CLOUD COMPUTING

Dr. D Muhammad Noorul Mubarak

Head, Department of Computer Science,

University of Kerala

Module V

Service Oriented Application

- The Service Oriented Architecture is an architectural design which includes collection of services in a network which communicate with each other.
- The complication of each service is not noticeable to other service.
- The service is a kind of operation which is well defined, self contained that provides separate functionality such as checking customer account details, printing bank statements etc and does not depend on the sate of other services.
- History: The first report published on SOA by the analysts **Roy W.Schulte** and **Yefim V.Natis** in 1996.

SOA


SERVICES
BEST PRACTICES
SERVICE ORIENTED ARCHITECTURE(SOA)
PLATFORM
PROCESS
USERS

Service Oriented Application

- Why SOA?
  - SOA is widely used in market which responds quickly and makes effective changes according to market situations.
  - The SOA keep secret the implementation details of the subsystems.
  - It allows interaction of new channels with customers, partners and suppliers.
  - It authorizes the companies to select software or hardware of their choice as it acts as platform independence.

Service Oriented Application

- **Features:**
  - SOA uses interfaces which solves the difficult integration problems in large systems.
  - SOA communicates customers, providers and suppliers with messages by using the XML schema.
  - It uses the message monitoring to improve the performance measurement and detects the security attacks.
  - As it reuses the service, there will be lower software development and management costs.

Service Oriented Application

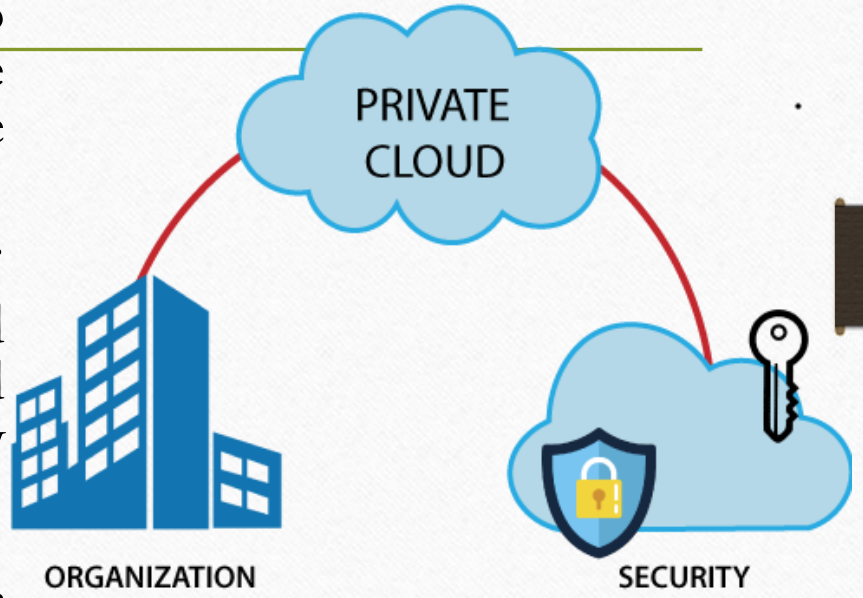
- **SOA Advantage**
  - SOA allows reuse the service of an existing system alternately building the new system.
  - It allows plugging in new services or upgrading existing services to place the new business requirements.
  - It can enhance the performance, functionality of a service and easily makes the system upgrade.
  - SOA has capability to adjust or modify the different external environments and large applications can be managed easily.
  - The companies can develop applications without replacing the existing applications.
  - It provides reliable applications in which you can test and debug the independent services easily as compared to large number of code.

Service Oriented Application

- **Disadvantages**
  - SOA requires high investment cost (means large investment on technology, development and human resource).
  - There is greater overhead when a service interacts with another service which increases the response time and machine load while validating the input parameters.
  - SOA is not suitable for GUI (graphical user interface) applications which will become more complex when the SOA requires the heavy data exchange.

Private Cloud

- Private cloud is also known as an **internal cloud** or **corporate cloud**.
- Private cloud provides computing services to a **private internal network (within the organization)** and **selected users** instead of the general public.
- Private cloud provides a **high level of security** and **privacy** to data through firewalls and internal hosting. It also ensures that operational and sensitive data are not accessible to third-party providers.
- HP Data Centers, Microsoft, Elastra-private cloud, and Ubuntu are the example of a private cloud.



Why Private Cloud?

- **Security**: Private cloud security is enhanced since traffic to a private cloud is typically limited to the organization's own transactions. Public cloud providers must handle traffic from millions of users and transactions simultaneously, thus opening a greater chance for malicious traffic. Since private clouds consist of dedicated physical infrastructure, the organization has better control over server, network, and application security.
- **Predictable performance**: Because the hardware is dedicated rather than multi-tenant, workload performance is predictable and unaffected by other organizations sharing infrastructure or bandwidth.
- **Long-term savings**: While it can be expensive to set up the infrastructure to support a private cloud, it can pay off in the long term. If an organization already has the hardware and network required for hosting, a private cloud can be much more cost-effective over time compared to paying monthly fees to use someone else's servers on the public cloud.
- **Predictable costs**: Public cloud costs can be very unpredictable based on usage, storage charges and data egress charges. Private cloud costs are the same each month, regardless of the workloads an organization is running or how much data is moved.
- **Regulatory governance**: Regulations such as the EU's GDPR may dictate where data resides and where computing occurs. In those regions where public cloud providers cannot offer service, a private cloud may be required. Additionally, organizations with sensitive data such as financial or legal firms may opt for private cloud storage to ensure they have complete control over personally identifiable or sensitive information.

Develop a Private Cloud

1. **Decide What You Want Out of a Cloud:** Journeying to the cloud is a huge trend in IT. The problem is that the term cloud computing means something different to everyone. In order to start your journey, your organization needs to be realistic about its cloud computing goals.
2. **Have Realistic Expectations of the Journey—and the Cloud:** You should expect that there cannot be true self-service IT within your organization. IT departments have spent years wrapping process and procedure around the act of creating and managing servers, usually with good reason. Often these processes are responsible for monitoring systems, determining sizing and dependencies, documenting system designs and responsibilities, handling licensing, and more.

Develop a Private Cloud

3. **Understand Your Workloads and Services:** Working toward a private cloud model is difficult when you don't understand the services your organization relies on. Documentation is key; without it the relationships between systems are hard to decipher, service-level agreements are unknown and it's easy to make false assumptions. The needs of the people using these services should also be documented so that new cloud services can be built to meet them.
4. **Get on the Path to Virtualization:** While it isn't required that a private cloud be based on virtualization, it is the common model. Virtualization usually drives certain knowledge and behaviors within organizations. For example, most virtualization software requires centralized storage. That same centralized storage will be a building block for a private cloud, so the knowledge gained in implementing virtualization is very beneficial to private clouds.

Develop a Private Cloud

5. **Understand That Standardization and Automation Go Hand in Hand:** Automation is one of the key goals that organizations have when moving to a private cloud. However, automation is incredibly difficult without standardization. For example, with standards for operating systems and server builds you can make assumptions about locations of files, sizes of file systems and authentication mechanisms. Based on those assumptions, you can script the installation of application software and middleware such as Web servers, application servers and firewall rules. This makes an installation easily repeatable, which anyone involved in rapid deployment or disaster recovery would be very happy about.
6. **Take a Look at Chargeback and Showback:** As clouds form and workloads centralize, it is important for organizations to keep track of resource usage and verify that resources are consumed fairly and organizational priorities are accounted for. A chargeback model is one of the most powerful yet most resisted forms of resource accounting.

Develop a Private Cloud

7. **Keep Everything in Its Proper Place:** Security is always a big part of IT, and when you're moving toward the cloud, it is a good time to reconsider your approaches to security. It's also a good time to consider new technologies.
8. **Know That Monitoring Is Crucial:** Centralization of services into a private cloud has many benefits, but it doesn't make performance monitoring any easier. Relocation of services often means more dependence on network performance, which, in turn, calls for extensive monitoring, plus the tools that perform that task.

Develop a Private Cloud

9. **Perform Future-Proofing:** Perform Future-Proofing Private clouds and virtualization technology decouple organizations from many problems that IT groups have been trying to solve for years. Centralizing, standardizing and automating workloads and workload management tasks frees time to do other things such as keeping an eye on new technologies. That, in turn, reduces reliance on external consultants and builds knowledge and expertise in-house.
10. **Remember, We're All in This Together:** One of the biggest changes an organization makes on the path to the cloud is internal cooperation. Years of building political and operational walls between parts of your organization serves only as a barrier to a cloud project.

Moving applications to the cloud

- Application migration to the cloud is the process of moving a software application from an on-premises server to a cloud provider's environment.
- The cloud migration process includes transferring the database, data, application, and IT processes to the cloud.

Moving applications to the cloud

- Step 1: Identify the Application and Assess its Cloud Readiness
  - Once you have identified a candidate application for cloud migration, take a look at its cloud readiness, including the application architecture, its dependencies on other systems and the underlying operating system, hardware, storage, and backing services, and its data and workload requirements. Consider how much data will need to be moved to the cloud. Keep benchmarks of workload performance on-premises that can be compared to trials in cloud environments.
- Step 2: Are Application Updates Required to Migrate to the Cloud?
  - When assessing the application for cloud readiness, you may decide to:
    1. Refactor the application for the cloud
    2. Lift and shift the application to the cloud
    3. Modernize the application with containers

Moving applications to the cloud

- **Refactoring** or rewriting: an application for the cloud is the recommended approach by public cloud vendors. The goal of refactoring an application is to take advantage of the features and functionality specific to native cloud computing environments. Your application will run faster and be more cost-efficient in the cloud, while your developers can leverage new cloud capabilities that can increase speed, flexibility, and quality. However, refactoring an application, while maintaining all of your rich features and high customer experience, can take years depending upon how complex the application is. Also, the cost, risk, and chance of failure are high. Keep in mind that shifting data from one public cloud to another, or back to on-prem, may require a rebuild and there is the potential to be locked-in to a single vendor.
- **Lifting and shifting:** is the process of copying your application (installer and file system data) and reinstalling it in the cloud on a platform (typically Windows or Linux). Lift and shift is faster and presents less risk and cost than refactoring, but you may not get the features and scalability that come with native cloud.
- **Containers:** can help you gradually migrate an application to the cloud without needing to refactor the entire application ahead of cloud migration. Containers allow you to lift and shift some of the application components while refactoring others to reduce Time to Mitigation (TTM), cost, and risk.

Moving applications to the cloud

- Step 3: Do a Cost Assessment and Review the Benefits Driving Cloud Adoption
  - This includes the time and effort it takes to refactor or update the application for cloud migration. You will need to look at your investments in hardware infrastructure and software and their associated costs, and the cost structure provided by cloud vendors. Compare the performance and cost of workloads running on-premises vs. in the cloud.
- Step 4: Identify the Right Cloud Environment
  - Enterprises have more than one cloud environment to choose from:
    - Public cloud – Access compute resources via the internet and shared across organizations
    - Private cloud – Access compute resources via the internet or private internal network and not shared across organizations
    - Hybrid cloud – Mixes public and private cloud models
    - Multi-cloud – Uses more than one public cloud vendor

Moving applications to the cloud

- Step 5: Identify the Right Cloud Service Provider and Make the Move
  - The big Cloud Service Providers (CSPs) are Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform (GCP), Oracle Cloud, and IBM Cloud Services. Note that no two CSPs are the same. When evaluating which CSP to use, you will want to evaluate the options that each provider offers and how they support your business plans.
  - Factors to consider:
    - Certifications and standards
    - Technologies and service roadmap
    - Data security, data governance, and business policies
    - Service dependencies and partnerships
    - Contracts, commercials, and SLAs
    - Reliability and performance
    - Migration support, vendor lock-in, and exit planning
    - Business health and company profile

Cloud Integration

- Cloud integration is the act of combining different cloud-based systems into an integral whole.
- The ultimate goal of cloud integration is to connect the disparate elements of various cloud and local resources into a single, ubiquitous environment that allows administrators to seamlessly access and manage applications, data, services and systems.
- The market for hybrid cloud integration platforms is expected to see a 14% compound annual growth rate between 2018 and 2023, according to a Research and Markets report.
- The drivers of this activity are many and varied.
- Some organizations are looking to use integrated resources to support real-time applications and services, while others are eyeing the increased automation capabilities for back-office and customer-facing platforms.
- Integrated clouds also provide better support for mobile applications and offer easier deployment and scalability options across the board.

Cloud Integration

- Examples of cloud integration tools include:
  - MuleSoft Anypoint Platform - includes various tools for developing, managing and testing application programming interfaces (APIs).
  - Dell Boomi - enables customers to design cloud-based integration processes called Atoms and transfer data between cloud and on-premises applications.
  - **IBM App Connect** - allows administrators to set up workflows that define how data is moved from one application to another.
  - **Cleo Integration Cloud** - provides digital integration agility across cloud and on-premise applications.
  - **Microsoft Azure Logic Apps** - allows administrators to automate workflows that integrate apps and data across cloud services and on-premises systems.
  - **Apache Libcloud** - this Python library allows administrators to manage different cloud resources through a unified application programming interface (API).

Cloud Integration

- Cloud integration can function:
  - Asynchronously: Asynchronous cloud integration communicates data and commands without having to wait on a response from the receiving application. This prevents unnecessary delays in sending -- or originating -- data because it does not need to wait for the receiving -- or target -- application to respond.
  - Synchronously: Synchronous cloud integration will wait for a response from the receiving application, which ensures the applications are fully synchronized before continuing.