PaaS : Platform as a Service

- Evolution of computing paradigms and related components
- Introduction to PaaS-What is PaaS,
- Service Oriented Architecture (SOA)

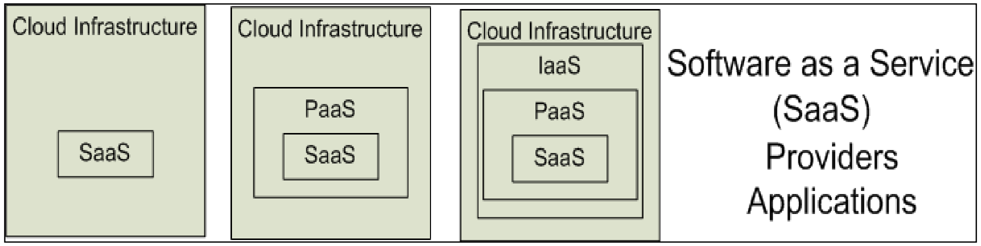
Cloud Services delivery Models

Software as a Service (SaaS)

Platform as a Service (PaaS)

Infrastructure as a Service (IaaS)

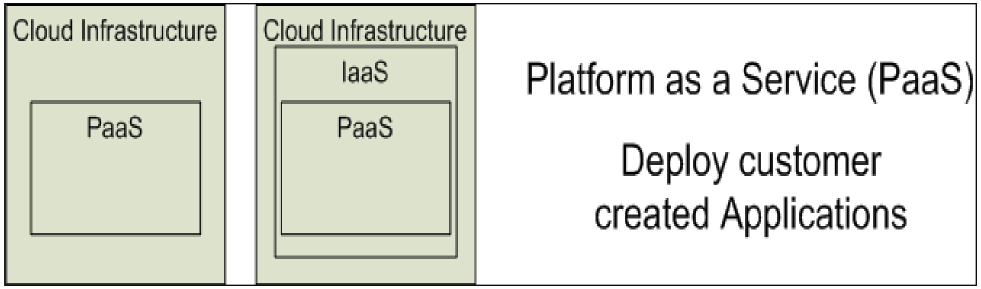
SalesForce
CRM
LotusLive



Software as a Service (SaaS)
Providers
Applications

Google
App

Windows Azure
The Future Made Familiar

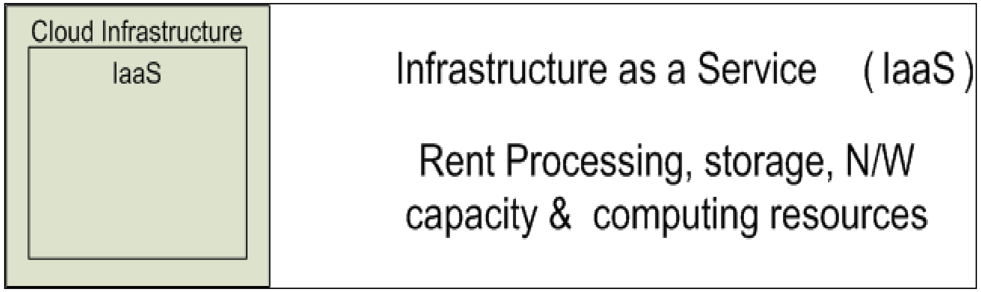


Platform as a Service (PaaS)

Deploy customer
created Applications

amazon
web services™

rackspace®
HOSTING



Infrastructure as a Service (IaaS)

Rent Processing, storage, N/W
capacity & computing resources

Platform as a Service PaaS

- Platform as a Service is a type of Cloud Computing which allows customers to develop, run, and manage the applications by providing them with the platform and diminishing the complexities of maintenance.

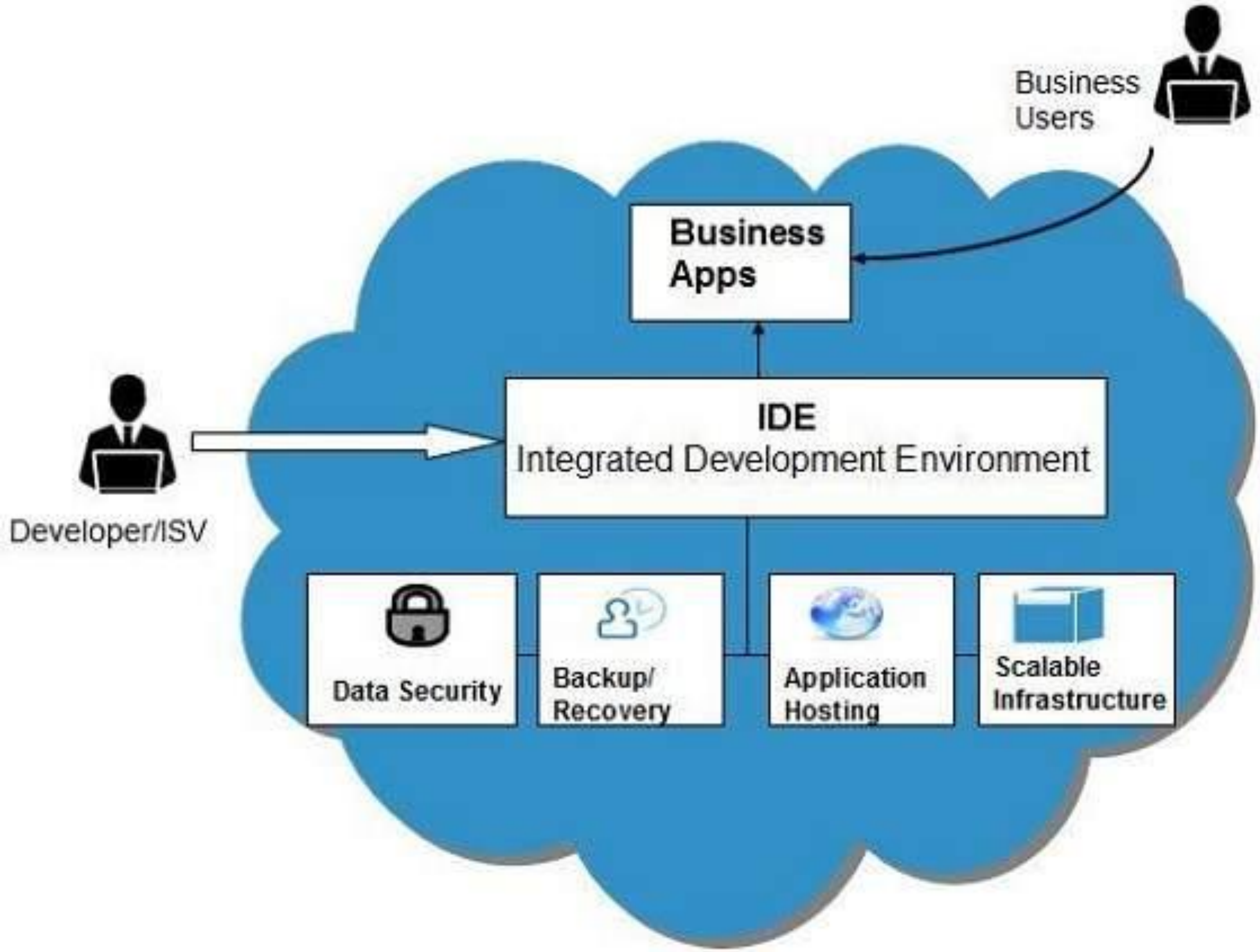
- PaaS enables to deliver from simple cloud-based applications to higher cloud-enabled applications.

- We can purchase the resources from the cloud service provider on a pay-as-you-go basis. These resources access with the help of internet.

- Platform as a service not only includes server, storage, and networking but also database, tools, business services, and many more.

- It is made to perform building, testing, deployment, managing, and modification of the application

Platform as a Service PaaS



PaaS offers an API and development tools to the developers and how it helps the end user to access business applications.

Platform as a Service PaaS

- Platform-as-a-Service offers the runtime environment for applications.

- It also offers development and deployment tools required to develop applications.

- PaaS has a feature of point-and-click tools that enables non-developers to create web applications.

- App Engine of Google and Force.com are examples of PaaS offering vendors.

- Developer may log on to these websites and use the built-in API to create web-based applications.

Platform as a Service PaaS

Characteristics

- PaaS offers browser based development environment. It allows the developer to create database and edit the application code either via Application Programming Interface or point-and-click tools.

- PaaS provides built-in security, scalability, and web service interfaces.

- PaaS provides built-in tools for defining workflow, approval processes, and business rules.

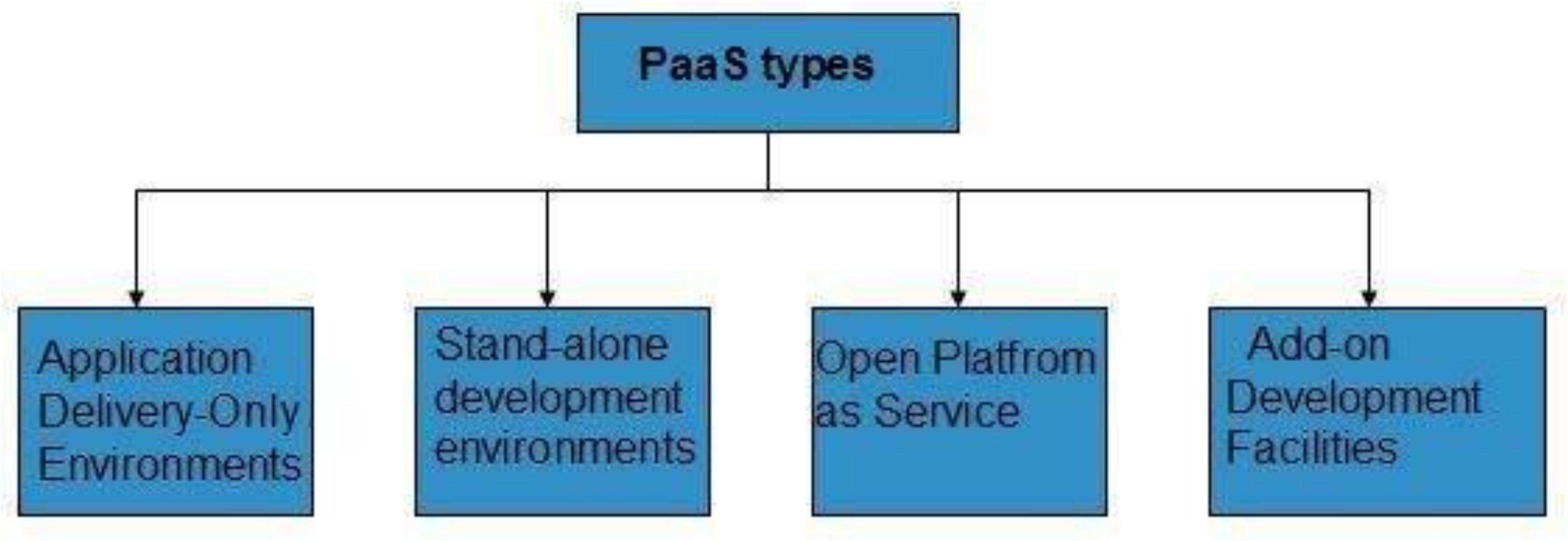
- It is easy to integrate PaaS with other applications on the same platform.

- PaaS also provides web services interfaces that allow us to connect the applications outside the platform.

Platform as a Service PaaS

PaaS Types

- Based on the functions, PaaS can be classified into four types as shown in the following diagram:


PaaS types
Application Delivery-Only Environments
Stand-alone development environments
Open Platfrom as Service
Add-on Development Facilities

Platform as a Service PaaS

PaaS Types

- **Stand-alone development environments:** The stand-alone PaaS works as an independent entity for a specific function. It does not include licensing or technical dependencies on specific SaaS applications.

- **Application delivery-only environments:** The application delivery PaaS includes on-demand scaling and application security.

- **Open platform as a service:** Open PaaS offers an open source software that helps a PaaS provider to run applications.

- **Add-on development facilities:** The add-on PaaS allows to customize the existing SaaS platform.

Platform as a Service PaaS

How Platform as a Service is used?

- Analytics and Business Intelligence
- Framework
- Additional Services

Platform as a Service PaaS

How Platform as a Service is used?

- Analytics and Business Intelligence: With the help of Platform as a Service, the companies can analyze the data by monitoring the demand of customers. It also helps to find insights, patterns which predicts the output to improve the service, investment, returns, and saves the time.

- Framework: With the help of the PaaS framework, the developers can build the cloud-based applications. There are several built-in software which allows customers to built their own application. The features such as scalability and high-availability save the extra cost and also reduce the time.

- Additional Services: There are several additional applications which enhance the working of the existing applications such as workflow, directory, security, and scheduling.

Platform as a Service PaaS

Advantages:

- Reduces the Coding Time: With the help of Platform as a Service, the coding time reduces as the time it takes to code new apps with pre-coded application parts design into the platform, like workflow, directory services, security measures, search then on.

- Enhances the Development Capabilities: Platform as Service parts will provide your development team new capabilities while you don’t need to add workers having the specified skills.

- Available at Multiple platforms: It can access from anywhere and from many devices such as mobile, tabs, and laptops. In addition, some service suppliers offer development choices for multiple platforms like computers and browsers creating cross-platform apps faster and easier to develop.

Platform as a Service PaaS

Advantages:

- Tools at economical costs: PaaS provides pay-as-you-go service which makes it potential for people or organizations to use subtle development package and business intelligence and analytics tools that they may not afford to get outright.

- Regular management of appliance lifecycle: There are several capabilities provided by Platform as a Service, which will support the whole net application lifecycle: building, testing, deploying, managing and change inside constant integrated setting.

- In addition, PaaS eliminates the expenses and complexity of purchasing new software and managing it. The tools which are provided by the Cloud providers manage this.

The Service Oriented Architecture (SOA)

- SOA is a logical way of designing a software system to provide services to either end-user applications or to other services distributed in a network, via published and discoverable interfaces.

- Once all the elements of an SOA are in place, existing and future applications can access the SOA-based services as necessary.

- This architectural approach is particularly applicable when **multiple applications running on varied technologies and platforms need to communicate with each other.**

The Service Oriented Architecture (SOA) Roles

- **Web Service Providers**
  - ✓ Create web services and web service descriptions

- **Web Service Requestor / Clients**
  - ✓ Users of a Web Service

- **Web Service Registry**
  - ✓ a service to connect the former
  - ✓ providers register services with registry
  - ✓ clients search registry for services they need

The Service Oriented Architecture (SOA) Roles

- **Web Service Providers**
  - ✓ From a business perspective the Web services provider is the organization that owns the Web service and implements the business logic that underlies the service.
  - ✓ From an architectural perspective this is the platform that hosts and controls access to the service.
  - ✓ The Web services provider is responsible for publishing the Web services it provides in a service registry hosted by a service discovery agency.

The Service Oriented Architecture (SOA) Roles

- **Web Service Requestor / Clients**
  - ✓ From a business perspective this is the enterprise that requires certain functions to be satisfied.
  - ✓ From an architectural perspective, this is the application that is looking for, and subsequently invoking, the service.
  - ✓ The Web services requestor searches the service registry for the desired Web services.
  - ✓ This effectively means discovering the Web services description in a registry provided by a discovery agency and using the information in the description to bind to the service.

The Service Oriented Architecture (SOA) Roles

- **Web Service Registry**
  - ✓ Web services registry, which is a searchable directory where service descriptions can be published and searched.
  - ✓ Service requestors find service descriptions in the registry and obtain binding information for services.
  - ✓ This information is sufficient for the service requestor to contact, or bind to, the service provider and thus make use of the services it provides.

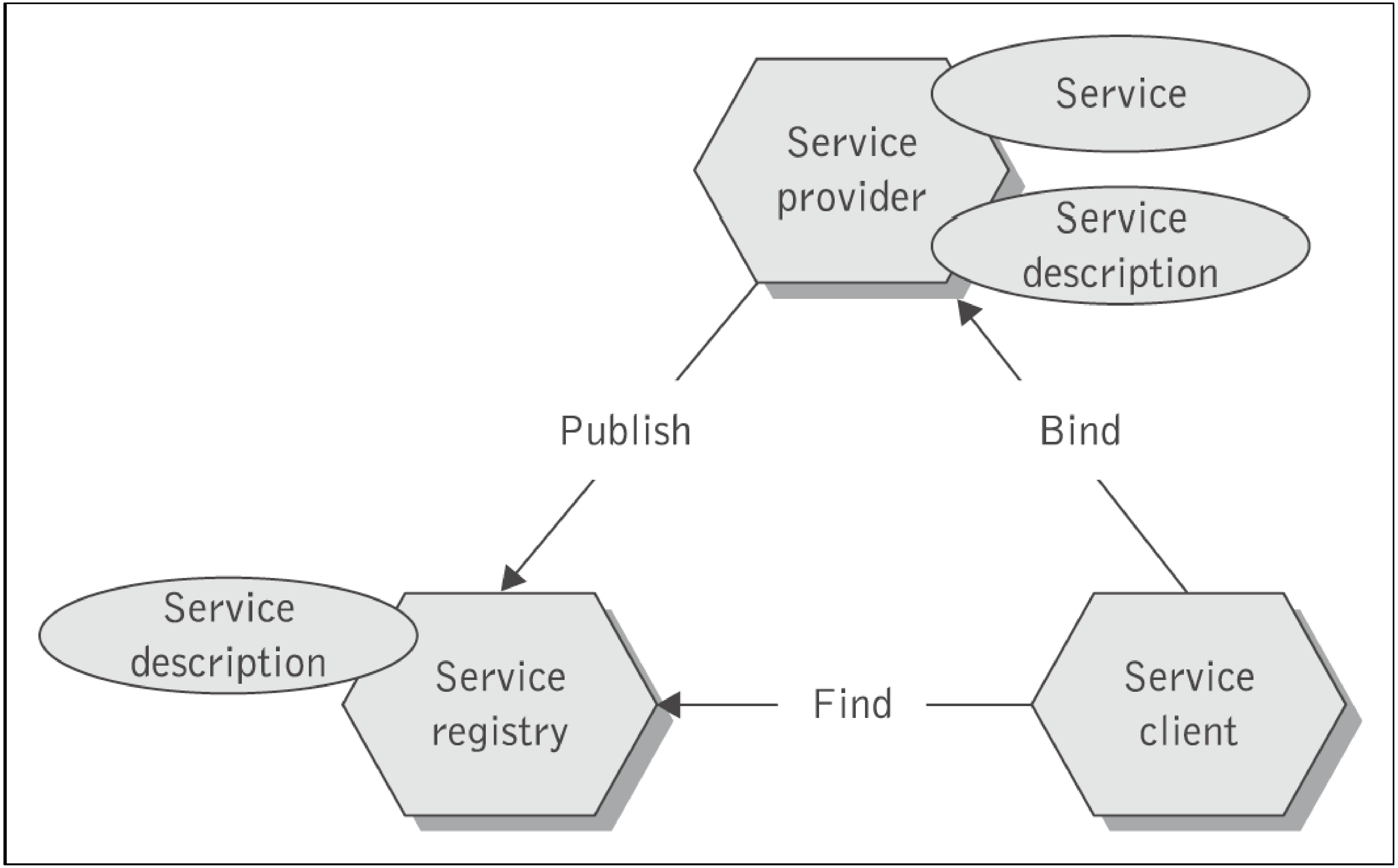
Operations in Service Oriented Architecture (SOA)

In the SOA three primary operations must take place:

- ✓ **Publication** of the service descriptions,
- ✓ **Finding** the service descriptions, and
- ✓ **Binding or invocation** of services based on their service description.

- The Web services provider publishes its Web service(s) with the discovery agency.
- Next, the Web services client searches for desired Web services using the registry of the discovery agency.
- Finally, the Web services client, using the information obtained from the discovery agency, invokes (binds to) the Web services provided by the Web services provider.

Operations in Service Oriented Architecture (SOA)



Operations in Service Oriented Architecture (SOA)

- **Publication :** First operation is *describing* the Web service itself; the other is the actual *registration* of the Web service.

  - ✓ The first requirement for publishing Web services with the service registry is for a service provider to properly describe them in WSDL.

  Three basic categories of information

  - ✓ **Business information:** information regarding the Web service provider or the implementer of the service;
  - ✓ **Service information:** information about the nature of the Web service;
  - ✓ **Technical information:** information about implementation details and the invocation methods for the Web service.

- Registration deals with storing the three basic categories of descriptive information about a service in the Web services registry.

Operations in Service Oriented Architecture (SOA)

- **Find Operation :** consists of first discovering the services in the registry of the discovery agency and then selecting the desired Web service(s) from the search results.

✓ **Discovering** Web services involves querying the registry of the discovery agency for Web services matching the needs of a Web services requestor.

  - A query consists of search criteria such as type of service, preferred price range, what products are associated with this service.

✓ **Selection** deals with deciding about which Web service to invoke from the set of Web services the discovery process returned.

  - *Manual selection* implies that the Web services requestor selects the desired Web service directly from the returned set of Web services after manual inspection.
  - *Automatic selection* Web services requestor has to specify preferences to enable the application to infer which Web service

Operations in Service Oriented Architecture (SOA)

- **Bind Operation :** The service requestor invokes or initiates an interaction at run-time using the binding details in the service description to locate and contract to the service.
  - ✓ The technical information entered in the registry by the Web services provider is used here.
  - ✓ *Direct invocation* of the Web service by the Web services requestor using the technical information included in the description of the service.
  - ✓ *Mediation* by the discovery agency when invoking the Web service.
  - ✓ The Web services requestor and the Web services provider goes through the Web services registry of the discovery agency.

SOA key Benefits

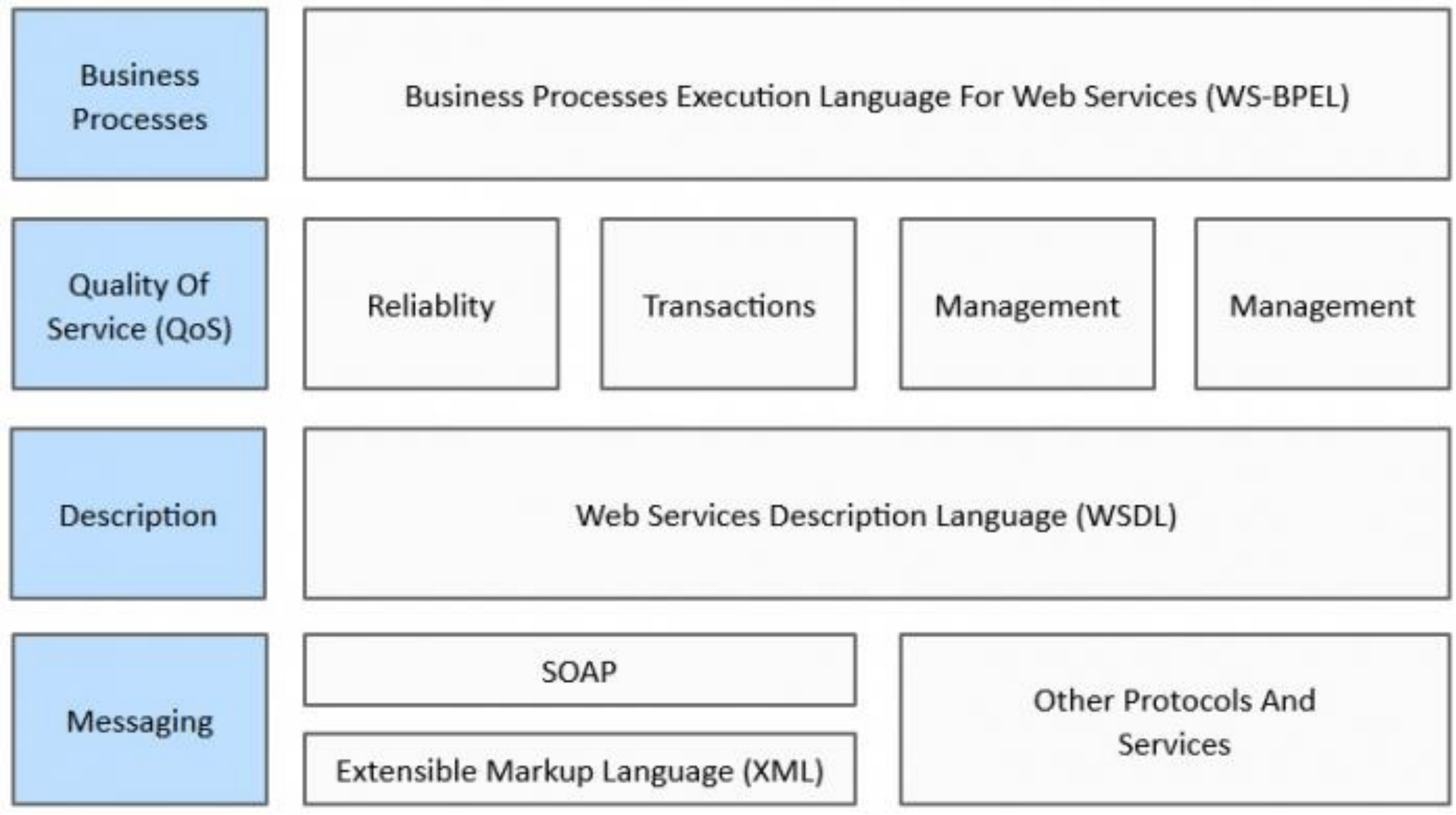
- Dependence on the network
- Provider cost
- Enterprise standards
- Agility

SOA Architecture

SOA architecture is viewed as five horizontal layers.

- Consumer Interface Layer: These are GUI based apps for end users accessing the applications.
- Business Process Layer: These are business-use cases in terms of application.
- Services Layer: These are whole-enterprise, in service inventory.
- Service Component Layer: are used to build the services, such as functional and technical libraries.
- Operational Systems Layer: It contains the data model.

SOA Architecture



Business Processes

Business Processes Execution Language For Web Services (WS-BPEL)

Quality Of Service (QoS)

Reliablity

Transactions

Management

Management

Description

Web Services Description Language (WSDL)

Messaging

SOAP

Extensible Markup Language (XML)

Other Protocols And Services

SOA Architecture

- Here lies the protocol stack of SOA showing each protocol along with their relationship among each protocol.

- These components are often programmed to comply with SCA (Service Component Architecture), a language that has broader but not universal industry support.

- These components are written in BPEL (Business Process Execution Languages), Java, C#, XML etc and can apply to C++ or FORTRAN or other modern multi-purpose languages such as Python, PP or Ruby.

- With this, SOA has extended the life of many all-time famous applications.

SOA Benefits

With high-tech engineering and enterprise point of view, various offers are provided by SOA which proved to be beneficial.

- Language Neutral Integration: Regardless of the developing language used, the system offers and invoke services through a common mechanism. Programming language neutralization is one of the key benefits of SOA's integration approach.

- Component Reuse: Once an organization built an application component, and offered it as a service, the rest of the organization can utilize that service.

- Organizational Agility: SOA defines building blocks of capabilities provided by software and it offers some service(s) that meet some organizational requirement; which can be recombined and integrated rapidly.

- Leveraging Existing System: This is one of the major use of SOA which is to classify elements or functions of existing applications and make them available to the organizations or enterprise.

Some Popular Paas Providers And Services That Are Provided By Them

Providers	Services
Google App Engine (GAE)	App Identity, URL Fetch, Cloud storage client library, Logservice
Salesforce.com	Faster implementation, Rapid scalability, CRM Services, Sales cloud, Mobile connectivity, Chatter.
Windows Azure	Compute, security, IoT, Data Storage.
AppFog	Justcloud.com, SkyDrive, GoogleDocs
Openshift	RedHat, Microsoft Azure.
Cloud Foundry from VMware	Data, Messaging, and other services.

When to Use PaaS:

- PaaS is often the most cost-effective and time-effective way for a developer to create a unique application.

- PaaS allows the developer to focus on the creative side of app development, as opposed to menial tasks such as managing software updates or security patches.

- All of their time and brainpower will go into creating, testing, and deploying the app.

When to Use PaaS:

PaaS Ecommerce Example:

- Magento Commerce Cloud (also known as Magento Enterprise Cloud Edition) is the most common example of PaaS for ecommerce.

- This enables the merchant to bundle their hosting as part of their package with Magento.

- Merchants evaluating Magento go through a scoping process to determine their hosting needs which is then bundled into their monthly plan.

- Merchants still have full access to edit the source code of their Magento store and can fully customize the application.

- Any platform updates, security patches and general maintenance to their store would be the responsibility of the merchant.

When to Use PaaS:

AWS Elastic Benstalk:

- With Elastic Beanstalk, you can quickly deploy and manage applications in the AWS Cloud without having to learn about the infrastructure that runs those applications.

- Elastic Beanstalk reduces management complexity without restricting choice or control.

- You simply upload your application, and Elastic Beanstalk automatically handles the details of capacity provisioning, load balancing, scaling, and application health monitoring.

- Elastic Beanstalk supports applications developed in Java, .NET, Node.js, PHP, Python, and Ruby.

- When you deploy your application, Elastic Beanstalk builds the selected supported platform version and provisions one or more AWS resources, such as Amazon EC2 instances, to run your application.

What is the difference between PaaS and traditional web hosting

Traditional web hosting usually consists of the following:

- a root folder.

  It’s a subfolder in a server created specifically for your account needs. All data and custom configurations are stored there. That may consist of an FTP folder for FTP uploads and downloads, your public folder serving your website and an email folder if email service is included

- Mail service tied to the accounts domain(s).
- FTP account for uploads and downloads
- DNS service to handle your DNS records
- SSH Access with a jailed user on that account root folder.
- Database
- All that is usually happening on a shared server, along with 10s or 100s of other clients sharing the same services.

What is the difference between PaaS and traditional web hosting

- PaaS providers on the other hand, provide a Platform in which depending on the implementation, instead of using a single instance to serve multiple accounts, multiple instances are used for a single account.

- All that is done seemingly without the user's intervention for the intermediate systems, hence the platform.

- Most commonly, the case scenarios where PaaS providers are used is when it's not a simple WordPress site with a moderate traffic, but a complex application with high traffic that would normally underperform in a shared environment.

- That doesn't necessarily mean that simpler applications are not the scope of PaaS providers, since in such scenarios, scaling up is a matter of clicks instead of painful and time consuming migrations.

What is the difference between PaaS and traditional web hosting

PaaS providers are the ones responsible for building and maintaining the infrastructure necessary for the applications development and support. That might consist of:

- Production environments
- Staging environments
- Networking (Load Balancers/VPC)
- Highly tested software stack with constant updates
- Statistics (not just google ads)
- Redundancy (for performance and security)
- Autoscalling
- Monitoring
- Professional Services on demand
- and most importantly a 24/7 support depending on the SLA
- Although you might use a PaaS provider to host a simpler application, rarely a demanding one would be hosted on a traditional hosting provider.

Reference Links:

- https://www.infoworld.com/article/3220669/what-is-iaas-your-data-center-in-the-cloud.html

- https://www.ibm.com/in-en/cloud/learn/iaas-paas-saas

- https://data-flair.training/blogs/platform-as-a-service-paas/

- Case Study: https://www.guru99.com/cloud-computing-for-beginners.html