Otras Arquitecturas y metodologías SOA

Pablo García Sánchez
pgarcia@atc.ugr.es
Departamento de Arquitectura y Tecnología de Computadores

Curso Web 2.0 Arquitectura Orientada a Servicios en Java
Escuela de Posgrado
Febrero/Marzo de 2010

¿Qué vamos a aprender?

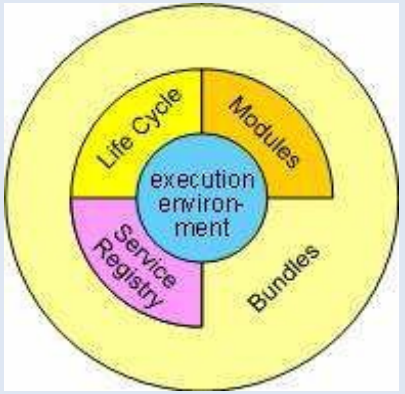
- Otras SOA en Java:
  - OSGI
  - ebXML
- Metodologías
  - RosettaNet
  - BCM
  - UMM
- Proyectos de la vida real

Introducción

- SOA no es sólo Web Services, SOAP y BPEL!
- Es cualquier sistema software que permita descubrir e invocar servicios (remotos o locales)
- Pueden basarse en una especificación pública o no

OSGi

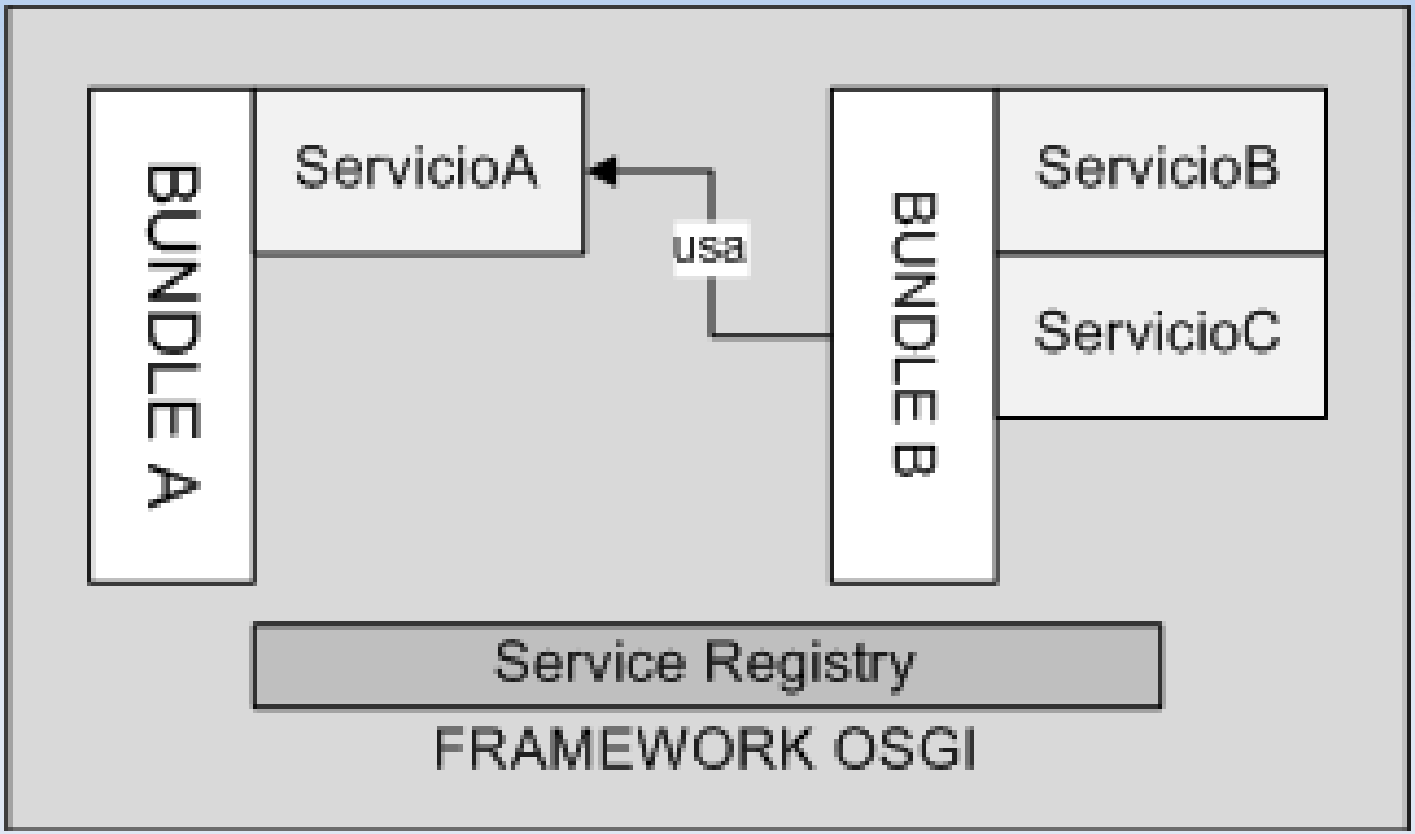
- **OSGi** (*Open Service Gateway Initiative*) define una arquitectura SOA dentro de una máquina virtual de Java para integración de sistemas heterogéneos. Además, proporciona características muy deseables:
  - Abstracción de paquetes
  - Gestión del ciclo de vida
  - Empaquetamiento
  - Versionado



OSGi (II)

- **Bundle**: Jar con el fichero MANIFEST adaptado
- **Servicio**: conecta bundles de manera dinámica
- **Componente**: Clase dentro de un bundle junto con una descripción en XML interpretada en tiempo de ejecución->Servicios Declarativos

OSGi (III)


BUNDLE A
ServicioA
usa
BUNDLE B
ServicioB
ServicioC
Service Registry
FRAMEWORK OSGI

OSGi (III)

Manifest-Version: 1.0
Bundle-ManifestVersion: 2
Bundle-Name: Tabusearch Plug-in
Bundle-SymbolicName: tabusearch
Bundle-Version: 1.0.0
Bundle-RequiredExecutionEnvironment: J2SE-1.5
Import-Package: ch.ethz.iks.r-osgi,
es.ugr.osgiliath.algorithms,
es.ugr.osgiliath.algorithms.types,
es.ugr.osgiliath.network,
es.ugr.osgiliath.problem,
es.ugr.osgiliath.utils,
org.osgi.framework
Export-Package: es.ugr.osgiliath.tabusearch,
es.ugr.osgiliath.tabusearch.distributed
Service-Component: OSGI-INF/distributedTS.xml

Ejemplo de MANIFEST.MF

OSGi (IV)

Ejemplo de Descriptor de Servicio

```
<?xml version="1.0"?>
<component name="tabusearch">
 <implementation
  class="es.ugr.osgiliath.tabusearch.TabuSearch" />
 <service>
  <provide
   interface="es.ugr.osgiliath.algorithms.Algorithm" />
 </service>
 <reference name="OBJECTIVEFUNCTION"
  interface="es.ugr.osgiliath.tabusearch.ObjectiveFunction"
  bind="setObjectiveFunction"
  unbind="unsetObjectiveFunction"
  cardinality="1..1" />
 <reference name="PARAMETERS"
  interface="es.ugr.osgiliath.algorithms.Parameters"
  bind="setParameters"
  unbind="unsetParameters"
  cardinality="1..1" />
 <reference name="TABULIST"
  interface="es.ugr.osgiliath.tabusearch.TabuList"
  bind="setTabuList"
  unbind="unsetTabuList"
  cardinality="1..1" />
 <!-- Resto de referencias ...-->
</component>
```

R-OSGi

Problema de OSGi: imposibilidad de invocar servicios remotos

- Uso de conectores basados en UPnP y Jini: INVASIVOS

R-OSGi es un midleware para distribuir de forma transparente servicios en red

Crea proxies que actúan como bundles normales que se encargan de comunicar con otros bundles distribuidos

ebXML

- Es una arquitectura completa para crear un mercado electrónico global donde las empresas puedan:
  - Encontrarse unas a otras
  - Realizar negocios mediante el intercambio de mensajes de negocio basados en XML

ebXML (II)

- SOAP, WSDL, UDDI por sí solos no son adecuados
  - WSDL no puede hacer frente a colaboración de negocio
  - SOAP (en su forma básica) no proporciona mensajería segura y confiable
  - UDDI no proporciona capacidad de repositorio para objetos de negocio

WS Vs. B2B

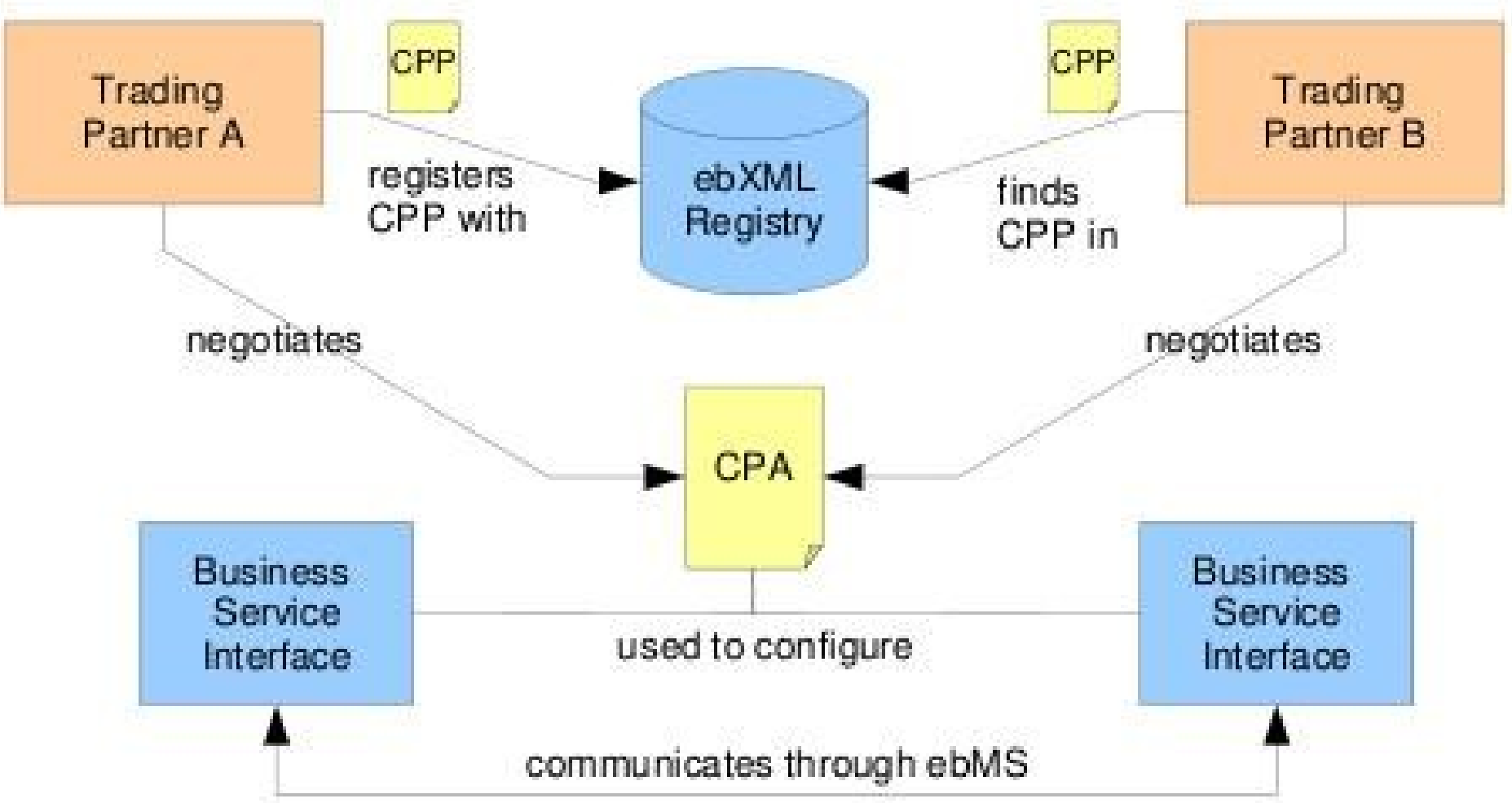
Web Services	B2B
Interacción simple	Interacción compleja
Orientados al consumidor	Orientados al negocio
Procesos “ligeros”	Procesos “pesados”
Sin colaboración de negocio	Con colaboración de negocio
Sin perfil de socio	Soporta perfil de socio
No seguro, no confiable	Seguro, confiable
No soporta no-repudiación	Soporta no-repudiación
No soporta repositorio	Registro y Repositorio

Modulos de la Arquitectura de ebXML

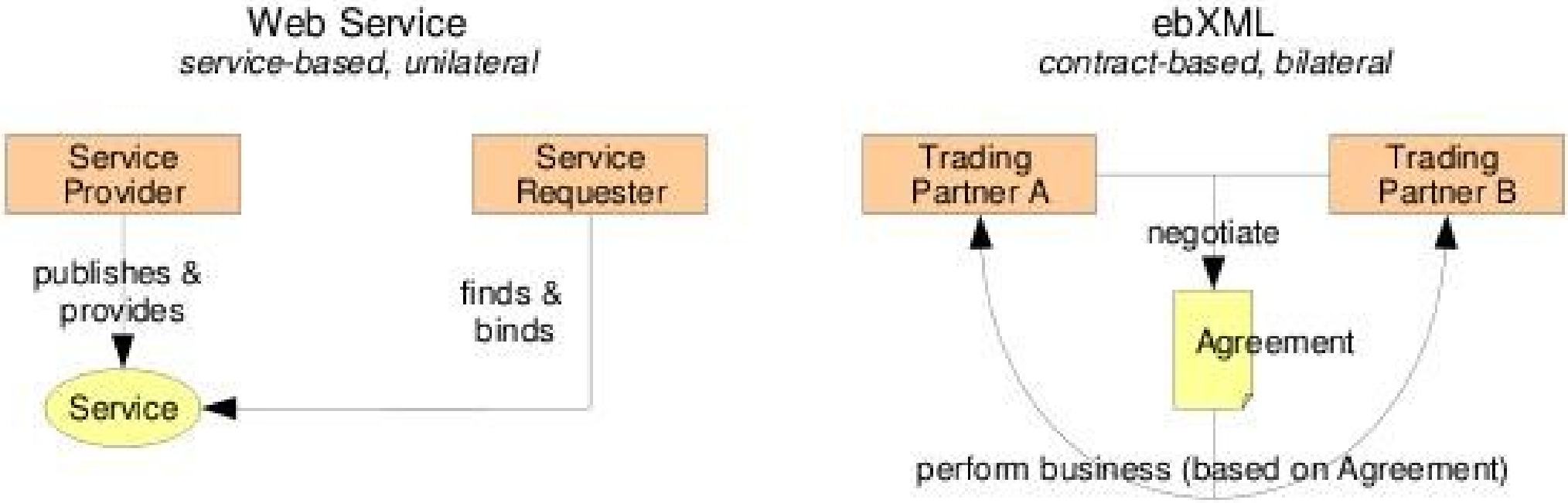
- Business Process Specification
- Partner Profile and Agreements **(ISO 15000-1:2004)**
- Registro y Repositorio **(ISO 15000-3:2004 e ISO 15000-2:2004)**
- Messaging Service **(ISO 15000-4:2004)**
- Core Components **(ISO 15000-5:2005)**

- ¡Pueden ser usados por separado!

Descripción general de un sistema ebXML



Aproximación basada en Servicios Vs basada en contrato



Modelado Top-down vs. Bottom-up

- ebXML
  - Utilizar metodologías para ebXML (como UMM) implica desarrollo Top-Down (el analista empieza desde arriba)
  - Pero se pueden utilizar servicios ya existentes
- WS
  - El desarrollo de WS sigue una filosofía Bottom-Up (se empieza con servicios pequeños y se agrupan)

Conclusiones

- ebXML y WS tienen sus ventajas e inconvenientes
  - ebXML es una solución todo en uno basada en estándares y lista para B2B, pero con poco soporte en la industria TI
  - WS se basa en especificaciones desarrolladas independientemente, con algunas lagunas, pero ampliamente aceptada

Metodologías para SOA

- RosettaNet
- UMM
- SOMA

RosettaNet

- *RosettaNet Implementation Framework*, RNIF
- Framework que define parte de la interacción:
  - Estructura de mensajes de negocio genérica
  - Pasos requeridos para transmitir el mensaje entre socios comerciales
  - Empaquetamiento y desempaquetamiento
  - Protocolos de transmisión
  - Manejo de errores
  - Validación de ciertas partes del contenido

RosettaNet (II)

Partner Interface Processes (PIPs):

Son documentos XML que especifican interacciones entre dos participantes para alcanzar una meta de negocio, como procesar una orden de compra o preguntar un precio

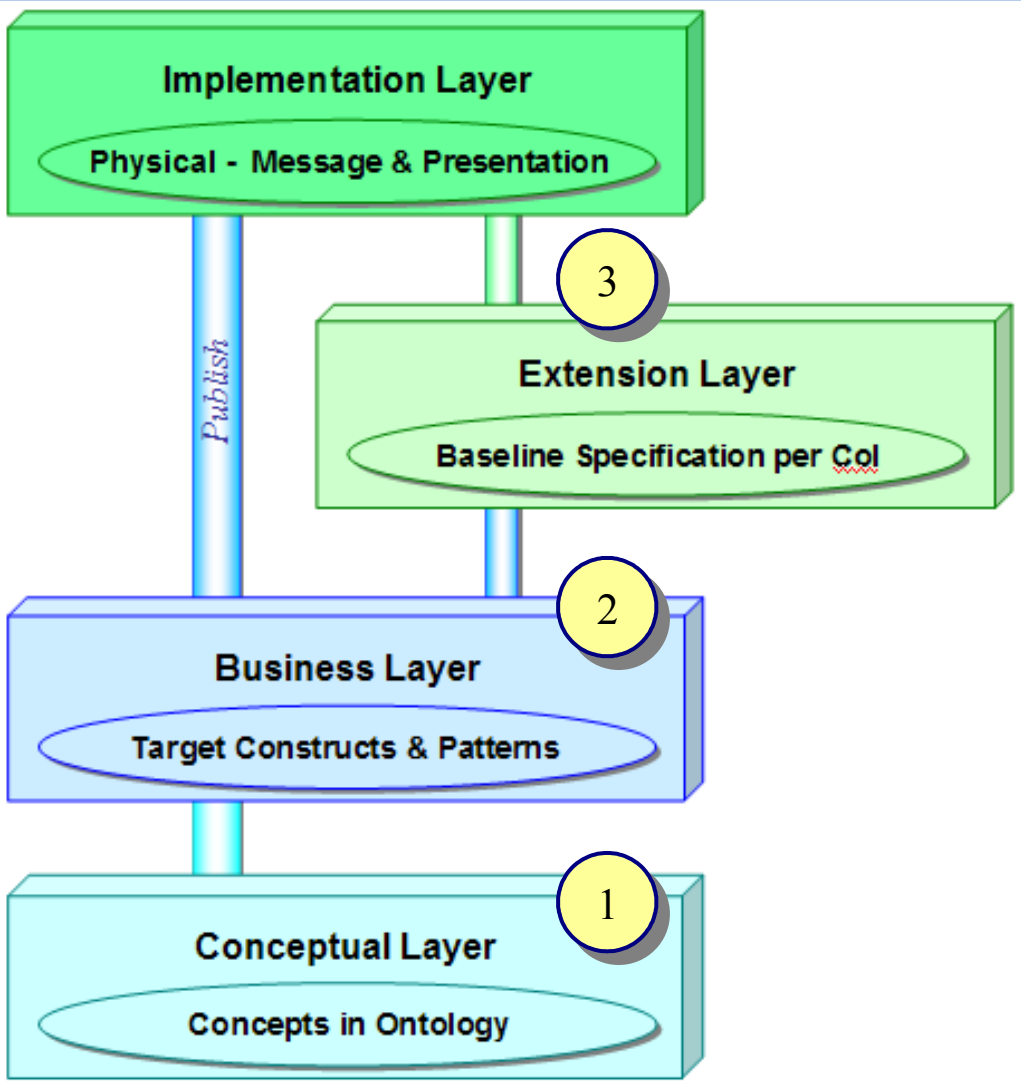
RosettaNet (y III)

- Productos software que utilizan RosettaNet:
- **Oracle Integration B2B:** La implementación de RosettaNet forma parte de este producto, Sin embargo delega la mayor parte de su funcionalidad en otros productos de Oracle (p.e. Oracle BPEL Process Manager u Oracle Business Activity Monitoring).
- **WebSphere Partner Gateway** (IBM)
- **RosettaNet Accelerator**: Microsoft propone un paquete enfocado a RosettaNet dentro de su servidor **BizTalk**
- **WebMethods** (Trading Networks)

Business Centric Methodology

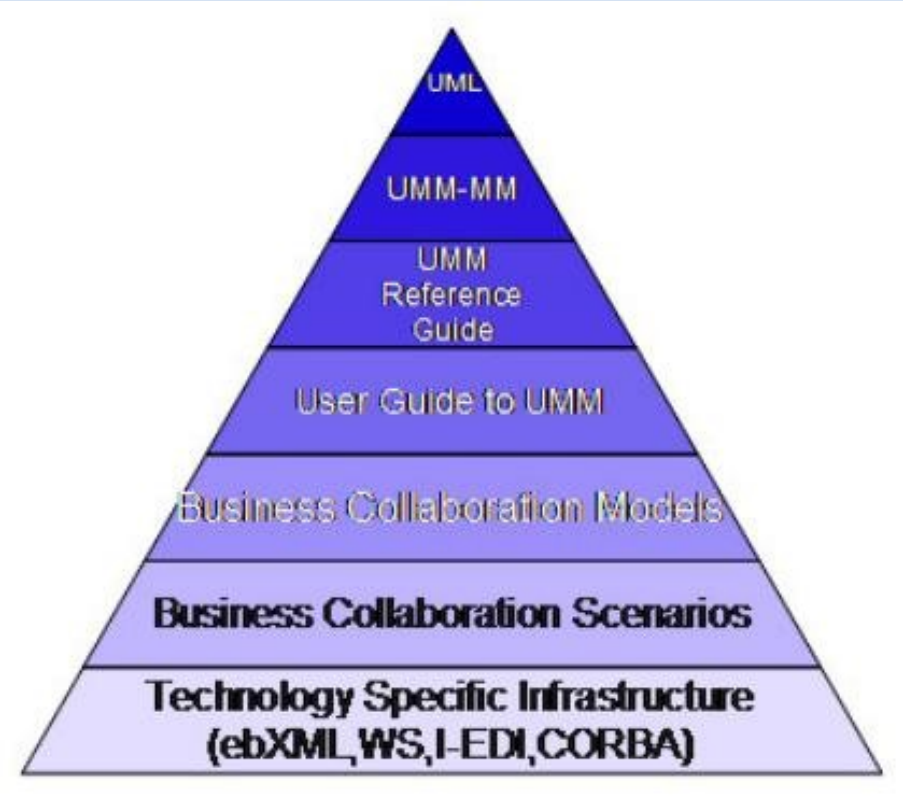
- Desarrollada por OASIS
- Dividida en tres partes
  - **Capas BCM**: Es un modelo de capas que soporta las Plantillas BCM y otros modelos opcionales para cualificar los aspectos de la solución en el que cada capa representa una segmentación definida del problema.
  - **Pirámide de información BCM**: Es una representación semántica de toda la información existente en el proyecto, cuya clasificación y detalles son desarrollados por los analistas de negocio.
  - **BCM Operacional**: Se asegura de que la tecnología del software de implementación trate directamente con esos mecanismos semánticos a través de una arquitectura consistente orientada al contexto.

BCM (y II)


Implementation Layer
Physical - Message & Presentation
Publish
3
Extension Layer
Baseline Specification per CoI
2
Business Layer
Target Constructs & Patterns
1
Conceptual Layer
Concepts in Ontology

UMM

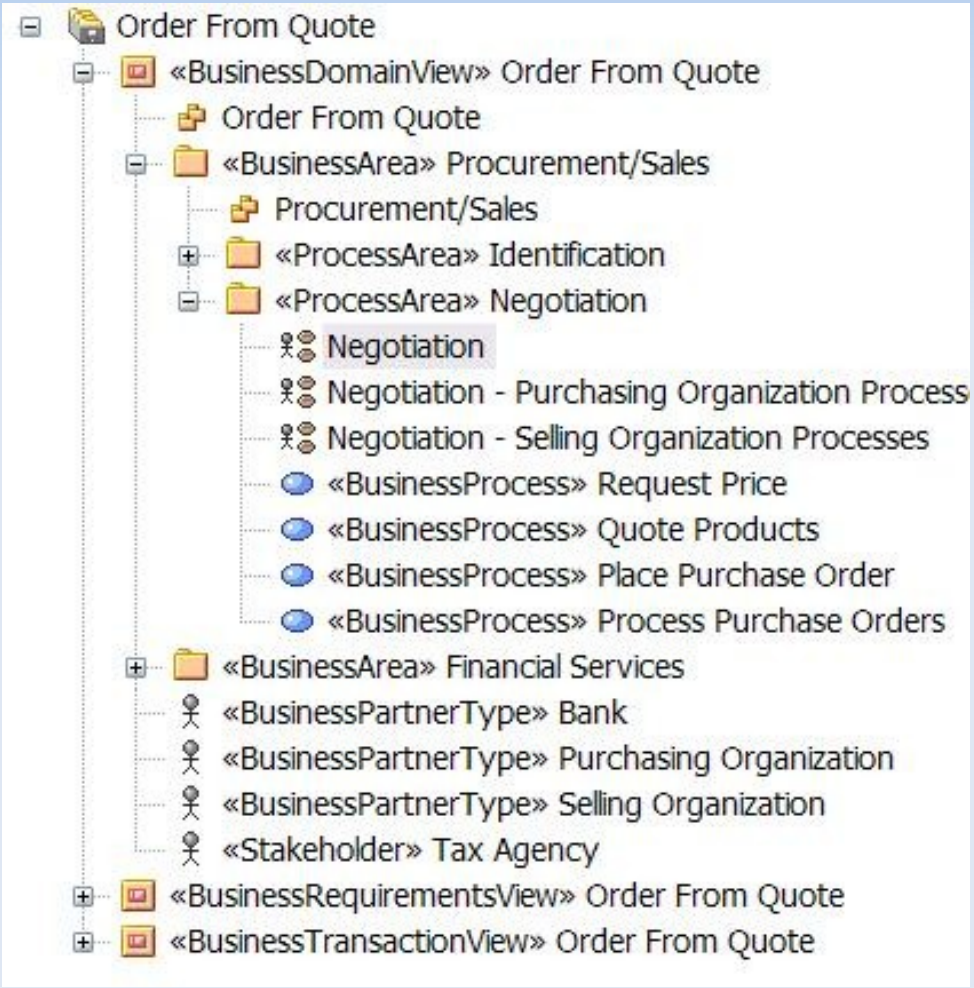
- Desarrollada por la UN/CEFACT
- Perspectiva Top-Down



UMM (II)

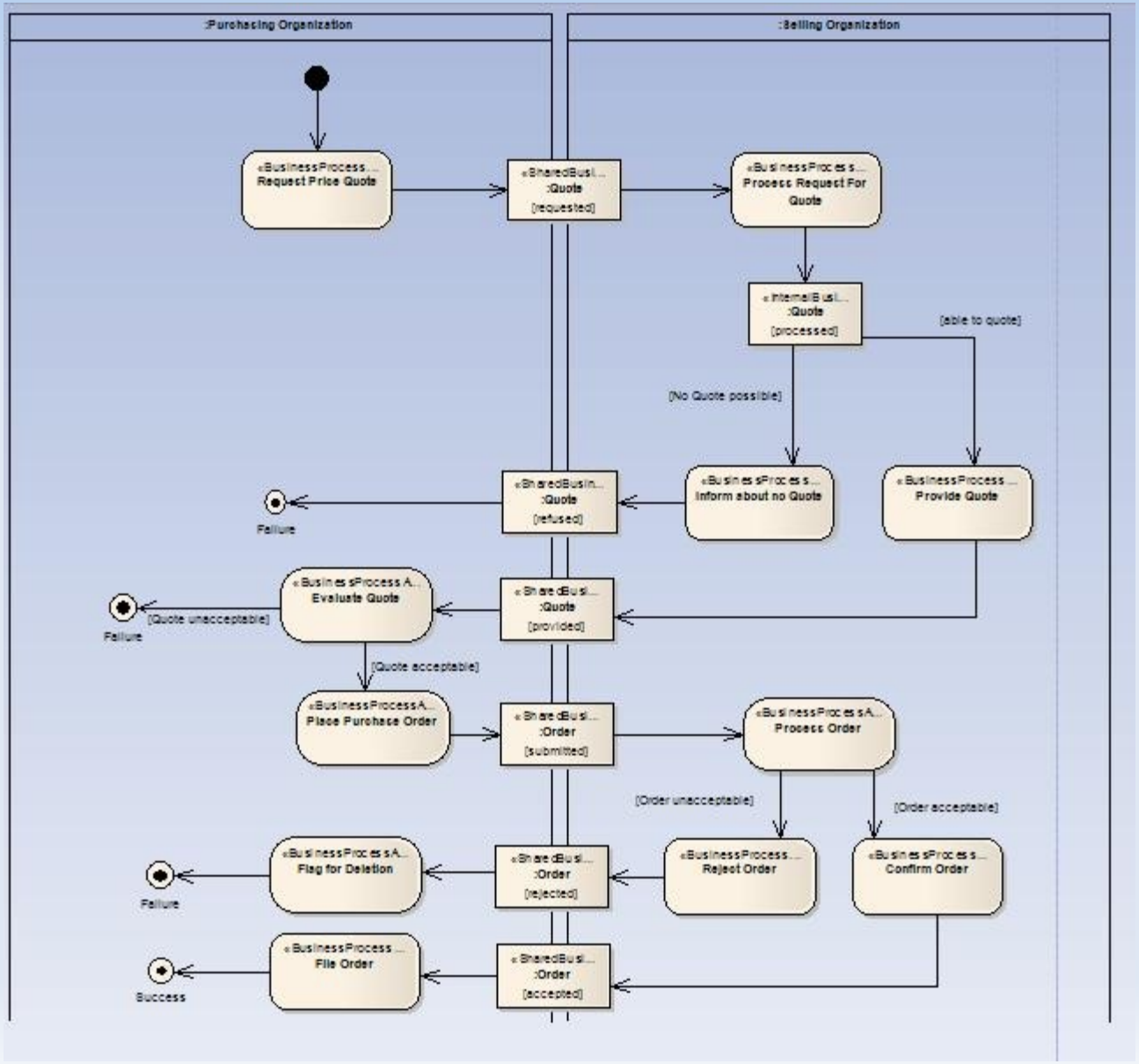
- **Vista de dominio de negocio**
  - Recolectar conocimiento de las partes interesadas
- **Vista de requisitos de negocio**
  - Describir procesos, entidades, transacciones, colaboraciones y realizaciones
- **Vista de transacciones de negocio**
  - Definir la coreografía global de intercambios de información
- **Vista de servicios de negocio**
  - Especificar los servicios compuestos, agentes e intercambio de mensajes expresada en conceptos técnicos de los desarrolladores software

UMM (III)

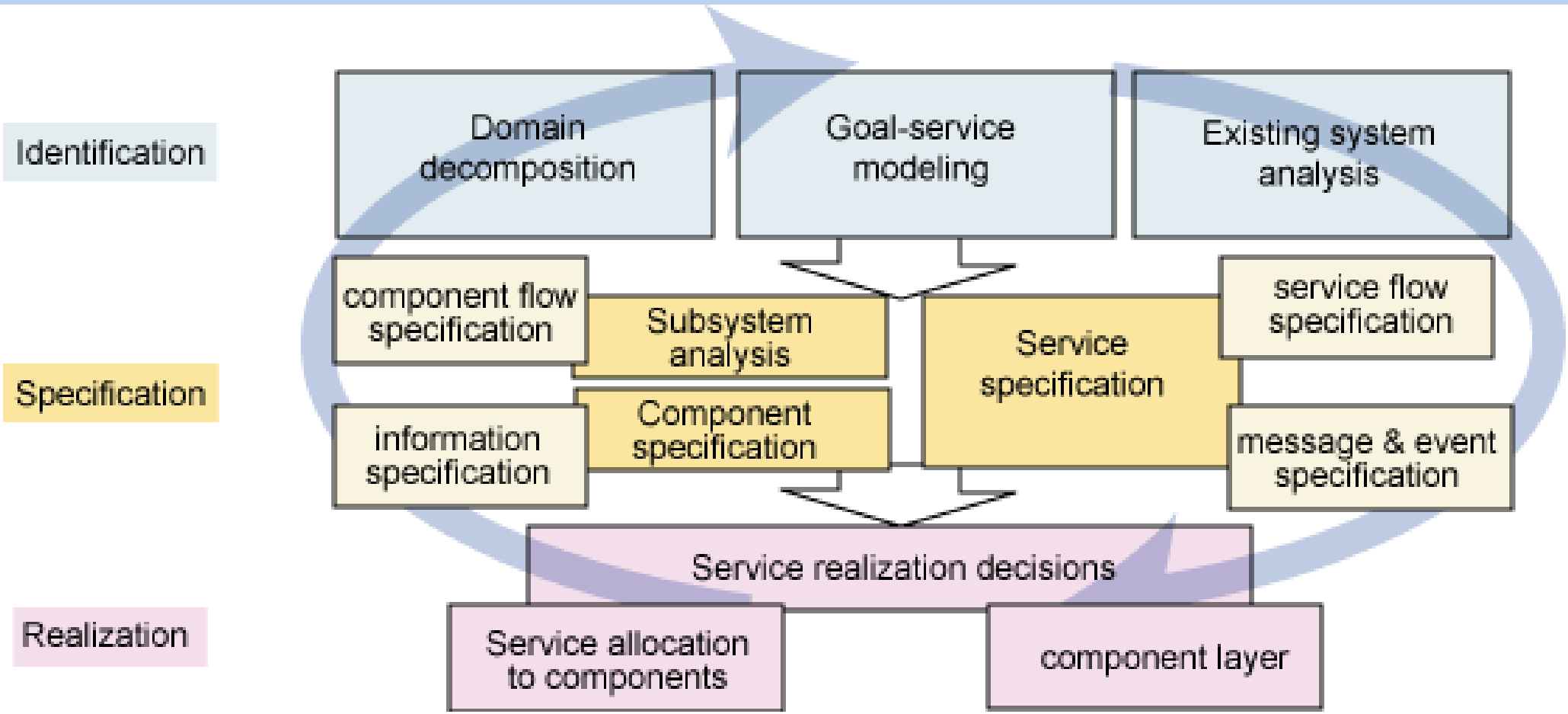

Order From Quote
«BusinessDomainView» Order From Quote
Order From Quote
«BusinessArea» Procurement/Sales
Procurement/Sales
«ProcessArea» Identification
«ProcessArea» Negotiation
Negotiation
Negotiation - Purchasing Organization Process
Negotiation - Selling Organization Processes
«BusinessProcess» Request Price
«BusinessProcess» Quote Products
«BusinessProcess» Place Purchase Order
«BusinessProcess» Process Purchase Orders
«BusinessArea» Financial Services
«BusinessPartnerType» Bank
«BusinessPartnerType» Purchasing Organization
«BusinessPartnerType» Selling Organization
«Stakeholder» Tax Agency
«BusinessRequirementsView» Order From Quote
«BusinessTransactionView» Order From Quote

UMM (IV)

- Ejemplo de sub-vista de la vista de requisitos



SOMA de IBM


Identification
Specification
Realization
Domain decomposition
Goal-service modeling
Existing system analysis
component flow specification
Subsystem analysis
Service specification
service flow specification
information specification
Component specification
message & event specification
Service realization decisions
Service allocation to components
component layer

Proyectos de la Vida Real

- eIntegr@
- GAD
- AmIVital
- OSGiLiath

eIntegr@

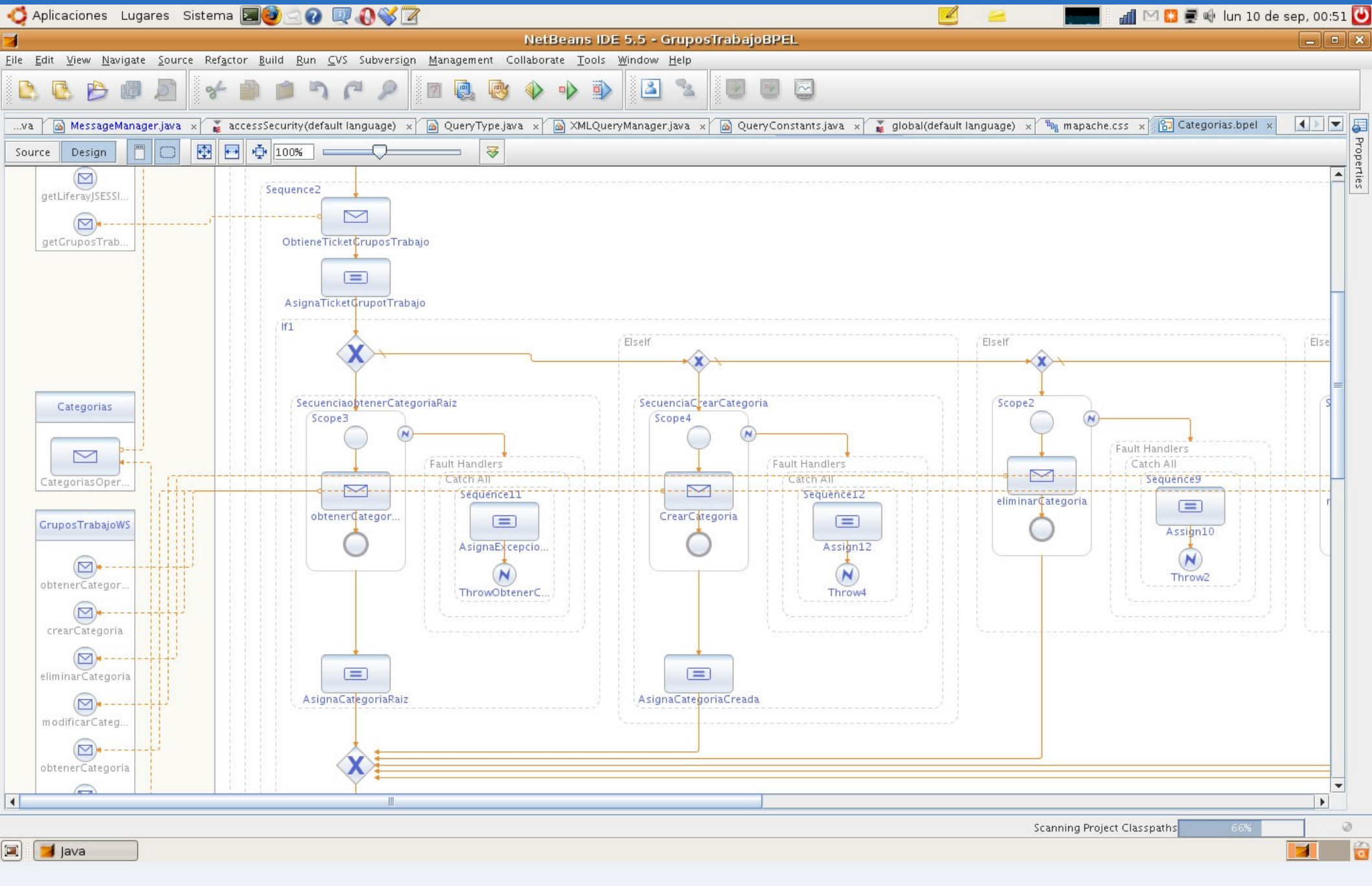
- Colaboración entre la UGR y la empresa Intecna Soluciones
- Single-Sign-On: Liferay, Alfresco
  - LDAP
  - CAS
  - Operaciones:
    - getTicket(user, password)
    - validateTicket(ticket)

eIntegr@ (II)

Alfresco (yeja con los pajaros!)

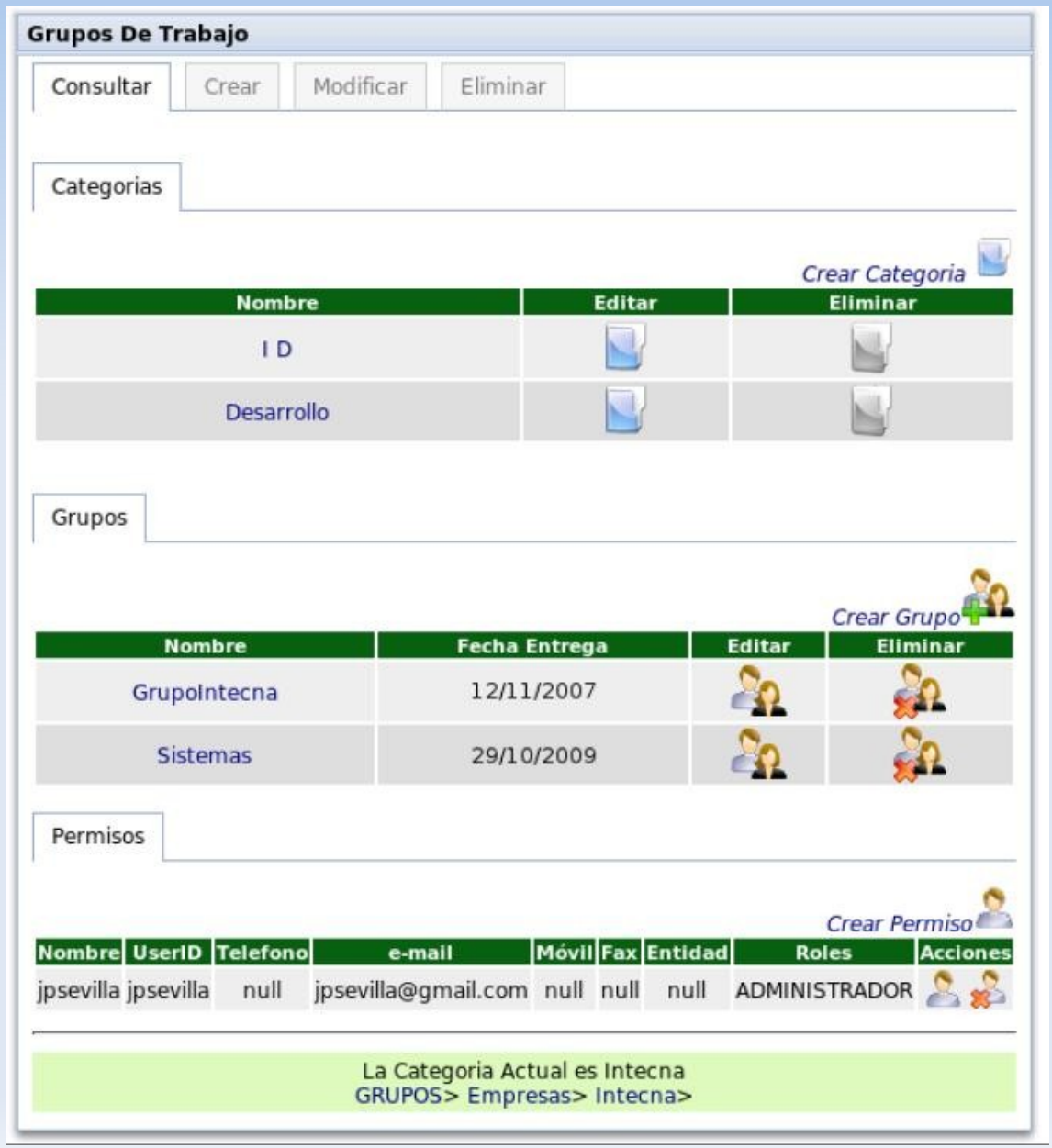
- ¿Qué es?
- ¿En qué se basa?
  - Alfresco
  - Operaciones
- ¿Cómo podemos utilizarlo?
- ¿Qué servicios hay que crear?
  - Gestión de usuarios
  - Gestión de documentos
  - Gestión de grupos
  - Gestión de categorías

eIntegr@ (III)


Aplicaciones Lugares Sistema
lun 10 de sep, 00:51
NetBeans IDE 5.5 - GruposTrabajoBPEL
File Edit View Navigate Source Refactor Build Run CVS Subversion Management Collaborate Tools Window Help
MessageManager.java
accessSecurity(default language)
QueryType.java
XMLQueryManager.java
QueryConstants.java
global(default language)
mapache.css
Categorias.bpel
Source Design
100%
Properties
Sequence2
ObtieneTicketGruposTrabajo
AsignaTicketGrupotTrabajo
If1
ElseIf
ElseIf
Else
SecuenciaobtenerCategoriaRaiz
SecuenciaCrearCategoria
Scope2
Scope3
Scope4
Fault Handlers
Catch All
Sequence11
Sequence12
Sequence9
AsignaExcepcio...
ThrowObtenerC...
Assign12
Throw4
Assign10
Throw2
obtenerCategor...
CrearCategoria
eliminarCategoria
AsignaCategoriaRaiz
AsignaCategoriaCreada
getLiferayJSESSI...
getGruposTrab...
Categorias
CategoriasOper...
GruposTrabajoWS
obtenerCategor...
crearCategoria
eliminarCategoria
modificarCateg...
obtenerCategoria
Scanning Project Classpaths
66%
Java

eIntegr@ (IV)

- Interfaz en Liferay



GAD

- Gestor de activos digitales diseñado para la Junta de Andalucía por la *Fundación I+D del Software Libre*, que permite almacenar,categorizar, agrupar y publicar vídeo de fuentes heterogéneas
- Basado en Python+Django
- También usa Alfresco
- Toda operación para trabajar con su modelo de datos sería un servicio web publico (interfaces=1)
- BPEL para publicación de usuarios identificados.

AmIVital

- Entorno de desarrollo para la salud y el bienestar
- Colaboran un montón de empresas y OPIs (Ericsson, Siemens, Telefónica, UGR, UMA...)
- Crear una plataforma de desarrollo de servicios para salud y teleasistencia
  - Servicios funcionales
  - Servicios tecnológicos

AmIVital (II)

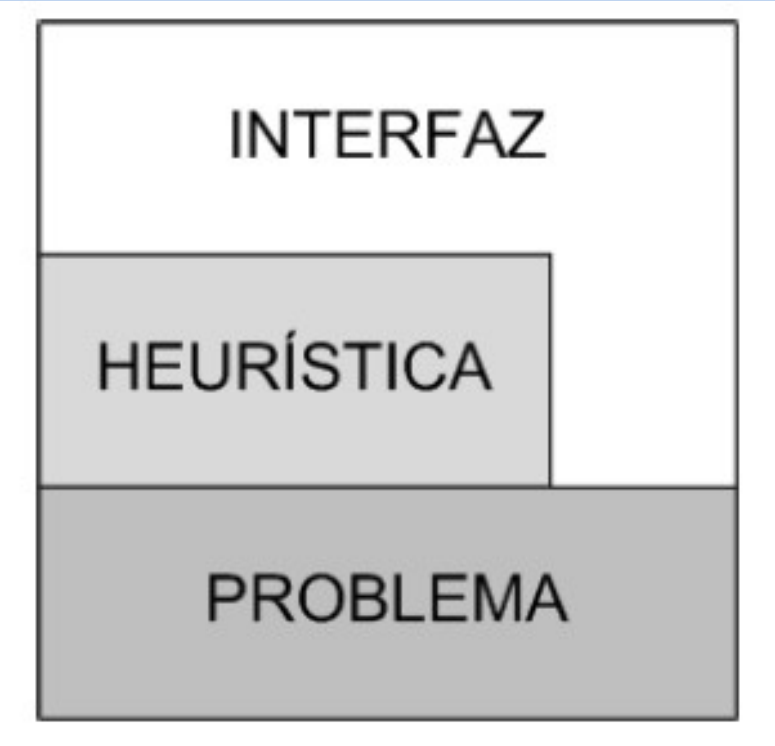
- Pasarela residencial móvil (UGR-TID-ÍTACA)
- Utiliza OSGi y servicios declarativos
- Utiliza Axis para desplegar Web Services
- Llama a servicios web de otras empresas:
  - Calendario Virtual
  - Envío de Monitorización
  - Alarmas
  - Gestión de contenidos

OSGiLiath

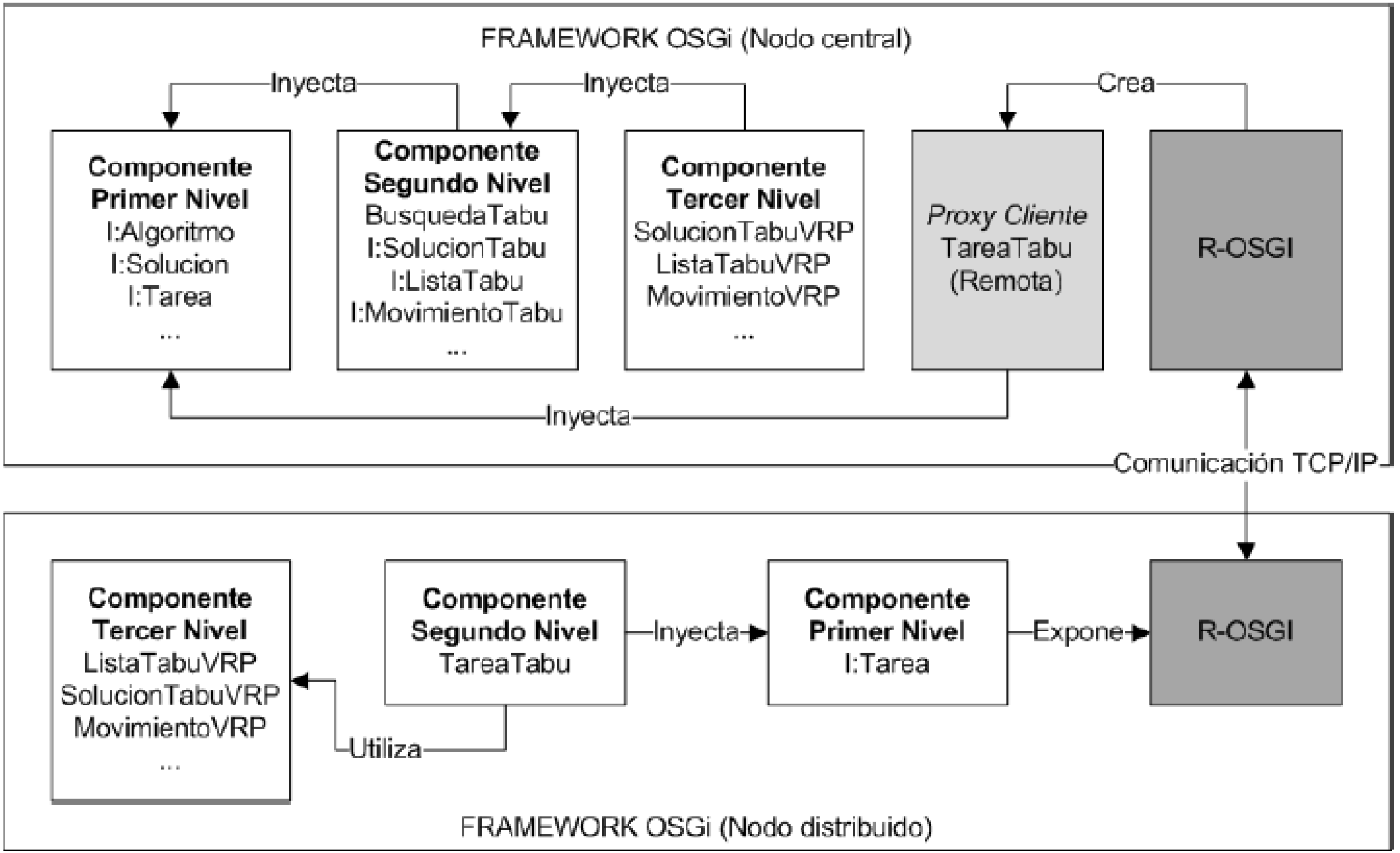
- Acrónimo un poco friki de "*OSGi Laboratory for Implementation and Testing of Heuristics*"
- Framework de desarrollo de heurísticas no centrado en ningún paradigma concreto y basado en plug-ins
- Usa:
  - Interfaz sencilla
  - Programación orientada a componentes
  - Activación dinámica de componentes
  - Servicios declarativos

OSGiLiath (II)

- **Interfaz:**
  - Interfaces para:
    - Algoritmo
    - Algoritmo distribuido
    - Solución
    - Problema
    - Datos de entrada
    - Parámetros
- **Heurística:**
  - Implementación de la heurística
- **Problema:**
  - Implementación del problema



OSGiLiath (III)



Conclusiones

- *Palabros* que recordar de esta presentación:
  - OSGi
  - ebXML (de pasada)
- Metodologías
  - SOMA o UMM
- Hemos visto que SOA se usan en el MundoReal (tm)!

Trabajo futuro en SOA

- Composición automática de servicios
- Enrutado ”inteligente”
- Extended SOA (xSOA)
  - Monitorización
  - QoS
  - ”Agreement”
  - Agregación compleja

Referencias

- Papazoglou, M.P. et al.: *Service Oriented Architectures: approaches, technologies and research issues*, VLDB Journal, 16, pp. 389-415
- García-Sánchez P. et al. *Plataforma de integración de servicios para la administración basada en BPEL y SOA.* Actas de las “III Jornadas en Servicios Web y SOA (JSWEB), 2007
- García-Sánchez, P. et al. *Entorno de desarrollo de heurísticas distribuidas utilizando OSGi*. Actas de las ”XX Jornadas de Paralelismo”, 2009.
- García-Sánchez, P. et al. *Using UN/CEFACT’S Modelling Methodology (UMM) in e-health projects*. Actas de International Work Conference on Artificial Neural Networks and Ambient Assisted Living (IWAAN'09), Salamanca, 2009.
- http://www.fidesol.org
- http://www.intecna.es
- http://www.amivital.es

No tiene que ver con SOA, pero...