

## Las tecnologías SOA y ESB como herramientas integradoras para el acceso unificado a servicios colaborativos heterogéneos

Boccalari Cristian

# Temario General

Visión global del informe (diagrama)

Free/Libre Open Source Software (FLOSS)

Motivación e influencias del informe

Integración de aplicaciones y patrones de integración

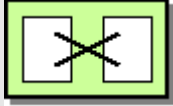
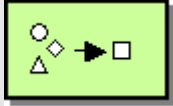
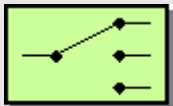
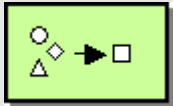
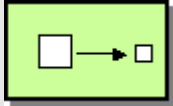
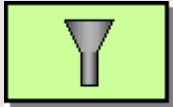
Service-oriented Architecture (SOA)

Enterprise Service Bus (ESB)

Java Business Integration (JBI) y jESBihca

# Visión global del informe

Patrones de Integración



## Integración de Aplicaciones

Service-oriented Architecture (SOA)

Enterprise Service Bus (ESB)

Java Business Integration (JBI)

jESBihca

Lucene

Apache Camel

Java

The Apache Software Foundation  
<http://www.apache.org/>

HIBERNATE

<?xml?>

ServiceMix

spring source

Free/Libre Open Source Software (FLOSS)

# El informe se desarrolla en 3 fases

## 1. Motivación

Experiencia Laboral  
+  
Software Libre

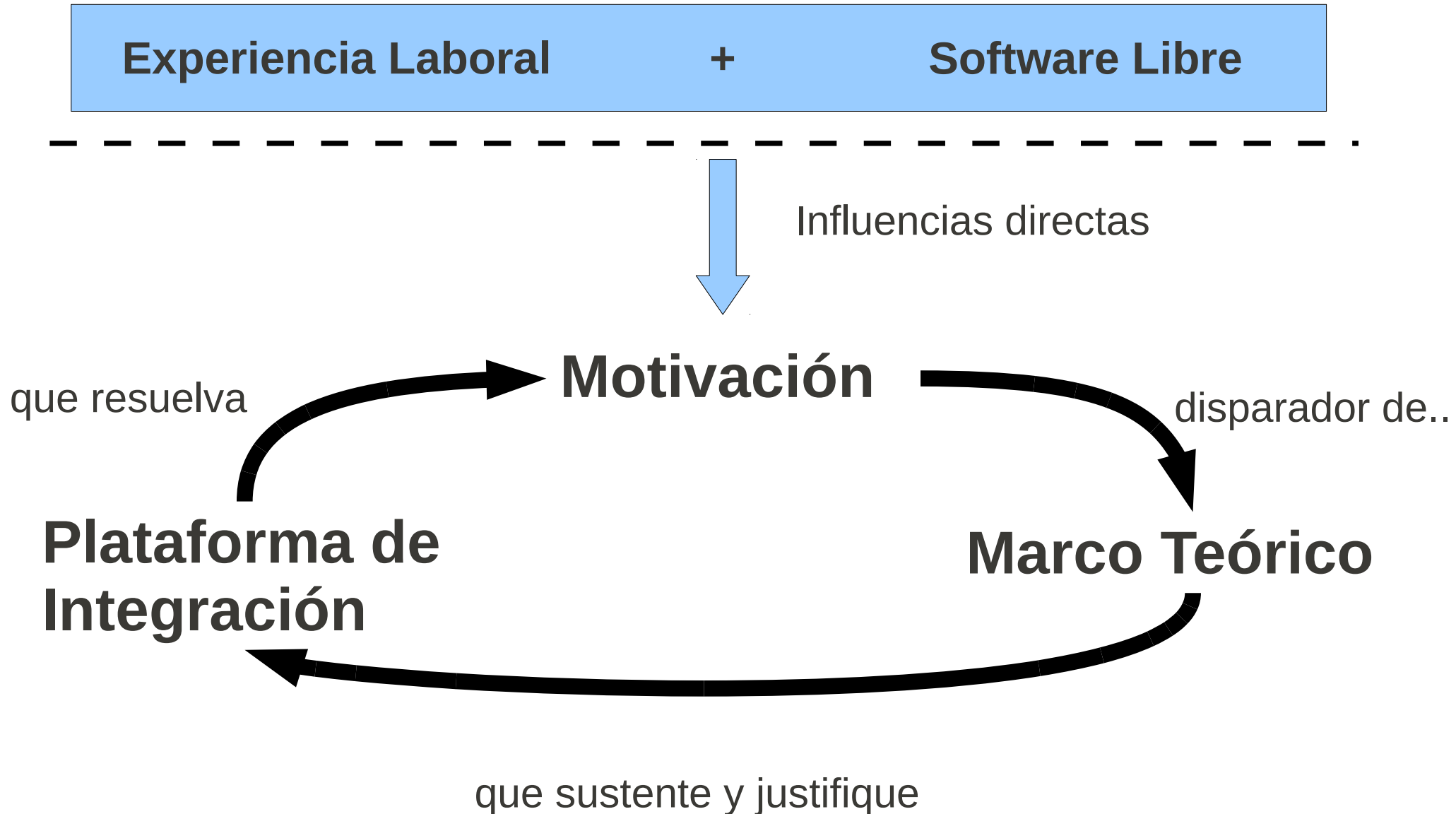
## 2. Marco Teórico

Integración de Aplicaciones  
Arquitecturas Orientadas a  
Servicios (SOA)  
Enterprise Service Bus (ESB)  
Java Business Integration (JBI)

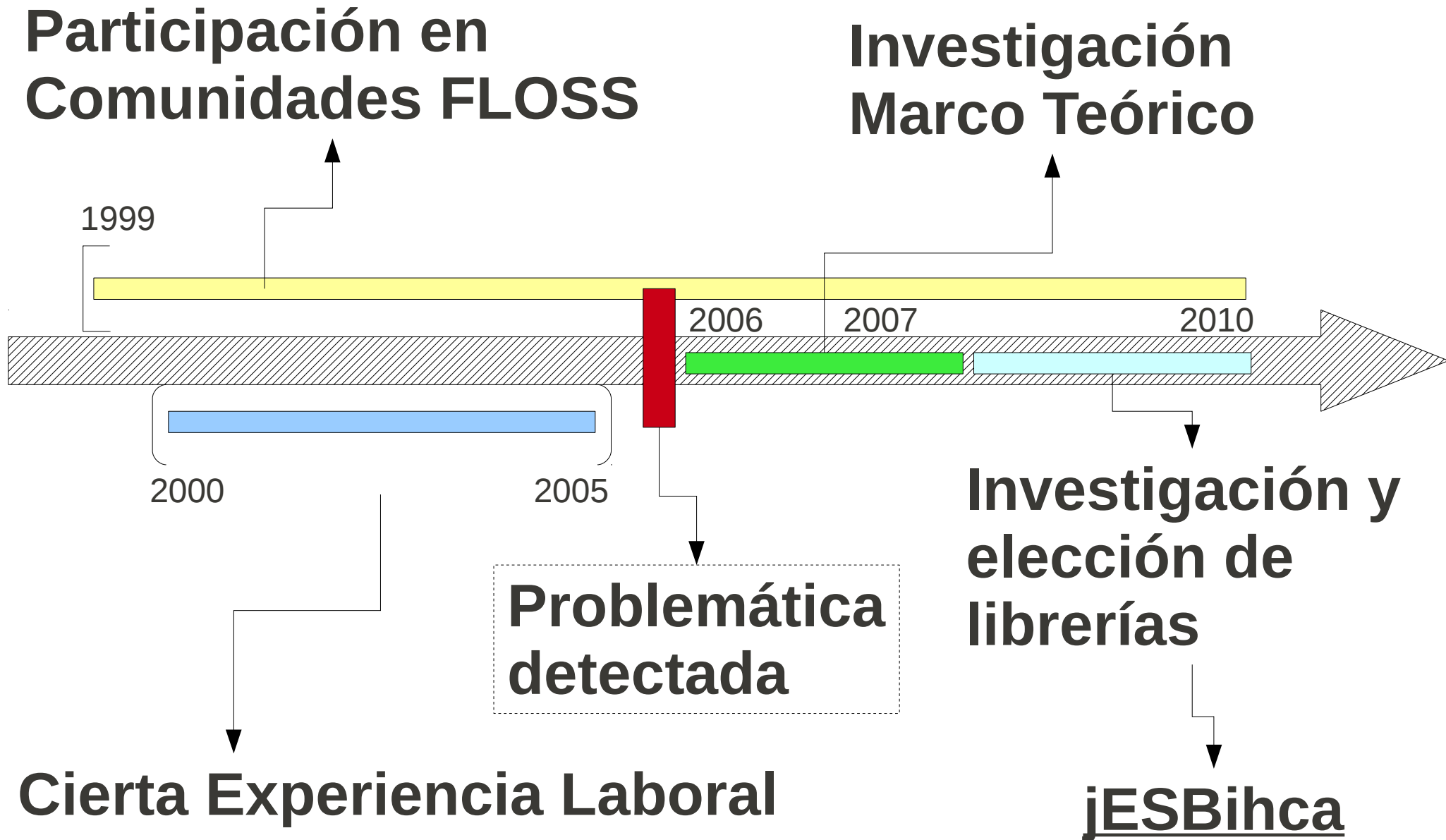
## 3. Implementación práctica

► jESBihca: plataforma FLOSS de integración

# Relación entre las 3 fases del informe



# Visión General: Línea de Tiempo



# Free/Libre Open Source Software (FLOSS)



# Motivación: Influencias

## Software Libre

↳ es el que una vez obtenido, puede ser...

**... usado ...**

**... estudiado y adaptado ... \***

**... copiado y distribuido ...**

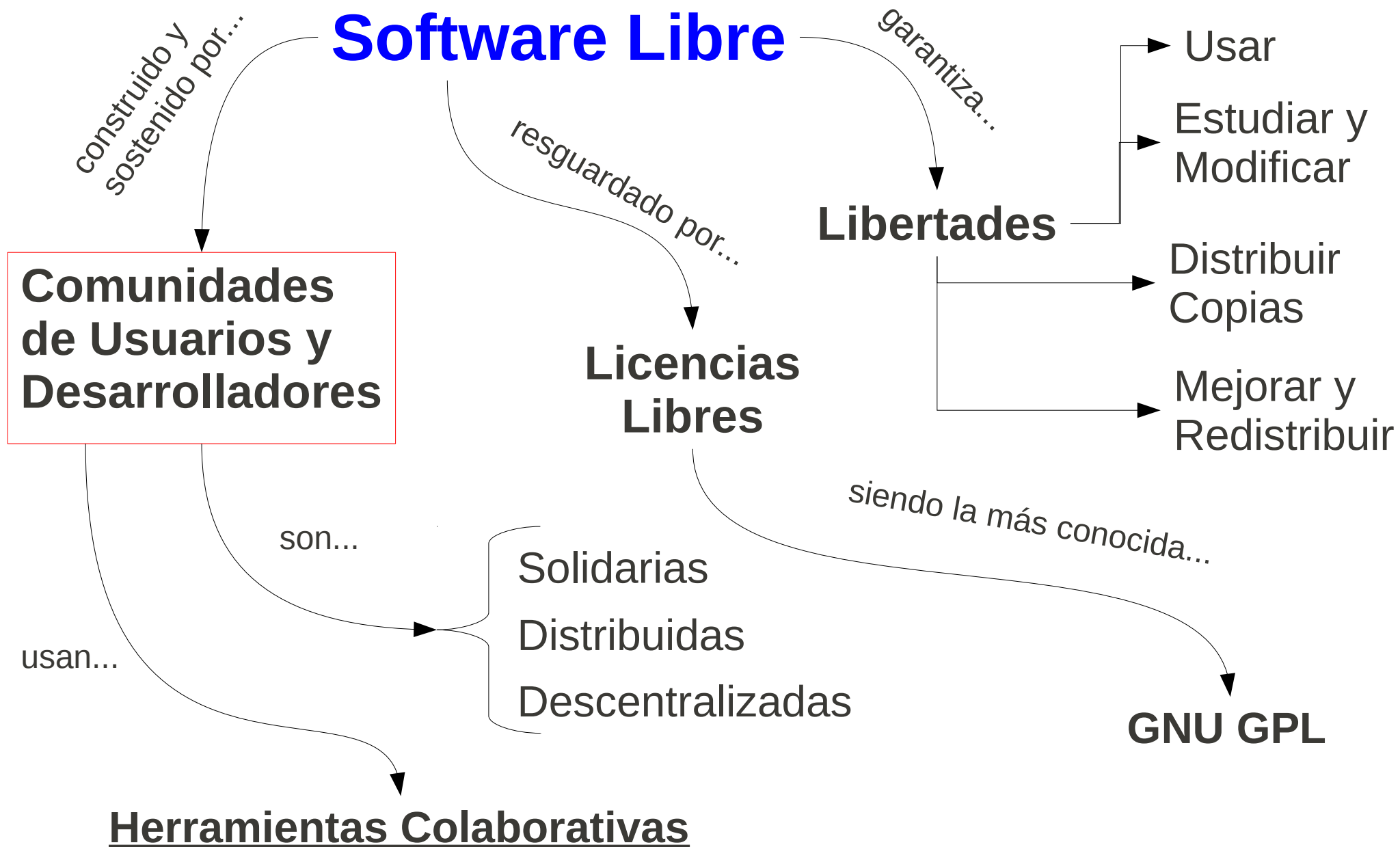
**... mejorado y redistribuido ... \***

**... LIBREMENTE !!!**

\* requiere acceso al código fuente



# Mapa del Software Libre



# Comunidades FLOSS

Desarrollo descentralizado y distribuido

Uso de diversas herramientas de comunicación, usualmente asincrónicas.

Asignación de roles definidos

Múltiples colaboradores con competencias diversas

Aplicación del “Modelo Bazar”

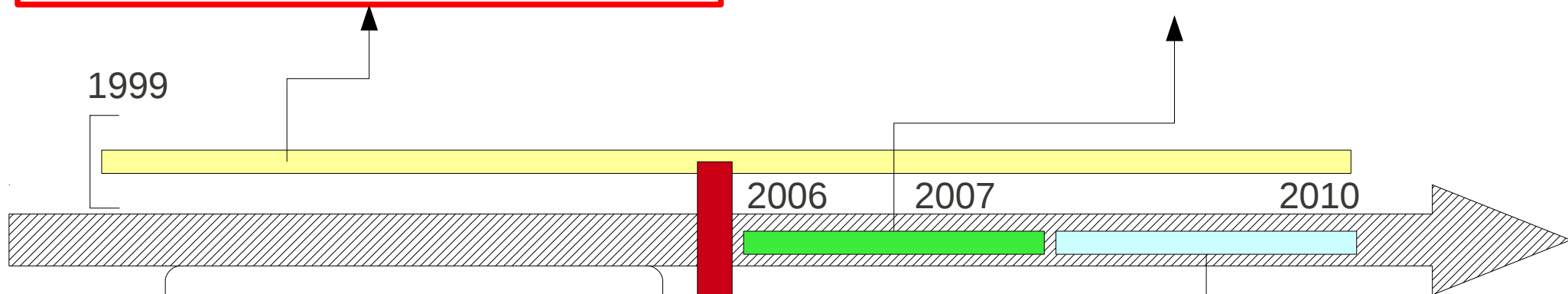
# Herramientas Colaborativas para la comunicación y coordinación

- Listas de Correo (Mailman)
- **WIKI** (Mediawiki, Docuwiki)
- **Bug Tracker** (Mantis BT)
- Manejadores de contenido (Drupal, OpenCMS)
- **FTP**, WebDAV
- Foros (PHPBB)
- Chat (IRC)
- Mensajería Instantánea (Jabber)
- **Control de versiones del código fuente** (CVS, SVN)
- Blogs (Wordpress, Drupal)

# Visión General: Línea de Tiempo

**Participación en Comunidades FLOSS**

**Investigación Marco Teórico**



2006

2007

2010

1999

2000

2005

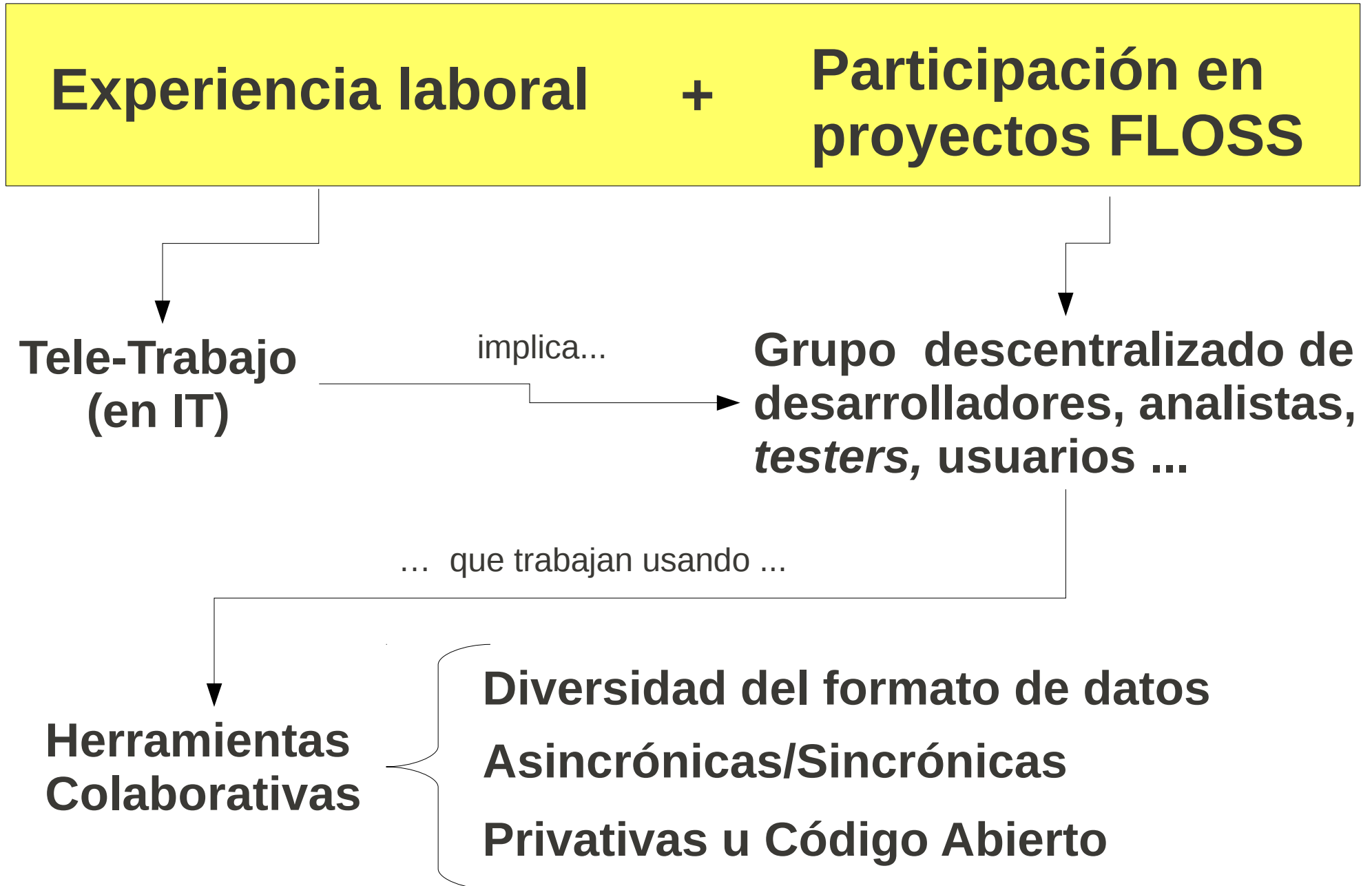
**Problemática detectada**

**Investigación y elección de librerías**

**Cierta Experiencia Laboral**

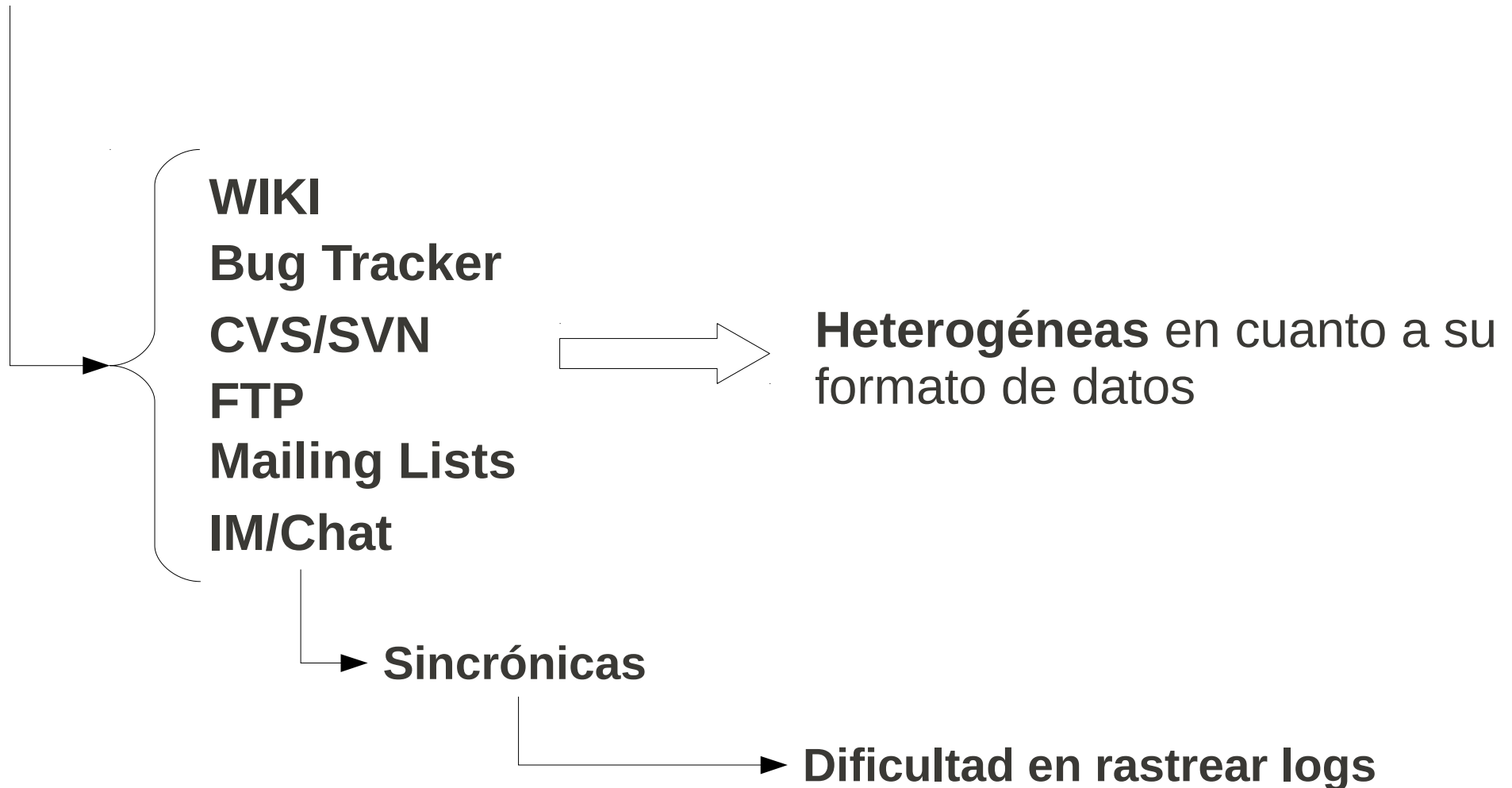
**jESBihca**

# Motivación: Influencias



# Motivación: problemática

## Existencia de múltiples Herramientas Colaborativas



# Motivación: problemática

La existencia de múltiples Herramientas Colaborativas Heterogéneas

producía...

→ Dispersión de la información

WIKI  
Bug Tracker  
CVS/SVN  
FTP  
Mailing Lists  
IM/Chat

generaba...

**Dificultad**

al momento de...

Buscar

+

Recuperar

→ información  
relevante

# Motivación: Resumiendo

Participaba en **comunidades FLOSS** públicas y privadas

Las comunidades usaban **herramientas colaborativas** donde almacenaban el conocimiento generado

Las herramientas colaborativas eran **heterogéneas**

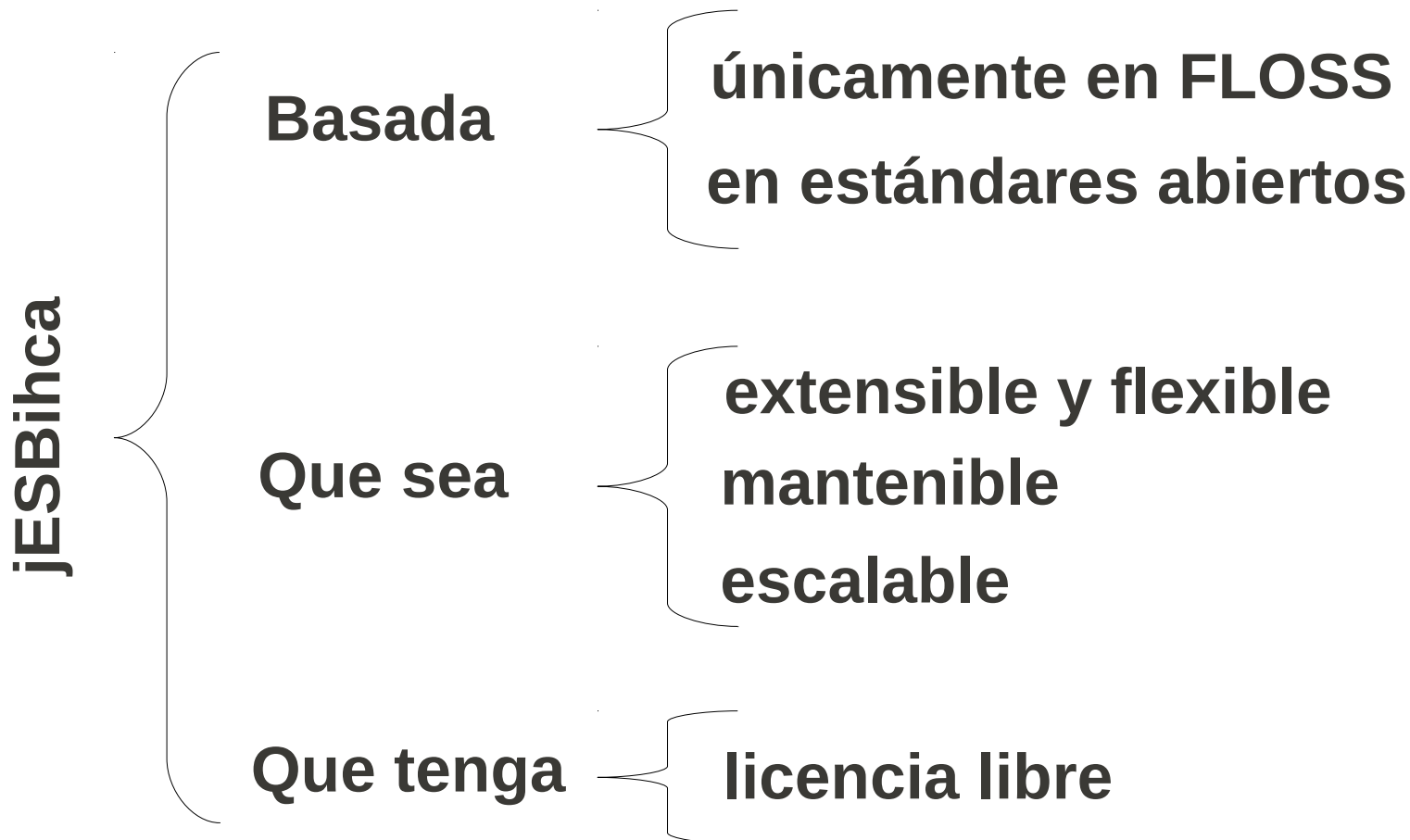
La **información** que almacenaban estaba **dispersa**

La **búsqueda** de información era **difícil**



# Motivación: Solución propuesta

**Desarrollar un prototipo de plataforma de integración y acceso a información almacenada en herramientas colaborativas asincrónicas**

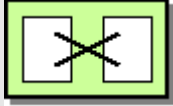
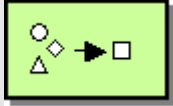
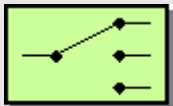
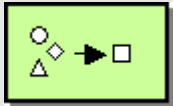
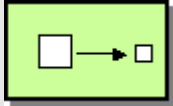
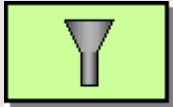


# Integración de Aplicaciones



# Visión global del informe

Patrones de Integración



## Integración de Aplicaciones

Service-oriented Architecture (SOA)

Enterprise Service Bus (ESB)

Java Business Integration (JBI)

jESBihca

Lucene

Apache Camel

Java

The Apache Software Foundation  
<http://www.apache.org/>

HIBERNATE

<?xml?>

ServiceMix

spring source

Free/Libre Open Source Software (FLOSS)

# Integración de Aplicaciones

## Necesidad



La integración de aplicaciones surge ante la necesidad de conectar múltiples sistemas ...

▶ ... generalmente **heterogéneos**

# Integración de Aplicaciones

## Principal Objetivo



Es que los sistemas integrados puedan proveer una visión unificada de funcionalidad y/o datos

# Integración de Aplicaciones

## Dificultades



Los sistemas pueden estar contruidos con **tecnologías diferentes**

Los sistemas pueden ser **aplicaciones heredadas** sin documentación

Los sistemas pueden pertenecer a **distintas organizaciones**

Los sistemas pueden utilizar **formatos de datos distintos**

# Integración de Aplicaciones

## Middleware

Es un software de conectividad que ofrece un conjunto de servicios que hacen posible el funcionamiento de aplicaciones distribuidas sobre plataformas heterogéneas

**Tecnologías**

Tecnologías de base de datos      Servidores de aplicaciones

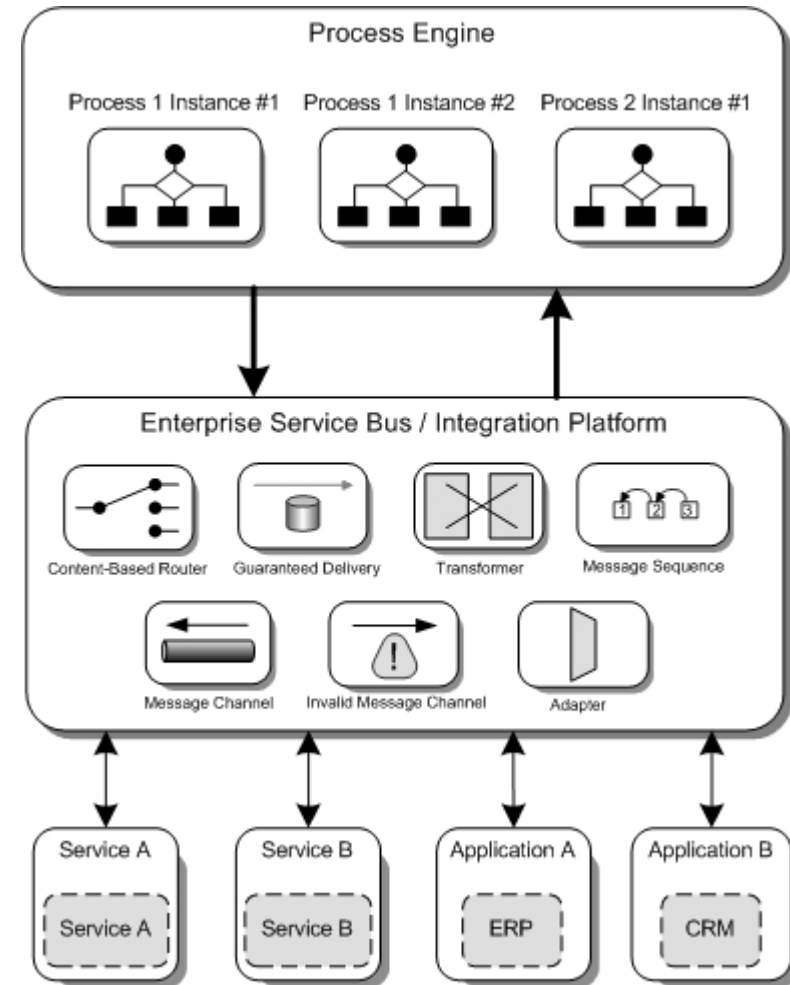
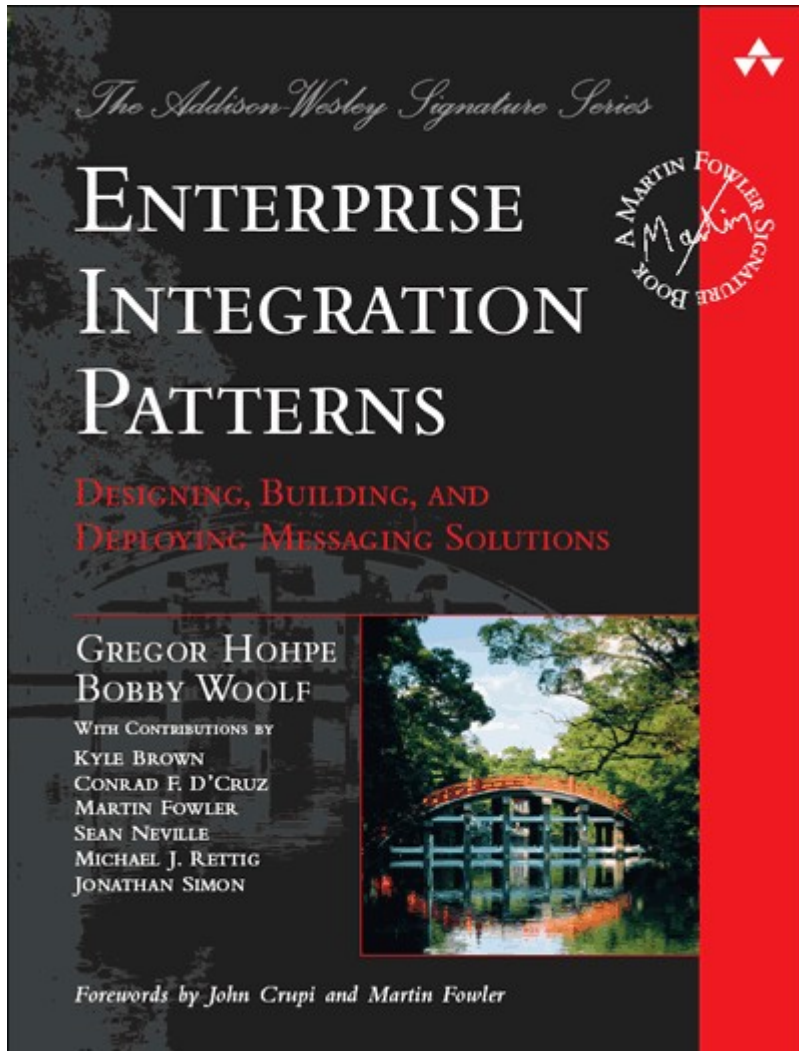
Message-oriented Middleware (MOM)

Remote Procedure Call (RPC)      Object Request Broker (ORB)

**Enterprise Service Bus (ESB)**

**Servicios Web**

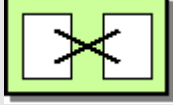
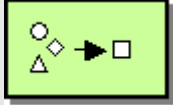
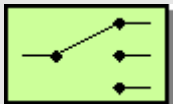
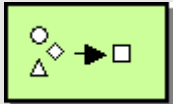
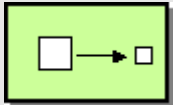
# Patrones de Integración





# Visión global del informe

Patrones de Integración



## Integración de Aplicaciones

Service-oriented Architecture (SOA)

Enterprise Service Bus (ESB)

Java Business Integration (JBI)

jESBihca

Lucene

Apache Camel

Java

The Apache Software Foundation  
<http://www.apache.org/>

HIBERNATE

<?xml?>

ServiceMix

spring source

Free/Libre Open Source Software (FLOSS)

# Patrones de Integración

## ¿Qué son?

Describen métodos para la resolución de problemas recurrentes

## ¿Qué no son?

No deben ser considerados una solución "lista para usarse"

## ¿Para qué sirven?

Usados adecuadamente pueden ayudar a disminuir la brecha existente entre la visión global y abstracta de la integración y la implementación real del sistema

# Patrones de Integración

## Principales patrones

### Canales de mensajería

Point-to-Point channel

Publish & Subscribe channel



### Ruteo de mensajes

Content-based Router (CBR)

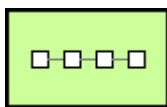
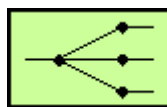
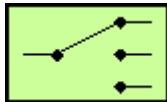
Message Filter

Dynamic router

Recipients List

Splitter

Routing Slip

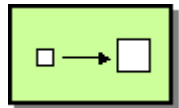


# Patrones de Integración

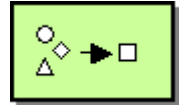
## Principales patrones (cont...)

**Transformación de mensajes**

Content Enricher



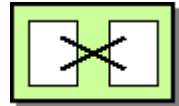
Normalizer



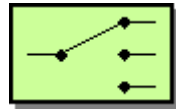
Envelope Wrapper

Content Filter

Message Translator



# Patrones de Integración



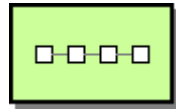
## Content-based Router

### Descripción

Utilizado por sistemas que requieren leer el contenido de los mensajes para, junto a ciertas reglas de ruteo, redirigirlos a los destinatarios adecuados

Demo: [pattern-cbr.ogv](#)

# Patrones de Integración



## Routing Slip

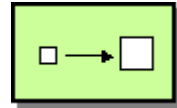
### Descripción

Este patrón es usado cuando se requiere conocer el camino completo que recorrerá un mensaje.

Cada mensaje entrante tendrá asociado la secuencia de pasos de procesamiento que necesitará atravesar

Demo: [pattern-routing-slip.ogv](#)

# Patrones de Integración



## Content Enricher

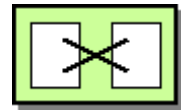
### Descripción

Este patrón es usado en situaciones que el destinatario del mensaje requiere más información que la que el remitente puede proveer.

En estos casos es necesario "enriquecer" el mensaje original con información adicional recuperada de fuentes de información externas

Demo: [pattern-content-enricer.ogv](#)

# Patrones de Integración



## Message Translator

### Descripción

Este patrón se utiliza entonces en los escenarios de integración en los cuales es necesario transformar los formatos de los datos de los sistemas que están interactuando

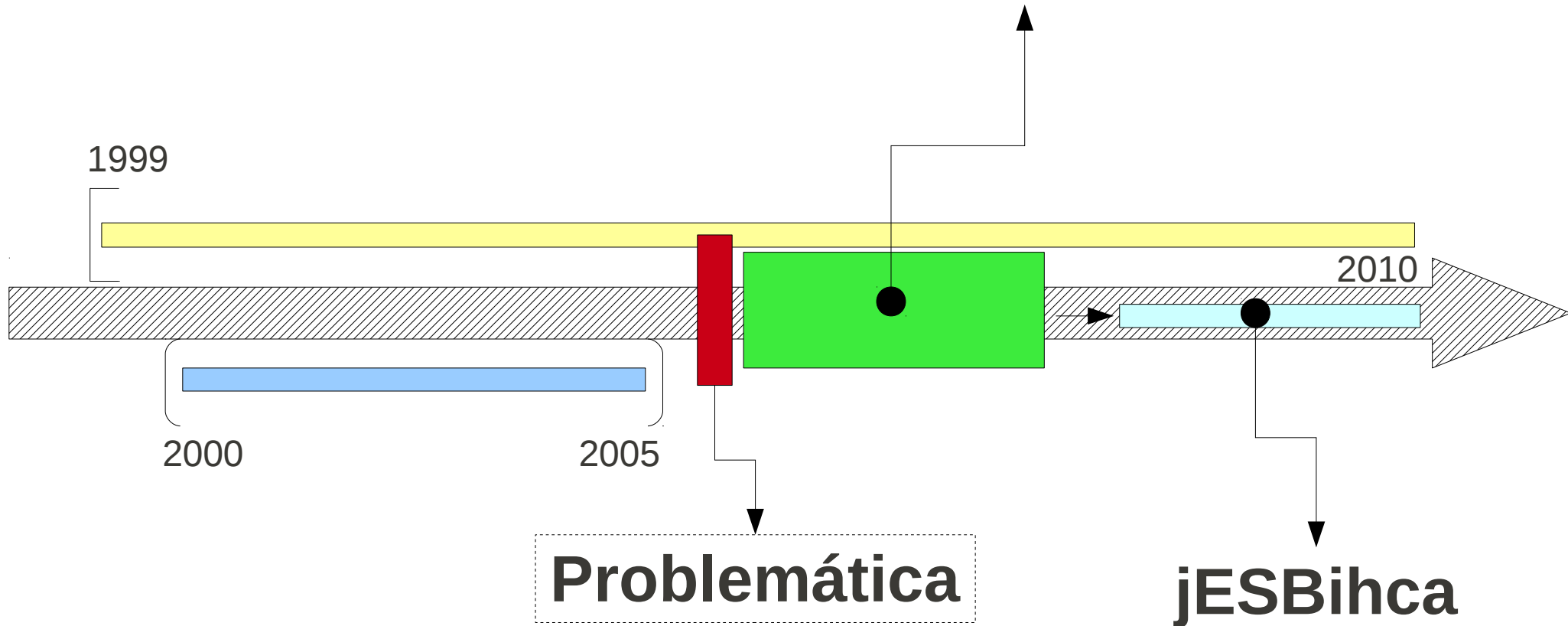


# Service-oriented Architecture (SOA)



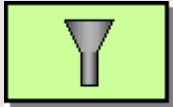
# Línea de Tiempo

## Investigación y Marco Teórico

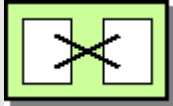
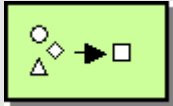
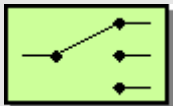
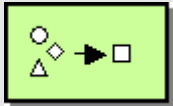
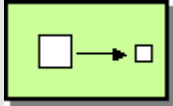


# Visión global del informe

Patrones de Integración



## Integración de Aplicaciones



**Service-oriented Architecture (SOA)**

**Enterprise Service Bus (ESB)**

**Java Business Integration (JBI)**

**jESBihca**

*Lucerne*

**Apache  
Camel**

**Java**

 **The Apache  
Software Foundation**  
<http://www.apache.org/>

 **HIBERNATE**

**<?xml?>**

*ServiceMix*

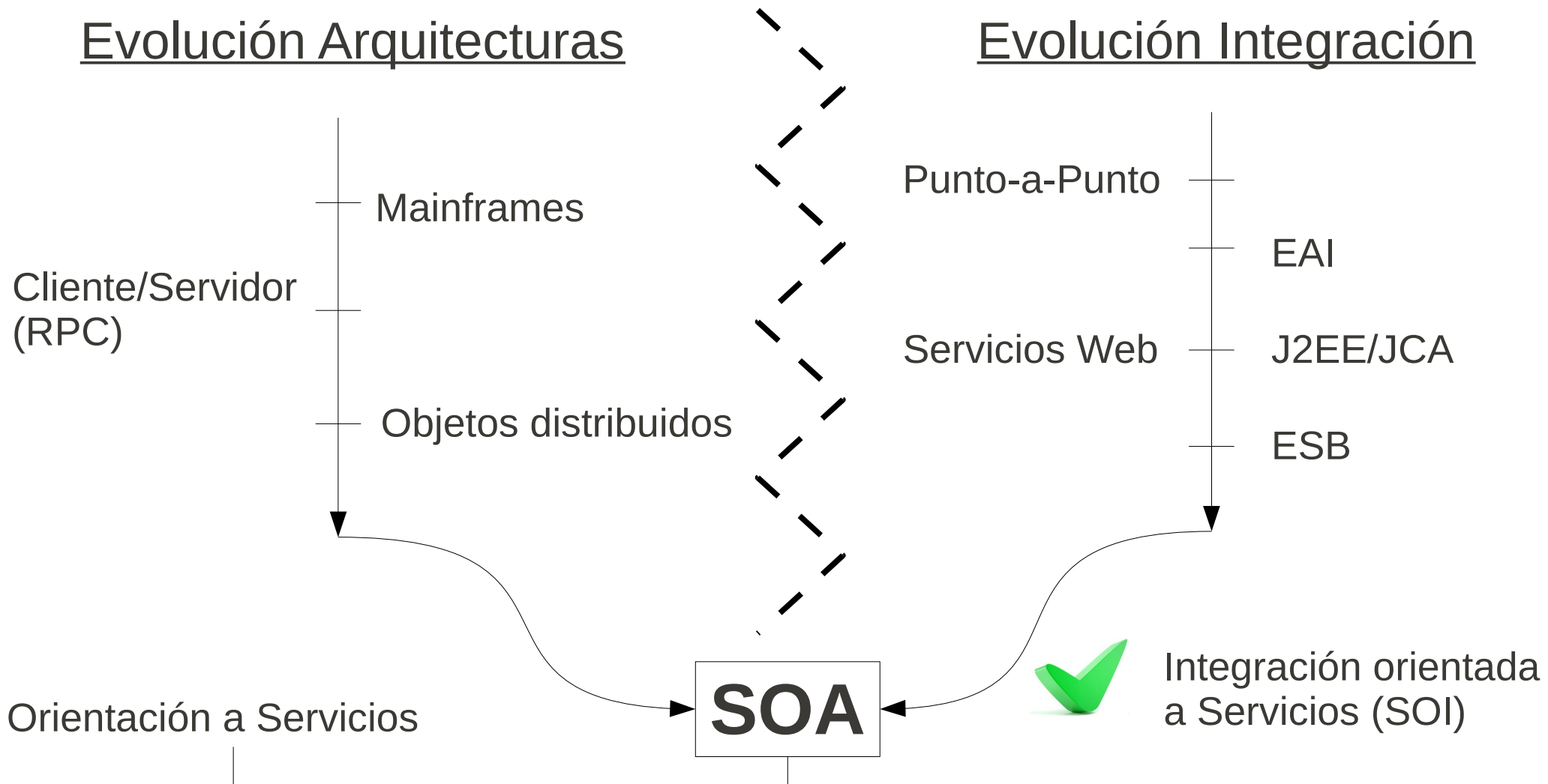
 **spring  
source**

**Free/Libre Open Source Software (FLOSS)**

# Convergencia en la evolución de arquitecturas e integración

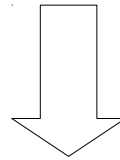
## Evolución Arquitecturas

## Evolución Integración

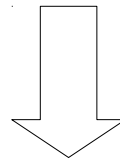


# SOA: Disgresión

Teniendo en mente el objetivo final de lograr implementar un prototipo de plataforma de integración orientada a servicios y basada en FLOSS

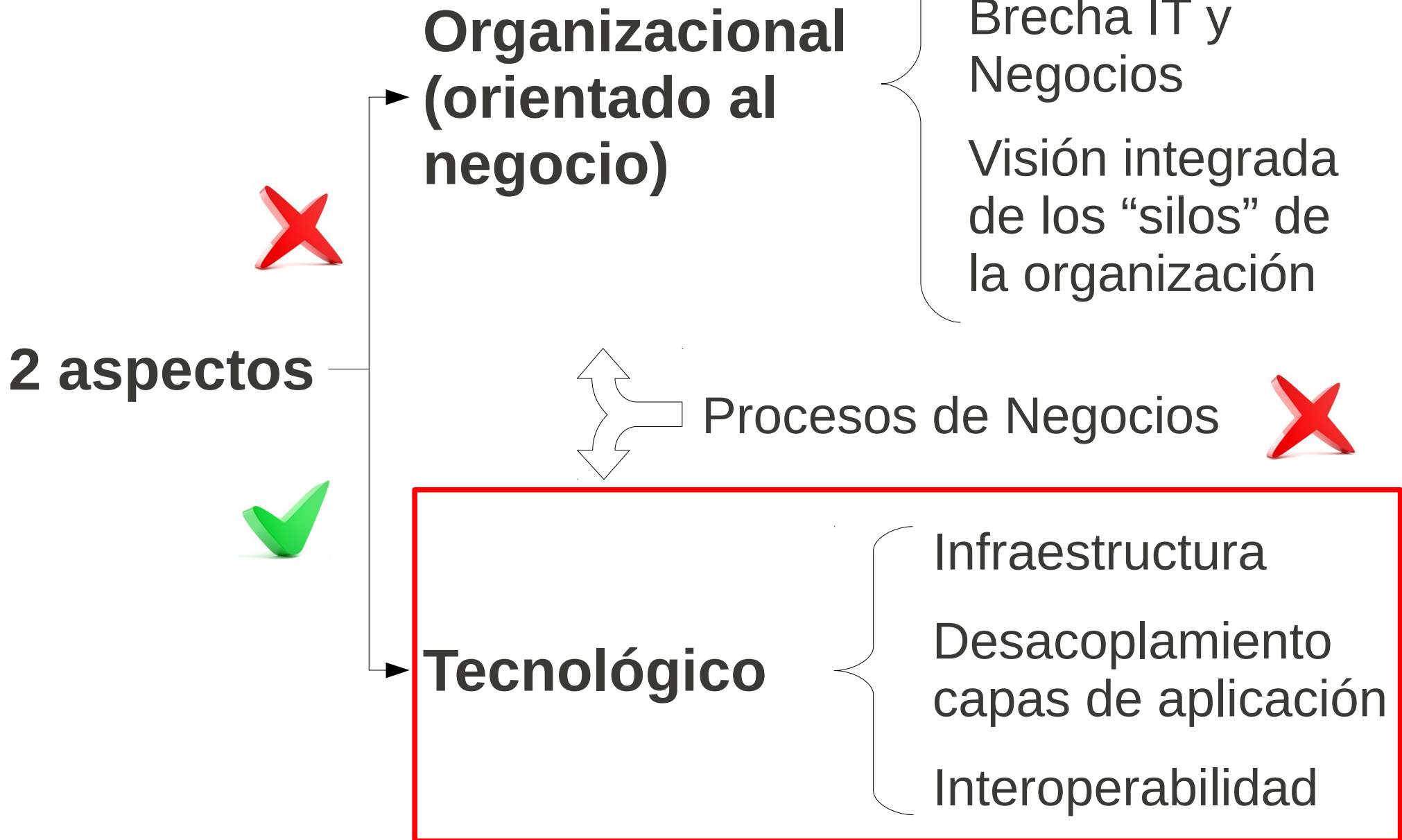


El informe focalizó únicamente en el **aspecto técnico** de SOA...

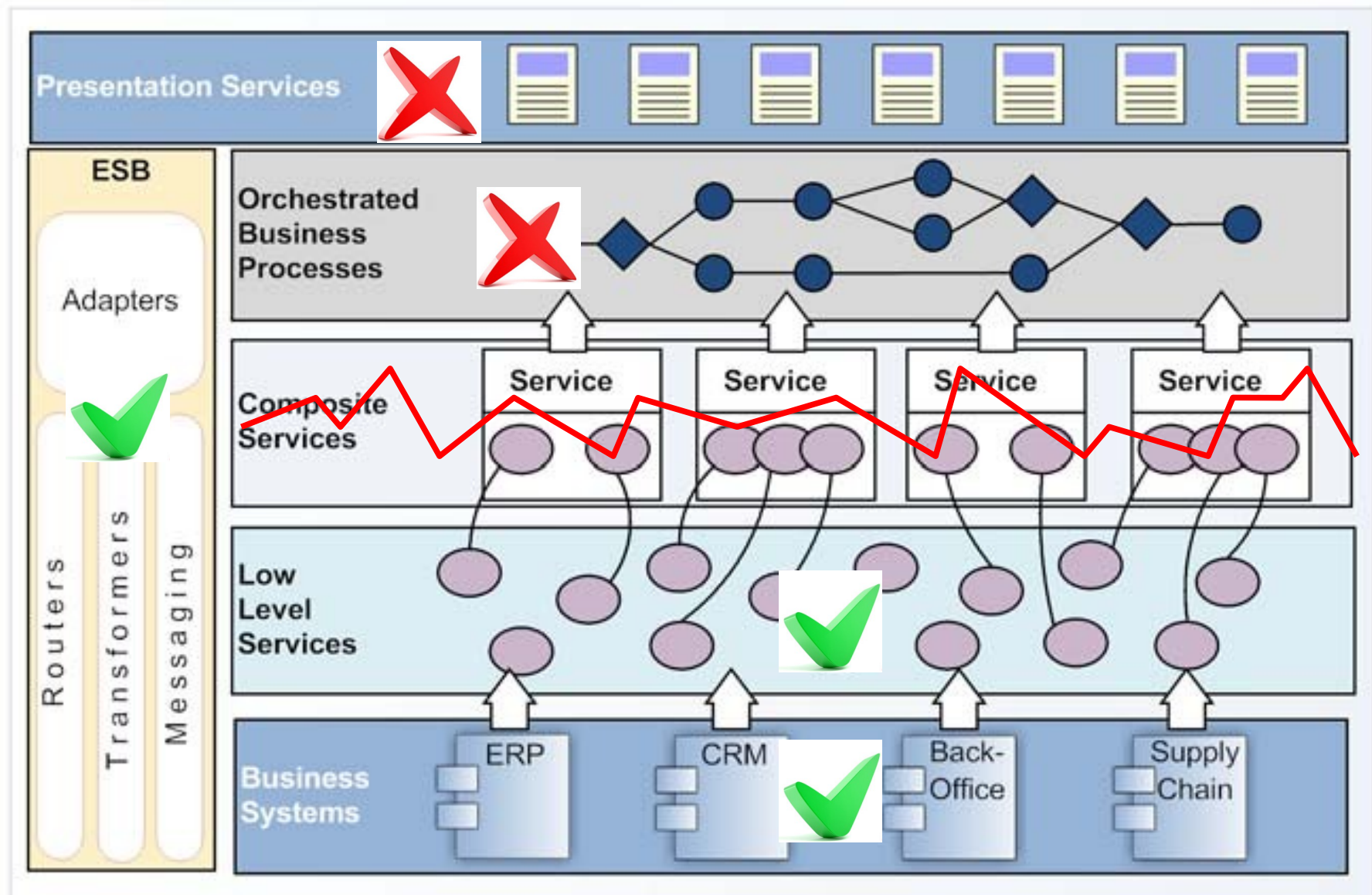


...que servirá de guía en la posterior la implementación de jESBihca

# SOA: Visiones



# Ambiente SOA: visión global



# Definiciones de SOA

Según el modelo de referencia OASIS, SOA es...

→ “... un **paradigma** para...”

Según Thomas Erl SOA...

→ “... representa una **arquitectura** abierta, extensible ....”

Según Eric Newcomer SOA es...

→ “... un **estilo de diseño** que guía...”



# Entonces... ¿Qué es SOA?

Elegimos ver a  
SO~~X~~ como ...

un paradigma   
una forma de pensar

SO~~X~~ no es algo que pueda comprarse



soa open source download

[Búsqueda avanzada](#)  
[Herramientas del idioma](#)

Buscar con Google

Voy a tener suerte

# Conceptos técnicos claves

- **Servicios**

Representación IT de alguna funcionalidad de negocios autónoma

- **Interoperabilidad**

Conectar fácilmente sistemas heterogéneos

Infraestructura

- **Débil acoplamiento**

Minimizar dependencias

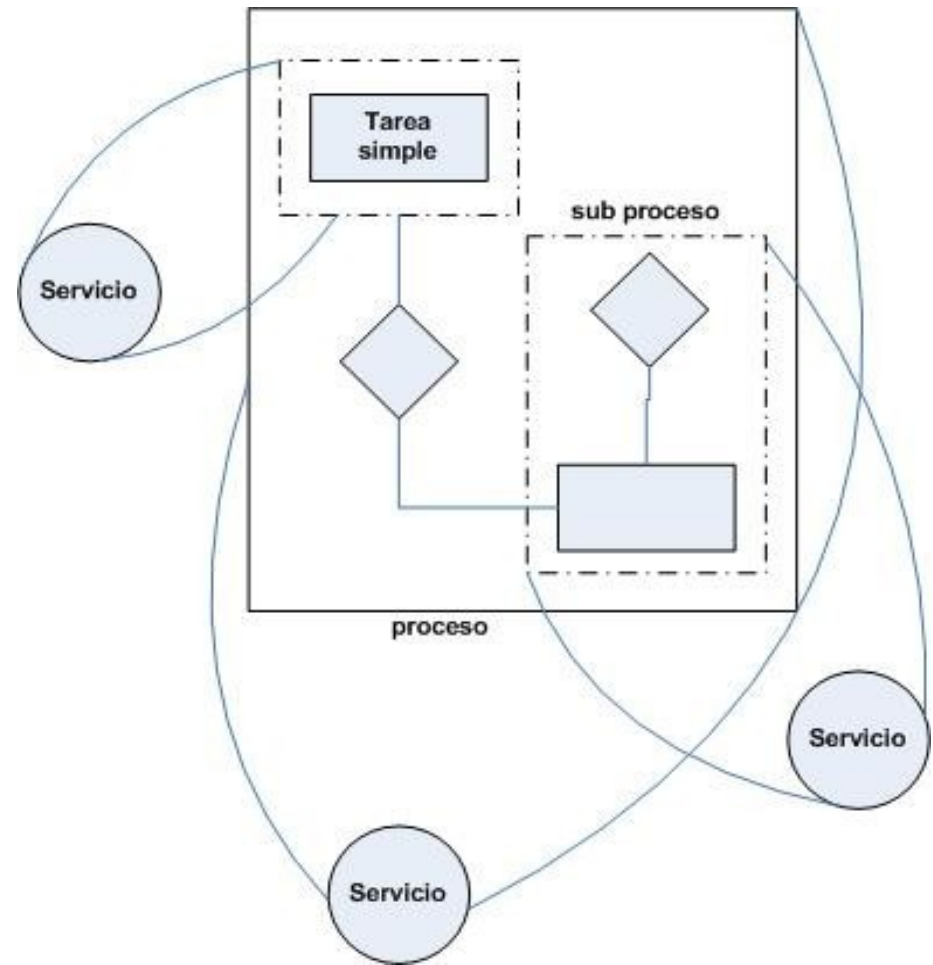
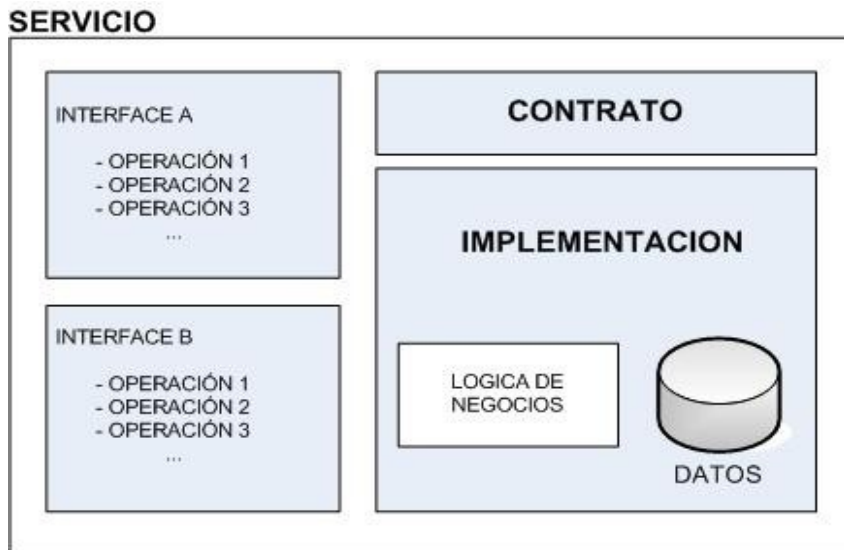
Flexibilidad

Escalabilidad

# Servicios

## Definición simple

Una unidad de funcionalidad que un proveedor de servicio deja disponible en el ambiente a través de una interface, para que los consumidores del servicio pueden hacer uso



# Principios de la Orientación a Servicios

Estos principios son independientes de cualquier tecnología específica, e implican que los servicios...

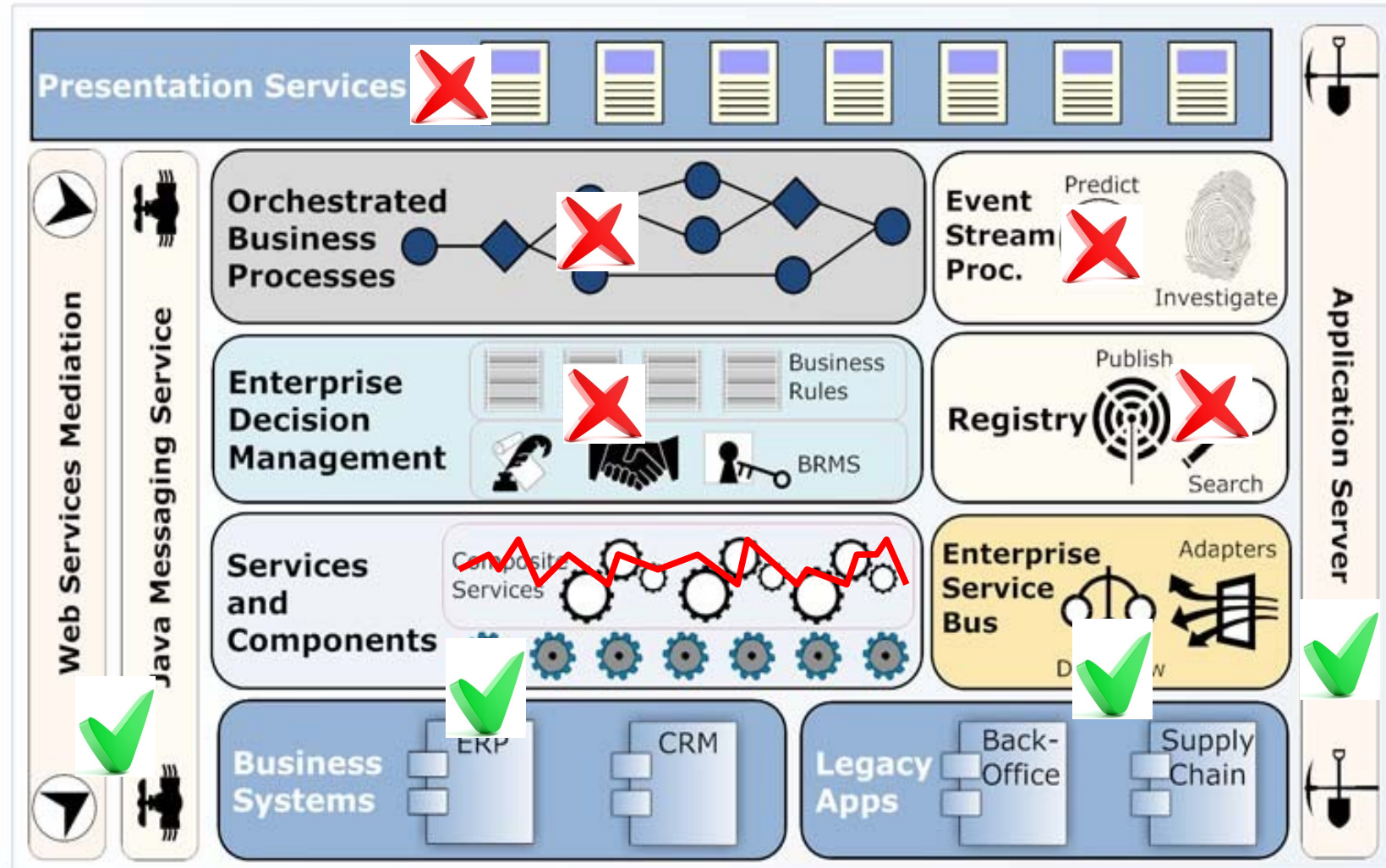
...deben...

- ser **reusables**
- proveer de un **contrato formal**
- ser **débilmente acoplados**
- permitir la **composición**
- ser **autónomos**
- poder ser **descubiertos**

...no deben

- └ tener **estado**

# SOA: Elección de Tecnologías



# SOA: Infraestructura

## Enterprise Service Bus (ESB)

Es la **parte técnica** de SOA



que permite la alta interoperabilidad

# jESBihca



**caso-de-estudio-1.ogv**

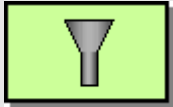
# Enterprise Service Bus (ESB)



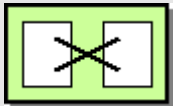
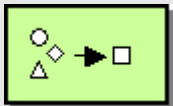
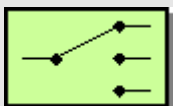
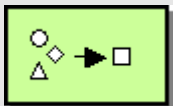
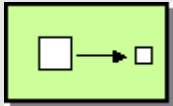


# Visión global del informe

Patrones de Integración



## Integración de Aplicaciones



Service-oriented Architecture (SOA)

Enterprise Service Bus (ESB)

Java Business Integration (JBI)

jESBihca

Lucene

Apache Camel

Java

The Apache Software Foundation  
<http://www.apache.org/>

HIBERNATE

<?xml?>

ServiceMix

spring source

Free/Libre Open Source Software (FLOSS)

# Temario

¿Qué es un ESB?

Características deseables

Infraestructura del ESB

# ¿ Qué es un ESB ?

Es un software de **middleware**

Actúa como **mediador** entre diferentes e incompatibles protocolos

Es parte fundamental de la **infraestructura de SOA**

Un SOA no puede comprarse, pero un ESB sí

# Características deseables de un ESB

**Basado en estándares abiertos**

**Basado en mensajes**

XML XSL JBI WSDL SOAP JMS

**Extensible**

**Distribuido**

Service Unit

**Transformación de datos**

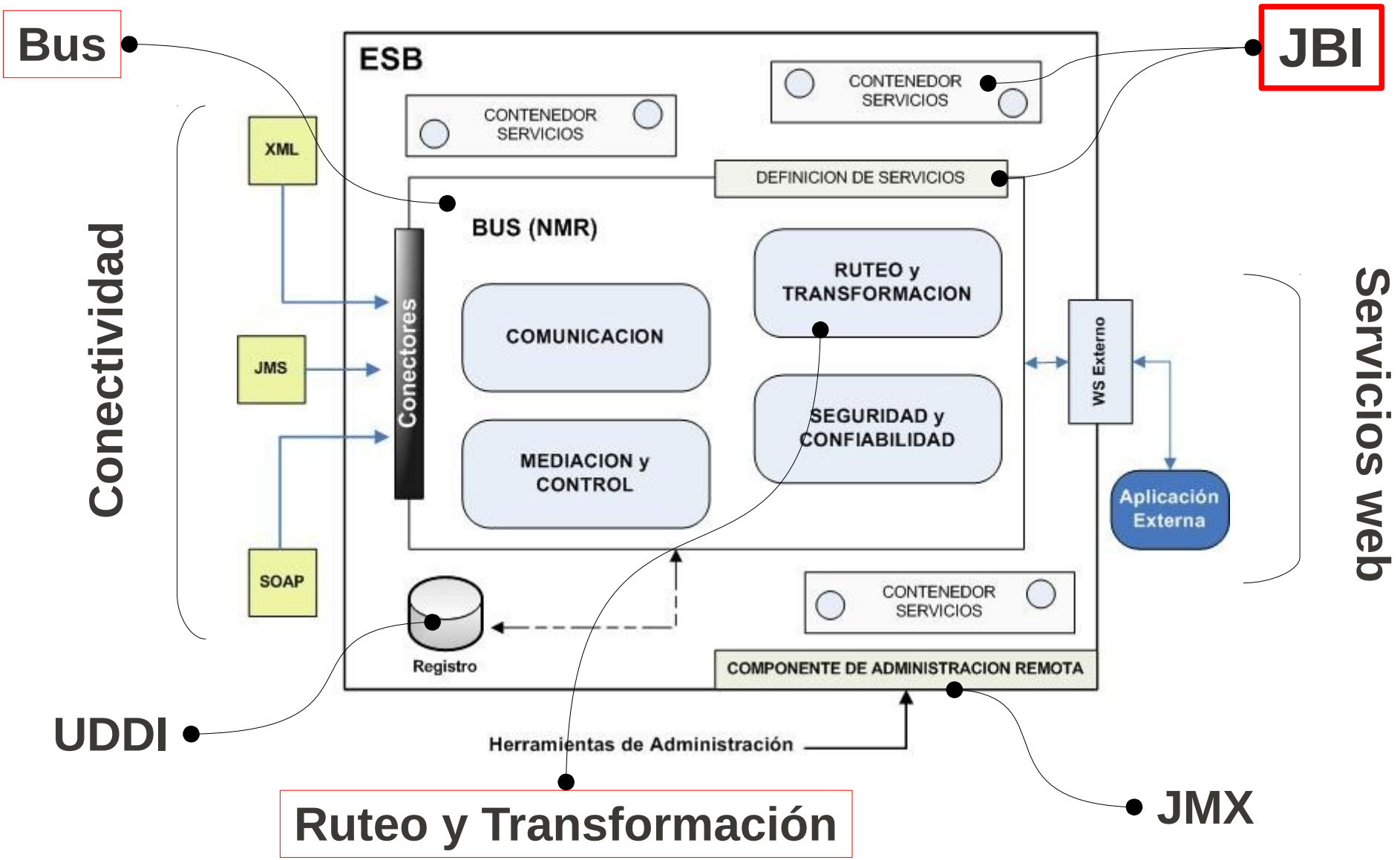
**Configurable**

**Débil acoplamiento**

JMX

**Tipo de datos nativo**

# Infraestructura del ESB



# JBI (Java Business Integration)

Especificación (JSR 208) definida por la JCP



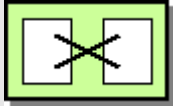
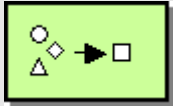
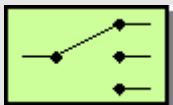
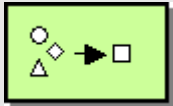
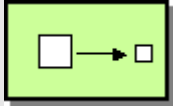
Especifica un framework orientado a *plugins* (componentes), basado en estándares abiertos

# Visión global del informe

Patrones de Integración



## Integración de Aplicaciones



### Service-oriented Architecture (SOA)

### Enterprise Service Bus (ESB)

### Java Business Integration (JBI)

**jESBihca**

*Lucene*

**Apache Camel**

**Java**

 **The Apache Software Foundation**  
<http://www.apache.org/>

 **HIBERNATE**

**<?xml?>**

**spring**  
source

*ServiceMix*

**Free/Libre Open Source Software (FLOSS)**

# Temario

Contenedor JBI

Arquitectura JBI

Principales conceptos JBI

Componentes BC y SE

Normalized Message Router (NMR)

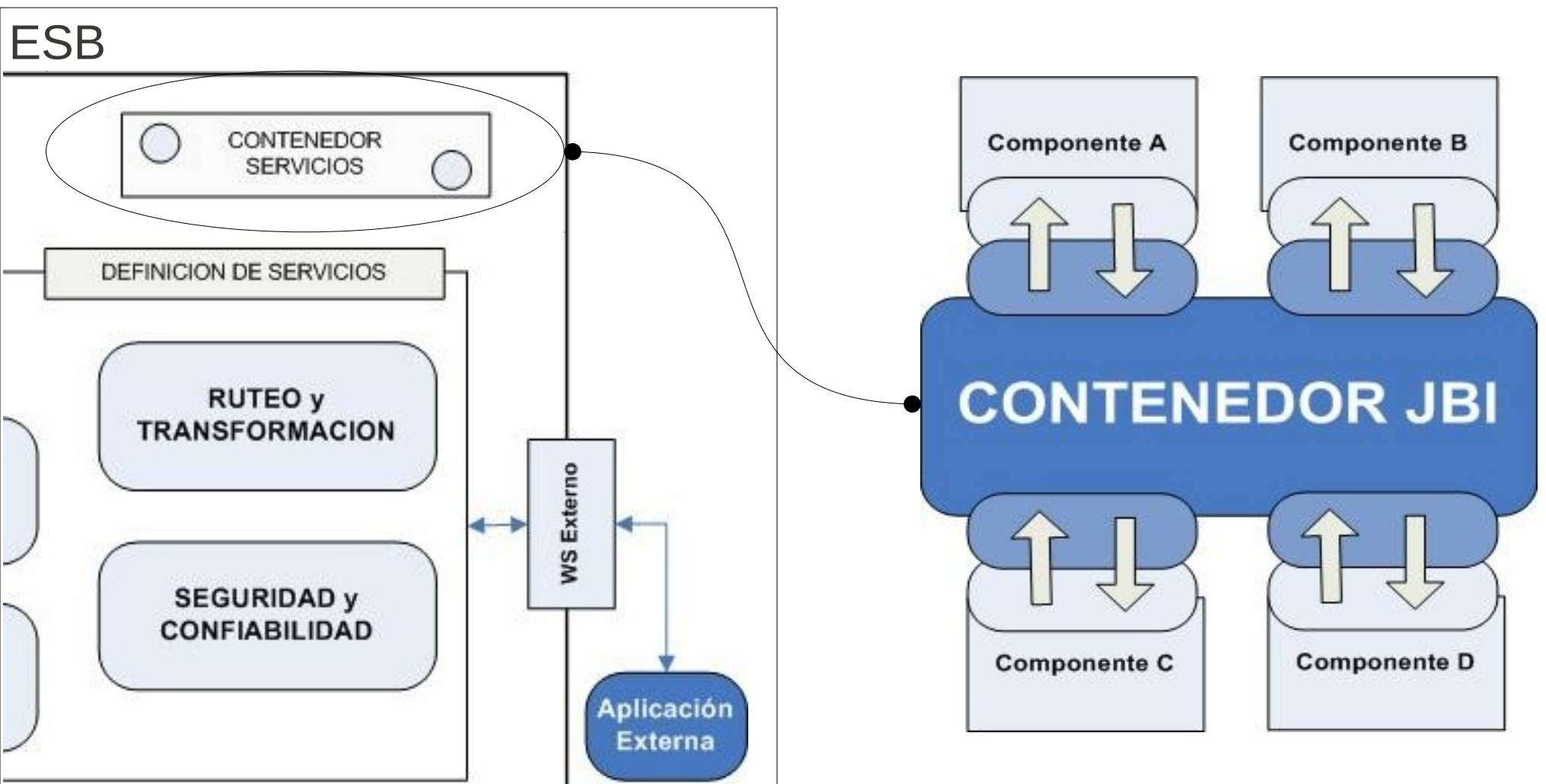
Delivery Channel (DC)



# Contenedor JBI

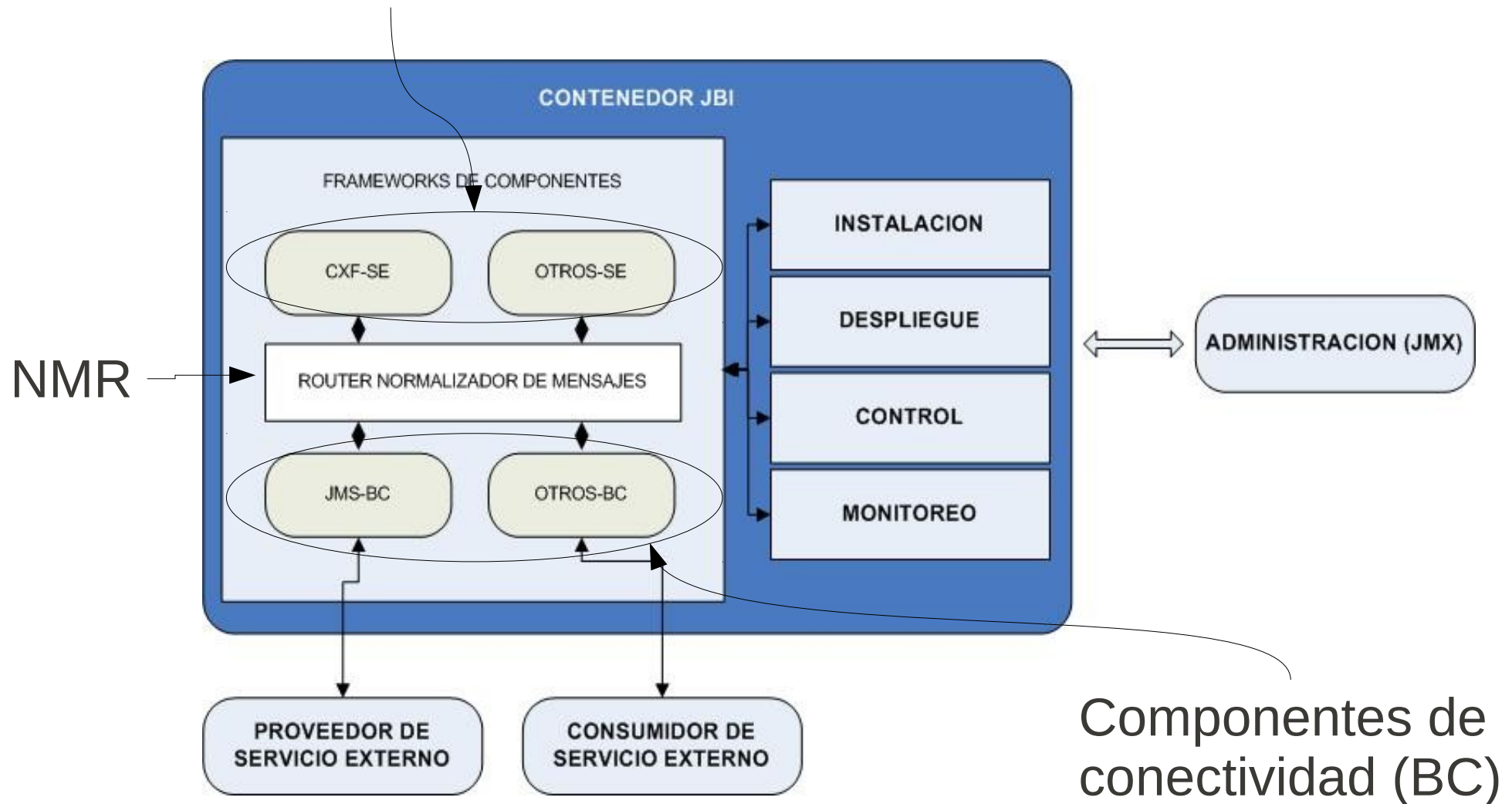
JBI es el **contenedor** de servicios seleccionado para **jESBihca**

→ Apache **ServiceMix** es la implementación de JBI elegida



# Arquitectura JBI

Componentes de negocios (SE)



# Conceptos JBI: componentes

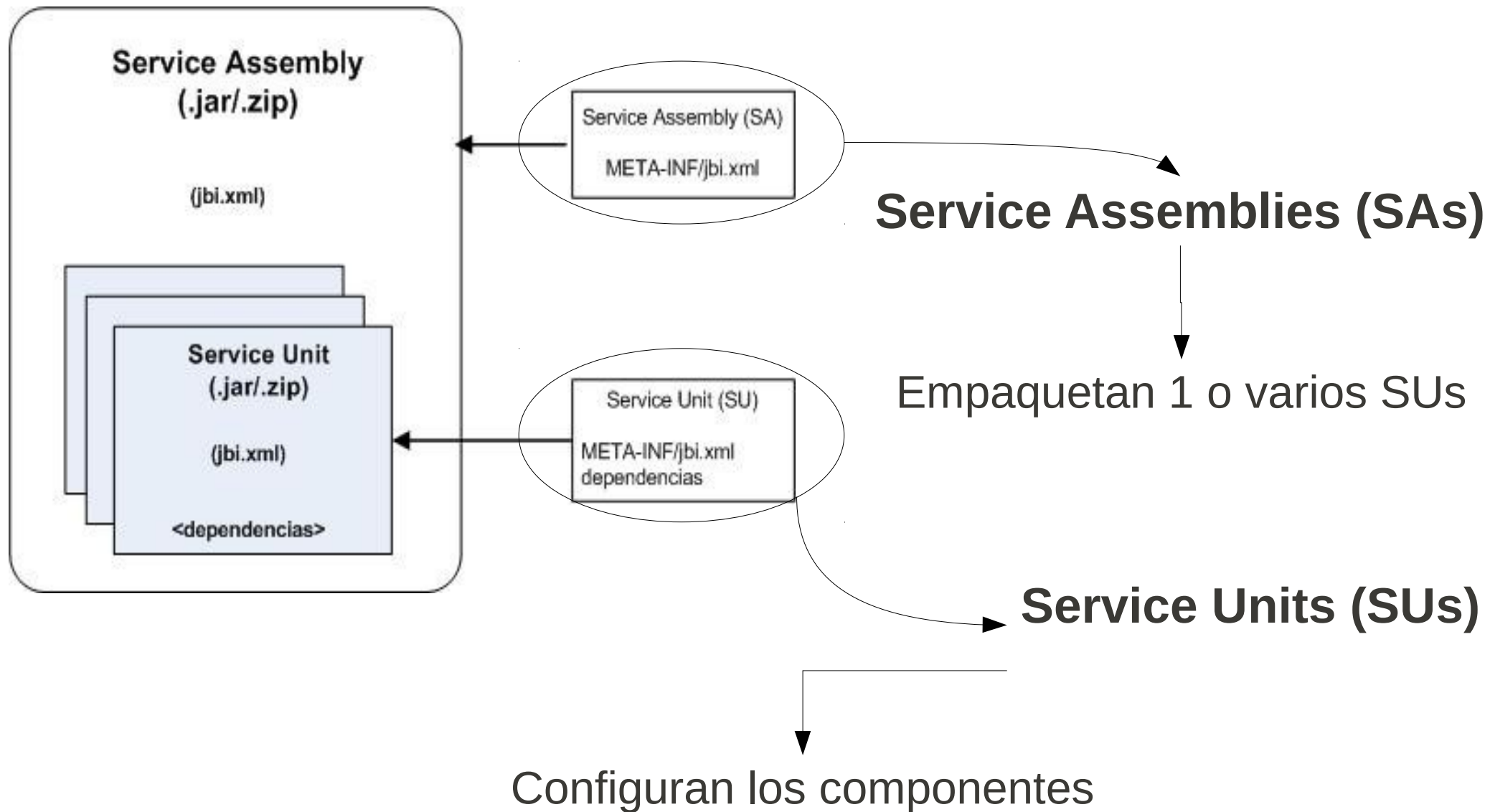
2 tipos de componentes (*plugins*)

**Service Engines (SE):** implementa la lógica de negocios y otros servicios (ruteo, transformación, scheduler)

**Binding Components (BC):** proveen conectividad a otros servicios/aplicaciones ubicadas fuera del ambiente JBI

Cada componente actúa en sí como un **contenedor**

# Conceptos JBI: SUs y SAs



# jESBihca

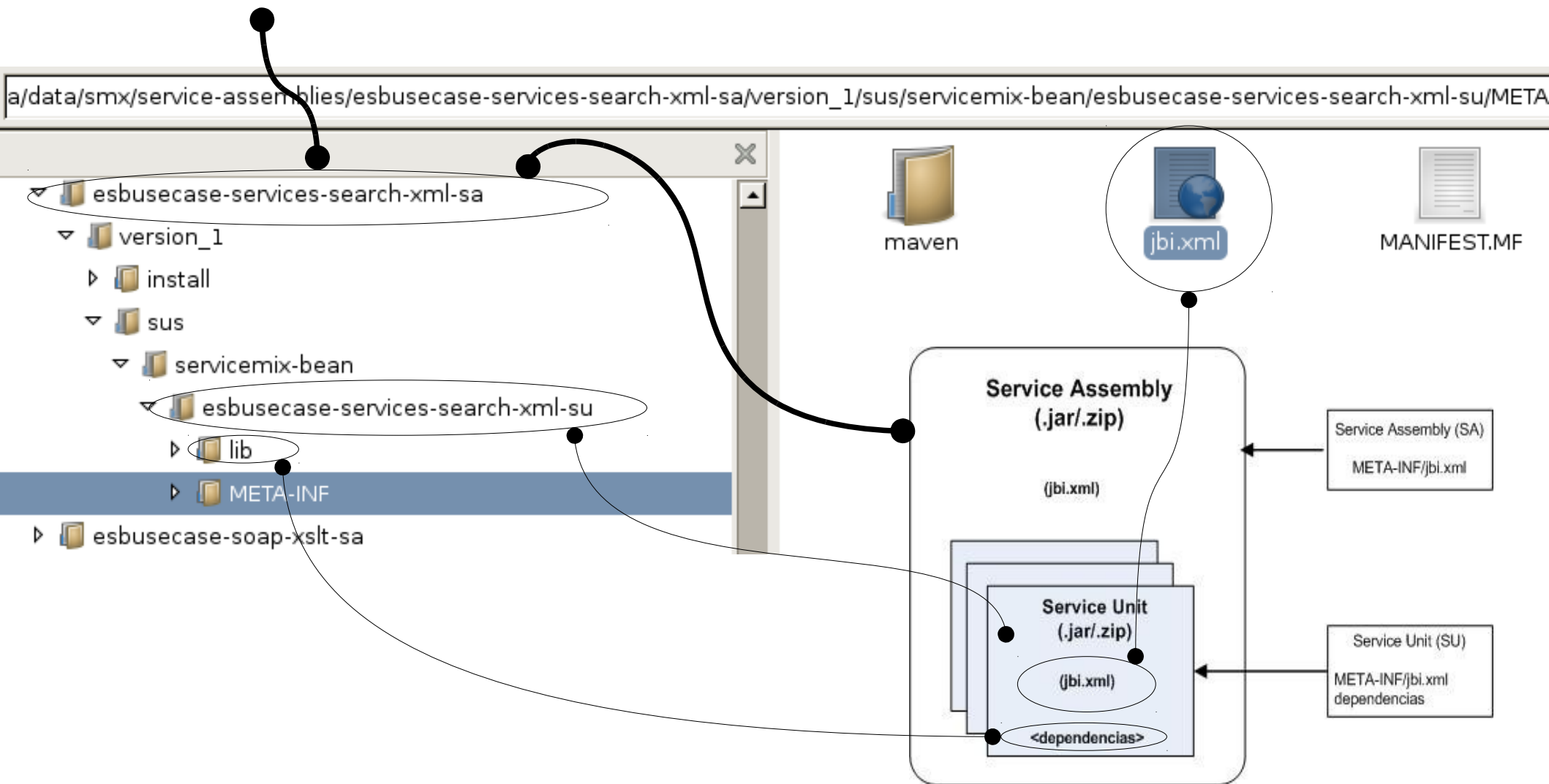


**fs-adapter.ogv**

**svn-mw-adapter.ogv**

# Conceptos JBI: SUs y SAs

Ejemplo en jESBihca de un **Service Assembly** (servicio XML)

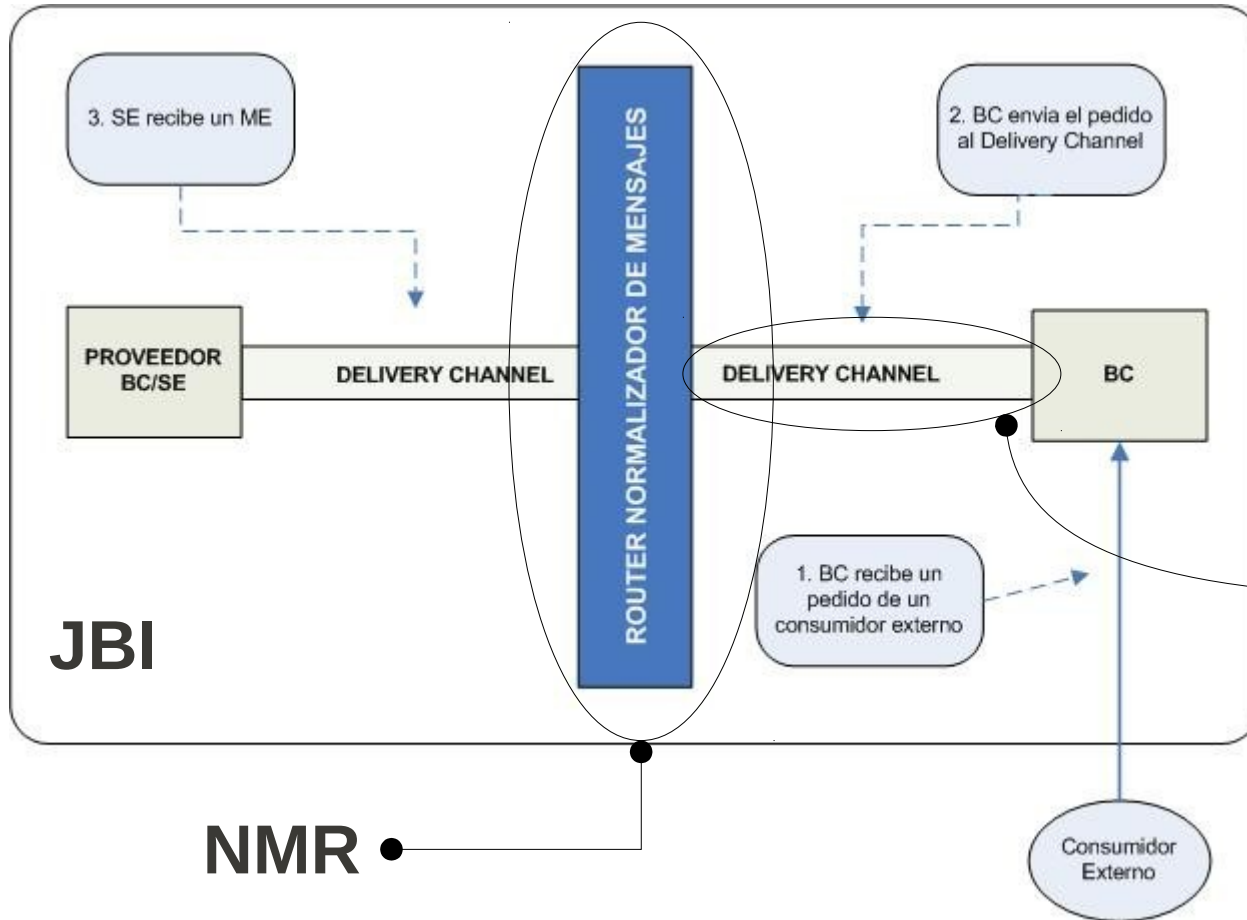


# Conceptos JBI: NMR

## Normalized Message Router (bus de mensajes)

- ▶ Infraestructura para el intercambio de mensajes
  - ▶ Desacopla servicios proveedores y consumidores
  - ▶ Interoperabilidad entre componentes

# Conceptos JBI: NMR



Permite la interacción bidireccional entre el NMR y los componentes

• **Delivery Channel**

En jESBihca

```
public class SearchSimple implements MessageExchangeListener {  
    @Resource  
    private DeliveryChannel channel;  
}
```



# Conceptos JBI: Delivery Channel

## Ejemplo: Servicio XML desplegado en jESBihca

```
searchXML = searchXML.replace("<?xml version=\"1.0\" encoding=\"UTF-8\"?>", "");  
logger.info("DOCUMENTO XML ENTRANTE: " + searchXML);
```

```
//SE PONE EN FUNCIONAMIENTO EL FRAMEWORK DE SERVICIOS XML  
String xmlOut = this.handleRequest(searchXML);
```

```
xmlOut = XMLUtils.stripNonValidXMLCharacters(xmlOut);  
logger.info("MENSAJE ENVIADO AL BUS: " + xmlOut);
```

```
StringSource ss = new StringSource(xmlOut);
```

```
nm.setContent(ss);  
nm.setProperty("xmlresponse", xmlOut);
```

```
//SE AGREGA EL MENSAJE XML DE SALIDA AL INTERCAMBIO IN/OUT  
inout.setOutMessage(nm);
```

```
//SE ENVAIA EL MENSAJE OUT AL DELIVERY CHANNEL  
channel.send(inout);
```

Mensaje Normalizado



Envia el mensaje al NMR



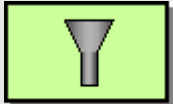
# jESBihca

“no reinventar la rueda”

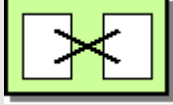
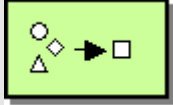
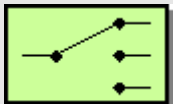
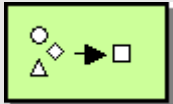
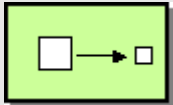


# Visión global del informe

Patrones de Integración



## Integración de Aplicaciones



### Service-oriented Architecture (SOA)

### Enterprise Service Bus (ESB)

### Java Business Integration (JBI)

**jESBihca**

*Lucene*

**Apache Camel**

**Java**

 **The Apache Software Foundation**  
<http://www.apache.org/>

 **HIBERNATE**

**<?xml?>**

*ServiceMix*

 **spring source**

**Free/Libre Open Source Software (FLOSS)**

# jESBihca

## Prototipo de una plataforma de integración

### Principales características ...

... que respeta los principios de la **orientación a servicios**

... basada en un ESB **JB1**-compatible

... implementada utilizando **únicamente** FLOSS

# Caso de estudio: Presentación

## Objetivo

Integrar los resultados de la búsqueda de información...

... dispersa en...

... un conjunto **específico** de Herramientas Colaborativas Asincrónicas

... que **coordinan** el trabajo colaborativo de ...

... un **hipotético** proyecto FLOSS

**jIntegra**

MediaWIKI  
Mantis BT  
SVN  
FileSystem

# Escenarios de búsqueda del Caso de Estudio

Se implementaron 3 escenarios de búsquedas

Basado en **texto simple**

Basado en **documentos XML** con formato acordado

Basado en **mensajes SOAP**

# jESBihca



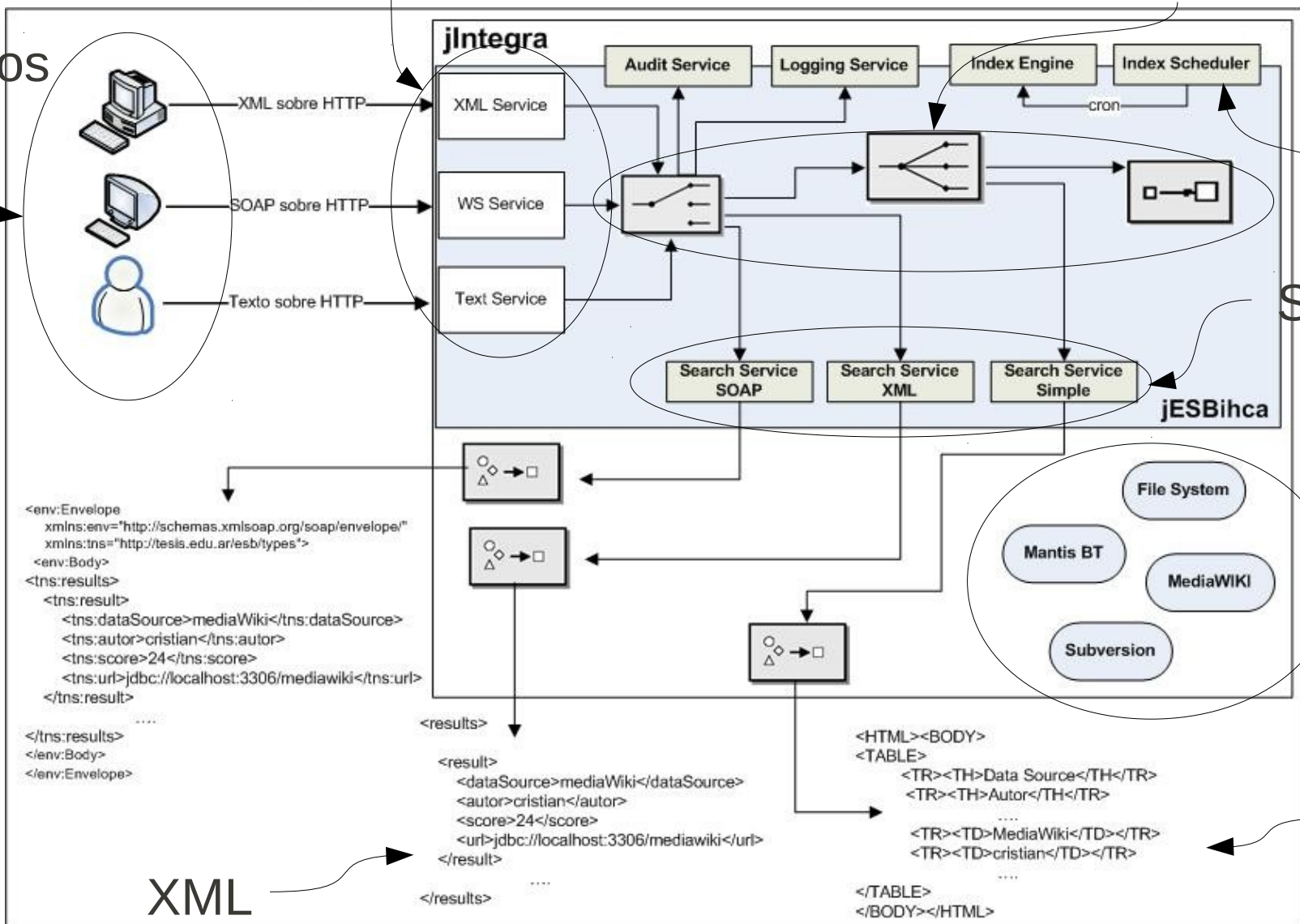
**caso-de-estudio-2.ogv**

# Caso de Estudio: Visión general

Binding Components

Patrones de integración

Escenarios



Servicios

Herramientas

HTML

SOAP

XML

```

<env:Envelope
  xmlns:env="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:tns="http://tesis.edu.ar/esb/types">
  <env:Body>
  <tns:results>
  <tns:result>
  <tns:dataSource>mediaWiki</tns:dataSource>
  <tns:autor>crislian</tns:autor>
  <tns:score>24</tns:score>
  <tns:url>jdbc://localhost:3306/mediawiki</tns:url>
  </tns:result>
  ....
  </tns:results>
  </env:Body>
  </env:Envelope>
  
```

```

<results>
  <result>
  <dataSource>mediaWiki</dataSource>
  <autor>crislian</autor>
  <score>24</score>
  <url>jdbc://localhost:3306/mediawiki</url>
  </result>
  ....
  </results>
  
```

```

<HTML><BODY>
<TABLE>
<TR><TH>Data Source</TH></TR>
<TR><TH>Autor</TH></TR>
  ....
<TR><TD>MediaWiki</TD></TR>
<TR><TD>crislian</TD></TR>
  ....
</TABLE>
</BODY></HTML>
  
```



# jESBihca



**servicio-get-pdf.avi**

... Gracias ...

... cada quien aporta un granito de arena (*jESBihca*), y luego todos disfrutamos de una playa maravillosa (*software libre*) \*

\* adaptado de PabloRizzo.com

