SFWR ENG 3A04: Software Design II

Dr. Ridha Khedri

Department of Computing and Software, McMaster University
Canada L8S 4L7, Hamilton, Ontario

Term 1

Outline of Part I

1 Model-View-Controller
  - MVC-I
  - MVC-II

2 Presentation-Abstraction-Control (PAC) Architecture

Outline of Part II

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Outline
Part I: Review of
Previous Lecture
Part II: Today's
Lecture

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Model-View-
Controller

Presentation-
Abstraction-
Control (PAC)
Architecture

Part I

Review of Previous Lecture

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Part II

Today's Lecture

Distributed Architecture   Overview

- A distributed system is a collection of computers connected through a communication network

Distributed Architecture   Overview

- A distributed system is a collection of computers connected through a communication network
  - Data is distributed

Distributed Architecture   Overview

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

- A distributed system is a collection of computers connected through a communication network
  - Data is distributed
  - Software is distributed

Distributed Architecture   Overview

- A distributed system is a collection of computers connected through a communication network
  - Data is distributed
  - Software is distributed
  - Users are distributed

Distributed Architecture   Overview

- A distributed system is a collection of computers connected through a communication network
  - Data is distributed
  - Software is distributed
  - Users are distributed

- The sub-systems or components within a distributed system communicate with each other via

Distributed Architecture   Overview

- A distributed system is a collection of computers connected through a communication network
  - Data is distributed
  - Software is distributed
  - Users are distributed

- The sub-systems or components within a distributed system communicate with each other via
  - message passing

Distributed Architecture   Overview

- A distributed system is a collection of computers connected through a communication network
  - Data is distributed
  - Software is distributed
  - Users are distributed

- The sub-systems or components within a distributed system communicate with each other via
  - message passing
  - remote procedure call

Distributed Architecture   Overview

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

- A distributed system is a collection of computers connected through a communication network
  - Data is distributed
  - Software is distributed
  - Users are distributed

- The sub-systems or components within a distributed system communicate with each other via
  - message passing
  - remote procedure call
  - remote method invocation

Distributed Architecture   Overview

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

- A distributed system is a collection of computers connected through a communication network
  - Data is distributed
  - Software is distributed
  - Users are distributed

- The sub-systems or components within a distributed system communicate with each other via
  - message passing
  - remote procedure call
  - remote method invocation
  - etc.

Distributed Architecture   Overview

Two important issues for designing a distributed system are:

- Topology: the way in which entities connect with each other

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture   Overview

Two important issues for designing a distributed system are:

- Topology: the way in which entities connect with each other

- Mode: the method by which they communicate with each other

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture   Overview

Two important issues for designing a distributed system are:

- Topology: the way in which entities connect with each other

- Mode: the method by which they communicate with each other
  - Synchronous

Distributed Architecture   Overview

Two important issues for designing a distributed system are:

- Topology: the way in which entities connect with each other

- Mode: the method by which they communicate with each other
  - Synchronous
  - Asynchronous

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture   Overview

Two important issues for designing a distributed system are:

- Topology: the way in which entities connect with each other

- Mode: the method by which they communicate with each other
    - Synchronous
    - Asynchronous
    - message driven

Distributed Architecture   Overview

Two important issues for designing a distributed system are:

- Topology: the way in which entities connect with each other

- Mode: the method by which they communicate with each other
  - Synchronous
  - Asynchronous
  - message driven
  - callback

Distributed Architecture   Overview

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Two important issues for designing a distributed system are:

- Topology: the way in which entities connect with each other

- Mode: the method by which they communicate with each other
    - Synchronous
    - Asynchronous
    - message driven
    - callback
    - event-driven

Distributed Architecture
Client/Server

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Figure: Examples of network topologies

Distributed Architecture   Overview

- A distributed system can be modeled as a
  - Client/server architecture

Distributed Architecture   Overview

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

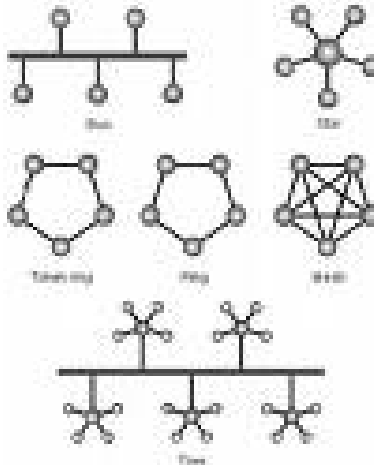Service-Oriented
Architecture (SOA)

- A distributed system can be modeled as a
  - Client/server architecture

  - Broker architecture

Distributed Architecture Overview

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

- A distributed system can be modeled as a
  - Client/server architecture

  - Broker architecture

  - Service-Oriented Architecture (SOA)

Distributed Architecture   Overview

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

- A distributed system can be modeled as a
    - Client/server architecture

    - Broker architecture

    - Service-Oriented Architecture (SOA)

- The important features of a distributed architecture include

Distributed Architecture   Overview

- A distributed system can be modeled as a
  - Client/server architecture

  - Broker architecture

  - Service-Oriented Architecture (SOA)

- The important features of a distributed architecture include

  - its service location transparency

Distributed Architecture   Overview

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

- A distributed system can be modeled as a
  - Client/server architecture

  - Broker architecture

  - Service-Oriented Architecture (SOA)

- The important features of a distributed architecture include

  - its service location transparency

  - service reliability and availability

Distributed Architecture Client/Server

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

- The client-server model is the most common distributed system

- It is based on two communicating subsystems (usually running on different processors)

Distributed Architecture Client/Server

- The client-server model is the most common distributed system

- It is based on two communicating subsystems (usually running on different processors)

  - Client issues a request to the second process server

Distributed Architecture   Client/Server

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

- The client-server model is the most common distributed system

- It is based on two communicating subsystems (usually running on different processors)

    - Client issues a request to the second process server

    - Server process receives the request, carries it out, and sends a reply to the client

Distributed Architecture
Client/Server

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Figure: Two tier client/server architecture

Distributed Architecture   Client/Server

- Advantages

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture   Client/Server

- Advantages
  - Separation of responsibilities such as user interface presentation and business logic processing

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

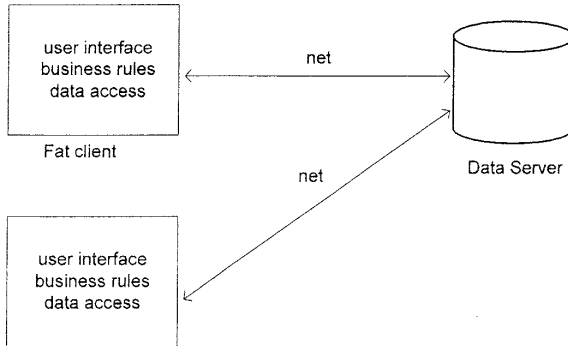# Distributed Architecture   Client/Server

- Advantages
  - Separation of responsibilities such as user interface presentation and business logic processing

  - Reusability of server components

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture   Client/Server

- Advantages
  - Separation of responsibilities such as user interface presentation and business logic processing

  - Reusability of server components

- Disadvantages

Distributed Architecture   Client/Server

- Advantages
  - Separation of responsibilities such as user interface presentation and business logic processing

  - Reusability of server components

- Disadvantages
  - Lack of heterogeneous infrastructure to deal with the requirement changes

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture   Client/Server

- Advantages
  - Separation of responsibilities such as user interface presentation and business logic processing

  - Reusability of server components

- Disadvantages
  - Lack of heterogeneous infrastructure to deal with the requirement changes

  - Security complications

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture   Client/Server

- Advantages
    - Separation of responsibilities such as user interface presentation and business logic processing

    - Reusability of server components

- Disadvantages
    - Lack of heterogeneous infrastructure to deal with the requirement changes

    - Security complications

    - Server availability and reliability

Distributed Architecture   Client/Server

- Advantages
  - Separation of responsibilities such as user interface presentation and business logic processing
  - Reusability of server components

- Disadvantages
  - Lack of heterogeneous infrastructure to deal with the requirement changes
  - Security complications
  - Server availability and reliability
  - Testability and scalability

Distributed Architecture   Client/Server

- Advantages
    - Separation of responsibilities such as user interface presentation and business logic processing
    - Reusability of server components

- Disadvantages
    - Lack of heterogeneous infrastructure to deal with the requirement changes
    - Security complications
    - Server availability and reliability
    - Testability and scalability
    - Fat-clients/ Thin-clients (depends on the application)

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture   Multi-tier

- The front tier in a multi-tier architecture is the user interface presentation tier

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture   Multi-tier

- The front tier in a multi-tier architecture is the user interface presentation tier
- The middle-tier(s) take(s) care of business logic, application decision, and execution

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture   Multi-tier

- The front tier in a multi-tier architecture is the user interface presentation tier
- The middle-tier(s) take(s) care of business logic, application decision, and execution
- The back-end tier usually works on database management, or on a (virtual) machine

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture Multi-tier

- The front tier in a multi-tier architecture is the user interface presentation tier
- The middle-tier(s) take(s) care of business logic, application decision, and execution
- The back-end tier usually works on database management, or on a (virtual) machine
- The advantages of multi-tier over the two-tier architecture are

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture   Multi-tier

- The front tier in a multi-tier architecture is the user interface presentation tier
- The middle-tier(s) take(s) care of business logic, application decision, and execution
- The back-end tier usually works on database management, or on a (virtual) machine
- The advantages of multi-tier over the two-tier architecture are
  - the enhancement of reusability

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture   Multi-tier

- The front tier in a multi-tier architecture is the user interface presentation tier
- The middle-tier(s) take(s) care of business logic, application decision, and execution
- The back-end tier usually works on database management, or on a (virtual) machine
- The advantages of multi-tier over the two-tier architecture are
  - the enhancement of reusability
  - scalability by the middle tier

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture   Multi-tier

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

- The front tier in a multi-tier architecture is the user interface presentation tier
- The middle-tier(s) take(s) care of business logic, application decision, and execution
- The back-end tier usually works on database management, or on a (virtual) machine
- The advantages of multi-tier over the two-tier architecture are
  - the enhancement of reusability
  - scalability by the middle tier
    - The middle tier can also provide multi-threading supports for scalability

Distributed Architecture   Multi-tier

- The front tier in a multi-tier architecture is the user interface presentation tier
- The middle-tier(s) take(s) care of business logic, application decision, and execution
- The back-end tier usually works on database management, or on a (virtual) machine
- The advantages of multi-tier over the two-tier architecture are
    - the enhancement of reusability
    - scalability by the middle tier
        - The middle tier can also provide multi-threading supports for scalability
    - Multi-tier architecture also reduces the traffic on the network

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)
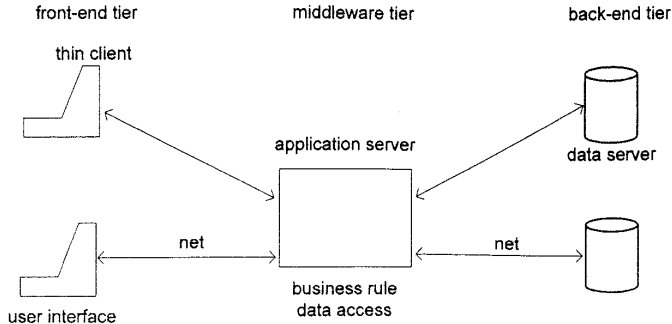
Distributed Architecture   Multi-tier

- The front tier in a multi-tier architecture is the user interface presentation tier
- The middle-tier(s) take(s) care of business logic, application decision, and execution
- The back-end tier usually works on database management, or on a (virtual) machine
- The advantages of multi-tier over the two-tier architecture are
  - the enhancement of reusability
  - scalability by the middle tier
    - The middle tier can also provide multi-threading supports for scalability
  - Multi-tier architecture also reduces the traffic on the network
- Disadvantage: complex testability

Distributed Architecture
Multi-tier

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Figure: Three tier architecture

Distributed Architecture
Broker Architectural Style

- The broker architecture is a middleware architecture widely used in distributed computing

Distributed Architecture
Broker Architectural Style

- The broker architecture is a middleware architecture widely used in distributed computing

- It is suitable for distributed computing that coordinates and facilitates communication

Distributed Architecture
Broker Architectural Style

- The broker architecture is a middleware architecture widely used in distributed computing

- It is suitable for distributed computing that coordinates and facilitates communication
  - brokering the service requests

Distributed Architecture
Broker Architectural Style

- The broker architecture is a middleware architecture widely used in distributed computing

- It is suitable for distributed computing that coordinates and facilitates communication
  - brokering the service requests
  - locating proper server

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

- The broker architecture is a middleware architecture widely used in distributed computing

- It is suitable for distributed computing that coordinates and facilitates communication
  - brokering the service requests
  - locating proper server
  - forwarding and dispatching requests

Distributed Architecture
Broker Architectural Style

- The broker architecture is a middleware architecture widely used in distributed computing

- It is suitable for distributed computing that coordinates and facilitates communication
    - brokering the service requests
    - locating proper server
    - forwarding and dispatching requests
    - sending responses or exceptions back to clients

Distributed Architecture
Broker Architectural Style

- The broker architecture is a middleware architecture widely used in distributed computing

- It is suitable for distributed computing that coordinates and facilitates communication
  - brokering the service requests
  - locating proper server
  - forwarding and dispatching requests
  - sending responses or exceptions back to clients

- It can be used to structure distributed software systems with decoupled components that interact by remote service invocations

Distributed Architecture
Broker Architectural Style

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**

Service-Oriented
Architecture (SOA)

- The broker architecture is a middleware architecture widely used in distributed computing

- It is suitable for distributed computing that coordinates and facilitates communication
    - brokering the service requests
    - locating proper server
    - forwarding and dispatching requests
    - sending responses or exceptions back to clients

- It can be used to structure distributed software systems with decoupled components that interact by remote service invocations

- The most important quality of this architecture = better decoupling between clients and servers

Distributed Architecture
Broker Architectural Style

- Servers make their services available to their clients by registering and publishing their interfaces with the broker

Distributed Architecture
Broker Architectural Style

- Servers make their services available to their clients by registering and publishing their interfaces with the broker

- Clients can request the services of servers from the broker statically or dynamically by look-up

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

- Servers make their services available to their clients by registering and publishing their interfaces with the broker

- Clients can request the services of servers from the broker statically or dynamically by look-up

- A broker acts as a policeman in a busy intersection who controls and interacts with the client components and server components

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

- Servers make their services available to their clients by registering and publishing their interfaces with the broker

- Clients can request the services of servers from the broker statically or dynamically by look-up

- A broker acts as a policeman in a busy intersection who controls and interacts with the client components and server components

- The connection between clients and servers is maintained by the broker

Distributed Architecture
Broker Architectural Style

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**

Service-Oriented
Architecture (SOA)

- A distributed client can access distributed services simply by calling a remote method of a remote object

Distributed Architecture
Broker Architectural Style

- A distributed client can access distributed services simply by calling a remote method of a remote object

- This concept is similar to unix Remote Procedure Call (RPC) and Java Remote Method Invocation (RMI)

Distributed Architecture
Broker Architectural Style

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

- A distributed client can access distributed services simply by calling a remote method of a remote object

- This concept is similar to unix Remote Procedure Call (RPC) and Java Remote Method Invocation (RMI)

- The clients can dynamically invoke the remote methods even if the interfaces of the remote objects are not available at the compilation time

Distributed Architecture
Broker Architectural Style

- Client has a direct connection to its client-proxy

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

- Client has a direct connection to its client-proxy

- Server has direct connection to its server-proxy

Distributed Architecture
Broker Architectural Style

- Client has a direct connection to its client-proxy

- Server has direct connection to its server-proxy

- The proxy talks to the mediator-broker

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

- Client has a direct connection to its client-proxy

- Server has direct connection to its server-proxy

- The proxy talks to the mediator-broker

- The proxy is a well known pattern for hiding low-level detailed communication processing

Distributed Architecture
Broker Architectural Style

- Client has a direct connection to its client-proxy

- Server has direct connection to its server-proxy

- The proxy talks to the mediator-broker

- The proxy is a well known pattern for hiding low-level detailed communication processing
  - It intercepts the client's request

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

- Client has a direct connection to its client-proxy

- Server has direct connection to its server-proxy

- The proxy talks to the mediator-broker

- The proxy is a well known pattern for hiding low-level detailed communication processing
  - It intercepts the client's request
  - gets all arguments

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

- Client has a direct connection to its client-proxy

- Server has direct connection to its server-proxy

- The proxy talks to the mediator-broker

- The proxy is a well known pattern for hiding low-level detailed communication processing
  - It intercepts the client's request
  - gets all arguments
  - packets it

Distributed Architecture
Broker Architectural Style

- Client has a direct connection to its client-proxy

- Server has direct connection to its server-proxy

- The proxy talks to the mediator-broker

- The proxy is a well known pattern for hiding low-level detailed communication processing
  - It intercepts the client's request
  - gets all arguments
  - packets it
  - marshals (streamlines) and formats the package in the format of communication protocol

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**

Service-Oriented
Architecture (SOA)

- Client has a direct connection to its client-proxy

- Server has direct connection to its server-proxy

- The proxy talks to the mediator-broker

- The proxy is a well known pattern for hiding low-level detailed communication processing
    - It intercepts the client's request
    - gets all arguments
    - packets it
    - marshals (streamlines) and formats the package in the format of communication protocol
    - sends it to the broker

Distributed Architecture
Broker Architectural Style

- Client has a direct connection to its client-proxy

- Server has direct connection to its server-proxy

- The proxy talks to the mediator-broker

- The proxy is a well known pattern for hiding low-level detailed communication processing
  - It intercepts the client's request
  - gets all arguments
  - packets it
  - marshals (streamlines) and formats the package in the format of communication protocol
  - sends it to the broker

- A broker system is also called proxy-based system

Distributed Architecture
Broker Architectural Style

Sub-components of a broker architecture

- Broker

Distributed Architecture
Broker Architectural Style

Sub-components of a broker architecture

- Broker
- Stub (client-side proxy): It mediates between client and broker

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

Sub-components of a broker architecture

- Broker
- Stub (client-side proxy): It mediates between client and broker
- Skeleton (server-side proxy)

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

Sub-components of a broker architecture

- Broker
- Stub (client-side proxy): It mediates between client and broker
- Skeleton (server-side proxy)
  - It is statically generated by the service interface compilation and then deployed to the server side

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

Sub-components of a broker architecture

- Broker
- Stub (client-side proxy): It mediates between client and broker
- Skeleton (server-side proxy)
    - It is statically generated by the service interface compilation and then deployed to the server side
    - It receives the requests, unpacks the requests, unmarshals the method arguments, and calls the appropriate service

Distributed Architecture
Broker Architectural Style

Sub-components of a broker architecture

- Broker
- Stub (client-side proxy): It mediates between client and broker
- Skeleton (server-side proxy)
  - It is statically generated by the service interface compilation and then deployed to the server side
  - It receives the requests, unpacks the requests, unmarshals the method arguments, and calls the appropriate service
  - It also marshals results from the sever before it sends it back to the client

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

Sub-components of a broker architecture

- Broker
- Stub (client-side proxy): It mediates between client and broker
- Skeleton (server-side proxy)
  - It is statically generated by the service interface compilation and then deployed to the server side
  - It receives the requests, unpacks the requests, unmarshals the method arguments, and calls the appropriate service
  - It also marshals results from the sever before it sends it back to the client
- Bridges (Optional)

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

Sub-components of a broker architecture

- Broker
- Stub (client-side proxy): It mediates between client and broker
- Skeleton (server-side proxy)
  - It is statically generated by the service interface compilation and then deployed to the server side
  - It receives the requests, unpacks the requests, unmarshals the method arguments, and calls the appropriate service
  - It also marshals results from the sever before it sends it back to the client
- Bridges (Optional)
  - Used to hide implementation details when two brokers interoperate

Distributed Architecture
Broker Architectural Style

Sub-components of a broker architecture

- Broker
- Stub (client-side proxy): It mediates between client and broker
- Skeleton (server-side proxy)
  - It is statically generated by the service interface compilation and then deployed to the server side
  - It receives the requests, unpacks the requests, unmarshals the method arguments, and calls the appropriate service
  - It also marshals results from the sever before it sends it back to the client
- Bridges (Optional)
  - Used to hide implementation details when two brokers interoperate
  - Can connect two different networks based on different communication protocols

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**

Service-Oriented
Architecture (SOA)

Figure: Broker model

Distributed Architecture
Broker Architectural Style

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**

Service-Oriented
Architecture (SOA)

Figure: Connected brokers with client/server proxy

Distributed Architecture
Broker Architectural Style

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**
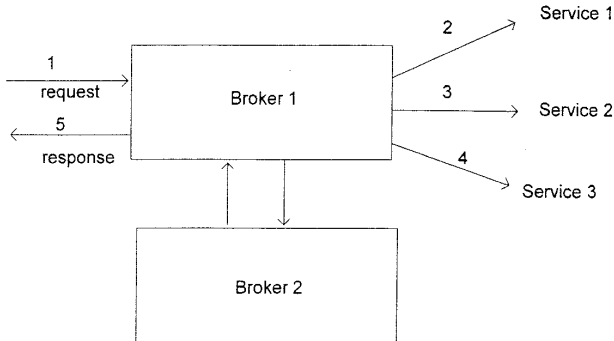
Service-Oriented
Architecture (SOA)

Figure: Class diagram for broker architecture

Distributed Architecture
Broker Architectural Style

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**
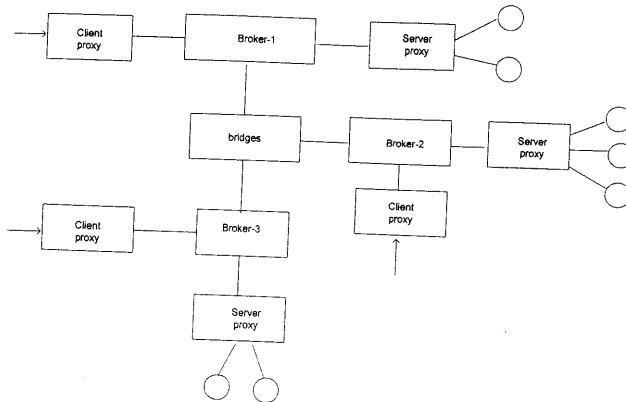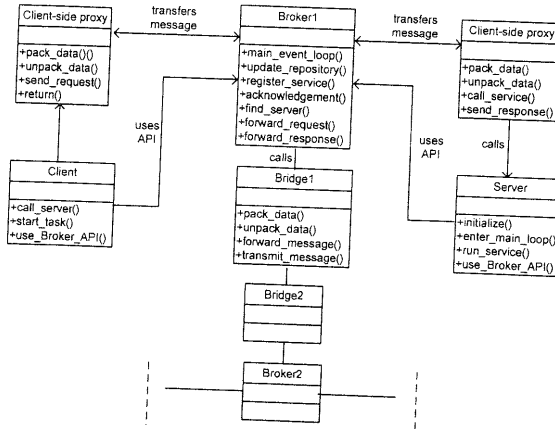
Service-Oriented
Architecture (SOA)

Figure: Sequence diagram for broker architecture

Distributed Architecture
Broker Architectural Style

- Advantages

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

- Advantages
  - Server component implementation and location transparency

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
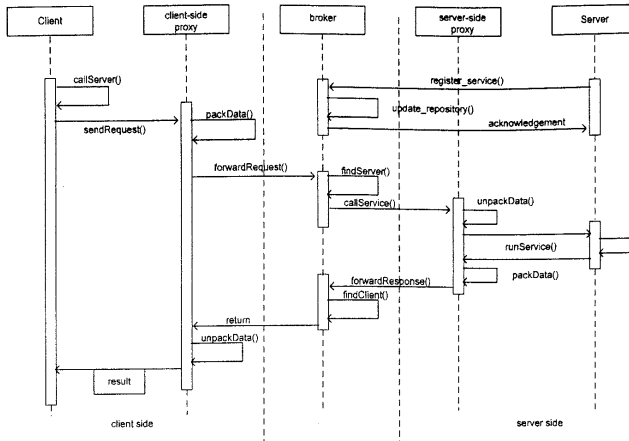Architectural Style**

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

- Advantages
  - Server component implementation and location transparency
  - Changeability and extensibility

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

- Advantages
    - Server component implementation and location transparency
    - Changeability and extensibility
    - Simplicity for clients to access server and server portability

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

- Advantages
    - Server component implementation and location transparency
    - Changeability and extensibility
    - Simplicity for clients to access server and server portability
    - Interoperability via broker bridges

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

- Advantages
  - Server component implementation and location transparency
  - Changeability and extensibility
  - Simplicity for clients to access server and server portability
  - Interoperability via broker bridges
  - Reusability

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**

Service-Oriented
Architecture (SOA)

Distributed Architecture
Broker Architectural Style

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**

Service-Oriented
Architecture (SOA)

- Advantages
  - Server component implementation and location transparency
  - Changeability and extensibility
  - Simplicity for clients to access server and server portability
  - Interoperability via broker bridges
  - Reusability
  - Feasibility of runtime changes of server components (add or remove server components)

Distributed Architecture
Broker Architectural Style

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**

Service-Oriented
Architecture (SOA)

- Advantages
  - Server component implementation and location transparency
  - Changeability and extensibility
  - Simplicity for clients to access server and server portability
  - Interoperability via broker bridges
  - Reusability
  - Feasibility of runtime changes of server components (add or remove server components)

- Disadvantages

Distributed Architecture
Broker Architectural Style

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**

Service-Oriented
Architecture (SOA)

- Advantages
  - Server component implementation and location transparency
  - Changeability and extensibility
  - Simplicity for clients to access server and server portability
  - Interoperability via broker bridges
  - Reusability
  - Feasibility of runtime changes of server components (add or remove server components)

- Disadvantages
  - Inefficiency due to the overhead of proxies

Distributed Architecture
Broker Architectural Style

- Advantages
    - Server component implementation and location transparency
    - Changeability and extensibility
    - Simplicity for clients to access server and server portability
    - Interoperability via broker bridges
    - Reusability
    - Feasibility of runtime changes of server components (add or remove server components)

- Disadvantages
    - Inefficiency due to the overhead of proxies
    - Low fault-tolerance

Distributed Architecture
Broker Architectural Style

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

**Broker
Architectural Style**

Service-Oriented
Architecture (SOA)

- Advantages
    - Server component implementation and location transparency
    - Changeability and extensibility
    - Simplicity for clients to access server and server portability
    - Interoperability via broker bridges
    - Reusability
    - Feasibility of runtime changes of server components (add or remove server components)

- Disadvantages
    - Inefficiency due to the overhead of proxies
    - Low fault-tolerance
    - Difficulty in testing

Distributed Architecture
Service-Oriented Architecture (SOA)

- A service is a business functionality that is

Distributed Architecture
Service-Oriented Architecture (SOA)

- A service is a business functionality that is
  - well-defined and self-contained

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Service-Oriented Architecture (SOA)

- A service is a business functionality that is
  - well-defined and self-contained
  - independent from other services

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Service-Oriented Architecture (SOA)

- A service is a business functionality that is
  - well-defined and self-contained
  - independent from other services
  - published and available to be used via an interface

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Service-Oriented Architecture (SOA)

- A service is a business functionality that is
  - well-defined and self-contained
  - independent from other services
  - published and available to be used via an interface

- SOA services can be reused extensively regardless of whether they are based on new or legacy applications

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Service-Oriented Architecture (SOA)

- A service is a business functionality that is
  - well-defined and self-contained
  - independent from other services
  - published and available to be used via an interface

- SOA services can be reused extensively regardless of whether they are based on new or legacy applications
- Loose coupling of service-orientation architecture provides a great flexibility for enterprises to make use of all available service resourses

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Service-Oriented Architecture (SOA)

- A service is a business functionality that is
  - well-defined and self-contained
  - independent from other services
  - published and available to be used via an interface

- SOA services can be reused extensively regardless of whether they are based on new or legacy applications
- Loose coupling of service-orientation architecture provides a great flexibility for enterprises to make use of all available service resourses

- The connections between services are conducted by common and universal message oriented protocols such as the SOAP Web service protocol

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Service-Oriented Architecture (SOA)

- A service is a business functionality that is
  - well-defined and self-contained
  - independent from other services
  - published and available to be used via an interface

- SOA services can be reused extensively regardless of whether they are based on new or legacy applications
- Loose coupling of service-orientation architecture provides a great flexibility for enterprises to make use of all available service resources

- The connections between services are conducted by common and universal message oriented protocols such as the SOAP Web service protocol
- A connection can be established statically or dynamically

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

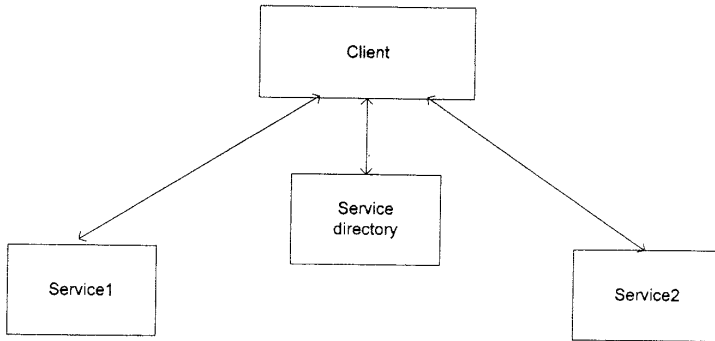# Distributed Architecture
Service-Oriented Architecture (SOA)

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Figure: Client with services and service directory

Distributed Architecture
Service-Oriented Architecture (SOA)

- A service-oriented application might make use of many available services

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

**Service-Oriented
Architecture (SOA)**

Distributed Architecture
Service-Oriented Architecture (SOA)

- A service-oriented application might make use of many available services
- For that one needs a flow control language

Distributed Architecture
Service-Oriented Architecture (SOA)

- A service-oriented application might make use of many available services
- For that one needs a flow control language
  - allows specifying the sequence and logical order of the business executions based on the business logic

Distributed Architecture
Service-Oriented Architecture (SOA)

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

- A service-oriented application might make use of many available services
- For that one needs a flow control language
  - allows specifying the sequence and logical order of the business executions based on the business logic

- Some services can be reused by other applications that they are not originally designed for

Distributed Architecture
Service-Oriented Architecture (SOA)

- A service-oriented application might make use of many available services
- For that one needs a flow control language
  - allows specifying the sequence and logical order of the business executions based on the business logic
- Some services can be reused by other applications that they are not originally designed for
- We can build a new service out of existing services

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Service-Oriented Architecture (SOA)

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

**Service-Oriented
Architecture (SOA)**

- A service-oriented application might make use of many available services
- For that one needs a flow control language
  - allows specifying the sequence and logical order of the business executions based on the business logic
- Some services can be reused by other applications that they are not originally designed for
- We can build a new service out of existing services
  - aggregation: extends one endpoint of a service to make a new interface of the new service

Distributed Architecture
Service-Oriented Architecture (SOA)

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

**Service-Oriented
Architecture (SOA)**

- A service-oriented application might make use of many available services
- For that one needs a flow control language
  - allows specifying the sequence and logical order of the business executions based on the business logic
- Some services can be reused by other applications that they are not originally designed for
- We can build a new service out of existing services
  - aggregation: extends one endpoint of a service to make a new interface of the new service
  - containment structure: has one interface that wraps all used services

Distributed Architecture
Service-Oriented Architecture (SOA)

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

- A service-oriented application might make use of many available services
- For that one needs a flow control language
  - allows specifying the sequence and logical order of the business executions based on the business logic
- Some services can be reused by other applications that they are not originally designed for
- We can build a new service out of existing services
  - aggregation: extends one endpoint of a service to make a new interface of the new service
  - containment structure: has one interface that wraps all used services
- Services can be recursively constructed to satisfy a more complex business needs (through aggr. and cont.)

Distributed Architecture
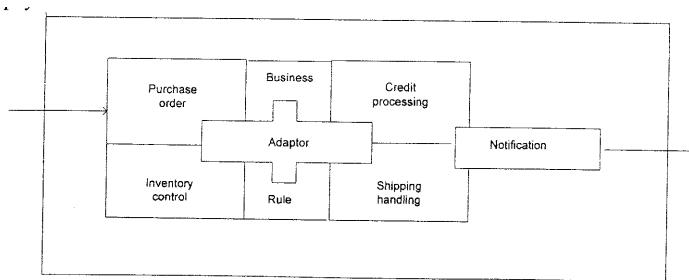## Service-Oriented Architecture (SOA)

Figure: Service composition
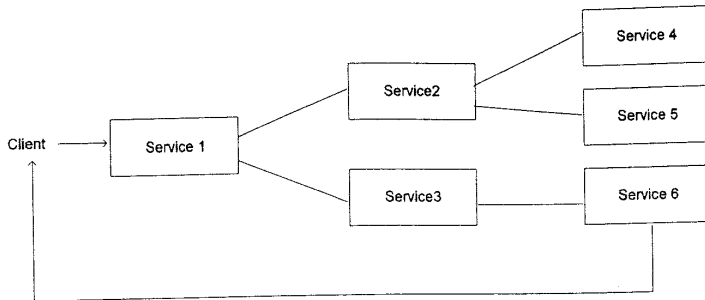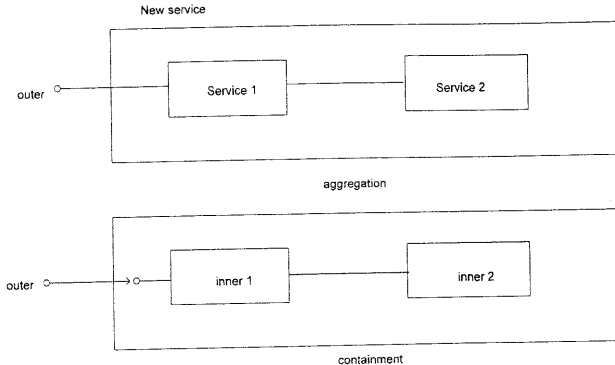
Distributed Architecture
Service-Oriented Architecture (SOA)

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Figure: Service reuse

Distributed Architecture
Service-Oriented Architecture (SOA)
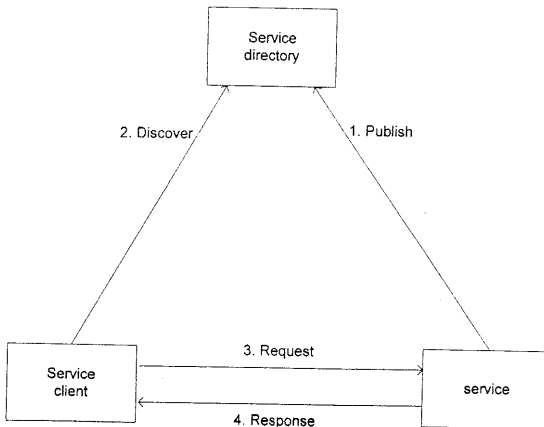
SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Figure: Service composition model

Distributed Architecture
Service-Oriented Architecture (SOA)

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Figure: Service working model

Distributed Architecture
Service-Oriented Architecture (SOA)

Advantages of SOA
- Loosely-coupled connection

Distributed Architecture
Service-Oriented Architecture (SOA)

Advantages of SOA

- Loosely-coupled connection

- Each service component is independent from other services due to the stateless service feature

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Service-Oriented Architecture (SOA)

Advantages of SOA

- Loosely-coupled connection

- Each service component is independent from other services due to the stateless service feature

- Interoperability: Technically any client or any service can access other services regardless of their platform, technology, vendors, or language implementations

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)

Distributed Architecture
Service-Oriented Architecture (SOA)

Advantages of SOA

- Loosely-coupled connection

- Each service component is independent from other services due to the stateless service feature

- Interoperability: Technically any client or any service can access other services regardless of their platform, technology, vendors, or language implementations

- Reusability: Any service can be reused by any other services and service is developed to be reused as well

Distributed Architecture
Service-Oriented Architecture (SOA)

Advantages of SOA

- Loosely-coupled connection

- Each service component is independent from other services due to the stateless service feature

- Interoperability: Technically any client or any service can access other services regardless of their platform, technology, vendors, or language implementations

- Reusability: Any service can be reused by any other services and service is developed to be reused as well

- Scalability: Loosely coupled services make themselves easy to scale

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Client/Server

Multi-tier

Broker
Architectural Style

Service-Oriented
Architecture (SOA)