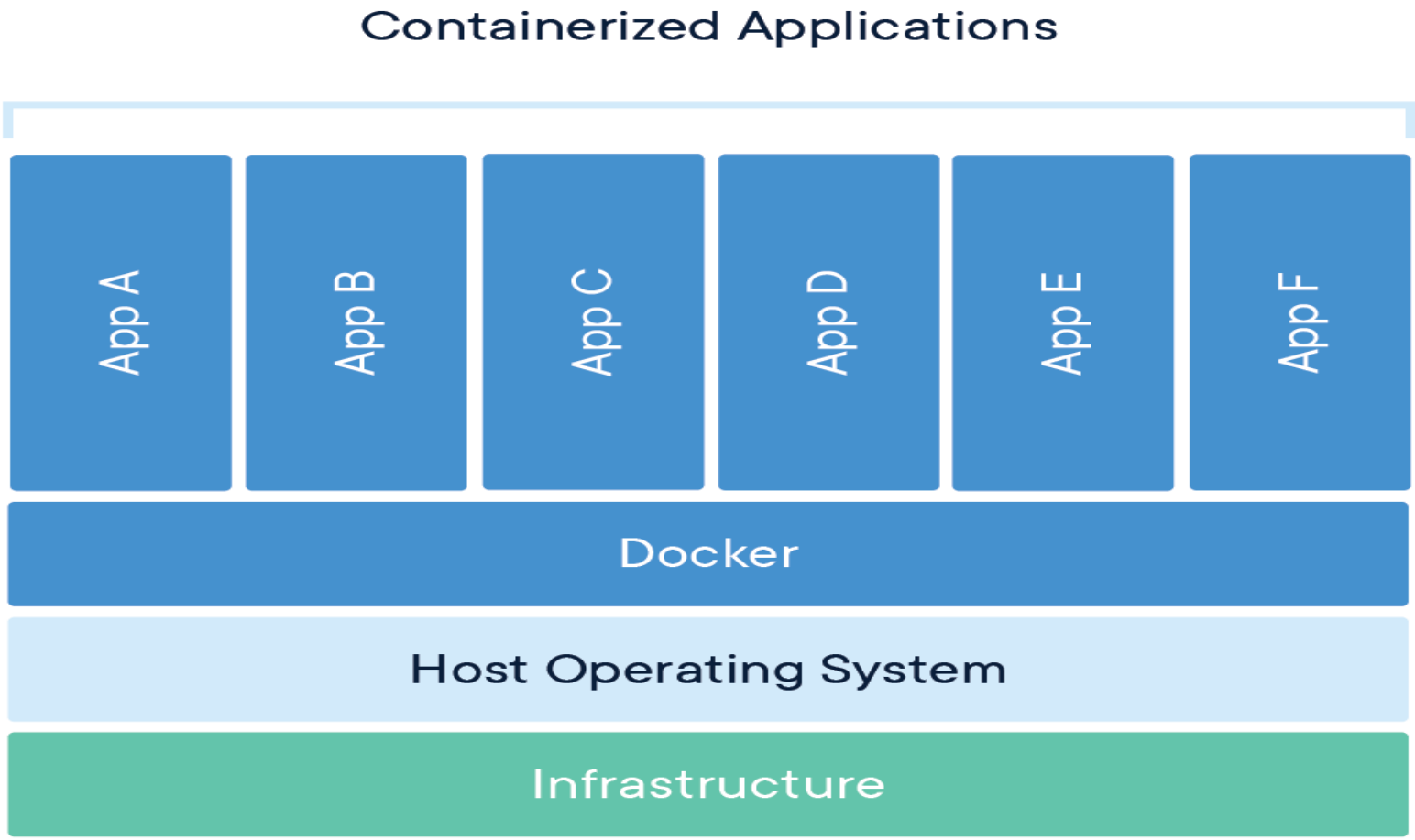

Virtual Machine
App A
Guest Operating System
Virtual Machine
App B
Guest Operating System
Virtual Machine
App C
Guest Operating System
Hypervisor
Infrastructure

Docker

- Docker is a tool that allows developers, sys-admins etc. to easily deploy their applications in a sandbox (called containers) to run on the host operating system i.e. Linux. The key benefit of Docker is that it allows users to package an application with all of its dependencies into a standardized unit for software development. Unlike virtual machines, containers do not have high overhead and hence enable more efficient usage of the underlying system and resources.

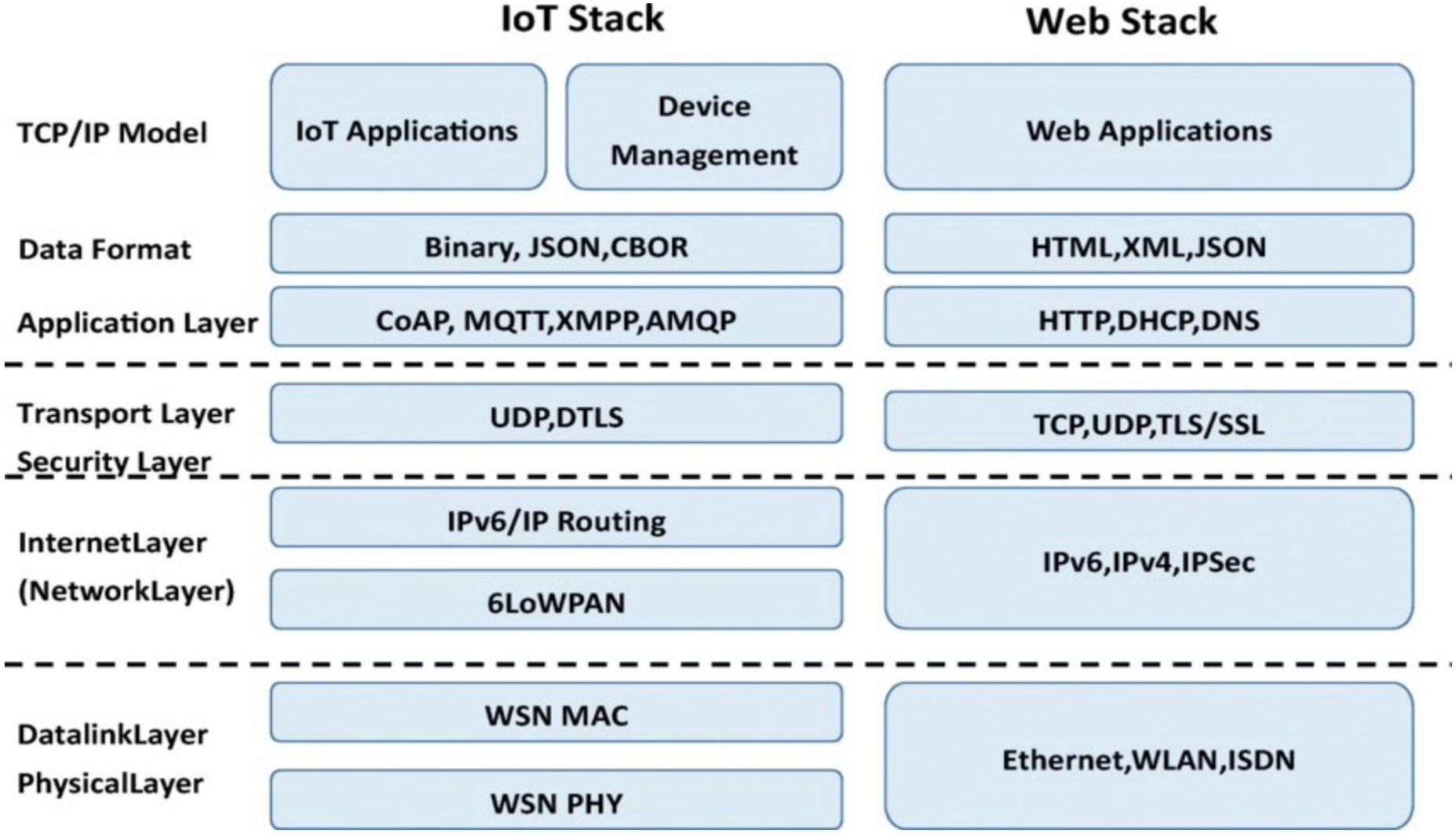
Container

- Containers offer a logical packaging mechanism in which applications can be abstracted from the environment in which they actually run. This decoupling allows container-based applications to be deployed easily and consistently, regardless of whether the target environment is a private data center, the public cloud, or even a developer’s personal laptop. This gives developers the ability to create predictable environments that are isolated from the rest of the applications and can be run anywhere.


Containerized Applications
App A
App B
App C
App D
App E
App F
Docker
Host Operating System
Infrastructure

Containers

- Containers are self-contained, isolated environments that include all the necessary software, libraries, and dependencies required to run an application. Containers are created from container images, which are read-only templates that define the application's file system, runtime environment, and configurations.
- Container Images: Container images are the building blocks of containers. They are created using specifications known as Dockerfiles, which define the steps needed to build the image. Dockerfiles contain instructions to install dependencies, copy application code, set environment variables, and configure the container. Images can be stored in registries, such as Docker Hub or private repositories, for easy distribution and sharing.


IoT Stack
Web Stack
TCP/IP Model
IoT Applications
Device Management
Web Applications
Data Format
Binary, JSON,CBOR
HTML,XML,JSON
Application Layer
CoAP, MQTT,XMPP,AMQP
HTTP,DHCP,DNS
Transport Layer
Security Layer
UDP,DTLS
TCP,UDP,TLS/SSL
InternetLayer
(NetworkLayer)
IPv6/IP Routing
6LoWPAN
IPv6,IPv4,IPSec
DatalinkLayer
PhysicalLayer
WSN MAC
WSN PHY
Ethernet,WLAN,ISDN

- https://learnembeddedsystems.co.uk/easy-raspberry-pi-iot-server