Detailed Notes on SOA, Web Services, QoS, SLA, Scalability, and Elasticity

Contents

1 Service-Oriented Architecture (SOA) 2
1.1 Definition 2
1.2 Key Characteristics 2
1.3 Approach 2

2 Web Services 3
2.1 Definition 3
2.2 Key Characteristics 3
2.3 Web Services Description Language (WSDL) 3
2.4 Types of Operations in WSDL 3

3 Quality of Service (QoS) and Service-Level Agreements (SLA) 4
3.1 Quality of Service (QoS) 4
3.1.1 Definition 4
3.1.2 Modern Use 4
3.2 Service-Level Agreement (SLA) 4
3.2.1 Definition 4
3.2.2 Key Components 4

4 Scalability and Elasticity 5
4.1 Scalability 5
4.1.1 Definition 5
4.2 Elasticity 5
4.2.1 Definition 5
4.3 Achieving Scalability and Elasticity 5
4.3.1 Methods 5

5 Dynamic Provisioning in Detail 5

6 Multi-Tenant Design in Detail 6

1 Service-Oriented Architecture (SOA)

1.1 Definition

- Service-Oriented Architecture (SOA) is a software design approach where applications are constructed as a collection of services.
- These services are modular, reusable, and can interact with each other to perform tasks.
- SOA emphasizes flexible design principles for scalable, maintainable, and reusable services.

1.2 Key Characteristics

1. **Collection of Services**:
   - Independent, modular components performing specific functions.
2. **Flexible Design Principles**:
   - Supports scalable and maintainable service development.
3. **Loosely Coupled**:
   - Services can be modified or replaced with minimal impact on others.
4. **Reusable Across Domains**:
   - Services are versatile and reduce development redundancy.

1.3 Approach

- Often implemented using the **Web Service model**.
- Utilizes standard protocols such as HTTP, SOAP, or REST for seamless communication between services.

2 Web Services

2.1 Definition

- Web Services are self-describing, stateless modules accessible over a network.
- They expose functionalities through APIs (Application Programming Interfaces).

2.2 Key Characteristics

1. **Self-Describing**:
   - Defined using standards like WSDL (Web Services Description Language).
2. **Stateless**:
   - No client state information is retained between requests.
3. **Accessible over the Network**:
   - Services are available over the internet.

2.3 Web Services Description Language (WSDL)

- An XML-based language describing the details of a web service.
- Specifies data types, messages, and operations provided by the service.

2.4 Types of Operations in WSDL

1. **One-Way**: A message is sent without expecting a response.
2. **Request & Response**: A client sends a request and waits for a response.
3. **Solicit Response**: The service sends a request and expects a response.
4. **Notification**: A service sends messages without expecting a response.

3 Quality of Service (QoS) and Service-Level Agreements (SLA)

3.1 Quality of Service (QoS)

3.1.1 Definition

- A set of technologies aimed at managing network traffic to improve user experience.

3.1.2 Modern Use

1. **Customer Care Evaluations**:
    - Focuses on QoS from customer experience perspective.
    - Only issues negatively affecting customer satisfaction, such as delays or interruptions in services, are considered.
2. **Technological Evaluations**:
    - Involves assessing the performance of technical systems.
    - Adjustments in procedures or software are made to optimize performance and resource utilization, ensuring smooth and efficient operations.

3.2 Service-Level Agreement (SLA)

3.2.1 Definition

- A formal contract between a service provider and a customer outlining service standards.

3.2.2 Key Components

1. **Performance Guarantee Metrics**:
    - Metrics like up-time ratio, system throughput, and response time.
    - **Up-time and Down-time Ratio**: Specifies the expected availability of the service.
    - **System Throughput**: Measures the amount of data processed by the system in a given time frame.
    - **Response Time**: The time it takes for the system to respond to user requests.
2. **Problem Management**:
    - Details the process for addressing issues.
3. **Penalties for Non-Performance**:
    - Specifies penalties or compensations if the service provider fails to meet agreed performance levels.
4. **Documented Security Capabilities**:

  - Outlines the security measures in place to protect the customer's data and ensure service reliability.

4 Scalability and Elasticity

4.1 Scalability

4.1.1 Definition

- Ability of a system to handle increased work or expand to manage growth.

4.2 Elasticity

4.2.1 Definition

- System's ability to automatically adapt to changing demand by scaling resources up or down.

4.3 Achieving Scalability and Elasticity

4.3.1 Methods

1. **Dynamic Provisioning**:

   - **Definition**: Automatically adjusting resources (like servers or storage) based on current demand.
   - **How it works**: Administrators or systems can allocate or deallocate resources in real-time to match user needs.

2. **Multi-Tenant Design**:

   - A single application serves multiple customers with individualized environments.
   - **Definition**: A software architecture where a single instance of the application serves multiple customers (tenants).
   - **How it works**: Each customer gets a virtual partition of the application, allowing them to customize their experience without affecting others.

5 Dynamic Provisioning in Detail

- **Problem in Traditional Computing**:

  1. Under-Provisioning: Insufficient resources leading to poor performance.
  2. Over-Provisioning: Excess resources leading to wasted costs.

- **Solution**: Adjust resources in real-time to match user demand.

6 Multi-Tenant Design in Detail

- **Definition**: One software application serves multiple customers securely.
- **Features**:
  1. Customization: Tailoring services without affecting others.
  2. Quality of Service: Ensures secure, reliable service.