Technology Architecture

CS 752 Software Architecture and Design Practices

Prof. Chandrashekar R

Ref: Software Architecture Patterns by Mark Richards

Outline

- Purpose
- Components and Elements of Technology Architecture
- Software Architecture Styles
- Deep Dive – Layered Architecture

Technology Architecture

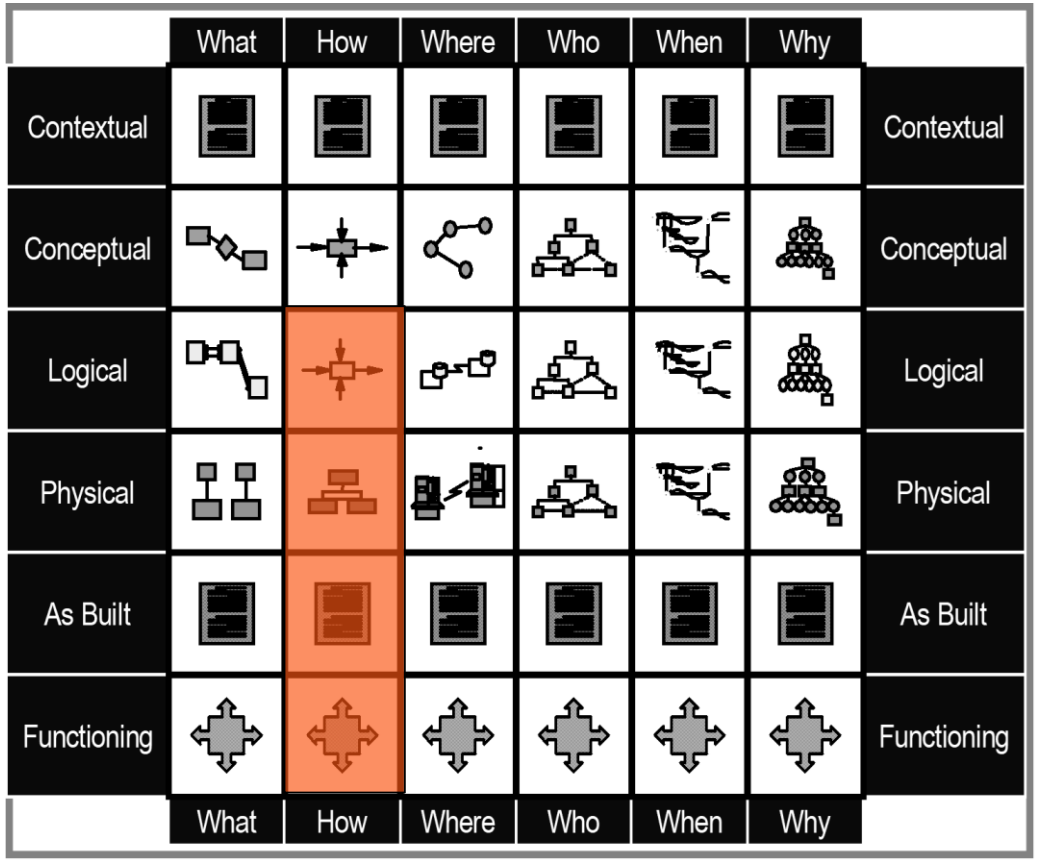
- The purpose of Technology Architecture is to specify a structure comprised of **hardware** and **software** for implementing an IT Solution

Same as Technical Architecture?

Contrast with Functional Architecture

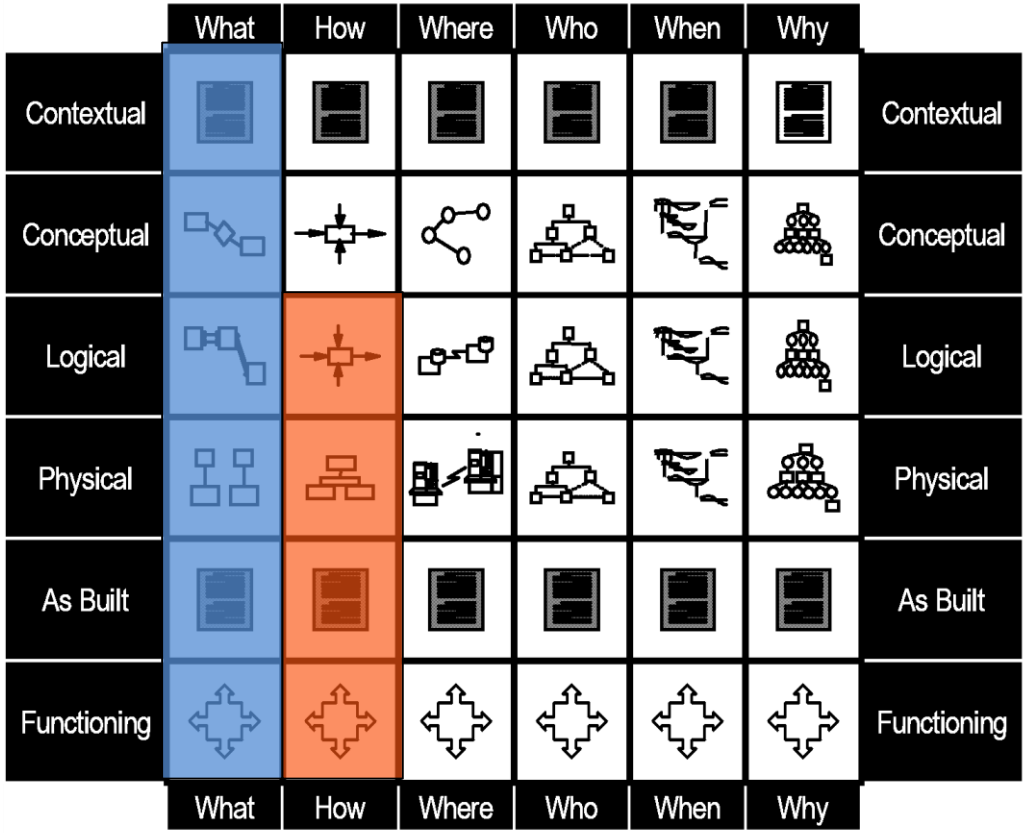
Relation with other architectures

- With Zachman Architecture Framework
  - Addresses the “How” column from the “Designer”/”Logical” row downwards



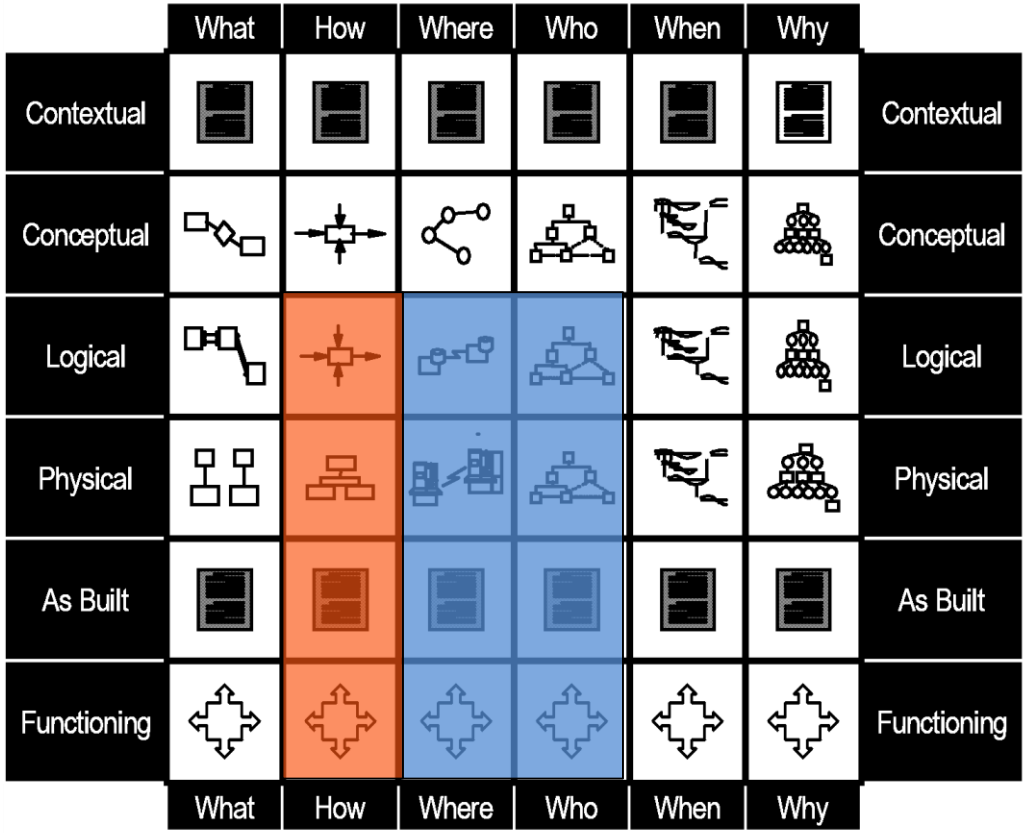
Relation with other architectures

- With Functional Architecture
  - The "WHAT" of Functional Architecture drives the specifics of "HOW" of Technical Architecture



Relation with other architectures

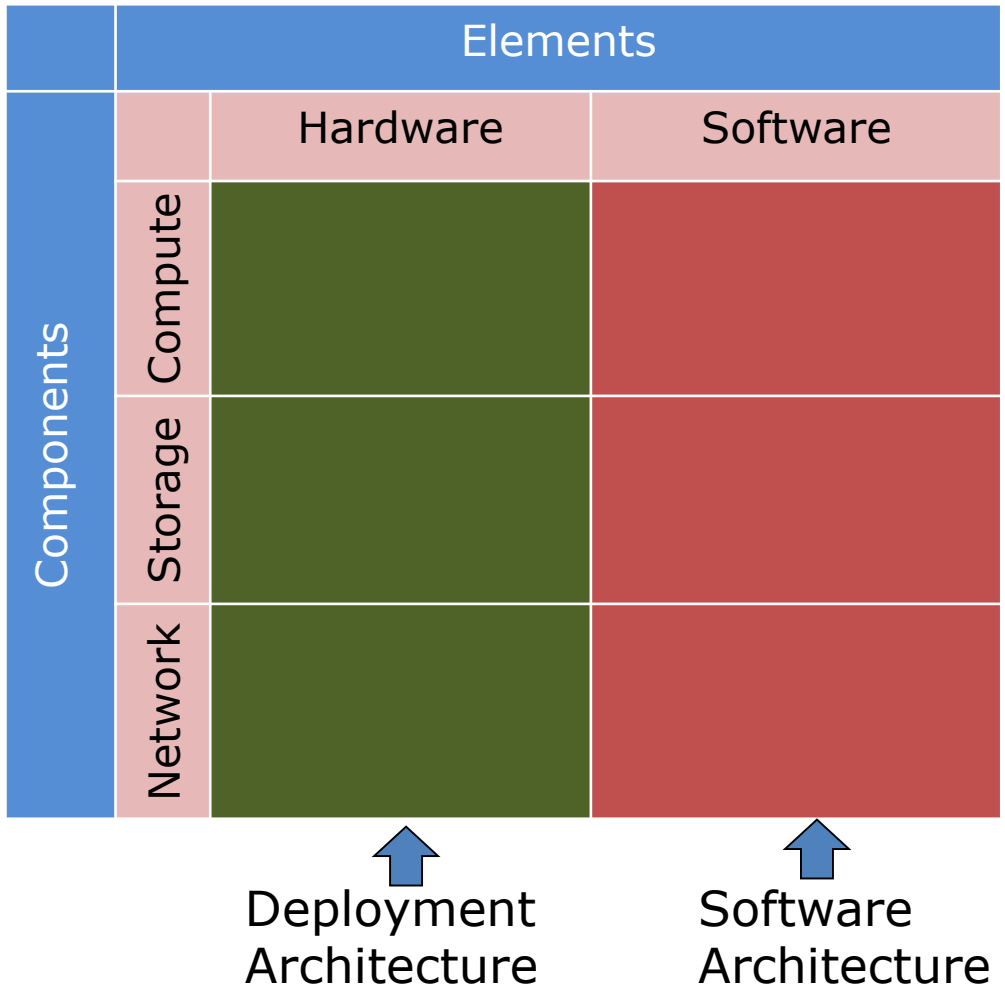
- With Deployment Architecture
  - The "HOW" of Technical Architecture drives the specifics of "WHERE" and "WHO" of Deployment Architecture



Technology Architecture Matrix

- Elements
  - Hardware
  - Software
- Components
  - Compute
  - Storage
  - Network

Technology Architecture Matrix



Components

Compute

How does the processing take place?

Storage

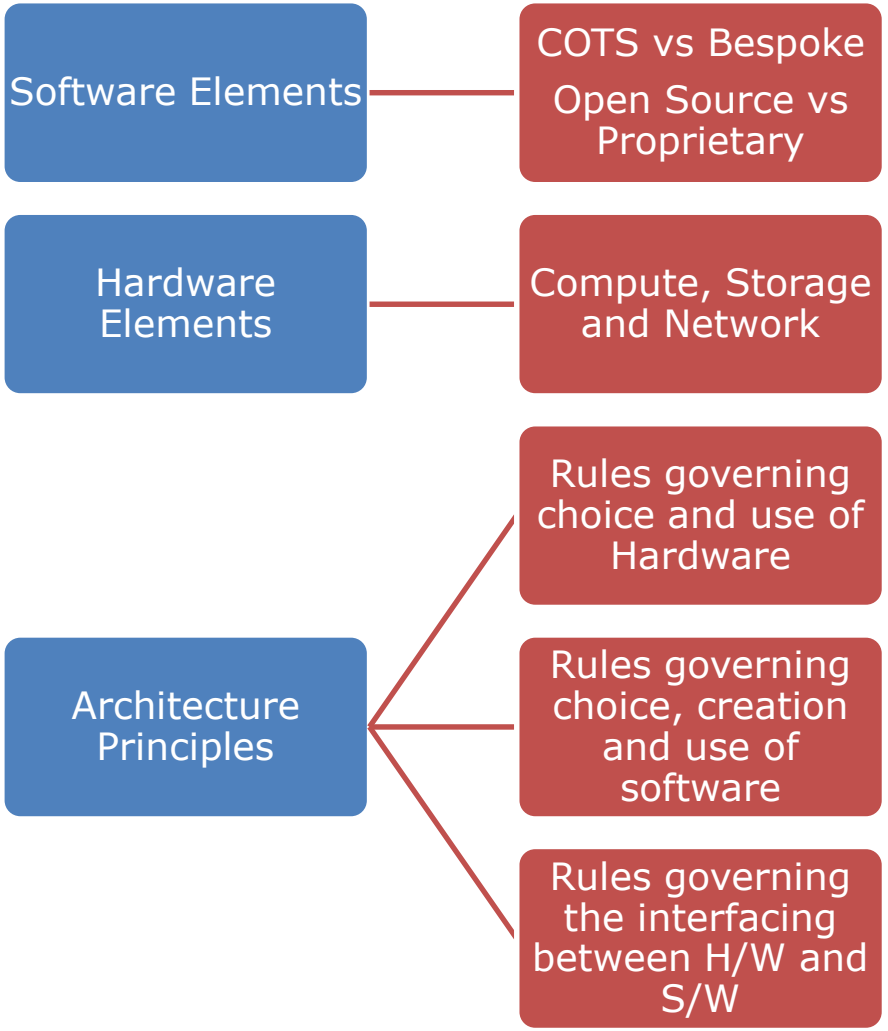
How is information persistence managed?

Network

How is the communication with users established

How does this compare with Von Neuman Architecture

Elements


Software Elements
COTS vs Bespoke
Open Source vs Proprietary
Hardware Elements
Compute, Storage and Network
Architecture Principles
Rules governing choice and use of Hardware
Rules governing choice, creation and use of software
Rules governing the interfacing between H/W and S/W

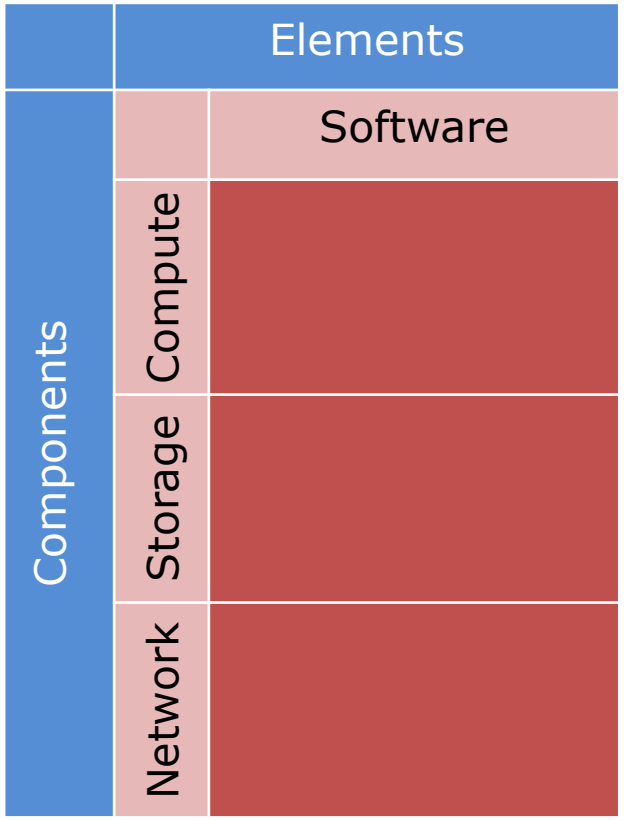
Example Elements and Components

		Elements	
		Hardware	Software
Components	Compute	• Web server • Database server	• Operating Systems • Web Server • Application Server • DBMS
	Storage	• SAN • NAS • SATA	• Storage O/S • Backup Software
	Network	• Routers • Switches • Cabling media	• Network O/S • Firewall • Proxy servers • Load Balancer

SOFTWARE ARCHITECTURE

Software Architecture

- Software Architecture is concerned with all necessary computing needs of a given IT solution
- It includes the software needed to provide compute, storage and networking capabilities



Software Architecture Styles

- A style specifies a consistent mechanism for designing software
- A given IT solution may choose a mixture of styles to suit the specific needs of the solution
- A given Application in an IT solution generally is based on only one style

Principle of Cohesion

Low

Degree to which different components of **WITHIN** a layer are focused on a single objective

Cohesion Characteristics

- Advantages of High Cohesion
  - Improves consistency
  - Increase reuse
  - Plug-and-play
  - Engineering eases (development and testing)
  - Localization of expertise

Binding forces of cohesion

Logical	Credit card payment + NEFT payment
Temporal	Init of DB + Init of Web server + Init O/S
Procedural	Loan verification + Loan evaluation + Loan underwriting
I/O oriented	PO printing + PO saving + PO mailing
Sequential	Search → Select → Add → Checkout
Functional	Self-contained (search to payment)

Separation of Concerns

- Governs relationships **ACROSS** layers
- Minimize overlap of responsibilities across layers
- Responsibilities are well-defined
- Advantages
  - Avoid duplication
  - Better maintainability
  - Reduces coupling / dependencies across layers

Architecture Styles to be Covered

- Layered
- Event-Driven
- Microkernel
- Micro-services

DEPLOYMENT ARCHITECTURE

Deployment

What?

- Deployment is the process implementing an IT solution comprised of all necessary hardware and software

Where?

- On-Premise
- Cloud

Who?

- Sys Admin, Hardware Specialists, Network Admins, etc.

When?

- Once a "ready-to-use" product is available

How?

- Manual
- Automated

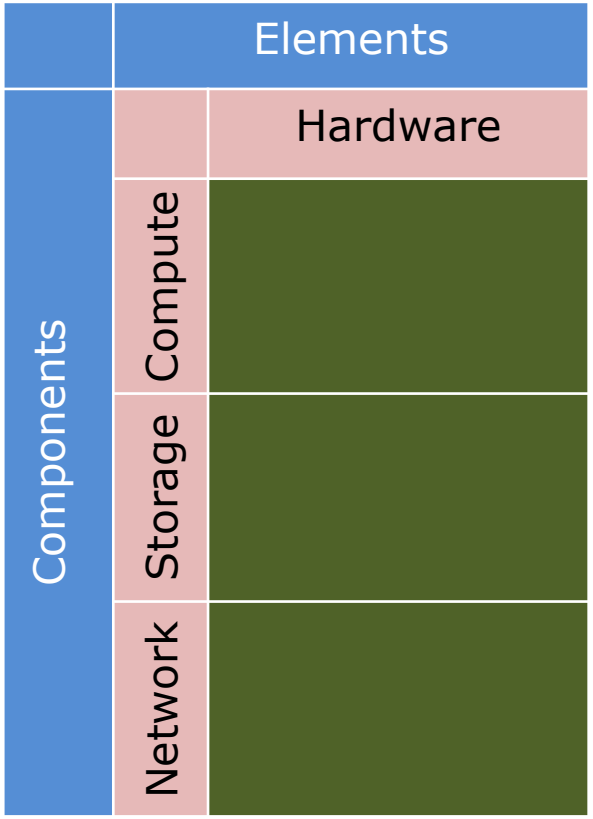
Why

- Easy of use
- Better agility
- Cost efficiency

Ready-to-use vs Finished?

WHAT – Components of Deployment

- Compute
  - Comprised of servers needed to carry out "computations"
  - Software gets deployed on "Compute" nodes
- Storage
  - Comprised of servers needed to store information
  - Storage hardware like HDD, Flash, Tapes get deployed
  - Software for managing the hardware gets deployed
- Network
  - Comprised of networking hardware components like routers, switches, connectivity media
  - Software for managing the network gets deployed



WHERE - Location of deployment

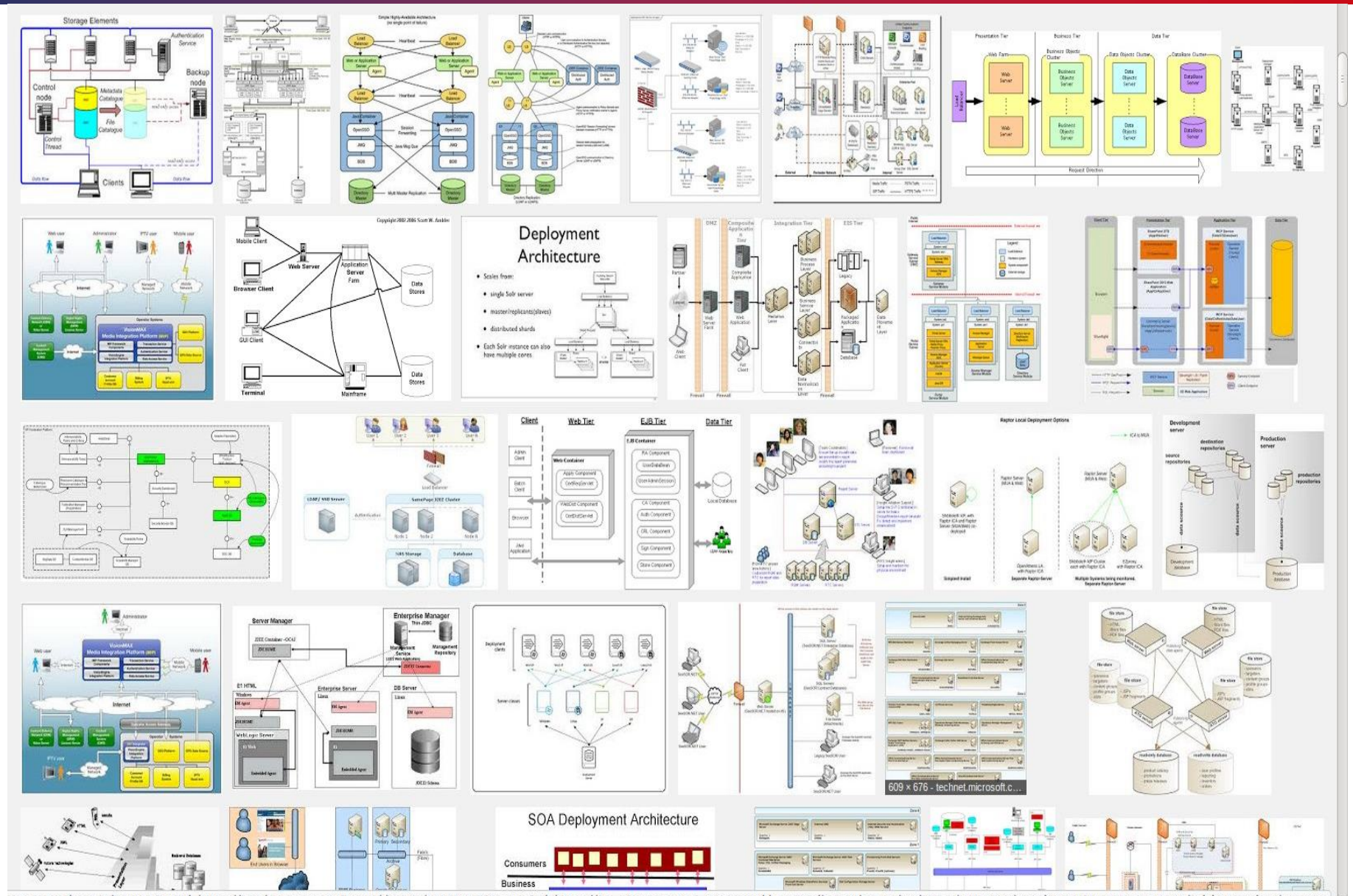
On-Premise ("On-Prem")

- **Dedicated** "Data center"
- **Owned** by the Enterprise
- Includes all physical infrastructure (space, cooling, power, cabling, hardware, monitoring tools)
- Managed by **locally** by the enterprise personnel
- Typically separate hardware stack dedicated to specific applications

Cloud

- **Shared** "Data Center"
- **Owned** by service provider and **used** by multiple enterprises
- Includes all physical infrastructure (space, cooling, power, cabling, hardware, monitoring tools)
- Managed **centrally** by service provider personnel
- Shared hardware stack across multiple applications

Examples: Zoom and see! ☺



www.google.co.in/imgres?imgurl=https://i-technet.sec.s-msft.com/dynimg/IC19166.gif&imgrefurl=https://te...vjp6BMLM&ei=Q9RJVuJGhbzQBKzxlbgM&tbm=isch&client=ubuntu&ved=0CEkQMygfMB9qFQoTCKL07diAlckCFQUelAodrHgFxw

GOING FORWARD

Looking ahead

Layered

Event-Driven

Microkernel

Micro-services