

SOA Diseño

SOA – Diseño

Como encontramos servicios?

Top Down - ADI

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

1. Transacciones
2. Aplicaciones
3. Fuente de Datos

Bottom Up



Visión consumidor

Basado en la Necesidad
(Radio) – Consumir Energía -> Batería



Visión Proveedor

Técnicos, expertos, juicio expertos,
conocen las transacciones, juntenlos
y digan que ofrecer, no hay mucho
A/D, (Radio proveer electricidad):
Provee Cables sueltos

Activo: Enchufe, otras triple.., alternativas

Clasificación de los Servicios

✓ Servicios Basicos:

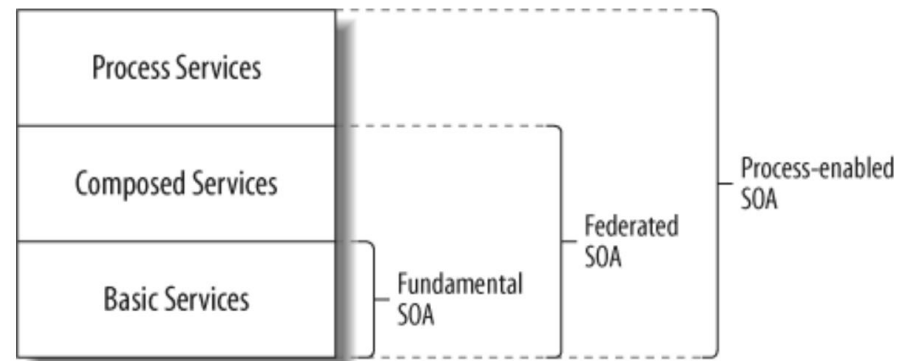
1ra etapa de expansion, SOA
Fundamental con una sola capa de
servicios básicos
(KrafzigBankeSlama, 2008)

✓ Servicios Compuesto:

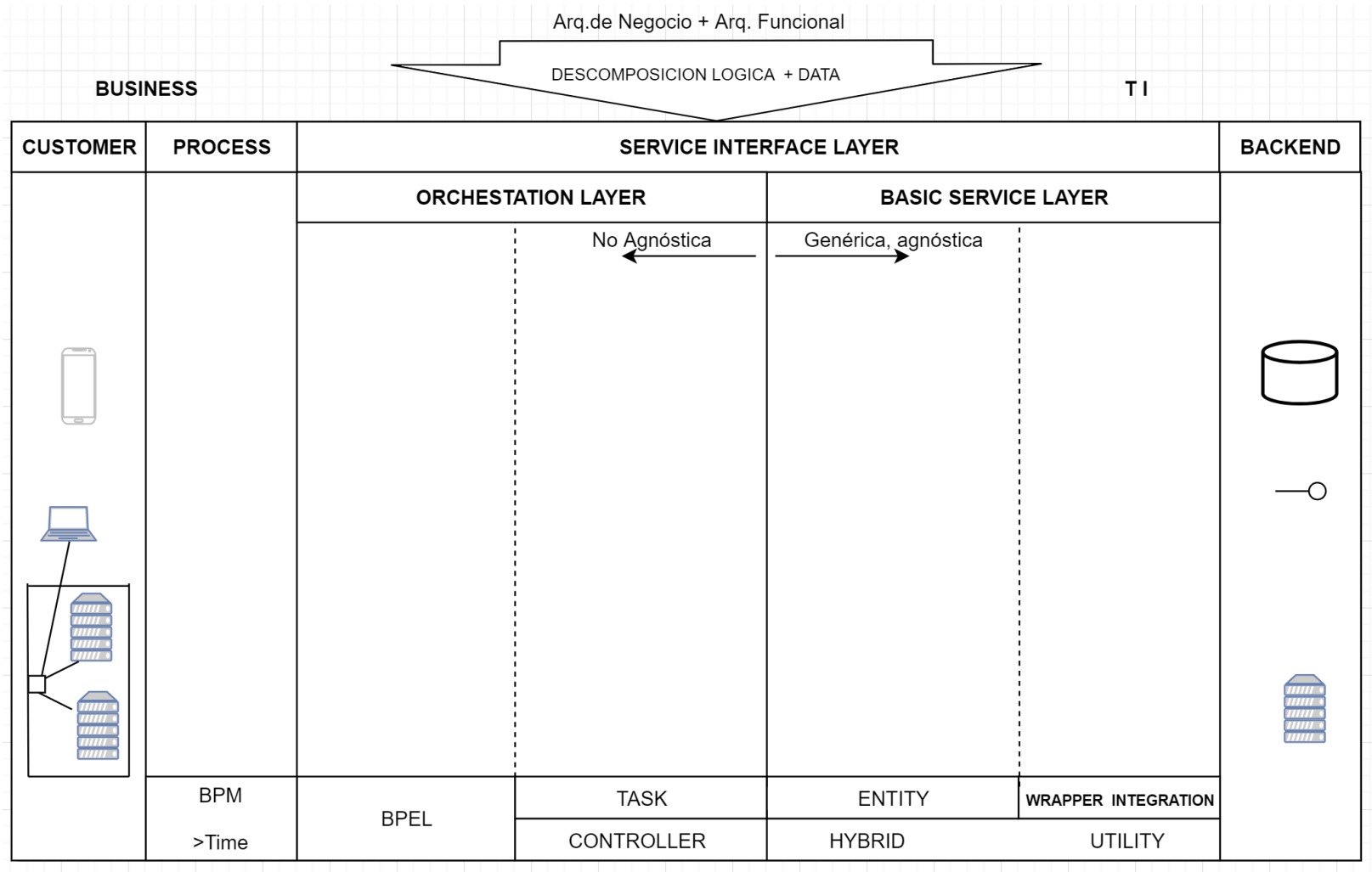
2da etapa de expansión, SOA
Federada, capa de servicios básicos
+ capa de servicios compuestos
(capa de orquestación o
composición)

✓ Servicios de Proceso:

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



Aquitectura SOA



Arquitectura SOA

BASIC SERVICE LAYER

- **Capa Agnóstica, Genérica**
 - **Capa Entity**: Capa que controla su propia entidad en la base de datos, generalmente los CRUDs o mantenimientos de su tabla, interactúa directamente con su propia base de datos **local**.
 - **Capa Integration**: Capa que invoca **servicios externos**, tales como webservices, servicios REST, servicios SAP, o legacy systems, también se consideran a los envíos de mensajes a servidores de correos, sms, servicios asíncronos de colas entre otros. Esta capa no tiene logica de negocio, es un wrapper a otros.

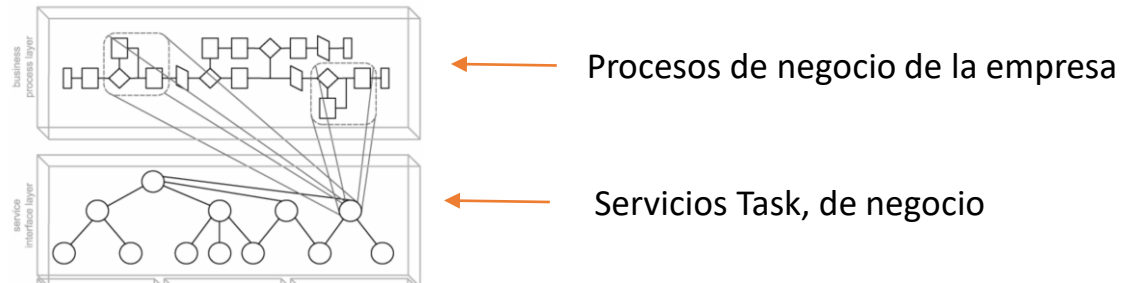
Arquitectura SOA

- **ORCHESTRATION SERVICE LAYER/NEGOCIO:**

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

- **Capa No Agnóstica**

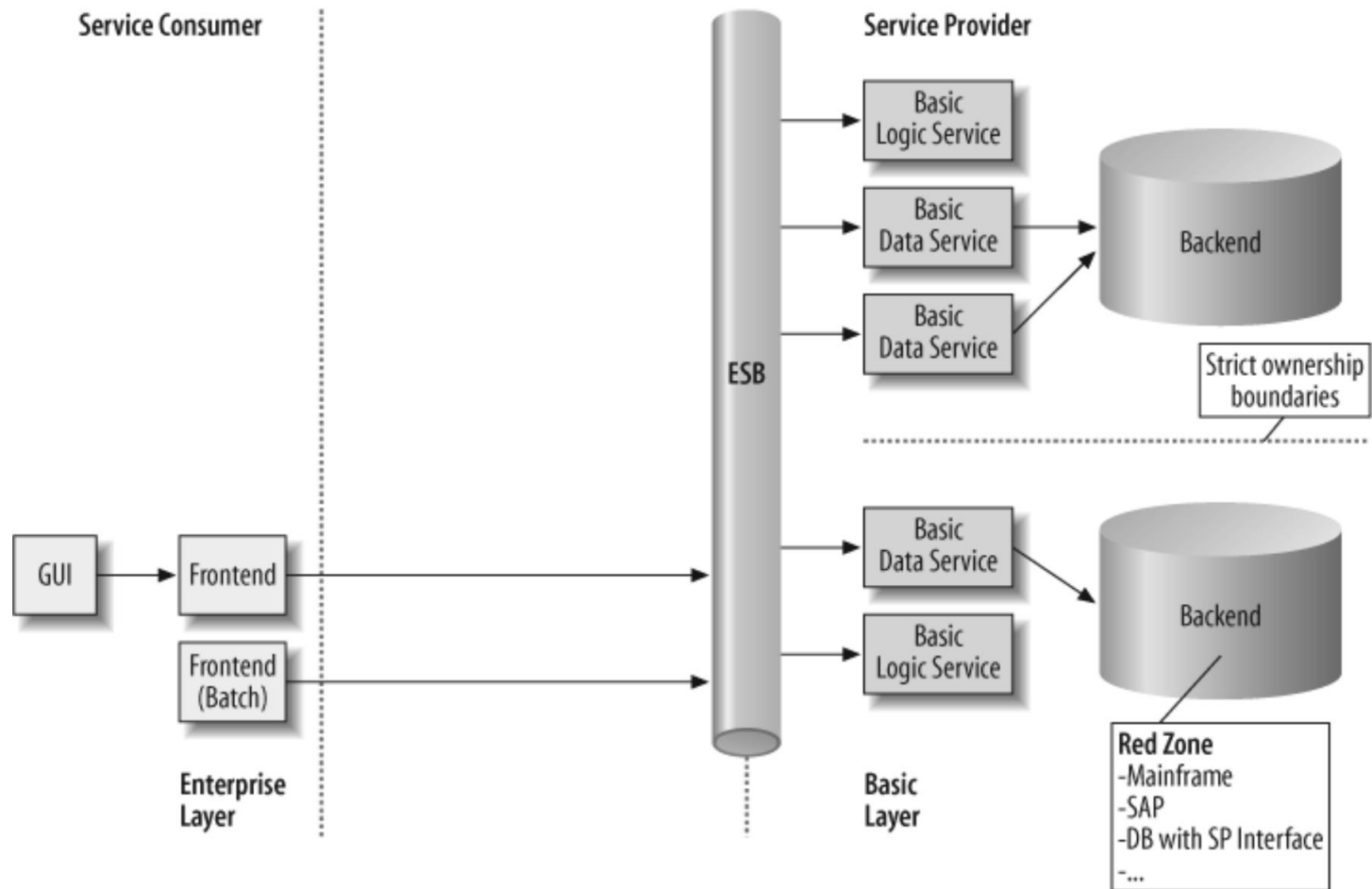
- **Capa Task** : Es la capa que tiene la lógica de negocio del core, es la capa que generalmente se compone o reutiliza a la capa de entidades o su misma capa, por eso se denomina también capa de orquestación. Los servicios de esta capa realizan algo específico del negocio de la empresa:



- **Capa BPEL**: También es una capa de orquestación, es un flujo de procesos donde está involucrado uno o más servicios pero con un factor de **tiempo significativo mayor al de un proceso síncrono o al anterior**. Ejemplo: evaluación crediticia, contratación de personal, representan muchas veces el mismo proceso de negocio de la empresa.

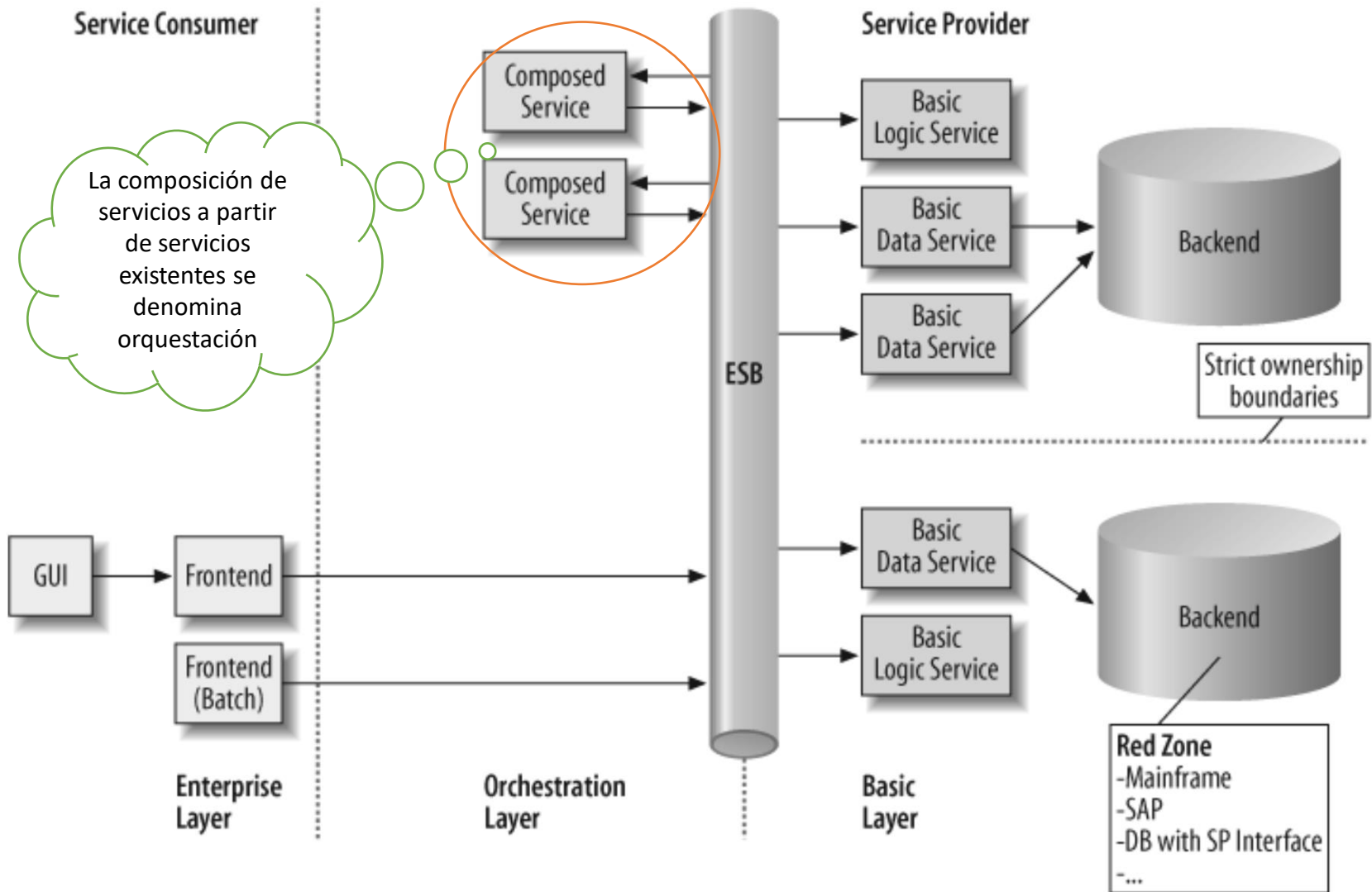
Capa de Interface al Cliente

- Sirve para **publicar o exponer algún servicio o lógica de la capa de orquestación hacia los clientes**, se le denomina **Service Client Interface** o simplemente **controladores** en el mundo de patrones de diseño como MVC.



SOA FUNDAMENTAL

Servicios Básicos – Basic Service – 1 Capa



SOA FEDERADA

Servicios Compuestos – Orchestration- 2 Capas

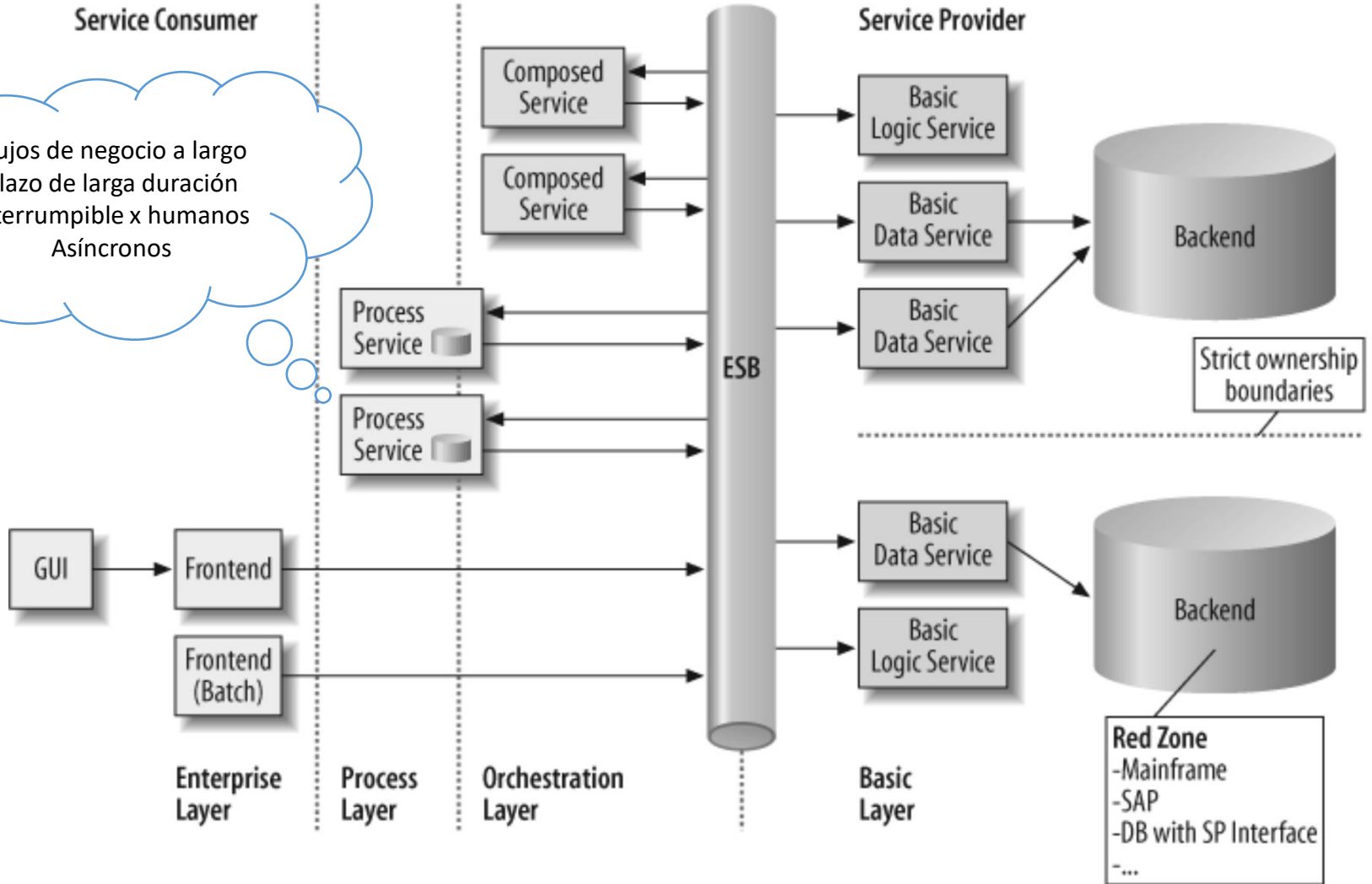
Orquestación vs. Coreografía

- **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.
- **Coreografía:** Es la opción que más encaja con la arquitectura de microservicios, pero también más compleja de gestionar. 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. Publish and Subscribers

Service Consumer

Service Provider

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



SOA Habilitado para Procesos / Process Enabled / BPEL

Método para Identificar Servicios

- Diseñar el diagrama de clases del Sistema
- Seleccionar los métodos de negocio y dividirlos:
 - Sólo entidades
 - Funcionalidades DAO que trabajan con (Repositorio), incluye CRUD
 - Funcionalidades que reutilizan servicios DAO o asimismos (Capa de Negocio u Orquestación, No Agnóstica)
 - Funcionalidades que invocan servicios o recursos externos (Capa de Integración)
 - Utilitarios (Capa de integración y Utilitarios)

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:

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

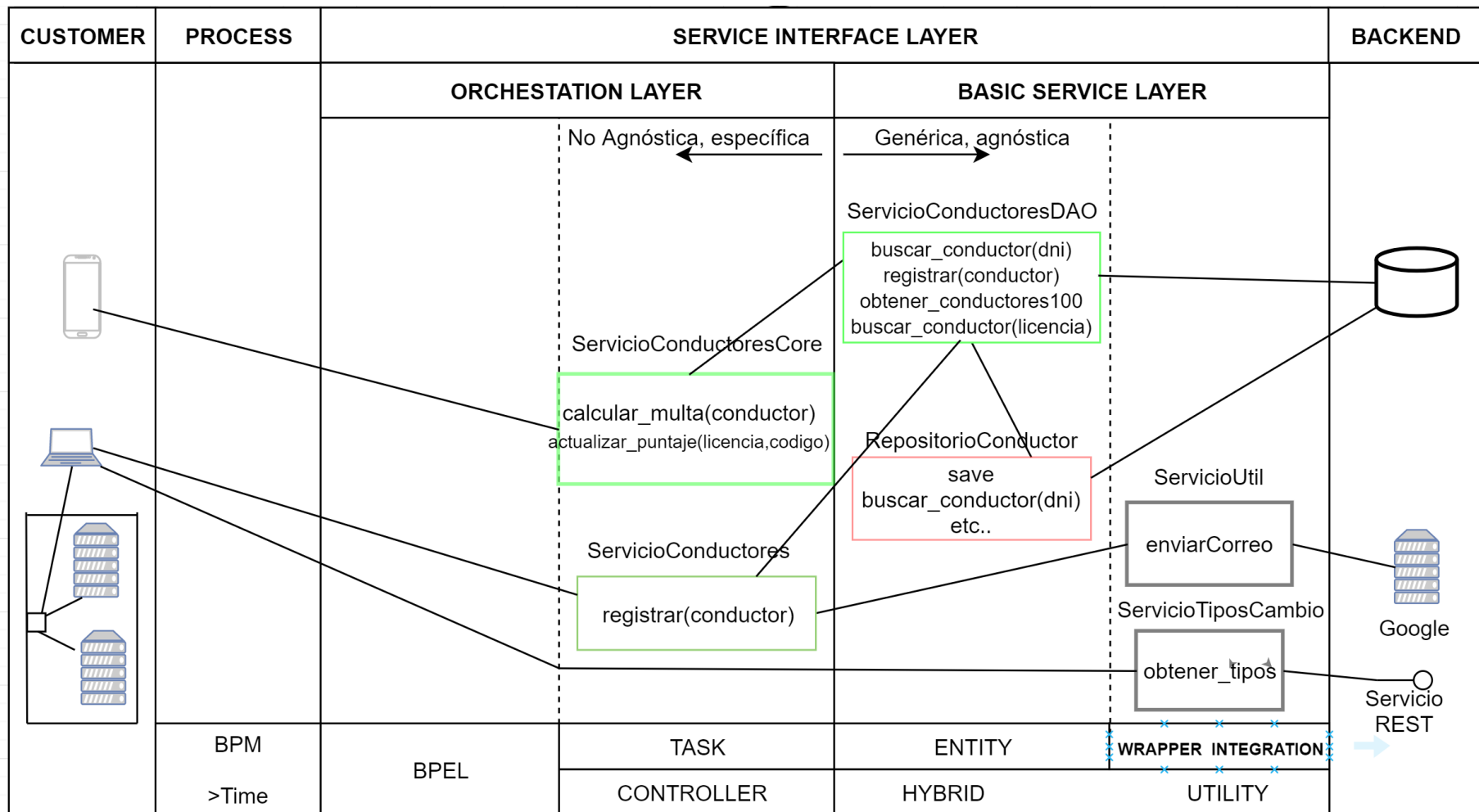
Para esta práctica no será necesario crear la tabla o entidad Penalidad.

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.



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