



ARQUITECTURA ORIENTADA A SERVICIOS - SOA

DSD



Arquitectura?

- Componentes
- Relaciones
- Mensajes
- Plan para construir
- Requisitos
- Actual y a futuro

Definiciones: Software Engineering Institute (SEI)
www.sei.cmu.edu/architecture/definitions.html .



Propósito de una Arquitectura

- Un propósito fundamental de la arquitectura de software es ayudar a gestionar la complejidad de los sistemas de software y las modificaciones que inevitablemente experimentan los sistemas en respuesta a cambios externos en los entornos empresarial, organizativo y técnico.

Estilos de Arquitectura

- Un estilo arquitectónico contiene un conjunto bien definido de patrones que constituyen una forma común para que los componentes de la solución empresarial interactúen entre sí.





Figura 8-1. Departamentos conducen a sistemas monolíticos.

Departamental a Distribuido

Tener procesos distribuidos significa que se necesita de:

- Planificación distribuida
- diseño distribuido
- realización distribuida
- operación distribuida.

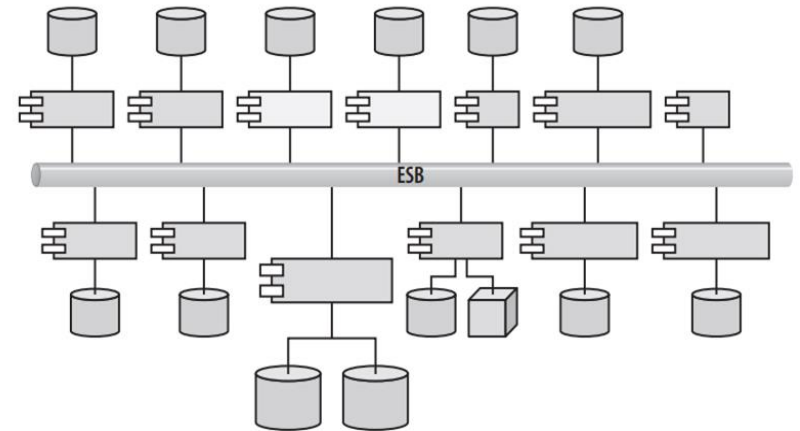
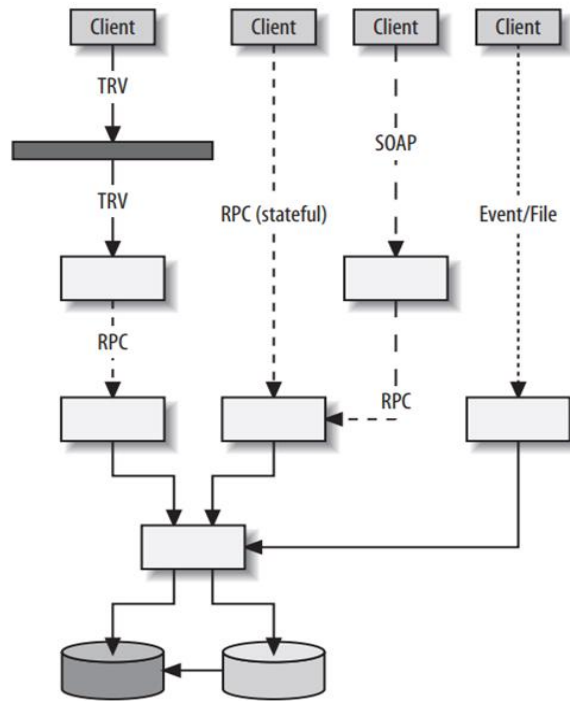
La integración y la distribución son cada vez más importantes



Problemática:

- Heterogeneidad y descentralización
- Cambios con altos impactos y caros
- Islas de Apps, desde el diseño
- La modularización extensible que no existe
- Proliferación de datos redundantes y lógica también

La realidad...



SOA (Service Oriented Architecture)

“Estilo arquitectónico que promueve el concepto de un **servicio** empresarial **alineado con el negocio** como la unidad fundamental de diseño, construcción y composición de soluciones empresariales.

Múltiples patrones que describen definiciones, implementaciones y despliegue de las soluciones SOA completan este estilo.”

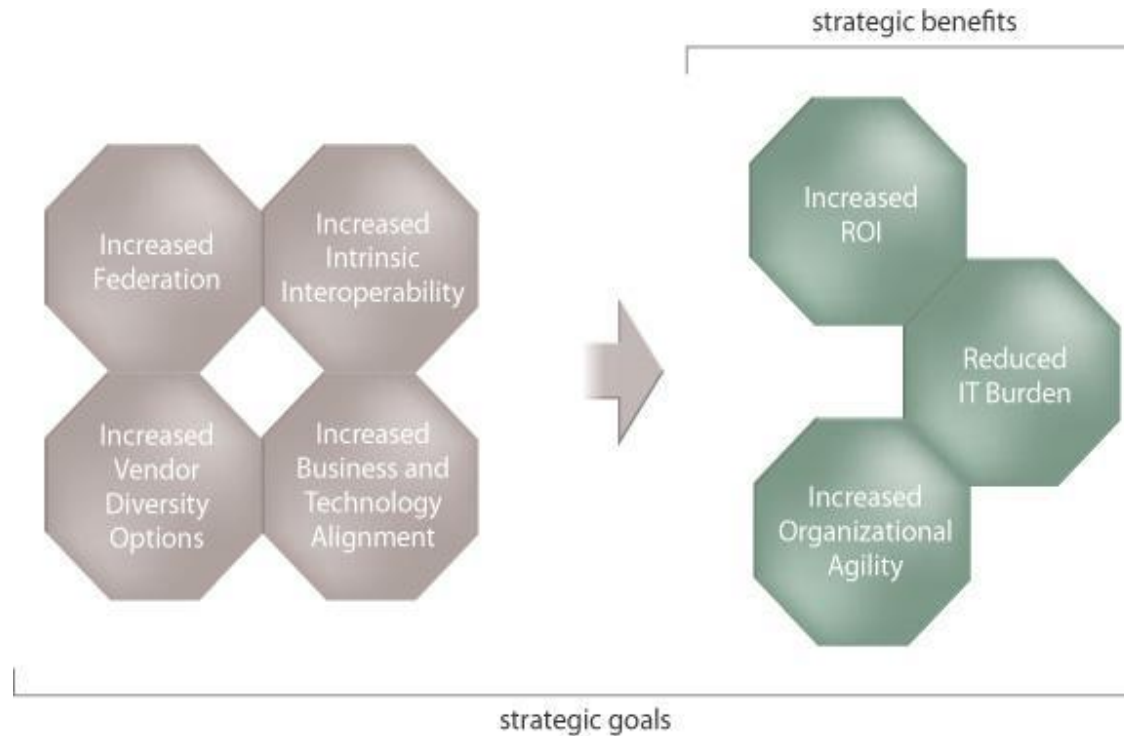
“SOA ofrece un marco de trabajo para **alinear los procesos de negocio con los sistemas** de TI.”



SOA History

- **1996:** Alexander Pasik de Gardner: Crean el concepto de SOA
- **2000:** Microsoft Web Services, complemento de SOA
- **2006:** Graddy y Boch, es una estrategia que requiere tiempo y esfuerzo, transformará las organizaciones y las harán más ágiles e innovadoras.
- **2008:** SOA proporcionará la base para el 80 por ciento de los proyectos de desarrollo

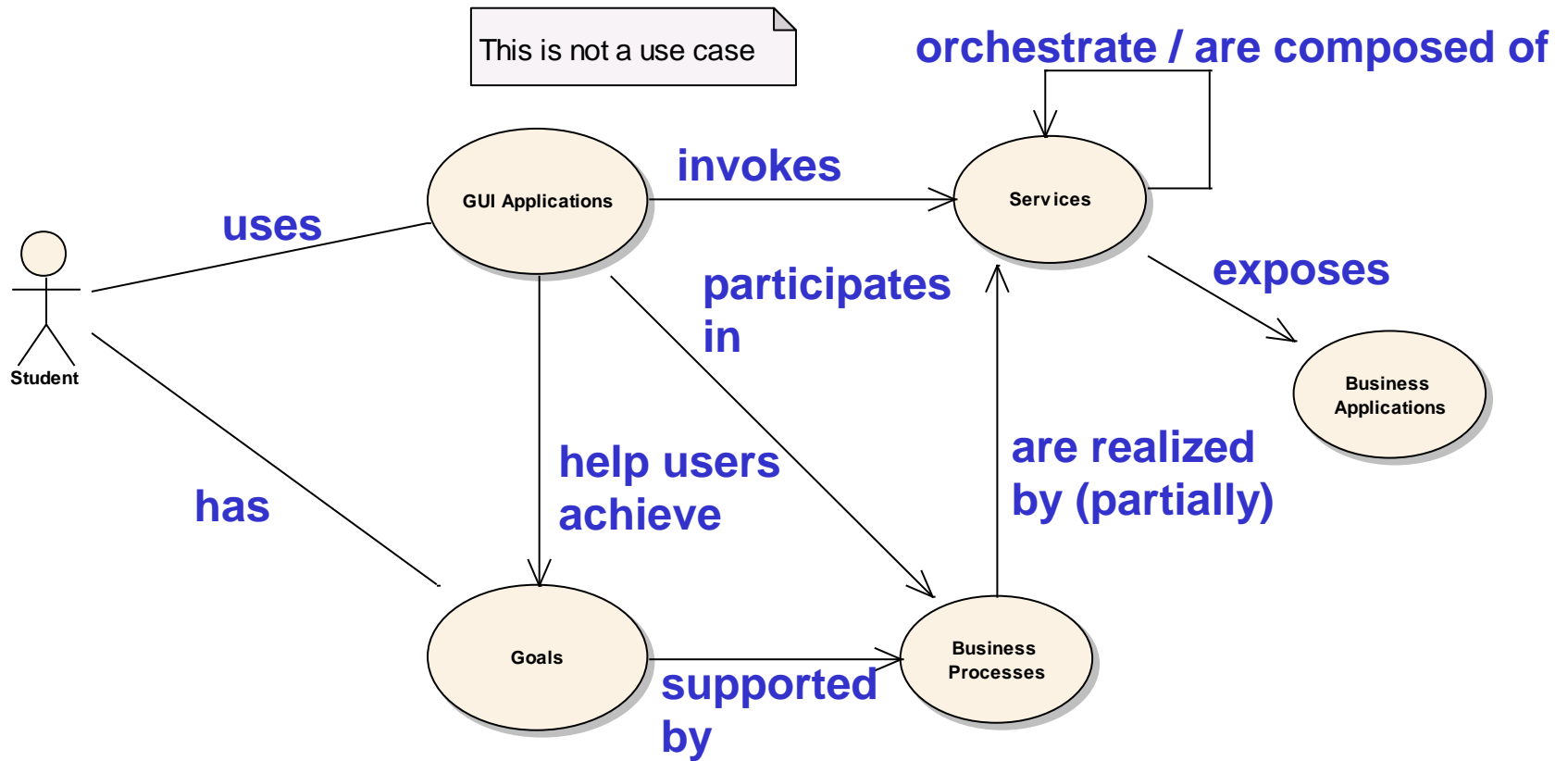
OBJETIVOS ORGANIZACIONALES



- Mayor interoperabilidad intrínseca
- Mayor federación
- Mayores opciones de diversificación de proveedores
- Mayor alineación empresarial y tecnológica

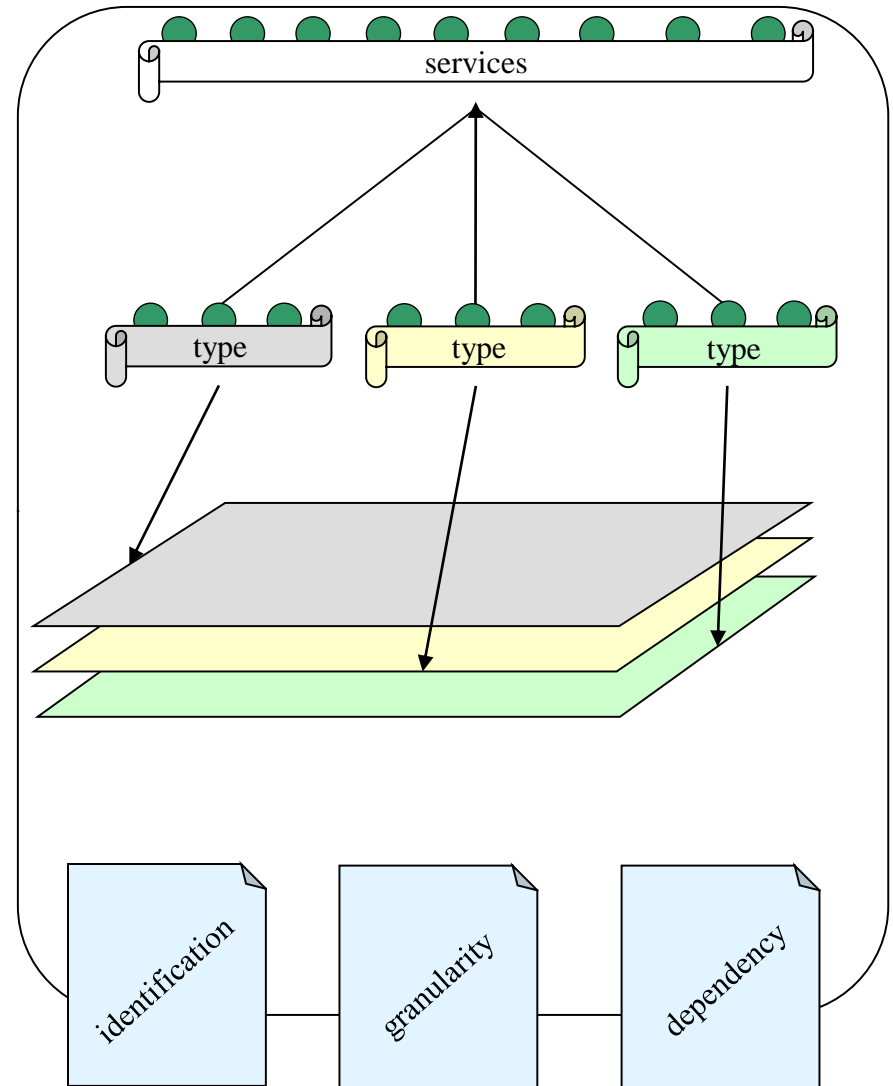
- Mayor ROI
- Mayor agilidad organizativa
- Reducción de la carga de TI

Service Relationships

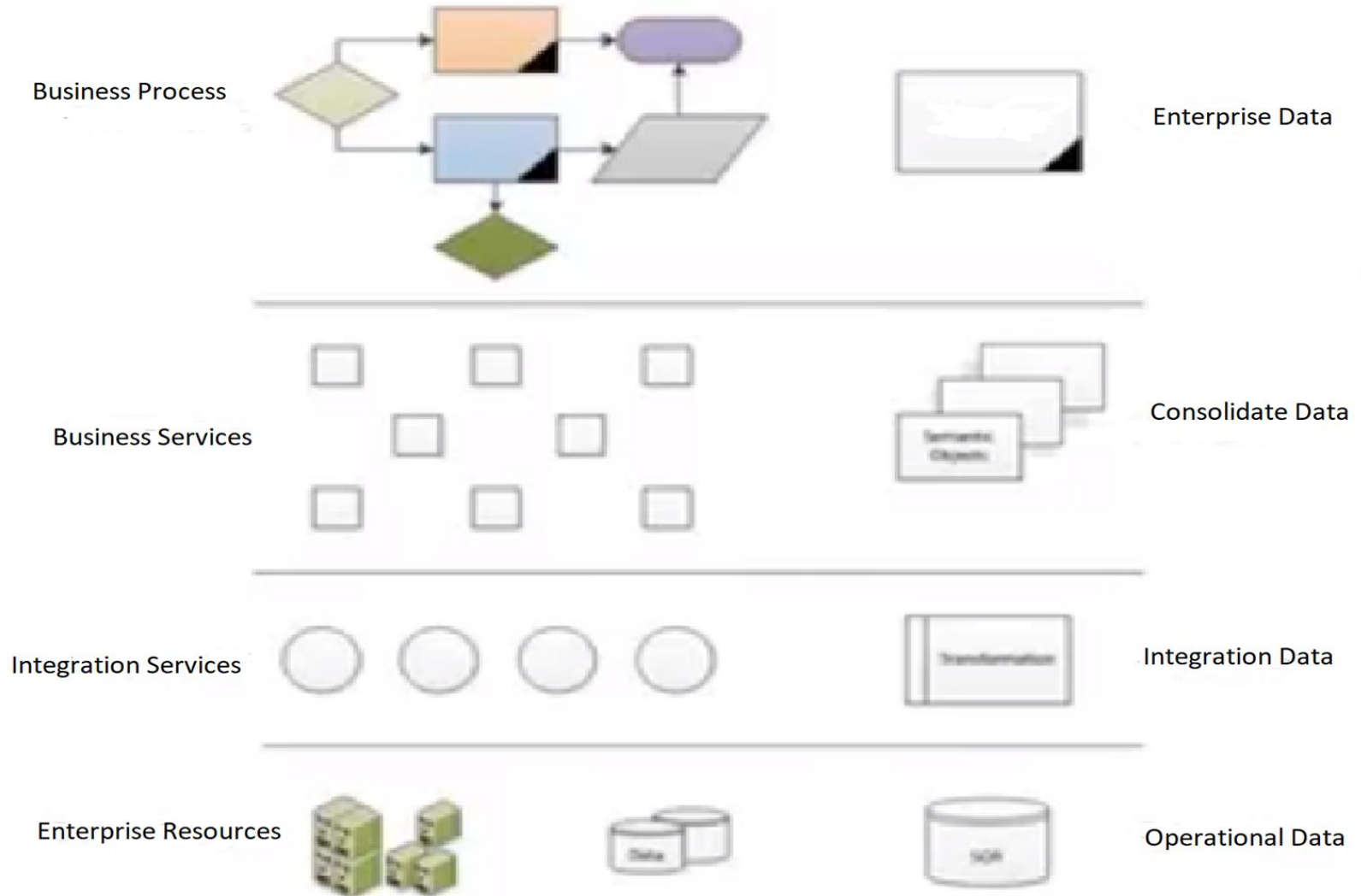


Que es una Arquitectura de Servicios?

- Una colección de Servicios
- Clasificado en Tipos
- Desplegado en capas
- Gobernado por patrones y políticas arquitectónicas



SOA en Capas



3-Tier Architecture



Homogenous

Language Dependent

Centralized Application Tiers

Code Centric Applications

Request/Reply Driven

HTML Pages

SOA



Heterogeneous

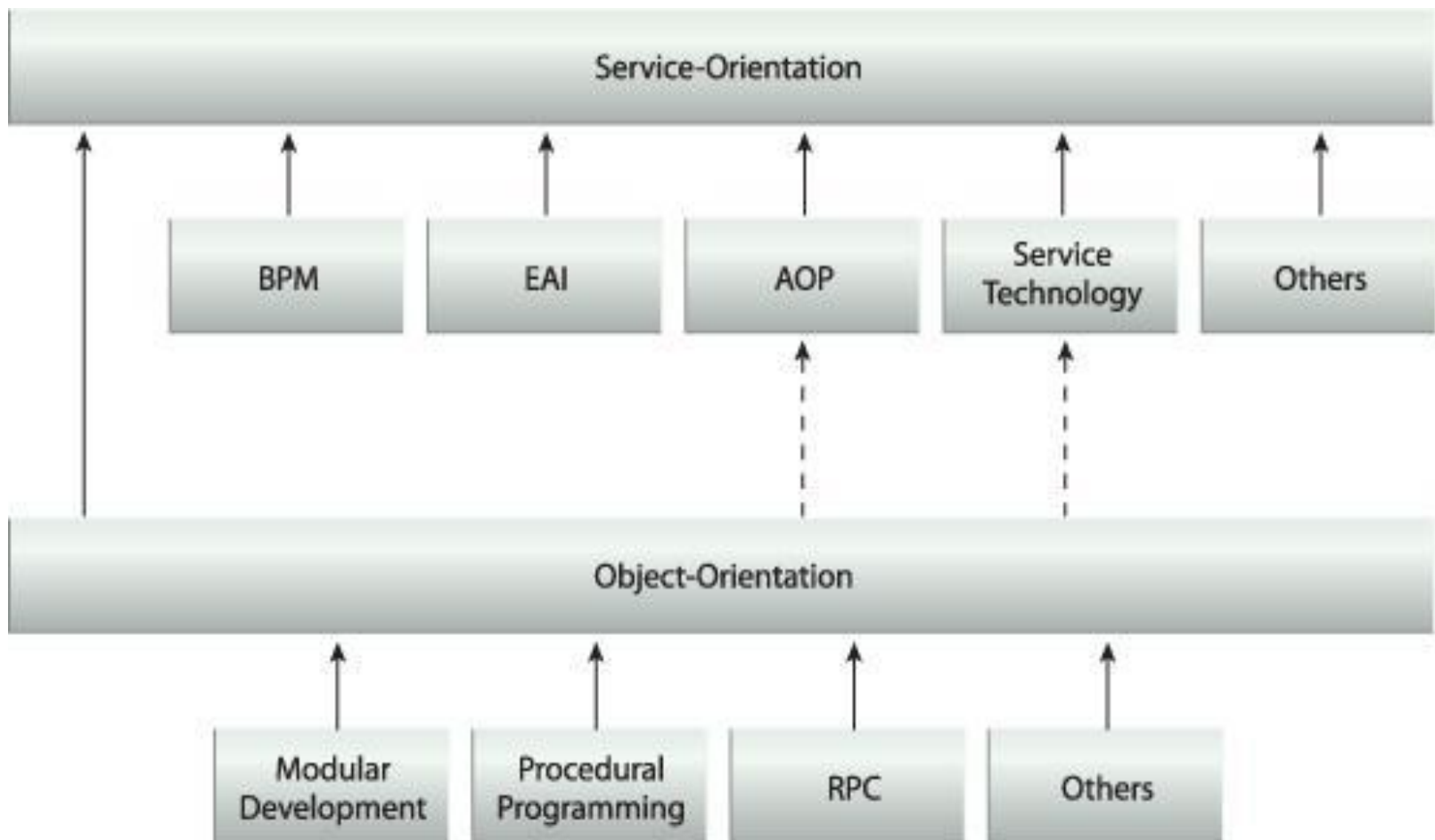
Language Independent

Massively Distributed Services

Flexible Composite Applications

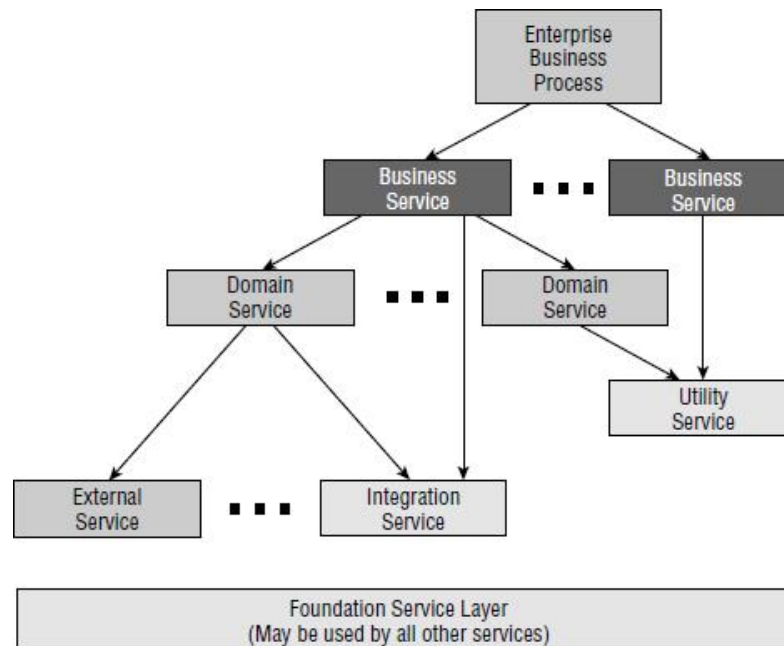
Request/Reply, Pub/Sub, Events

AJAX Rich Internet Applications



Granularidad del Servicio

- Cantidad de función comercial que se realiza en un solo intercambio de mensajes de solicitud / respuesta.

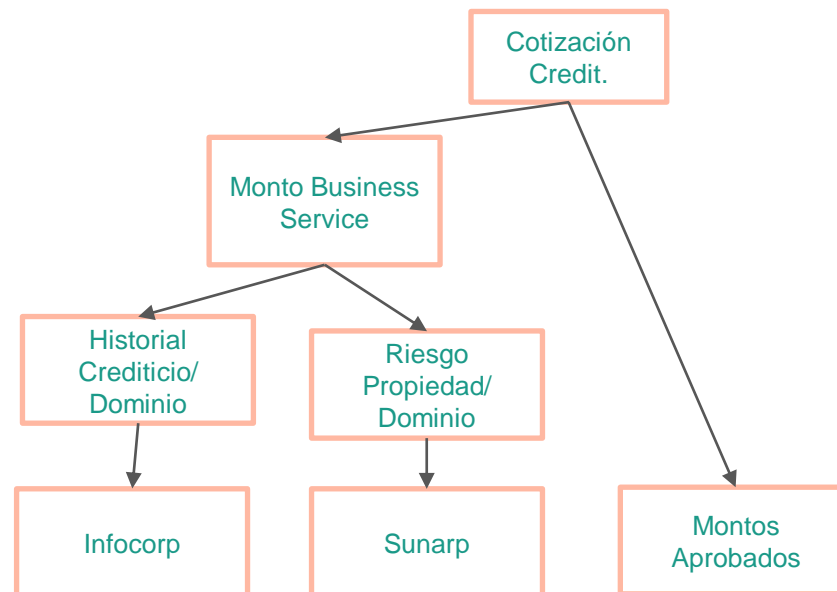


coarse-grained

Ejemplo

Crear un nuevo proceso comercial para para aprobar cotizaciones ante solicitudes de crédito

- Determinar su deuda o historial crediticio en Infocorp
- Determinar el riesgo asociado a sus propiedades
- Determinar el tipo de crédito ofrecido asociado a los solicitado

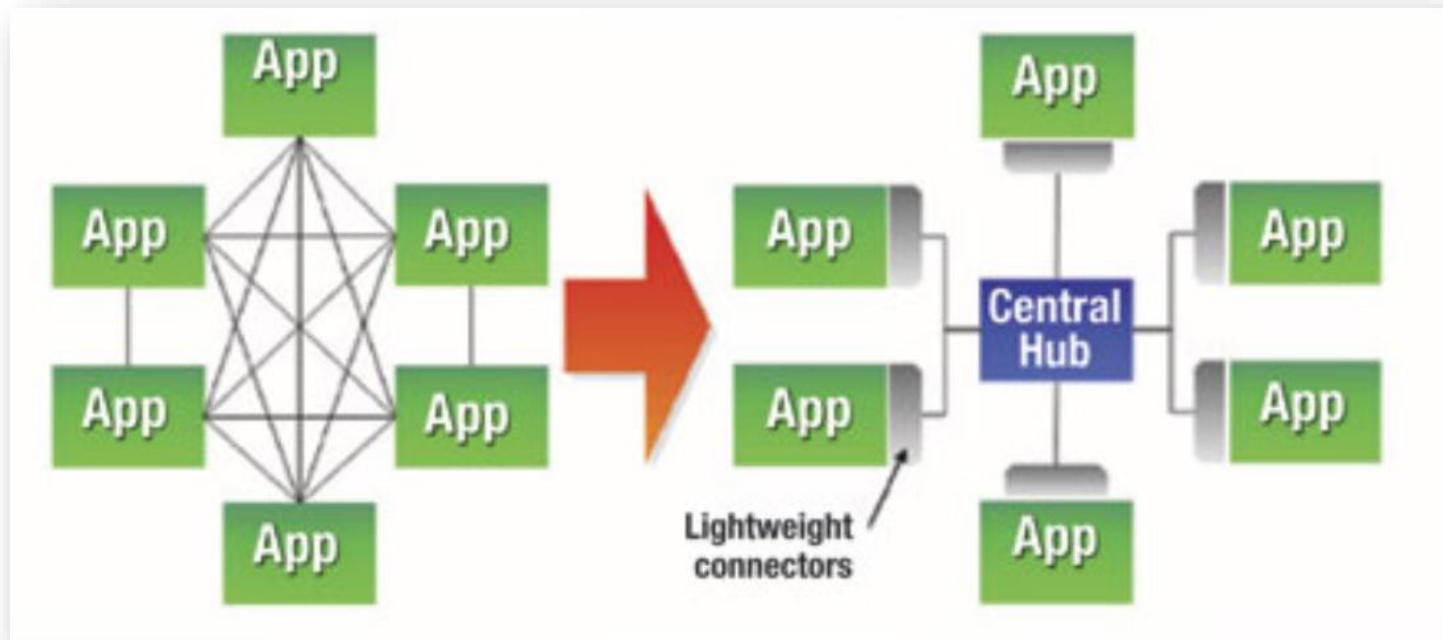




Tipos de Arquitectura

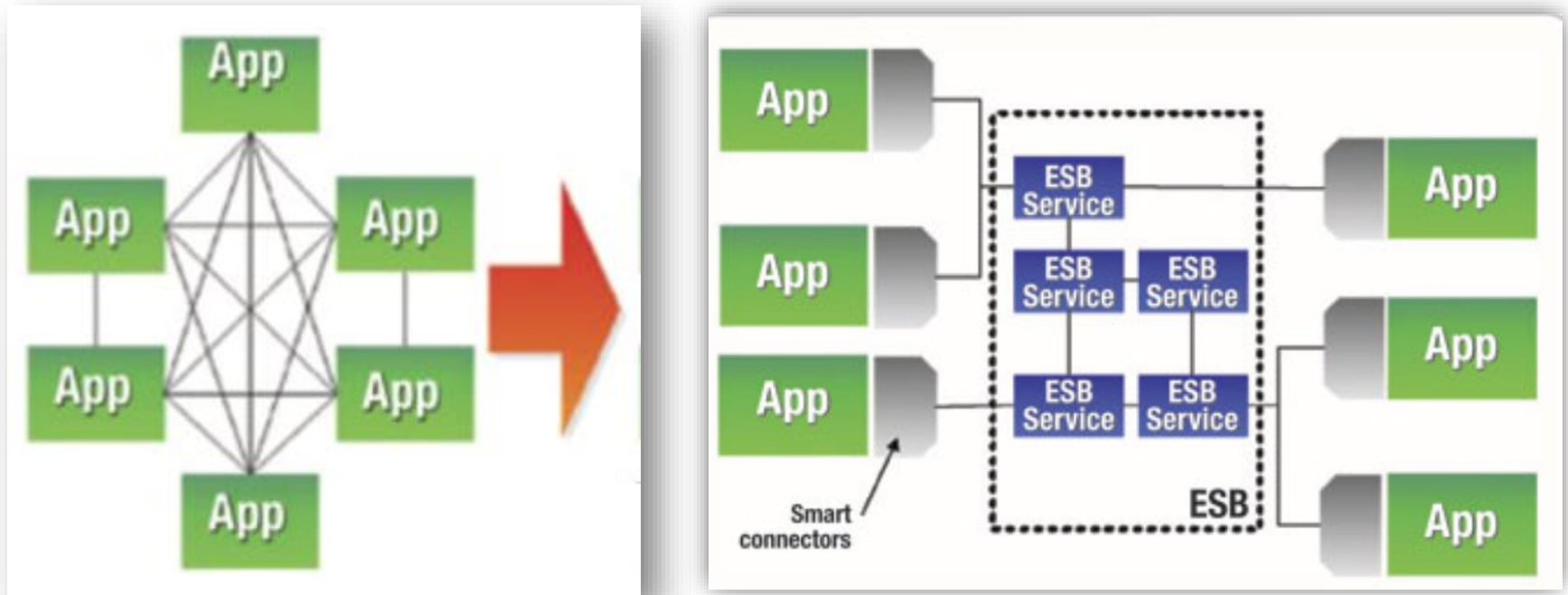
1. Tipo de Arquitectura Hub and Spoke

Biztalk Server
MOM Broker



- Uno de los objetivos clave de la arquitectura de Hub and Spoke con conectores es dejar los sistemas actuales intactos y sin cambios tanto como sea posible.
- Administración Centralizada.
- Hub se Cae se Cae todo

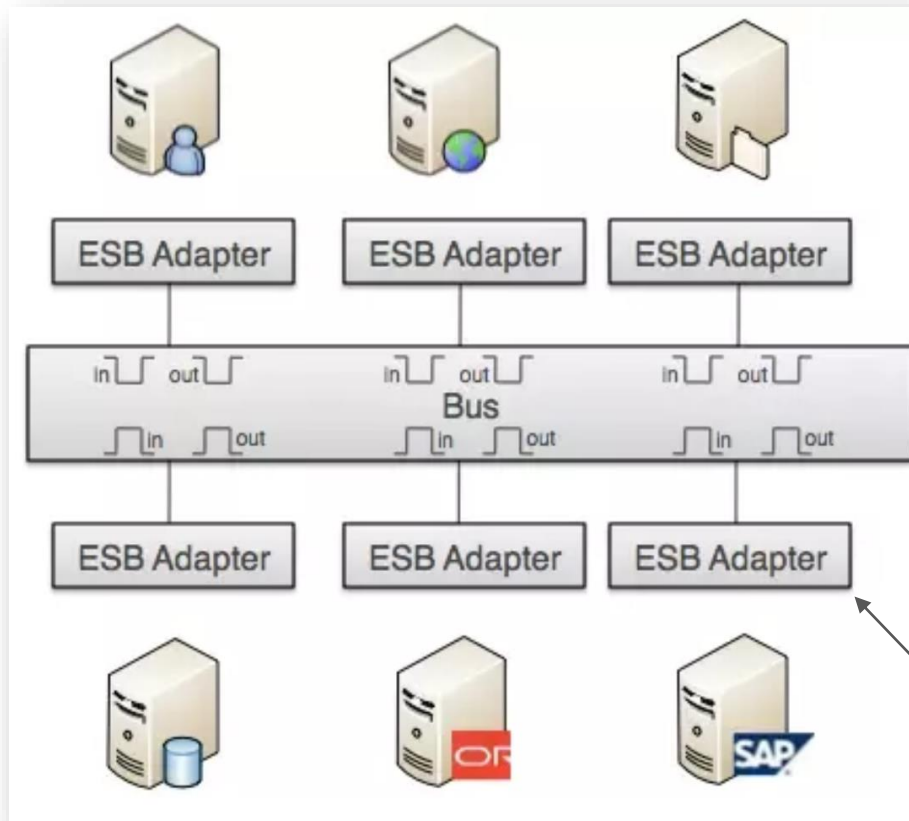
2.Tipo de Arquitectura: ESB – SOA (Distribuido)



BUS: Es una colección de servicios de Middleware que proporciona capacidades de integración

- Ofrece middleware de mensajería, enrutamiento inteligente, y transformación, interfaces XML/JSON
- Marco de seguridad flexible y una infraestructura de administración descentralizada para configurar, implementar y monitorear los servicios.
- Smart Connectors: La transformación de trama a xml/json se hace en el mismo Conector Inteligente

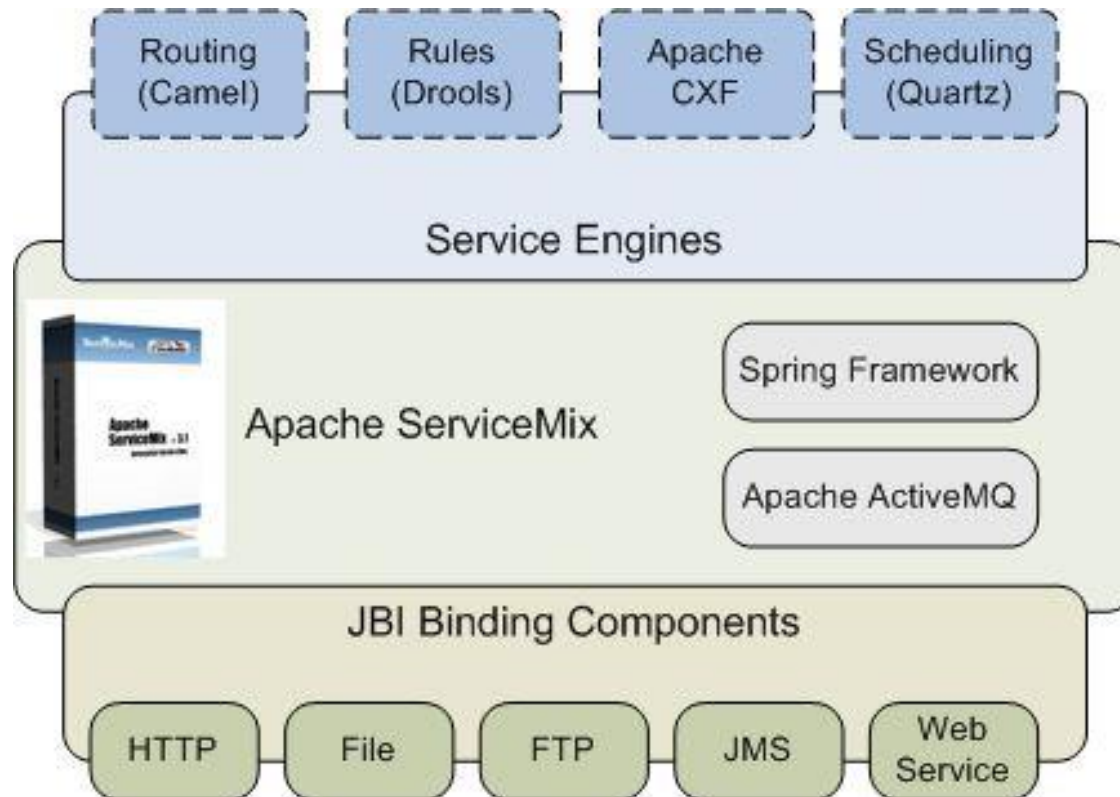
BUS



- ✓ OpenESB implementación en Java.
- ✓ Oracle ESB
- ✓ Oracle Service Bus (BEA AquaLogic Service Bus)
- ✓ Microsoft BizTalk Server
- ✓ Windows Azure Service Bus
- ✓ IBM WebSphere ESB
- ✓ IBM Integration Bus (IBM WebSphere Message Broker)
- ✓ JBoss Fuse
- ✓ Spring Integration
- ✓ Phoenix Service Bus en C#.
- ✓ Apache ServiceMix

SmartConnector

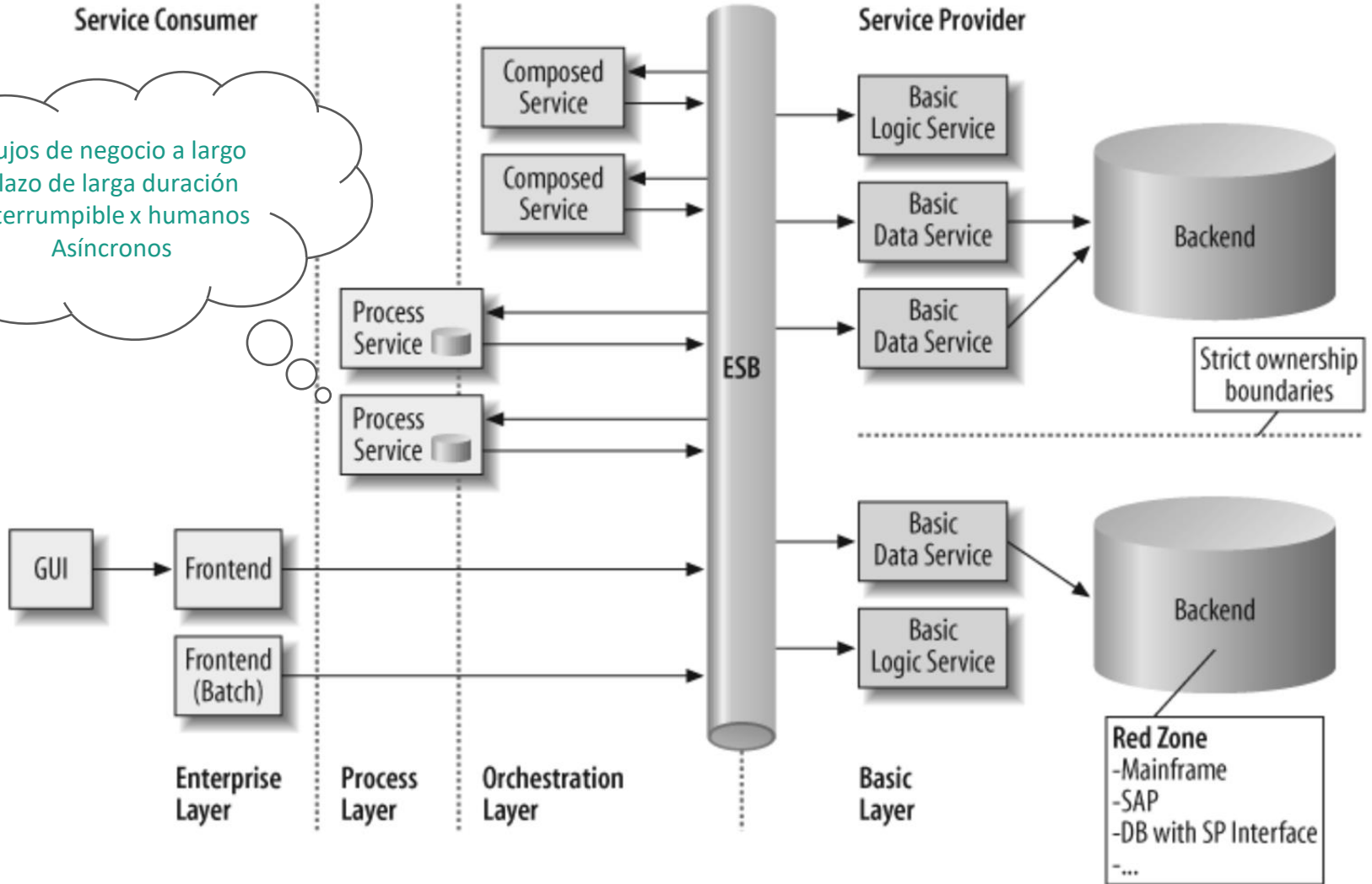
ESB



Service Consumer

Service Provider

Flujos de negocio a largo
plazo de larga duración
interrumpible x humanos
Asíncronos



SOA Habilitado para Procesos / Process Enabled / BPEL

Concepciones erróneas sobre SOA

- Basta utilizar Web Services para estar "service-oriented"
- SOA es una moda o estrategia comercial para renombrar los Web Services
- SOA es una simplificación de la computación distribuida
- Basta entender Web Services para implementar SOA
- Basta SOA para lograr que todo sea interoperable



TIPOS DE ARQUITECTURA

- Modelo Vista Controlador
- Entre pares o peer to peer /Blockchain
- Sistemas expertos/inteligentes
- **Orientada a servicios (SOA *Service-Oriented Architecture*).**
- Arquitectura de microservicios (MSA del inglés *MicroServices Architecture*). Algunos consideran que es una especialización de una forma de implementar SOA.
- Dirigida por eventos.
- En pipeline
- Máquinas virtuales o Cloud



SOA Diseño

Como encontramos servicios?

Top Down - ADI

1. Stake Holders
2. Plan Estratégico – Visión/Misión
3. Business Functions
4. Organigrama (Est. Fija)
5. Business Process (Est. Dinámica)
6. Business Entities

1. Transacciones
2. Aplicaciones
3. Fuente de Datos

Botton Up



Visión consumidor

Basado en la Necesidad



Visión Proveedor

Técnicos, expertos, juicio expertos, conocen las transacciones, juntarlos y digan que ofrecer, no hay mucho A/D



ORIENTACION A SERVICIOS SOA

Agenda

- ❖ Orientación a Servicios
- ❖ Principios de Orientación a Servicio
- ❖ SOA
- ❖ Arquitectura
- ❖ Tipos de Arquitectura
- ❖ Concepciones erróneas sobre SOA
- ❖ SOA Diseño
- ❖ Descomposición de Procesos en Servicios
- ❖ Ejercicios

Orientación al Servicio

La orientación al servicio es una forma de pensar en términos de servicios y desarrollo basado en servicios y los resultados de los servicios.



Orientación a Servicios

- Service Orientation, es un paradigma de diseño destinado a la creación de unidades lógicas de solución que tienen forma individual para que puedan usarse de forma colectiva y repetida en apoyo de las realizaciones de los objetivos estratégicos de la empresa.

- Servicios

- Las unidades de lógicas de la solución se denominan “Servicios”
 - El paradigma de diseño de Orientación a Servicios se compone de ocho principios planteados por Thomas Erl.

Servicio

- Un "Servicio" es (idealmente) una **función comercial** autónoma que acepta una o más solicitudes y devuelve una o más respuestas a través de una interfaz estándar bien definida.



Clasificación de los Servicios



Servicios Basicos:

1ra etapa de expansión, SOA
Fundamental con una sola capa de servicios básicos (KrafzigBankeSlama, 2008)



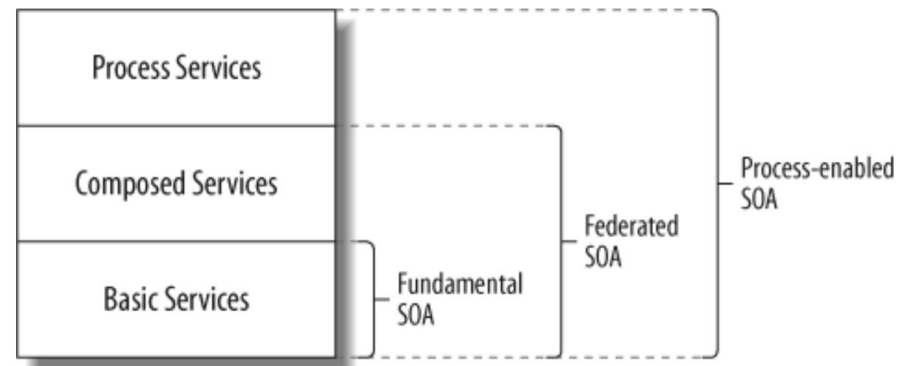
Servicios Compuesto:

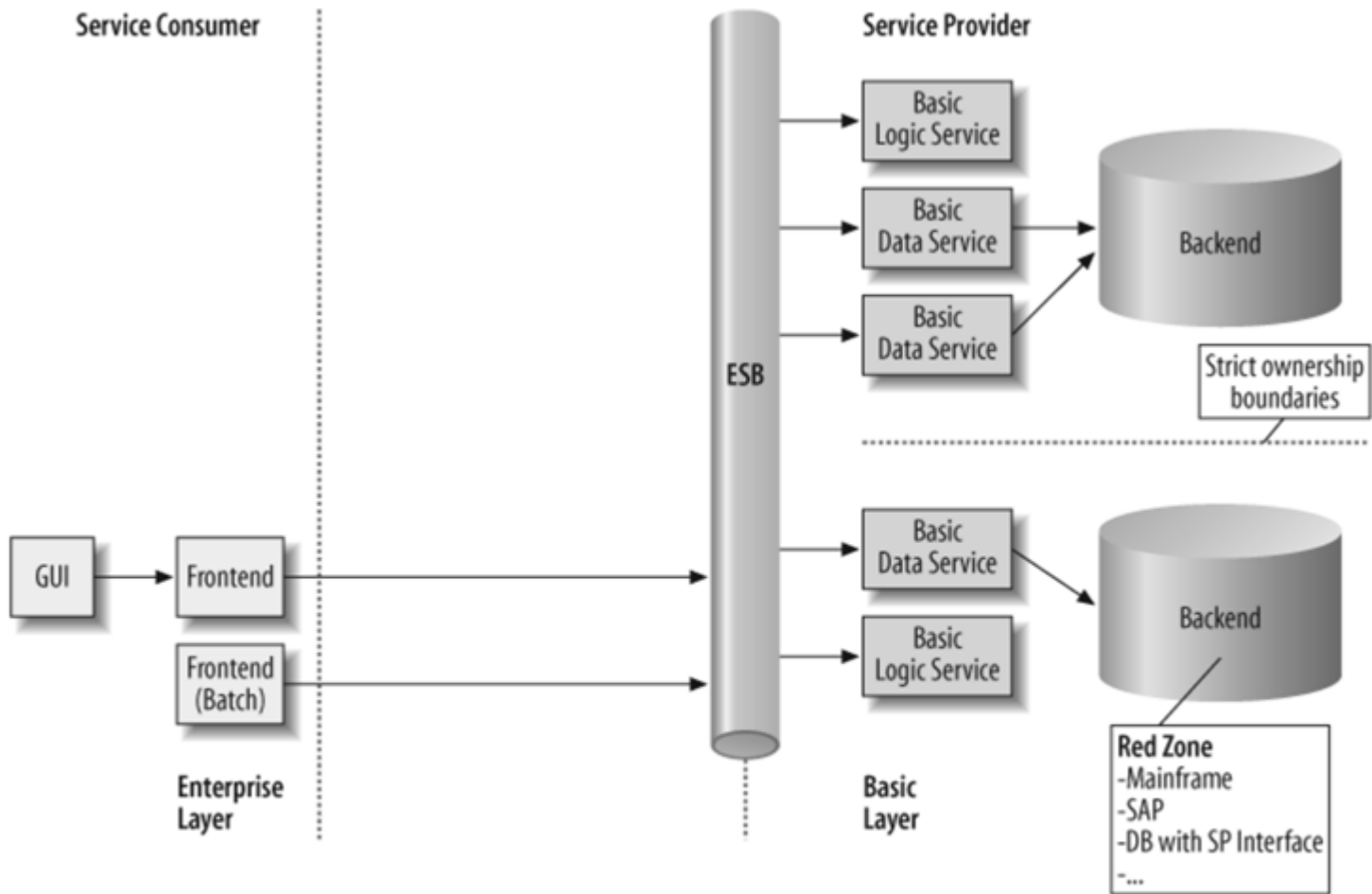
2da etapa de expansión, SOA
Federada, capa de servicios básicos + capa de servicios compuestos (capa de orquestación o composición)



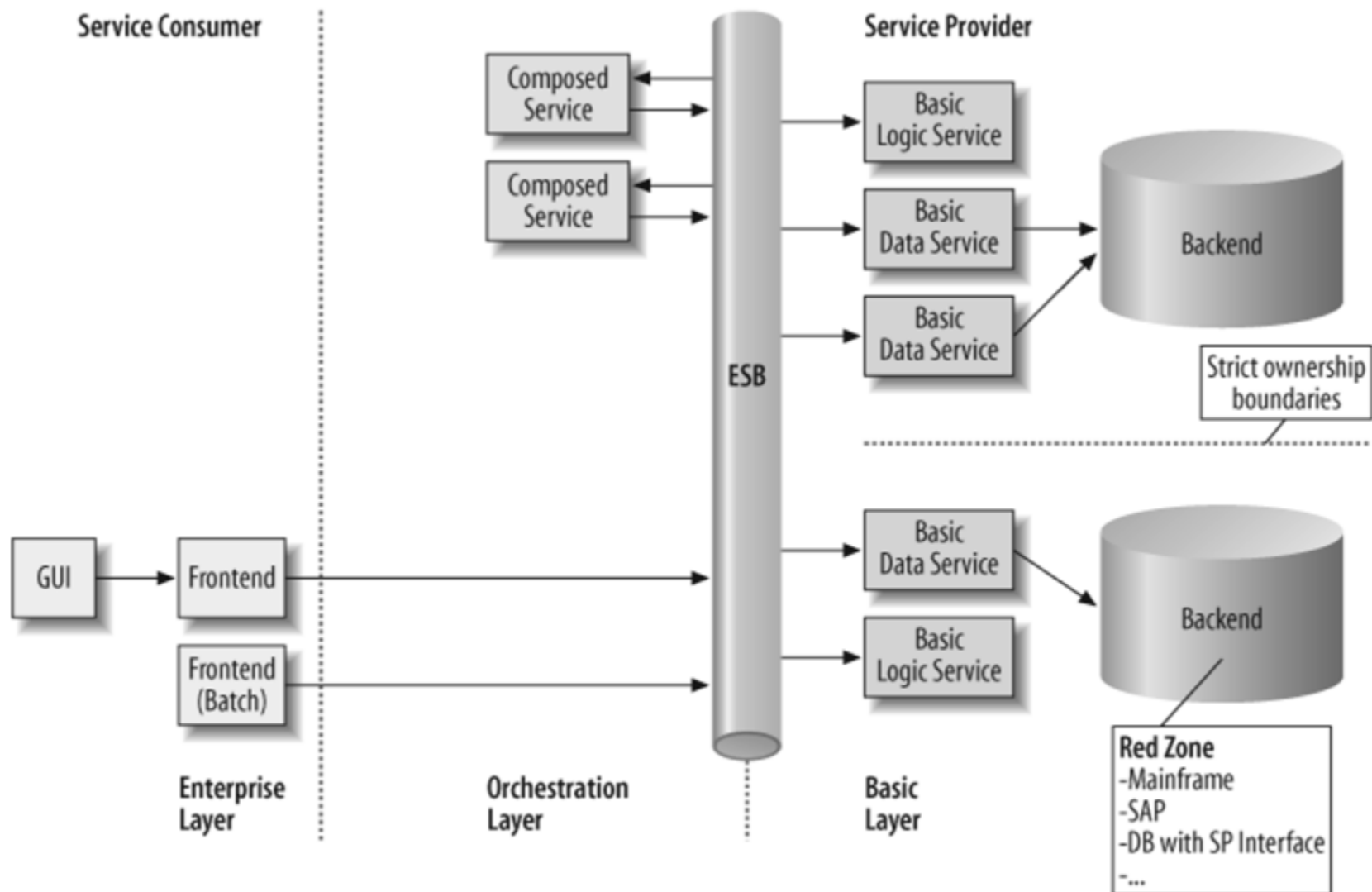
Servicios de Proceso:

3era etapa de expansión SOA
habilitada para procesos + capa adicional de servicios de procesos.

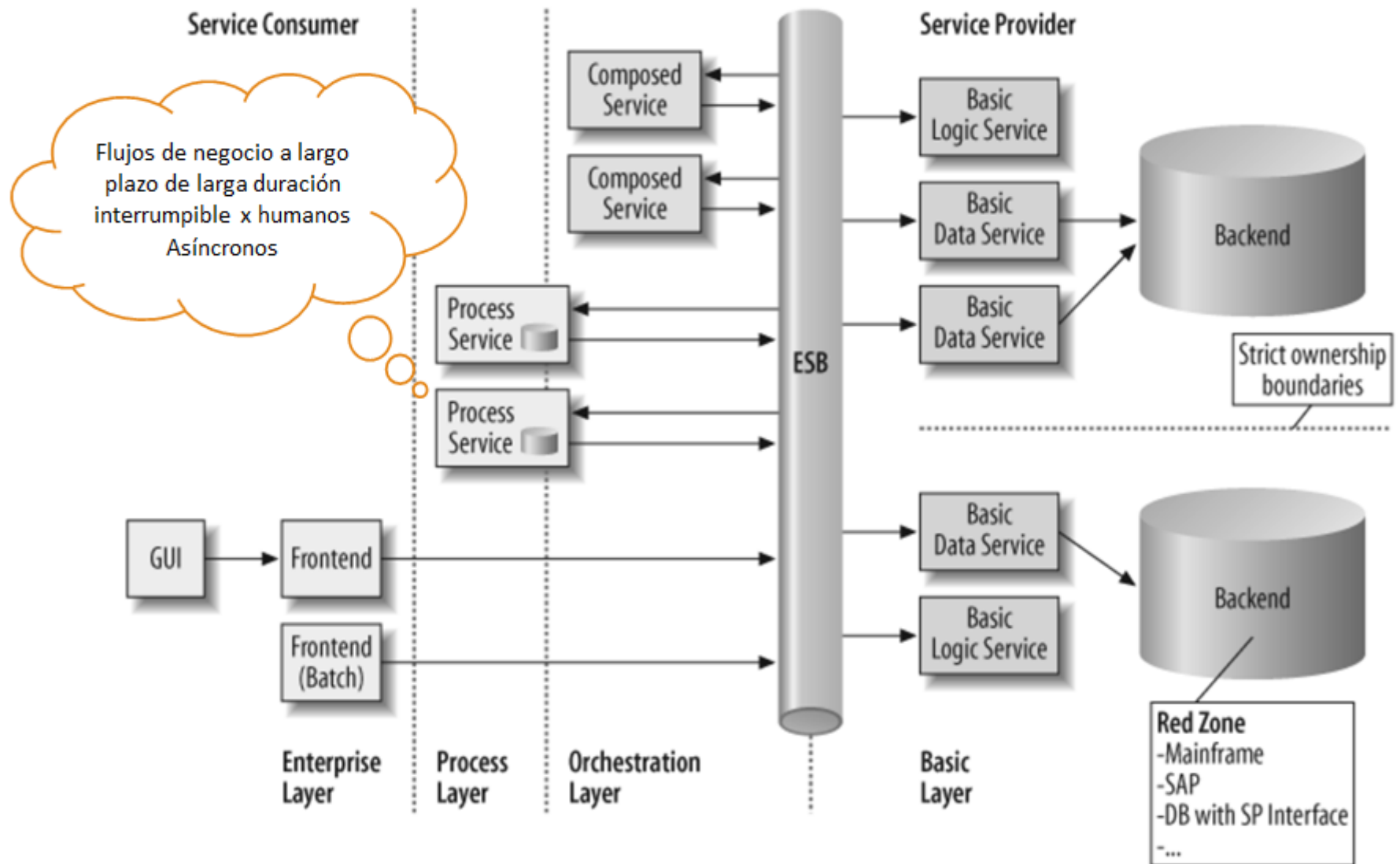




SOA FUNDAMENTAL

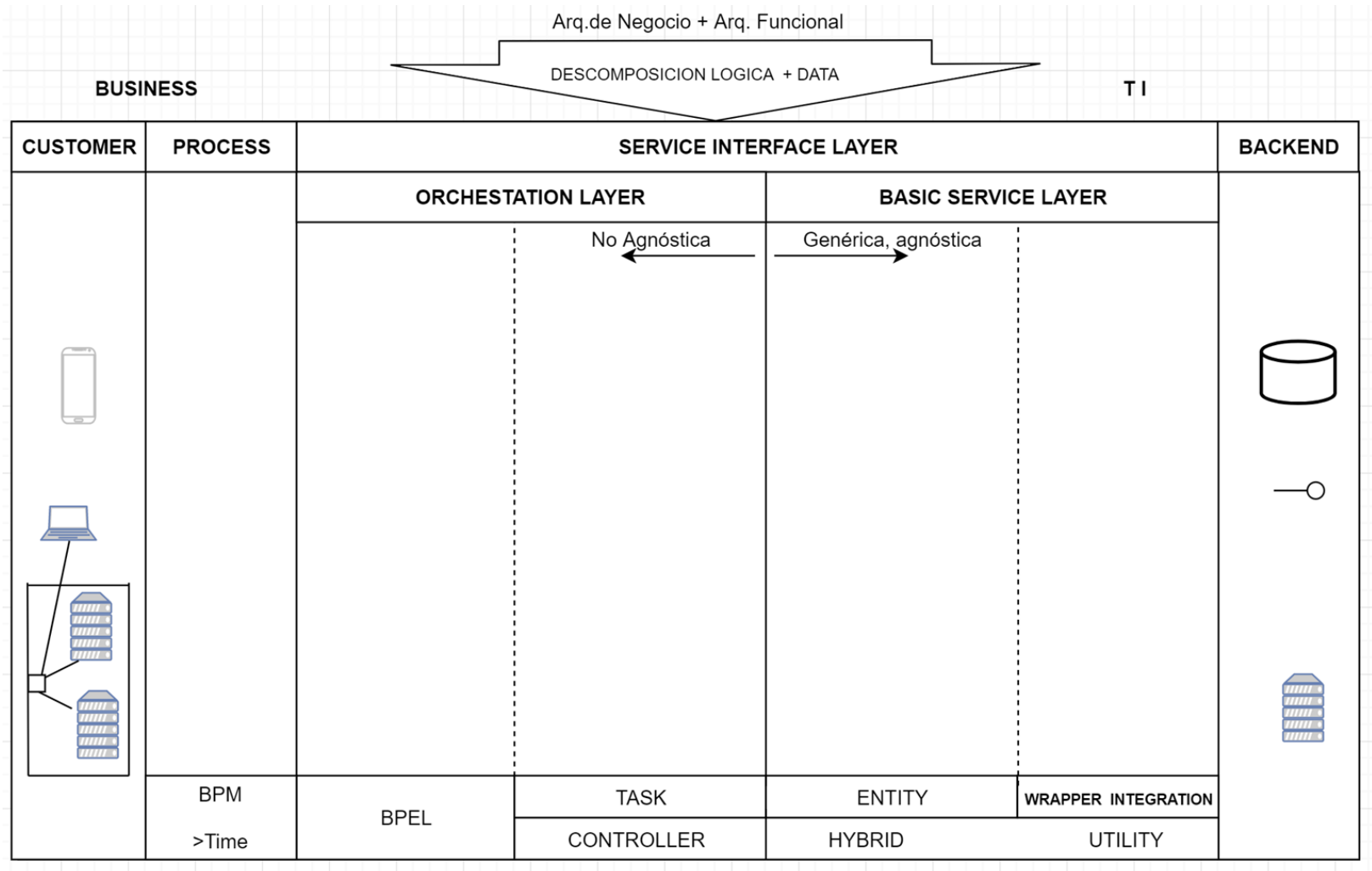


SOA FEDERADA



SOA Habilitado para Procesos / Process Enabled

Aquitectura SOA



Arquitectura SOA- Detalle Clasificación de los Servicios

1) BASIC SERVICE LAYER

- **Capa Agnóstica, Genérica**
 - Capa Entity
 - Capa Integration

Arquitectura de Servicios está compuesto de ...

➤ Services

- Unidades lógicas de procesamiento
- Ejemplo: Servicio de Tarjeta de Crédito

➤ Messages

- Unidades de comunicaciones entre servicios
- Necesario para que los servicios hagan su trabajo
-

➤ Operations

- Unidades de trabajo
- Example: Determinar el coste de atención

➤ Processes

- Compuesto / grupos de servicios orquestados
- Example: Evaluación Crediticia

Método para Identificar Servicios

1. Diseñar el diagrama de clases POO del Sistema

**2. Seleccionar los métodos de negocio y separar:
Entidades**

- Sólo entidades

Capa Agnóstica

- Funcionalidades DAO que trabajan con (Repositorio), incluye CRUD
- Funcionalidades que invocan servicios o recursos externos (Capa de Integración)
- Utilitarios (Capa de integración y Utilitarios)

3. Capa de Negocio u Orquestación, No Agnóstica

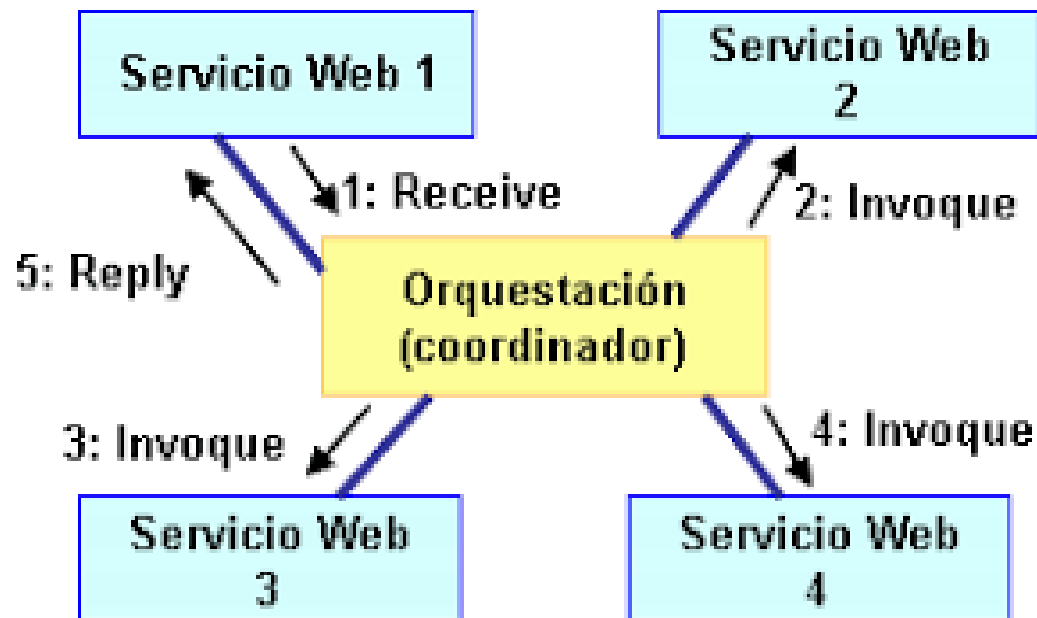
Funcionalidades que reutilizan servicios DAO o así mismos

4. Aplicar Principios SOA

Orquestación vs. Coreografía

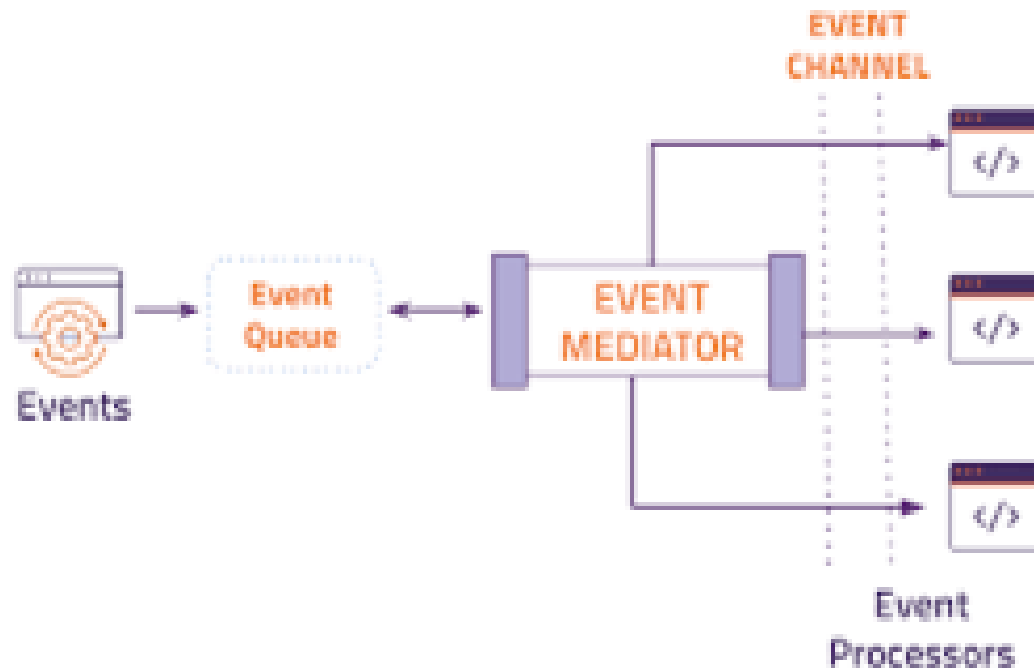


- **Orquestación:** Es la opción más clásica. Se basa en que **un componente** será el que coordine las llamadas a los servicios que necesita de forma secuencial, típicamente mediante llamadas de petición/respuesta. Este componente se encargaría de gestionar los errores.



Orquestación vs. Coreografía

- **Coreografía:** Los servicios no se llaman entre sí, sino que se utiliza un sistema de **eventos**, de forma que cuando un servicio termina su tarea, deja un mensaje y todos aquellos servicios suscritos a ese canal son notificados, de forma que puedan realizar su trabajo.

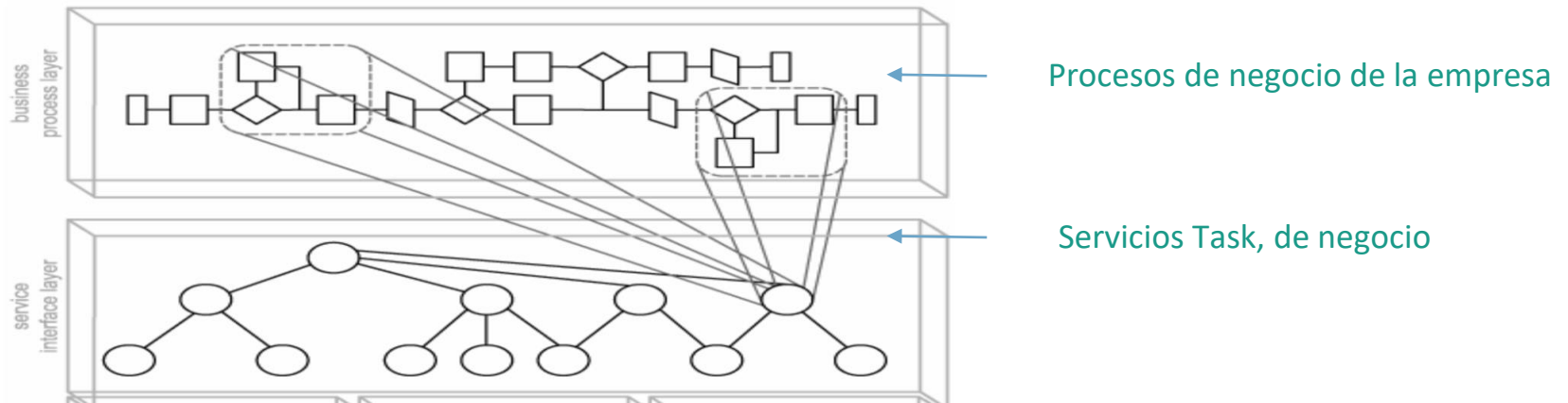


Arquitectura SOA- Detalle Clasificación de los Servicios

2) ORCHESTRATION SERVICE LAYER/NEGOCIO:

Representa la capa de negocio de servicios compuestos, SOA federada que se divide en las siguientes capas:

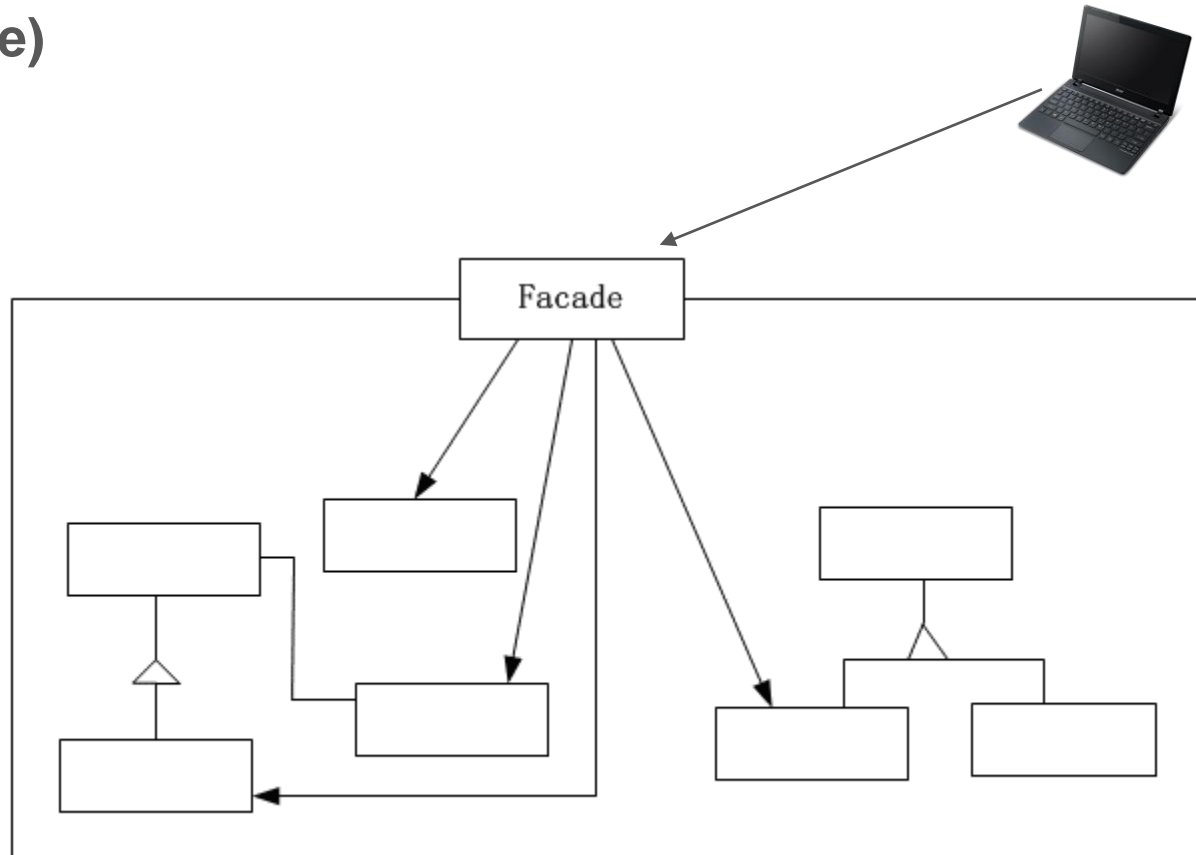
- **Capa No Agnóstica**
 - Capa Task - Especializada




- **Capa BPEL**

Arquitectura SOA- Detalle Clasificación de los Servicios

3) CAPA DE INTERFACE CLIENTE Service Client Interface (Facade)





Métodos para Identificar Servicios

Ejercicio

La Dirección de Tránsito de Lima necesita un programa que permita gestionar las multas por infracciones de tránsito de los conductores de Lima Metropolitana, así como la actualización de su puntaje. Para esto el sistema permitirá registrar los datos de los conductores de vehículos: DNI, número de licencia de conducir, nombre, dirección, email, año de registro y un puntaje de 300 puntos por defecto el cual indica que no tiene infracciones.

Las penalidades en puntaje por infracciones se aplican según:



Código Infracción	Descripción	Puntaje
A	Simple	-20
B	Básico	-50
C	Intermedio	-100
D	Alto	-200
E	Crítico	-300



La multa de infracciones al conductor se calcula de la siguiente manera:

$$0.20 * (\text{año de registro} - \text{año actual}) * (\text{puntaje actual}) \text{ en soles}$$

Para esta práctica no será necesario crear la tabla o entidad Penalidad.

Se pide:

- a) Registrar 3 conductores. (al finalizar cada registro debe enviarse un correo electrónico de bienvenida.
- b) Dado un DNI obtener los datos de conductor.
- c) Dado una licencia de conducir y código de penalidad actualice su récord de puntaje penalizándolo.
- d) Un listado de todos los conductores cuyo vehículo tengan menos de 100 puntos.
- e) Dado un número de DNI obtener la multa de infracción del conductor.
- f) Invocar un servicio externo de <https://currencylayer.com/> para obtener una lista de tipos de cambio de un dólar en todas las monedas extranjeras y peruana.

-
- g) Diseñe su arquitectura SOA incluya sus entradas y salidas de todos los servicios, mencione los principios SOA que aplica en su diseño/programa.

Adjunte sus pantallazos de resultados de pruebas unitarias y base de datos por cada prueba, también su proyecto con código fuente e imagen del diagrama SOA.

Administrador

+arregloConductores<>

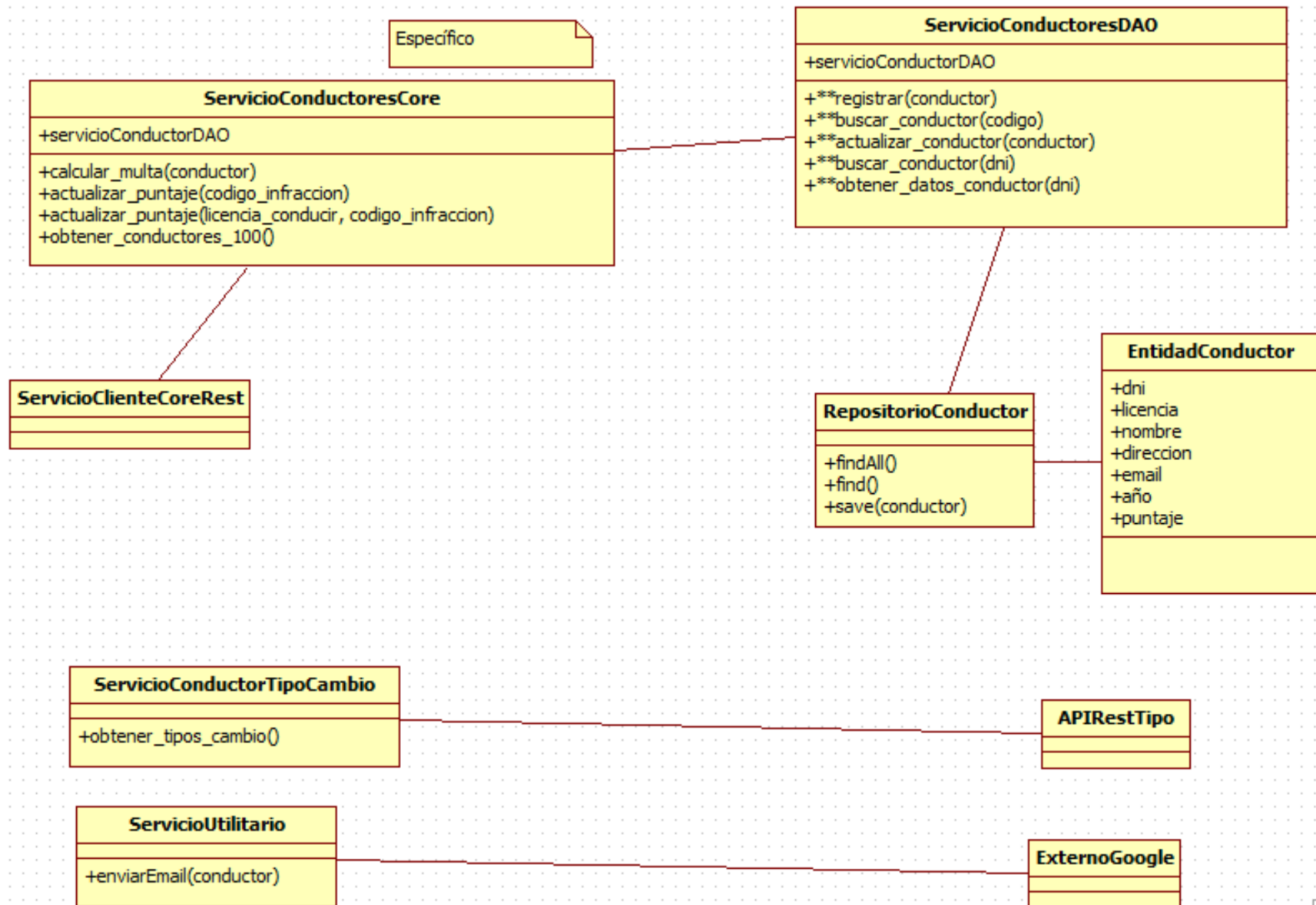
+**registrar(conductor)
+**buscar_conducto(dni)
+** buscar_conductor(codigo)
+**obtener_conductores_100()
+**actualizar_conductor(conductor)
+** obtener_datos_conductor(dni)
+obtener_pago_anual_impuesto(dni)
+enviarEmail(dni)
+obtenerTiposCambio()

1..*

Conductor

+dni
+licencia
+nombre
+direccion
+email
+año
+puntaje

+calcular_multa()
+enviarEmail()
+actualizar_puntaje(codigo_infraccion)

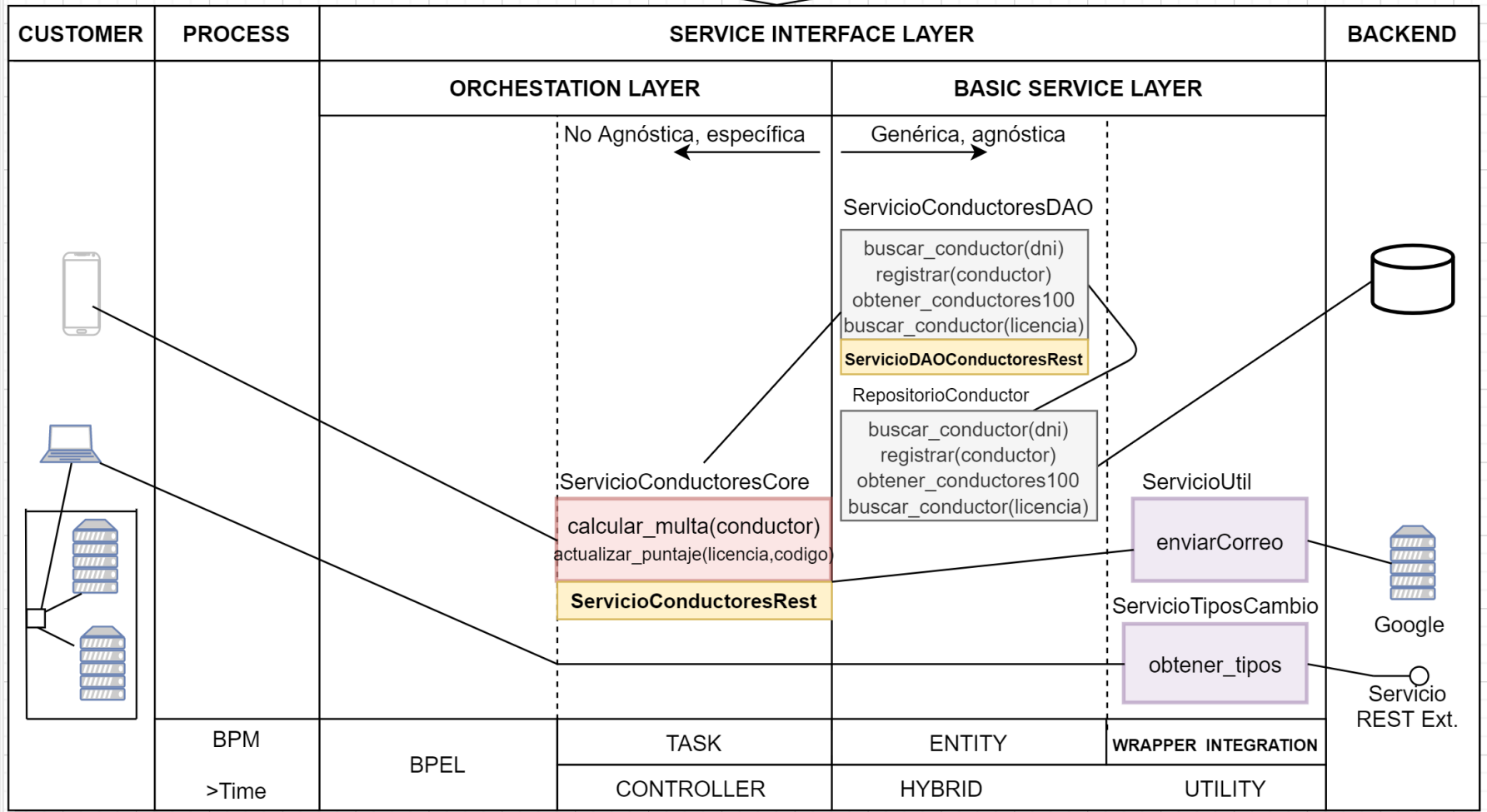


Arq.de Negocio + Arq. Funcional

DESCOMPOSICION LOGICA + DATA

BUSINESS

TI



Otro método de Diseño SOA

- Descomposición de Procesos en Servicios
(ver video del Aula Virtual y Referencias)

Referencias

- ISO 12207 (<http://www.12207.com/>).
- The Open Group Architecture Framework (<http://www.opengroup.org/>).
- OASIS Reference Model for Service Oriented Architecture (http://www.oasis-open.org/committees/tc_cat.php?cat=soa).
- Enterprise SOA: Service-Oriented Architecture Best Practices. Dirk Krafzig, Karl Banke, Dirk Slama. Prentice Hall PTR. November 09, 2004. ISBN 0131465759.
- IBM SOA (<http://www-306.ibm.com/software/solutions/soa/>).
- Oracle SOA (<http://www.oracle.com/technologies/soa/index.html>).
- Identificación de servicios por descomposición
https://www.youtube.com/watch?feature=player_detailpage&v=TauAUHGtH_w#t=433
- Arquitectura
<https://youtu.be/y46TBEf21Gs>
- https://www.ted.com/talks/ole_scheeren_why_great_architecture_should_tell_a_story/transcript?language=es