

# ESB

Norberto Fernández  
Departamento de Ingeniería Telemática  
<http://www.it.uc3m.es/berto/>



Tecnologías de Distribución de  
Contenidos - UC3M



1

## Motivación

- EAI (Enterprise Application Integration)
  - Una organización tiene distintas suborganizaciones con distintos sistemas
  - Colaboraciones entre empresas (ej: externalización de servicios como CRM)
  - Fusiones corporativas (ej: en España las cajas de ahorros)
  - Evolución (migración de sistemas legado -legacy- a otros más modernos)

Tecnologías de Distribución de  
Contenidos - UC3M

2

## Motivación

- N sistemas distintos que hay que interconectar (integrar)...
  - Integración de datos
  - Integración de procesos
- Problemas:
  - N puede ser grande
  - Heterogeneidad
  - Distribución
  - El time-to-market debe ser reducido

Tecnologías de Distribución de  
Contenidos - UC3M

3

## Enfoques a EAI (I)

- Point-to-point
  - Cada aplicación habla directamente con todas las otras con las que interactúa
- Problemas
  - La solución es específica de cada canal
    - Difícil de mantener ¿qué pasa si una aplicación nueva?
  - N aplicaciones máximo de  $N(N-1)/2$  conexiones
  - Alto acoplamiento entre aplicaciones
    - ¿Qué pasa si hay que hacer un cambio?

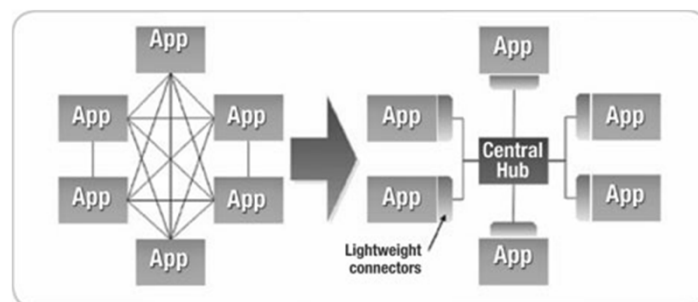
Tecnologías de Distribución de  
Contenidos - UC3M

4

## Enfoques a EAI (II)

- Hub-and-spoke
  - Centralizada en un nodo gestor “*broker*”
  - Uso de interfaces estandarizadas
    - Reduce acoplamiento, la interfaz puede abstraerlo
  - Fácil integrar nuevos sistemas (conectarlos al *broker*)
    - Cómo máximo N conexiones con *broker*
- Problema
  - *Broker* = Punto central de fallo / cuello de botella
  - Más tiempo de procesamiento que conexión directa
    - Procesamiento en el *broker*

## Enfoques a EAI (III)



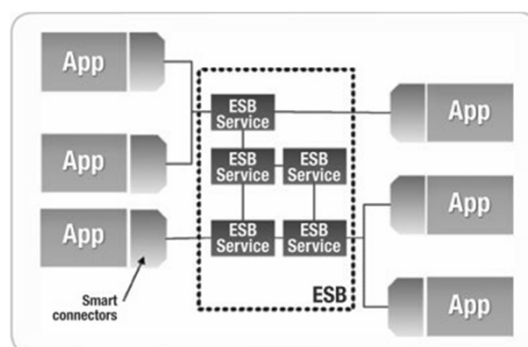
**f1** Figure 1: Point-to-point and hub-and-spoke architecture

Fuente: <http://gemsres.com/story/aug05/121831/bakker-fig1.jpg>

## Enfoques a EAI (IV)

- Bus (*Enterprise Service Bus*, ESB)
  - Uso de interfaces
  - Fácil integrar sistemas (conectar al bus)
  - No hay punto de fallo central
- Problema:
  - Necesario coordinar de algún modo los distintos elementos

## Enfoques a EAI (V)



 **Figure 2:** The enterprise service bus architecture

Fuente: <http://gemsres.com/story/aug05/121831/bakker-fig2.jpg>

## Bus vs Hub-and-spoke

- Bus es evolución de *hub-and-spoke*
  - Bus conjunto de componentes interconectados
    - Que son a su vez servicios
- Los dos ofrecen servicios básicos
  - Ej.: Transacciones involucrando varios elementos
- ESB hace énfasis en SOA
  - Basado en estándares, *hub-and-spoke* solían ser soluciones propietarias

## ESB: ¿Qué es?

- No hay una única definición comúnmente aceptada (*buzzword*)

*“An Enterprise Service Bus (ESB) is software infrastructure that enables SOA by acting as an intermediary layer of middleware through which a set of reusable business services are made widely available.”*

Fuente: Mike Gilpin, Forrester Research, August 2004

## ESB: ¿Qué es?

*“An Enterprise Service Bus (ESB) is a distributed infrastructure used **for enterprise integration**. It consists of a set of service containers, which integrate various types of IT assets. The containers are interconnected with a reliable messaging bus. Service containers adapt IT assets to a **standard services model**, based on XML message exchange using standardized message exchange patterns. The ESB **provides services** for transforming and routing messages, as well as the ability to centrally administer the distributed system.”*

Fuente: [https://blogs.oracle.com/rtenhove/entry/what is enterprise service bus](https://blogs.oracle.com/rtenhove/entry/what_is_enterprise_service_bus)

## ESB: ¿Qué es?

- Uso ambiguo...
  - A) Un patrón de arquitectura de aplicaciones empresariales
  - B) Herramientas que implementan ese patrón arquitectónico...
    - Comerciales: Microsoft BizTalk Server, Oracle Enterprise Service Bus, IBM WebSphere ESB, TIBCO ActiveMatrix, etc
    - Código libre: Open ESB, Apache ServiceMix, Mule ESB, Apache Synapse, etc

## Funcionalidades básicas (I)

- **Transparencia de localización**
  - El consumidor de un servicio no tiene necesidad de conocer la localización del proveedor
- **Conversión de protocolos**
  - HTTP, TCP, SMTP, FTP, ...
- **Transformación de mensajes**
  - Por ejemplo usando XSLT. Objetivos:
    - Añadir información para el consumidor del mensaje
    - Adaptar el formato

## Funcionalidades básicas (II)

- **Enrutamiento de mensajes**
  - En base al contenido, seleccionar dinámicamente destino
- **Aspectos de calidad de servicio**
  - Seguridad
    - Autenticación, autorización, cifrado, etc.
  - Mensajería fiable

## Funcionalidades básicas (III)

- Soporte a procesos de negocio
  - Orquestración de servicios (WS-BPEL)
  - Transacciones involucrando varios servicios
- Apertura
  - Posibilidad de conectar el ESB y sus servicios con otros ESBs
- Monitorización y gestión

## Referencias

- **Enterprise Service Bus: Theory in Practice**
  - David Chappell
  - Editorial O'Reilly, 2004
  - ISBN: 978-0596006754
- **Open-Source ESBs in Action**
  - Tijs Rademakers, Jos Dirksen
  - Editorial Manning Publications, 2008
  - ISBN: 978-1933988214

