

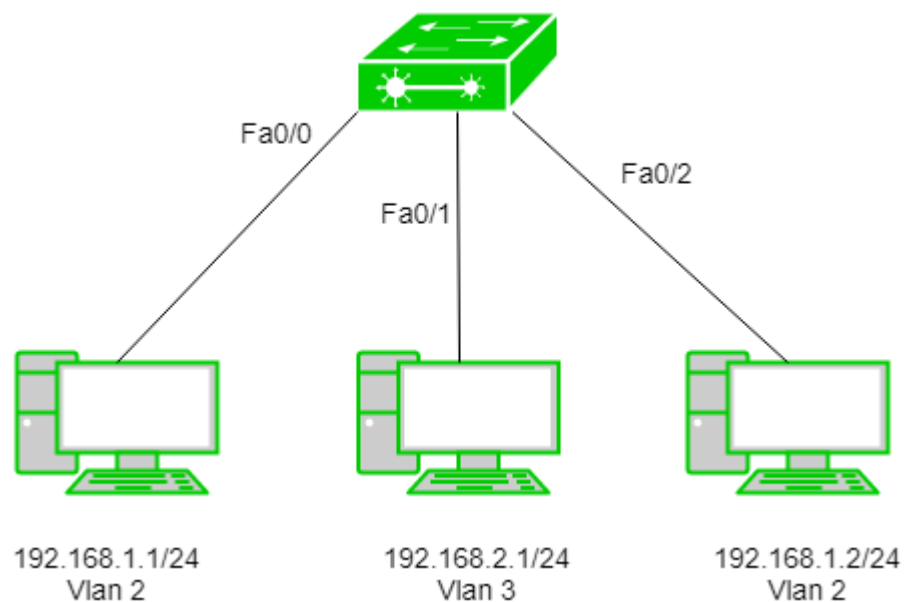
VLAN (Virtual Local Area Network)

Virtual LAN (VLAN) is a concept in which we can divide the devices logically on layer 2 (data link layer). Generally, layer 3 devices divide broadcast domain but broadcast domain can be divided by switches using the concept of VLAN.

VLAN ranges –

VLAN 1: It is the default VLAN of switches. By default, all switch ports are in VLAN. This VLAN can't be deleted or edit but can be used.

VLAN 2-1001: This is a normal VLAN range. We can create, edit and delete these VLAN.



How VLAN works

Here is step by step details of how VLAN works:

- VLANs in networking are identified by a number.
- A Valid range is 1-4094. On a VLAN switch, you assign ports with the proper VLAN number.
- The switch then allows data which needs to be sent between various ports having the same VLAN.
- Since almost all networks are larger than a single switch, there should be a way to send traffic between two switches.
- One simple and easy way to do this is to assign a port on each network switch with a VLAN and run a cable between them.

Benefits of Vlan

- Limit the number of broadcast.
- Better performance.
- Security.

Types of Vlan

- Static Vlan.
- Dynamic Vlan.

Static Vlan

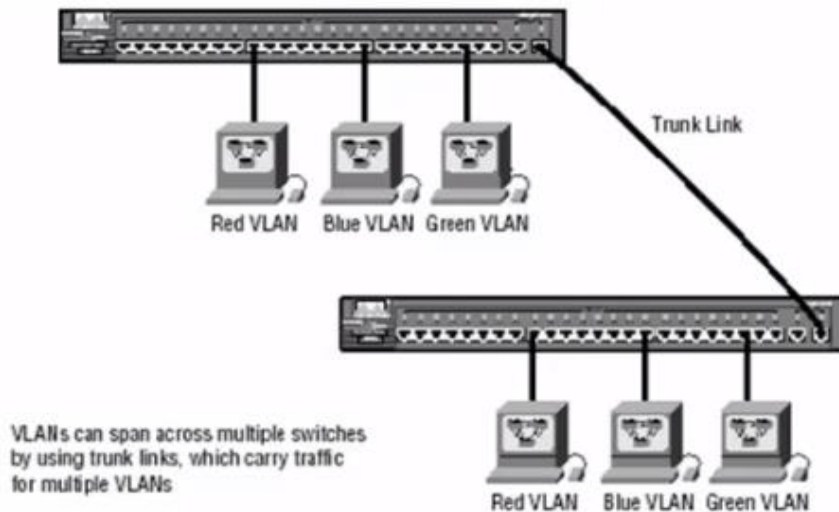
- Based on port number.
- Manually assign a port on a switch to a vlan.
- One port can be member of only one vlan.
- Also called port based vlan

Dynamic Vlan

- Based on the MAC address of the PC.
- Switch automatically assign a port to a vlan.
- Each port can be member of multiple vlans.
- A software called VMPS(Vlan Membership Policy Server) is needed.

Types of Ports in Vlan

- **Access links:**
 1. Connecting to end devices.
 2. Part of one vlan only.
- **Trunk links:**
 1. Do not belong to any vlan.
 2. Carry multiple vlan traffic.
 3. Link between two switches.



Advantages of VLAN

Here are the important pros/benefits of VLAN:

- It solves a broadcast problem.
- VLAN reduces the size of broadcast domains.
- VLAN allows you to add an additional layer of security.
- It can make device management simple and easier.
- You can make a logical grouping of devices by function rather than location.

Disadvantages of VLAN

Here are the important cons/ drawbacks of VLAN:

- A packet can leak from one VLAN to other.
- An injected packet may lead to a cyber-attack.
- Threat in a single system may spread a virus through a whole logical network.
- You require an additional router to control the workload in large networks.
- You can face problems in interoperability.

RAID (redundant array of independent disks)

RAID or redundant array of independent disks is a data storage virtualization technology that combines multiple physical disk drive components into one or more logical units for data redundancy, performance improvement, or both.

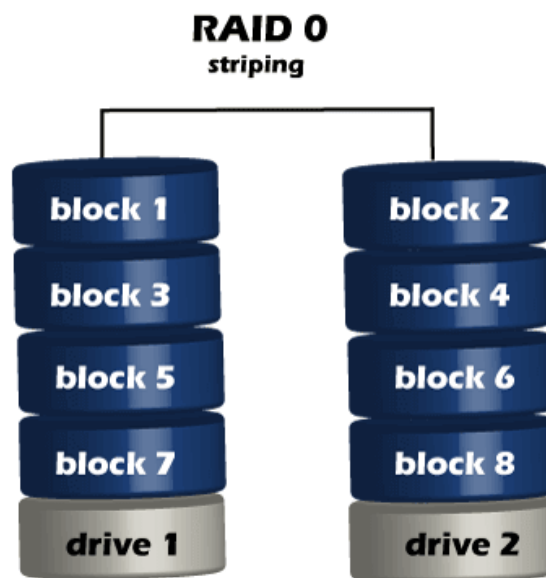
RAID combines several independent and relatively small disks into single storage of a large size. The disks included in the array are called array members. The disks can combine into the array in different ways, which are known as RAID levels.

Levels of RAID

Many different ways of distributing data have been standardized into various RAID levels. Each RAID level is offering a trade-off of data protection, system performance, and storage space.

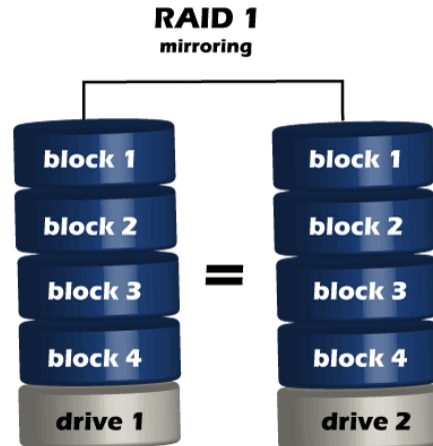
1. RAID 0 (striped disks)

- We can use RAID 0 in the public cloud.
- Minimum No. of Disks required for RAID 0 is 2.
- All the space of both disks is available.
- It only provides speed as it splits the data between disks.
- It should only be used in a cluster setup, not in single node setup.
- Recommended using only on the non-critical environment or multi-node setup.



2. RAID 1 (mirrored disks)

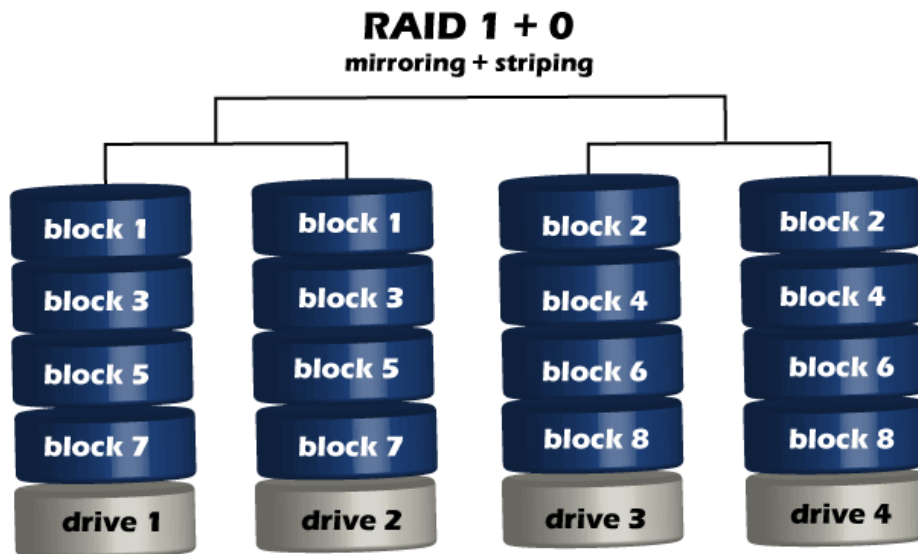
- RAID 1 is used where redundancy is required. It doesn't offer speed as it mirrors the data between two disks.
- Minimum No. of Disks required for RAID 1 is 2.
- Storage Efficiency is reduced to 50% of the available disk.
- Can be used in both single node and multinode cluster.



3. RAID 10 (1+0) (Nested RAID Level)

This level Combines RAID 1 and RAID 0 in a single system, which offers higher performance than RAID 1, but at a much higher cost.

This is a nested or hybrid RAID configuration. It provides security by mirroring all data on secondary drives while using striping across each set of drives to speed up data transfers.



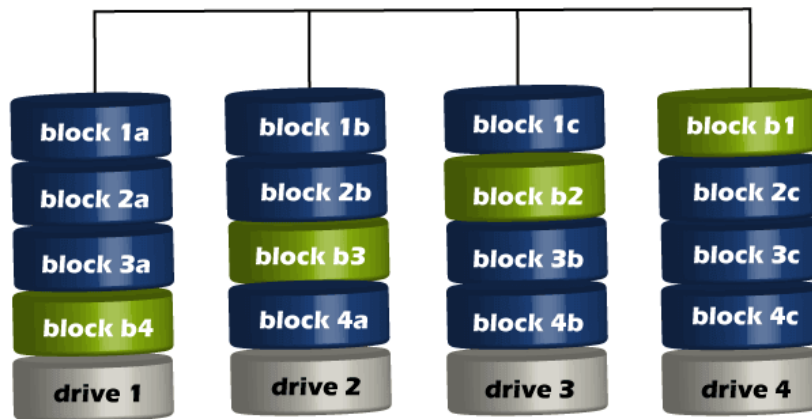
RAID-3 (Block-Level Striping with Dedicated Parity)

- Instead of duplicating data, this adopts a parity-based approach.

RAID 5 (Block-Level Striping with Distributed Parity)

- This is a slight modification of the RAID-3 system where the only difference is that the parity rotates among the drives.

RAID 5 **Striping with parity across drives**



PaaS

- The Platform as a Service model describes a software environment in which a developer can create customized solutions within the context of the development tools that the platform provides.
- A PaaS offering provides the tools and development environment to deploy applications on another vendor's application
- To be useful as a cloud computing offering, PaaS systems must offer a way to create user interfaces, and thus support standards such as HTML, JavaScript, or other rich media technologies.
- The vendor is responsible for all the operational aspects of the service, for maintenance, and for managing the product(s) lifecycle.
- For example, Google's App Engine platform, Force.com, Windows Azure Platform to systems like Drupal, Squarespace, Wolf.
- The difficulty with PaaS is that it locks the developer (and the customer) into a solution that is dependent upon the platform vendor.
- There is considerable vendor lock-in associated with a PaaS solution.

Service-oriented architecture (SOA)

Service-oriented architecture (SOA) is a method of software development that uses software components called services to create business applications. Each service provides a business capability, and services can also communicate with each other across platforms and languages. Developers use SOA to reuse services in different systems or combine several independent services to perform complex tasks.

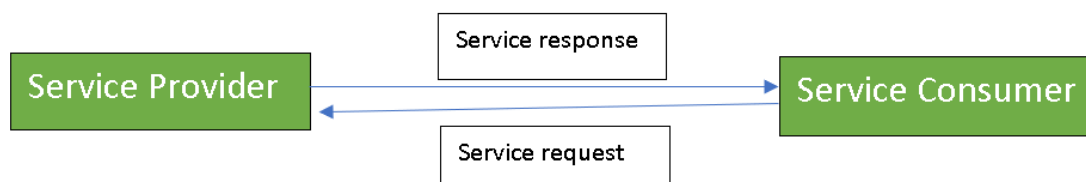
For example, multiple business processes in an organization require the user authentication functionality. Instead of rewriting the authentication code for all business processes, you can create a single authentication service and reuse it for all applications. Similarly, almost all systems across a healthcare organization, such as patient management systems and electronic health record (EHR)

systems, need to register patients. These systems can call a single, common service to perform the patient registration task.

There are two major roles within Service-oriented Architecture:

Service provider: The service provider is the maintainer of the service and the organization that makes available one or more services for others to use. To advertise services, the provider can publish them in a registry, together with a service contract that specifies the nature of the service, how to use it, the requirements for the service, and the fees charged.

Service consumer: The service consumer can locate the service metadata in the registry and develop the required client components to bind and use the service.



Guiding Principles of SOA or SOA Platform Features:

1. Standardized service contract: Specified through one or more service description documents.
2. Loose coupling: Services are designed as self-contained components, maintain relationships that minimize dependencies on other services.
3. Abstraction: A service is completely defined by service contracts and description documents. They hide their logic, which is encapsulated within their implementation.
4. Reusability: Designed as components, services can be reused more effectively, thus reducing development time and the associated costs.
5. Autonomy: Services have control over the logic they encapsulate and, from a service consumer point of view, there is no need to know about their implementation.
6. Discoverability: Services are defined by description documents that constitute supplemental metadata through which they can be effectively discovered. Service discovery provides an effective means for utilizing third-party resources.
7. Composability: Using services as building blocks, sophisticated and complex operations can be implemented. Service orchestration and choreography provide a solid support for composing services and achieving business goals.

Advantages of SOA:

- **Service reusability:** In SOA, applications are made from existing services. Thus, services can be reused to make many applications.
- **Easy maintenance:** As services are independent of each other they can be updated and modified easily without affecting other services.
- **Platform independent:** SOA allows making a complex application by combining services picked from different sources, independent of the platform.

- **Availability:** SOA facilities are easily available to anyone on request.
- **Reliability:** SOA applications are more reliable because it is easy to debug small services rather than huge codes
- **Scalability:** Services can run on different servers within an environment, this increases scalability

Disadvantages of SOA:

- **High overhead:** A validation of input parameters of services is done whenever services interact this decreases performance as it increases load and response time.
- **High investment:** A huge initial investment is required for SOA.
- **Complex service management:** When services interact they exchange messages to tasks. the number of messages may go in millions. It becomes a cumbersome task to handle a large number of messages.

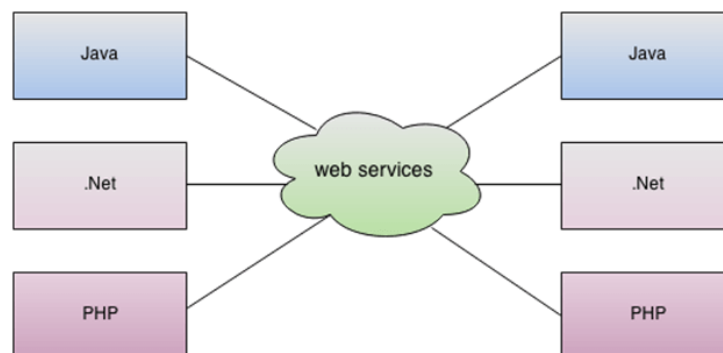
Practical applications of SOA: SOA is used in many ways around us whether it is mentioned or not.

1. SOA infrastructure is used by many armies and air forces to deploy situational awareness systems.
2. SOA is used to improve healthcare delivery.
3. Nowadays many apps are games and they use inbuilt functions to run. For example, an app might need GPS so it uses the inbuilt GPS functions of the device. This is SOA in mobile solutions.
4. SOA helps maintain museums a virtualized storage pool for their information and content.

Web Services

A web service is a set of open protocols and standards that allow data to be exchanged between different applications or systems. Web services can be used by software programs written in a variety of programming languages and running on a variety of platforms to exchange data via computer networks such as the Internet in a similar way to inter-process communication on a single computer.

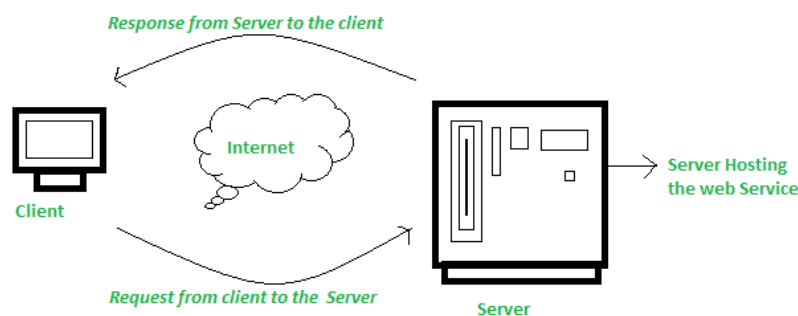
Let's understand it by the figure given below:



As you can see in the figure, Java, .net, and PHP applications can communicate with other applications through web service over the network. For example, the Java application can interact with Java, .Net, and PHP applications. So web service is a language independent way of communication.

How Does Web Service Work?

The diagram depicts a very simplified version of how a web service would function. The client would use requests to send a sequence of web service calls to a server that would host the actual web service.



Remote procedure calls are what are used to make these requests. Calls to methods hosted by the relevant web service are known as Remote Procedure Calls (RPC). Example: Flipkart offers a web service that displays prices for items offered on Flipkart.com. The front end or presentation layer can be written in .Net or Java, but the web service can be communicated using either programming language.

Web Service Components

There are three major web service components.

SOAP

WSDL

UDDI

SOAP

- SOAP is an acronym for Simple Object Access Protocol.
- SOAP is a XML-based protocol for accessing web services.
- SOAP is a W3C recommendation for communication between applications.

- SOAP is XML based, so it is platform independent and language independent. In other words, it can be used with Java, .Net or PHP language on any platform.

WSDL

- WSDL is an acronym for Web Services Description Language.
- WSDL is a xml document containing information about web services such as method name, method parameter and how to access it.
- WSDL is a part of UDDI. It acts as a interface between web service applications.

UDDI

- UDDI is an acronym for Universal Description, Discovery and Integration.
- UDDI is a XML based framework for describing, discovering and integrating web services.
- UDDI is a directory of web service interfaces described by WSDL, containing information about web services.

What is SaaS?

Software as a service (or SaaS) is a way of delivering applications over the Internet—as a service. Instead of installing and maintaining software, you simply access it via the Internet, freeing yourself from complex software and hardware management.

SaaS applications are sometimes called Web-based software, on-demand software, or hosted software. Whatever the name, SaaS applications run on a SaaS provider's servers. The provider manages access to the application, including security, availability, and performance.

Examples:

Social Networks - As we all know, social networking sites are used by the general public, so social networking service providers use SaaS for their convenience and handle the general public's information.

Mail Services - To handle the unpredictable number of users and load on e-mail services, many e-mail providers offering their services using SaaS.

SaaS Benefits

- Software cost reduction and total cost of ownership (TCO) were paramount.
- Service-level improvements.
- Rapid implementation.
- Standalone and configurable applications.
- Rudimentary application and data integration.

Subscription and pay-as-you-go (PAYG) pricing.

Open SaaS

When open source software is used in a SaaS, you may hear it referred to as Open SaaS. The advantages of using open source software are that systems are much cheaper to deploy because you don't have to purchase the operating system or software, there is less vendor lock-in, and applications are more portable.

What is Web 1.0?

It was the first stage of the evolution of the World Wide Web. Only a few content creators were available for Web 1.0 with a vast majority of content consumers. Personal web pages were the most common, which consisted of various static pages that are either hosted on free web hosting services or on the web servers run on an ISP.

Essential design elements of Web 1.0

1. Static pages
2. Frames and Tables for aligning and positioning the elements on any page
3. Server Side Includes or CGI (Common Gateway Interface) for building pages
4. The server's file system for serving the content.

What is Web 2.0?

These refer to the worldwide websites that highlight the user-generating content (UGC), interoperability, and usability for their end-users. Web 2.0 is also known as the participative social web. Now, this does not refer to any modifications to the technical specification. Instead, these modify how the web pages must be designed or even used. This transition is important, but these changes are not very much visible when they occur. Web 2.0 allows collaboration and interaction in the social media dialogue with the content creators of the virtual community. Thus, Web 2.0 is like a comparatively enhanced version of Web 1.0.

Web 1.0	vs.	Web 2.0
Read only		Read/write/collaborate
Reading platform		Publishing platform
Static		Dynamic
Developer		Public
Text-based		Multimodal



What are Microservices?

Microservice architecture, often known as Microservices, creates software systems in which a single application is structured as a series of loosely linked services. Applications usually start as a monolithic architecture and then evolve into a series of interconnected microservices over time.

It allows you to take a large application and decompose or break it into easily manageable small components with narrowly defined responsibilities. It is considered the building block of modern applications.

