Module 8 Part 4 Review/The Road Ahead

Harvinder S Jabbal

SE ZG651/ SS ZG653 Software Architectures

BITS Pilani

Pilani|Dubai|Goa|Hyderabad

innovate achieve lead

BITS Pilani

Pilani|Dubai|Goa|Hyderabad

The Road Ahead

Items for Preview

- SOA
- Cloud
- CAP Theorem.

innovate achieve lead

BITS Pilani

Pilani|Dubai|Goa|Hyderabad

SOA

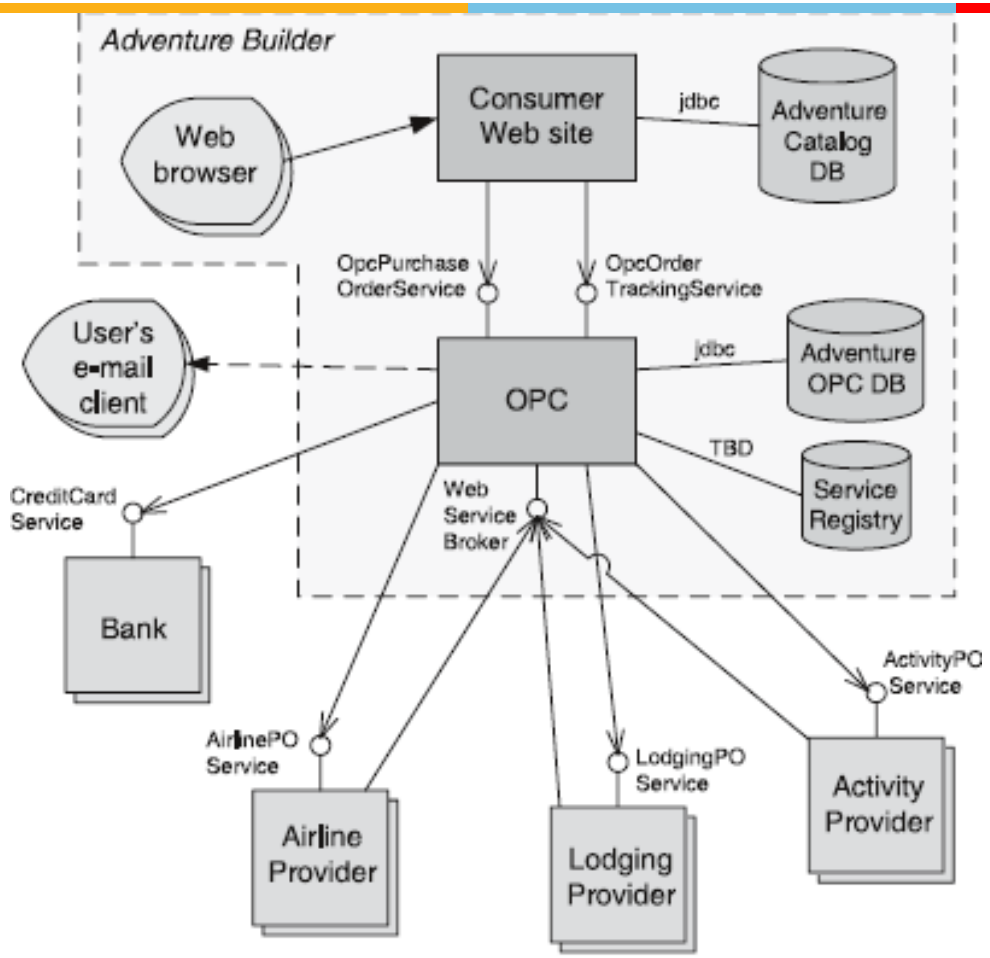
Service Oriented Architecture Pattern

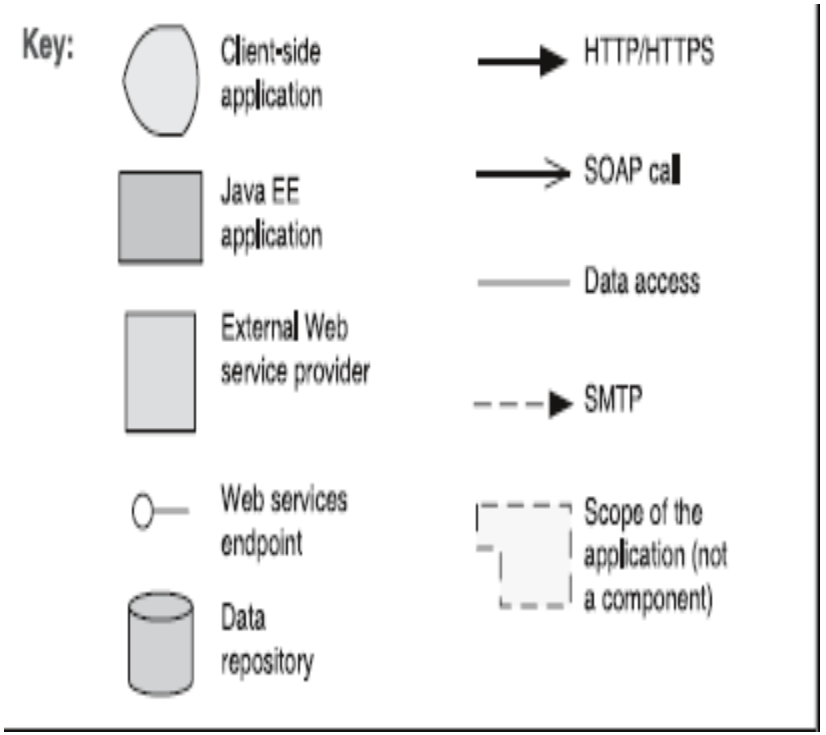
- **Context**: A number of services are offered (and described) by service providers and consumed by service consumers. Service consumers need to be able to understand and use these services without any detailed knowledge of their implementation.

Problem: How can we support interoperability of distributed components running on different platforms and written in different implementation languages, provided by different organizations, and distributed across the Internet?

Solution: The service-oriented architecture (SOA) pattern describes a collection of distributed components that provide and/or consume services.

Service Oriented Architecture Example





Service Oriented Architecture Solution - 1

Overview: Computation is achieved by a set of cooperating components that provide and/or consume services over a network.

Elements:

- Components:
  - *Service providers,* which provide one or more services through published interfaces.
  - *Service consumers,* which invoke services directly or through an intermediary.
  - *Service providers* may also be service consumers.
- *ESB,* which is an intermediary element that can route and transform messages between service providers and consumers.
- *Registry of services,* which may be used by providers to register their services and by consumers to discover services at runtime.
- *Orchestration server,* which coordinates the interactions between service consumers and providers based on languages for business processes and workflows.

Service Oriented Architecture Solution - 2

innovate achieve lead

- Connectors:
  - *SOAP connector,* which uses the SOAP protocol for synchronous communication between web services, typically over HTTP.
  - *REST connector,* which relies on the basic request/reply operations of the HTTP protocol.
  - *Asynchronous messaging connector,* which uses a messaging system to offer point-to-point or publish-subscribe asynchronous message exchanges.

Service Oriented Architecture Solution - 3

Relations: Attachment of the different kinds of components available to the respective connectors

Constraints: Service consumers are connected to service providers, but intermediary components (e.g., ESB, registry, orchestration server) may be used.

Weaknesses:

- SOA-based systems are typically complex to build.
- You don’t control the evolution of independent services.
- There is a performance overhead associated with the middleware, and services may be performance bottlenecks, and typically do not provide performance guarantees.

innovate achieve lead

BITS Pilani

Pilani|Dubai|Goa|Hyderabad

Cloud Computing Deployments Should Begin With Service Definition

- Dennis Smith

Cloud Computing Strategies

- Infrastructure and operations leaders often struggle to understand the role of cloud computing and develop strategies that exploit its potential.
- Infrastructure & Operations leaders should complete the prerequisites before making the technology decisions required for successful, service-centered cloud computing strategies.

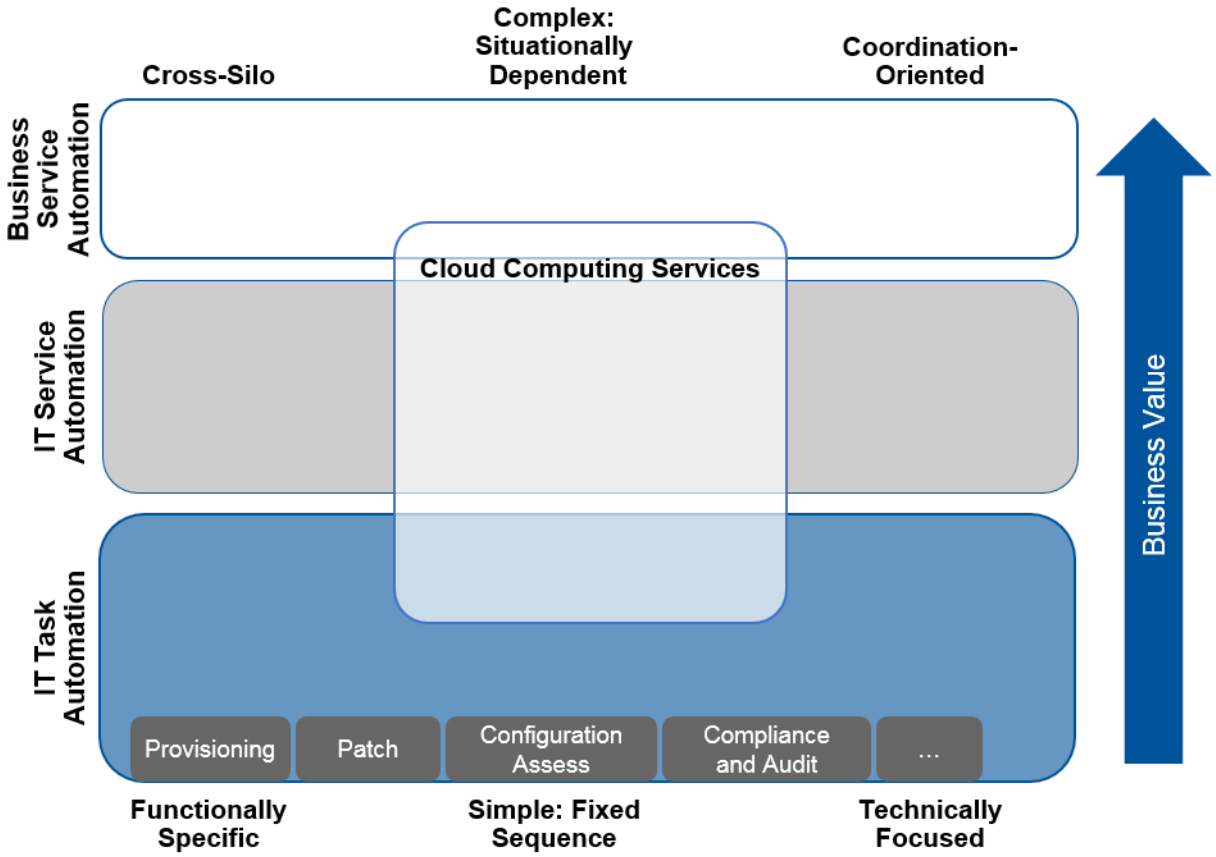
Key Challenges

- Many enterprises have failed to achieve success with cloud computing, because they failed to develop a cloud strategy rooted in the definition and delivery of IT services linked to business outcomes.
- Many companies are unsure how to initiate their cloud projects, which could cause them to miss chances to capitalize on business opportunities.

Recommendations

- Identify the cloud-computing-related IT services you will offer or procure.
- Document the internal processes that will be affected by the identified cloud services.
- Map applications and workloads to the associated cloud services.

Source: Gartner Report (July 2016)



innovate achieve lead

BITS Pilani

Pilani|Dubai|Goa|Hyderabad

CAP Theorem

CAP Theorem

- Described the *trade-offs involved in distributed system*

- It is impossible for a web service to provide following *three guarantees at the same time*:
  - **Consistency**
  - **Availability**
  - **Partition-tolerance**

CAP Theorem

Consistency:

- All nodes should see the same data at the same time

Availability:

- Node failures do not prevent survivors from continuing to operate

Partition-tolerance:

- The system continues to operate despite network partitions

- A distributed system can satisfy any two of these guarantees at the same time **but not all three**

Not Consistent- AP Database

innovate achieve lead

- Data A is being read a node in one partition.
- Data B is being written into a node another partition.

Return Older version of data.

- This assures:

Availability:

  - Node failures do not prevent survivors from continuing to operate

Partition-tolerance:

  - The system continues to operate despite network partitions

- CouchDB, Cassandra, ScyllaDB

Not Available- CP Database

- Data A in one partition is locked while….
- Data B is being written into a node another partition.

Inconsistent Node is not available.

- This assures:

Consistency:

  - All nodes should see the same data at the same time

Partition-tolerance:

  - The system continues to operate despite network partitions

- MongoDB, Redis

Not Partition Tolerant- CA database

- Data A in one partition is being read based on update in …. a node another partition
- Data B is being written into the node in the other partition.

Only theoretical. Not practically CA database currently in use.

- This assures:

Availability:

  - Node failures do not prevent survivors from continuing to operate

Consistency:

  - All nodes should see the same data at the same time

Thank you