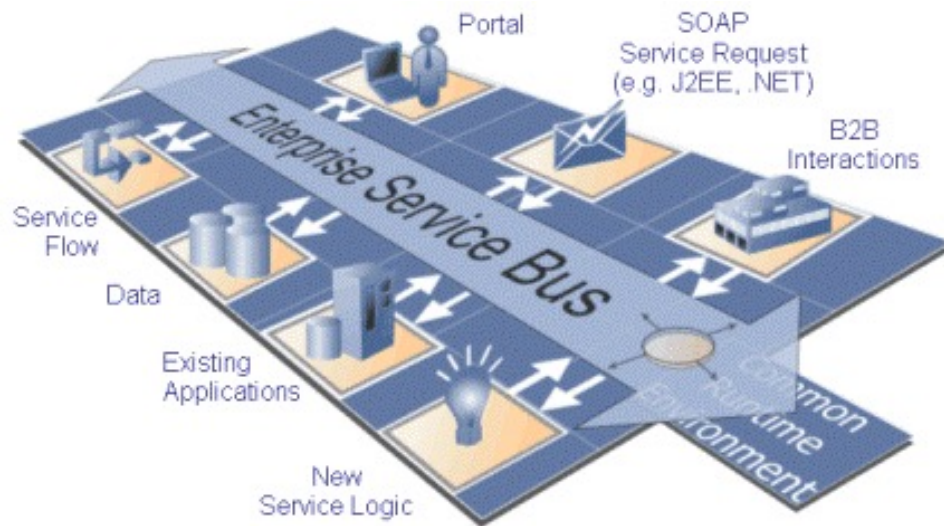


Enterprise Service

► Bus

Diogo Gomes dgomes@ua.pt

ESB ?

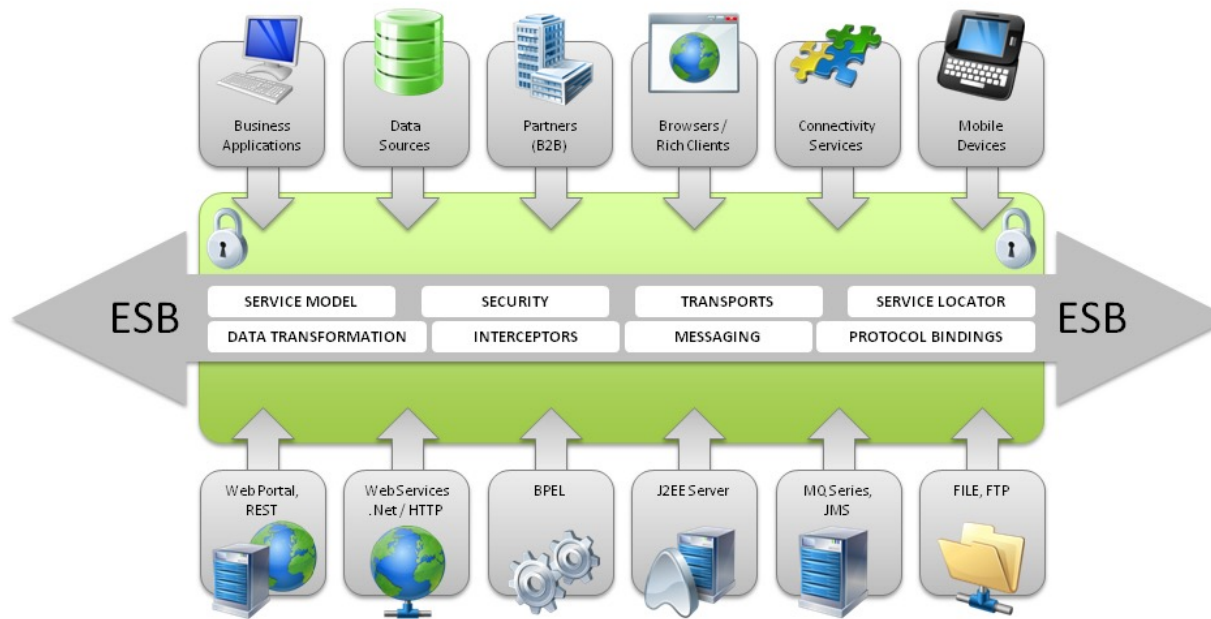


- ▶ An Enterprise Service Bus (ESB) is a software architecture for a middleware that provides fundamental services for more complex architectures.
- ▶ ESB can be used to implement SOA architectures
- ▶ ESB manages access to applications and services to present a single, simple, and consistent interface to end-users/services

Functionalities

- ▶ Distributes information across an enterprise quickly and easily.
- ▶ Masks differences among underlying platforms, software architectures, and network protocols.
- ▶ Ensures information delivery even when some systems or networks may go off-line.
- ▶ Re-routes, logs, and enriches information without requiring applications to be rewritten.
- ▶ Provide incremental solution implementations so all enterprise services and applications do not need to be changed immediately or all at once.

ESB Architecture



Capabilities

► Routing

- The ability to channel a request to a particular service provider based on deterministic or variable criteria.
- Types of routing to consider:
 - Static or deterministic Routing
 - Content-based Routing
 - Policy-based Routing
 - Complex Rules-based Routing

► Message Transformation

- The ability to convert the structure and format of the incoming business service request to the structure and format expected by the service provider

Capabilities

- ▶ Message Enhancement
- ▶ Protocol Transformation
- ▶ Service Mapping
- ▶ Message Processing
- ▶ Process Choreography
- ▶ Service Orchestration
- ▶ Transaction Management
- ▶ Security



Capabilities

- ▶ Message Enhancement

- ▶ The ability to add or modify the information contained in the message as required by the service provider.
- ▶ Types of Message Enhancement:
 - ▶ Date format conversion
 - ▶ Supplement data not included in original message
 - ▶ Data conversion(i.e. spaces to zero)
 - ▶ Rules-based enhancement

Capabilities

► Protocol Transformation

- The ability to accept one type of protocol from the consumer as input (i.e. SOAP/JMS) and communicate to the service provider through a different protocol (i.e. IIOP)
- A form of message transformation concerned with the message structure, not the message payload.
- Has both physical connection attributes as well as logical connectivity attributes
- Examples:
 - SOAP/JMS □ IIOP
 - XML/HTTP □ CICS MQ
 - XML/HTTP □ RMI/IIOP

Capabilities

► Service Mapping

- The ability to translate a business service into the corresponding service implementation and provide binding and location information
- Could be implemented through XML, a database, or embedded within the Mediator ESB component
- Usually contains the following core information
 - Implementation Service Name
 - Service Protocol and binding information
 - Protocol specific info (i.e. timeouts, failover location)
 - Service-specific routing information

► Message Processing

- The ability to manage state and perform request management by accepting an input request and ensuring delivery back to the client via message synchronization

Capabilities

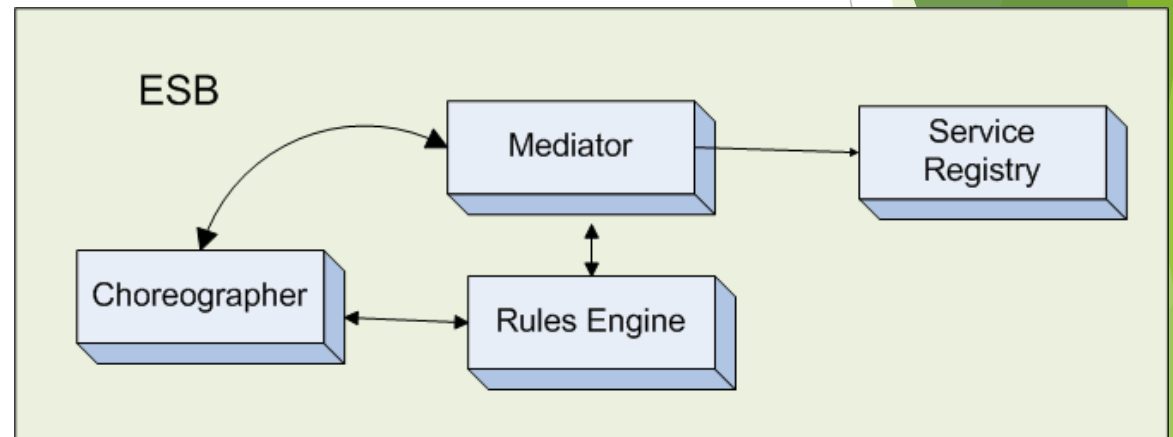
- ▶ Process Choreography
 - ▶ The ability to manage complex business process that requires the coordination of multiple business service to fulfill a single business service request
- ▶ Service Orchestration
 - ▶ The ability to manage the coordination of multiple implementation services
- ▶ Transaction Management
 - ▶ The ability to provide a single unit of work for a business service request by providing a framework for the co-ordination of multiple resources across multiple disparate services

Capabilities

- ▶ Security
 - ▶ The ability to protect enterprise services from unauthorized access
 - ▶ In SOA there are no more silos; services become visible to the entire enterprise through ESB
 - ▶ ESB should provide A4C mechanisms
 - ▶ Authentication, Authorization, Auditing, Accounting, Charging
 - ▶ ESB should access a security manager or authentication and authorization rather than have the direct responsibility

Components

- ▶ An ESB can be broken down into the following components
 - ▶ Mediator
 - ▶ Service Registry
 - ▶ Choreographer
 - ▶ Rules Engine



Mediator

- ▶ Routing
- ▶ Communication
- ▶ Message Transformation
- ▶ Message Enhancement
- ▶ Protocol Transformation
- ▶ Message Processing
- ▶ Error Handling
- ▶ Service Orchestration
- ▶ Transaction Management
- ▶ Security



Service Registry

- ▶ Service Catalog
- ▶ Provides Service Mapping
- ▶ Enables Service Discovery
- ▶ UDDI (Universal Description, Discovery and Integration)



Choreography

- ▶ Message Processing
- ▶ Process Choreography
- ▶ Transaction Management
- ▶ Security



Rules Engine

- ▶ Routing
- ▶ Message Transformation
- ▶ Message Enhancement

