# Service Layers

UNIT-III

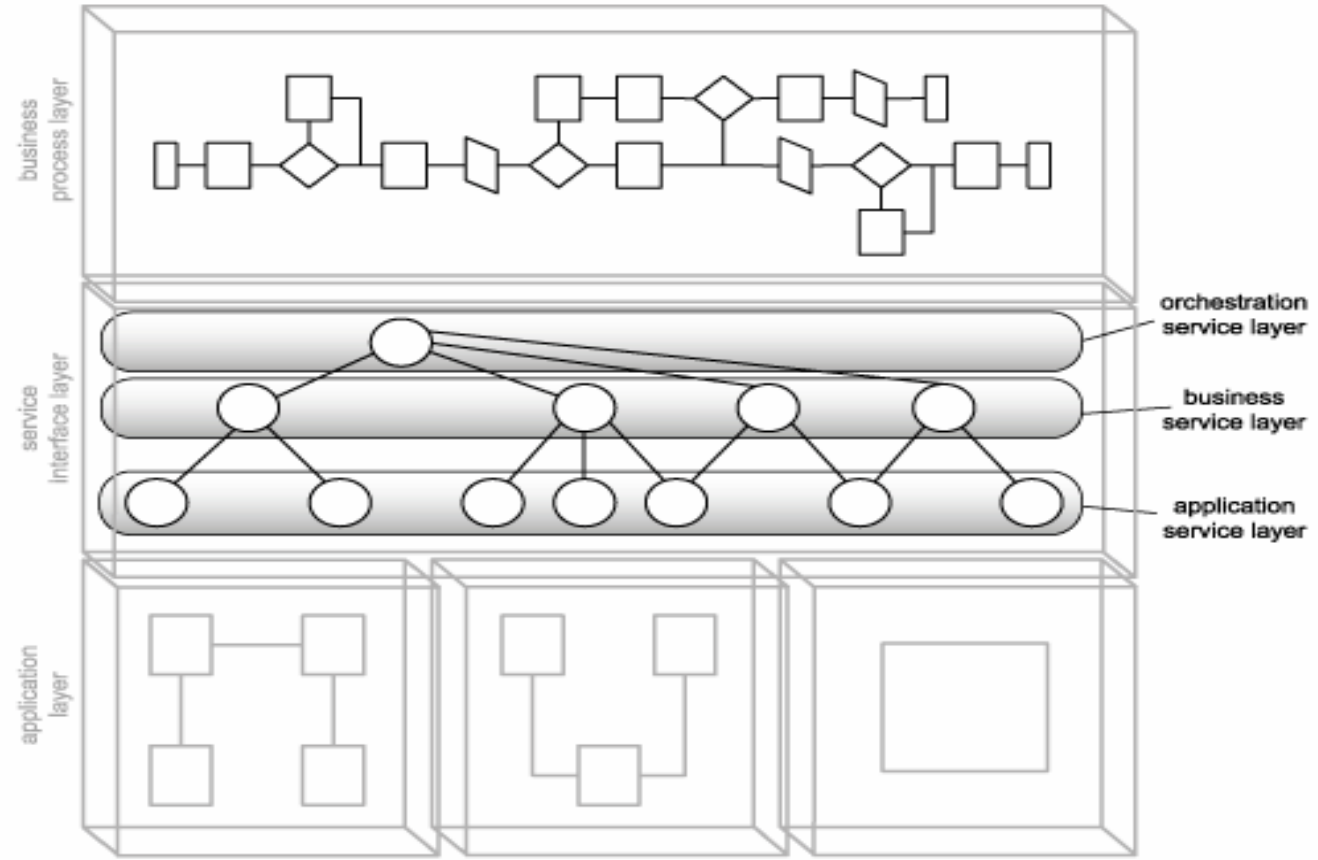
# Service Layer Abstraction

- Service layer positioned between business process and application layers
- Place where service connectivity resides in enterprise
- Where principles of service orientation prevalent
- Layering solves many issues in achieving this

- Need for seperate business and application layer:
- Business logic – represent core logic specific to the business domain
- Application logic – technology part that helps to execute business logic eg. Utility service, wrapper service

- **Separate layers of business and application logic allows loosely coupling and evolve independently**

# Service Layers



# Service Layers Contd...

- Application service layer
  - to address legacy application logic to be exposed or new logic developed to support of services
- Business service layer
  - responsible for creating service representing business logic be aligned with existing business model
- Orchestration Layer
  - to promote agility
  - a controller layer on top of more specialized service layers, defines composition logic, the sequence which service are executed

# Application Service Layer

- Services in this layer express technology-specific functionality
- Called as **application services**
- Provide reusable functionality to legacy / new application

- **Characteristics:**
  - Expose specific functionality to that context
  - Draws resources within given platform
  - Solution agnostic
  - Point-to-point integration can be acheived
  - Interface granularity is inconsistent
  - Consist mixture of custom+3rd party services

# Contd...

- Example: utility service, wrapper service
- When a separate business service layer exists, need to make application services into generic utility services –solution-agnostic and reusable
- if business logic does not reside in a separate layer, application services implement service models associated with the business service layer
- Services that contain both application and business logic referred to as **hybrid application services** – *not desirable, found in* traditional distributed architectures

# Contd...

- Application services  can be aggregated or solely enable integration  between system (*application integration services)*
- Wrapper services often used for integration purposes
- Ex. service adapter provided by legacy vendors
- variation of the wrapper service model proxy service, also called auto-generated WSDL

# Business Service Layer

- Business services implements of business service model
- Purpose - to represent business logic in the purest form (without application logic)
- Can also implement service model – Ex. a business service also can be classified as a controller service and a utility service

- **Business service layer abstraction**
  - *Task-centric business service*
  - *Entity-centric business service*

# Contd...

- **Task-centric business service**
  - A service that encapsulates business logic specific to a task or business process
  - business process logic is not centralized as part of an orchestration layer, so limited reuse
- **Entity-centric business service**
  - A service that encapsulates a specific business entity
  - highly reusable and services are composed by an orchestration layer or by a service layer consisting of task-centric business services (or both)

  - When a separate application service layer exists, two types of business services can be positioned to compose application services to carry out their business logic

# Orchestration service layer

- Orchestration allows to directly link process logic to service interaction within given workflow logic
- Orchestration service layer introduces a parent level of abstraction for other services (as a controller)
- It ensure that service operations are executed in a specific sequence / workflow
- Language used – WS-BPEL