PROYECTO FIN DE CARRERA

DESARROLLO DE UN PROTOTIPO SOFTWARE PARA LA EJECUCIÓN DE SERVICIOS GRID SEMÁNTICOS BASADO EN TECNOLOGÍA DE AGENTES

ANEXO 2:

MODELADO Y CASOS DE USO

Índice

1. INTRODUCCIÓN **6**

2. FASES DE ANÁLISIS Y DISEÑO **8**

2.1. Casos de uso 9

2.2. Modelo de organización 10

2.3. Modelo de agentes 11

2.4. Modelo de objetivos y tareas 12

2.5. Modelo de interacción 13

2.6. Modelo de entorno 15

Índice de figuras

1. CDU de nivel 0 9
2. CDU de nivel 1 9
3. Modelo de organización 10
4. Modelo facilitador de organización 11
5. Modelo de Agentes 12
6. Modelo de Objetivos 13
7. Modelo de Interacción general 14
8. Modelo de Interacción de *Nuevo Servicio* 14
9. Modelo de Interacción 15
10. Modelo de Entorno 16

Índice de tablas

1. Asociación entre INGENIAS y RUP 6
2. Fases de la metodología INGENIAS 8

1. INTRODUCCIÓN

La metodología elegida para el análisis y diseño del sistema basado en agentes incluido en este proyecto, ha sido INGENIAS. INGENIAS concibe el SMA como [1] la representación computacional de un conjunto de modelos. Cada uno de estos modelos muestra una visión parcial del SMA: los agentes que lo componen, las interacciones que existen entre ellos, cómo se organizan para proporcionar la funcionalidad del sistema, qué información es relevante en el dominio y cómo es el entorno en el que se ubica el sistema a desarrollar.

INGENIAS proporciona herramientas gráficas para generar especificaciones, validar los diseños, generar código de forma automática y generar documentos, entre otras cosas.

Una de las ventajas de INGENIAS es que se integra fácilmente con RUP (Rational Unified Process). En la Tabla 1 podemos ver las asociaciones entre los elementos del RUP y los elementos de los meta-modelos.

Tabla 1: Asociación entre INGENIAS y RUP

Entidad MAS	**Entidad RUP**
Agente	Clase
Organización	Arquitectura
Grupo	Subsistema
Interacción	Escenario
Roles, tareas y flujos de trabajo	Funcionalidad

Con los meta-modelos, es sencillo representar todos los aspectos de un sistema de agentes. Cada uno de estos meta-modelos se centran en aspectos concretos del sistema:

- **Meta-modelo de organización:** es el equivalente a la arquitectura del sistema. Describe como se agrupan y coordinan los distintos componentes del sistema, identifica las tareas relevantes para la organización, así como sus objetivos globales, y define restricciones en las interacciones entre los agentes.

- **Meta-modelo de agente:** describe agentes particulares, sus tareas, objetivos, los roles que desempeñan y los estados mentales en que se encontrarán a lo largo de su vida.

- **Meta-modelo de objetivos y tareas:** tiene como propósito recoger las motivaciones del sistema de agentes, definir las relaciones entre objetivos y tareas, indicar las entradas y salidas de cada tarea, y describir cuales son sus efectos, tanto en el entorno como en el estado mental de sus agentes responsables.

- **Meta-modelo de interacción:** describe como se coordinan y comunican los agentes

del sistema. Esto incluye a los actores implicados, las unidades de interacción, los protocolos utilizados y cómo afecta la interacción en el contexto.

- **Meta-modelo de entorno:** En este meta-modelo el propósito no es generar representaciones del mundo en el que se ubica el sistema, sino, desde un punto de vista más pragmático, categorizar el tipo de entidades relevantes en el entorno y restringir la interacción con ellas. Así, el entorno contendrá sólo recursos, aplicaciones y agentes, y se limitará la percepción y actuación de los agentes.

2. FASES DE ANÁLISIS Y DISEÑO

INGENIAS propone una metodología que divide el esfuerzo del análisis y diseño del sistema de agentes en tres fases: inicio, elaboración y construcción. Y dentro de cada una de ellas se desarrollan las iteraciones completas de desarrollo que construyen gradualmente el sistema: análisis, diseño, implementación y pruebas. La Tabla 2 muestra el conjunto de actividades que se llevan a cabo en cada una de las fases [1]. La principal diferencia, entre las fases, que se puede apreciar es el nivel de detalle alcanzado al final de las iteraciones que se realizan en cada una de dichas fases.

Tabla 2: Fases de la metodología INGENIAS [1]

		FASES		
		Inicio	**Elaboración**	**Construcción**
FLUJOS DE TRABAJO FUNDAMENTALES	**Análisis**	- Generar casos de uso e identificar realizaciones de los casos de uso con modelos de interacciones. - Esbozar la arquitectura con un modelo de organización. - Generar modelos del entorno para trasladar la captura de requisitos a los modelos.	- Refinar casos de uso. - Generar modelos de agente para detallar los elementos de la arquitectura. - Continuar con los modelos de organización identificando flujos de trabajo y tareas. - Modelos de tareas y objetivos para generar restricciones de control (objetivos principales, descomposición de objetivos). - Refinar modelo de entorno para incluir nuevos elementos.	- Estudiar resto de casos de uso.
	Diseño	- Generar un prototipo con herramientas de prototipado rápido, como ZEUS o Agent Tool.	- Centrar el modelo de organización en el desarrollo de flujos de trabajo. - Llevar las restricciones identificadas a modelos de tareas y objetivos para dar detalles acerca de las necesidades y resultados de las tareas y su relación con los objetivos del sistema. - Expresar la ejecución de tareas dentro de modelos de interacción. - Generar modelos de agente para detallar patrones de estado mental.	- Generar nuevos modelos de agente o refinar los existentes. - Depurar la organización centrando el desarrollo en las relaciones sociales.

El los siguientes apartados se explicarán los modelos más importantes de cada una de las fases comentadas anteriormente. Como resultado de estos modelos se obtendrá una visión clara y detallada de la plataforma de agentes sobre la que trata este proyecto.

2.1. Casos de uso

Los casos de uso (CDU) reflejan la funcionalidad del sistema. Primero se construye el caso de uso de más bajo nivel, que identificará la funcionalidad general del sistema. Este se muestra en la Figura 1, y como se puede observar incluye dos participantes: consumidores, que requieren servicios del sistema; y proveedores, que proveen servicios al sistema.

Figura 1: CDU de nivel 0

El siguiente diagrama (Figura 2) de CDU muestra un nivel de detalle mayor, e incluye las funcionalidades que ofrecen los roles y agentes.

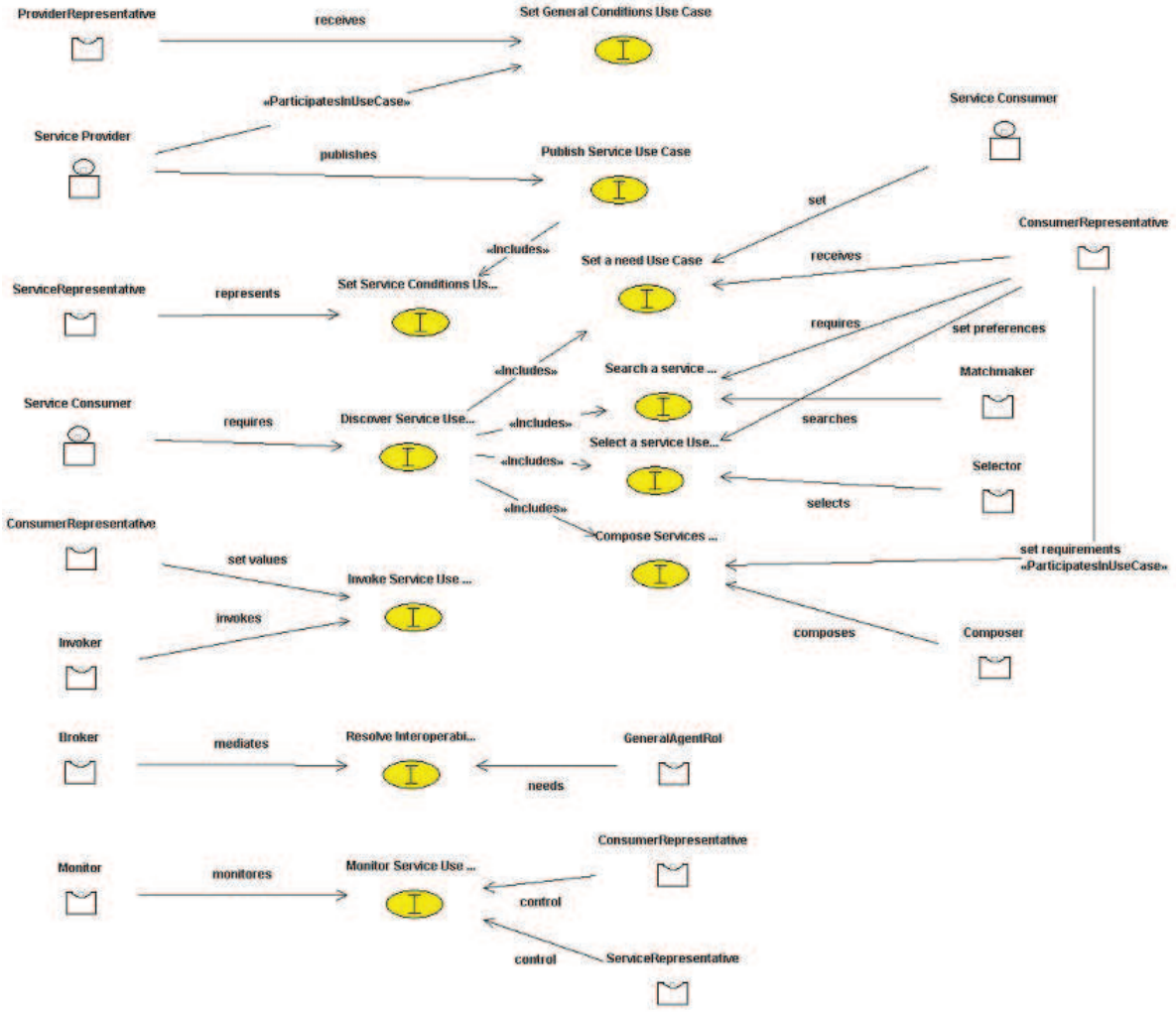

Figura 2: CDU de nivel 1

2.2. Modelo de organización

El modelo de organización es comparable a la estructura del sistema de agentes. En el se ofrece una visión de las relaciones existentes entre las entidades activas (agentes y roles), las pasivas (aplicaciones y recursos) y los elementos que proporcionan la funcionalidad del sistema (flujos de trabajo y tareas). La Figura 3 muestra la estructura general de la aplicación. *Company* debe satisfacer las necesidades de sus clientes proporcionándoles servicios.

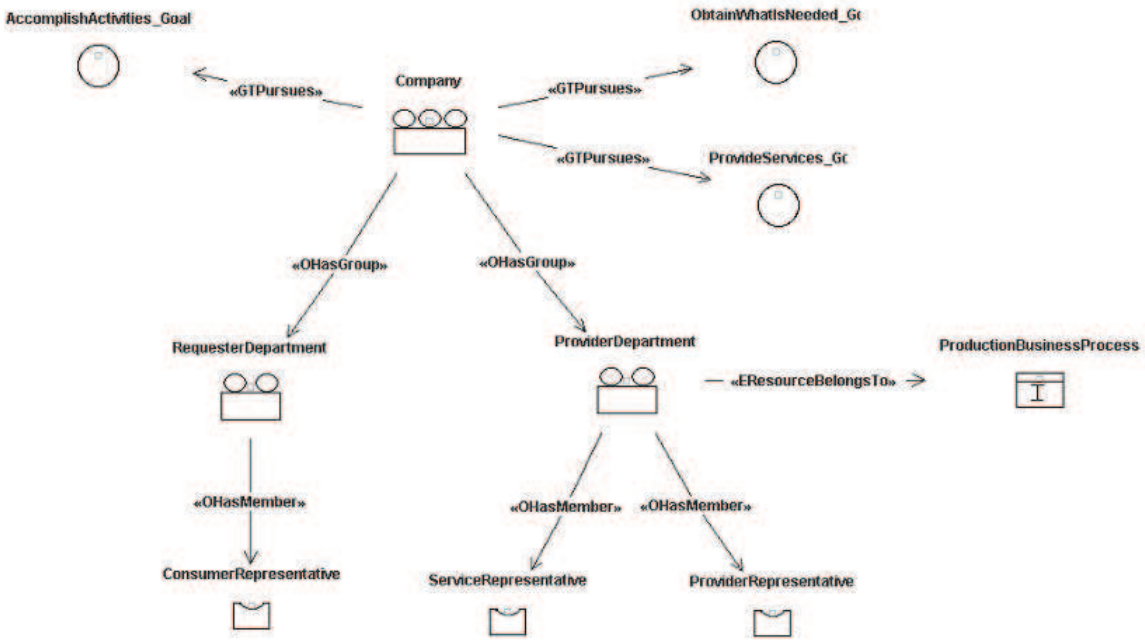

Figura 3: Modelo de organización

A continuación, en la Figura 4, se descompone la organización en grupos para ver con más detalle la labor que estos desempeñan. Concretamente, el modelo representa qué grupos necesita la organización y qué roles deben desempeñar cada uno de los grupos para satisfacer el objetivo global.

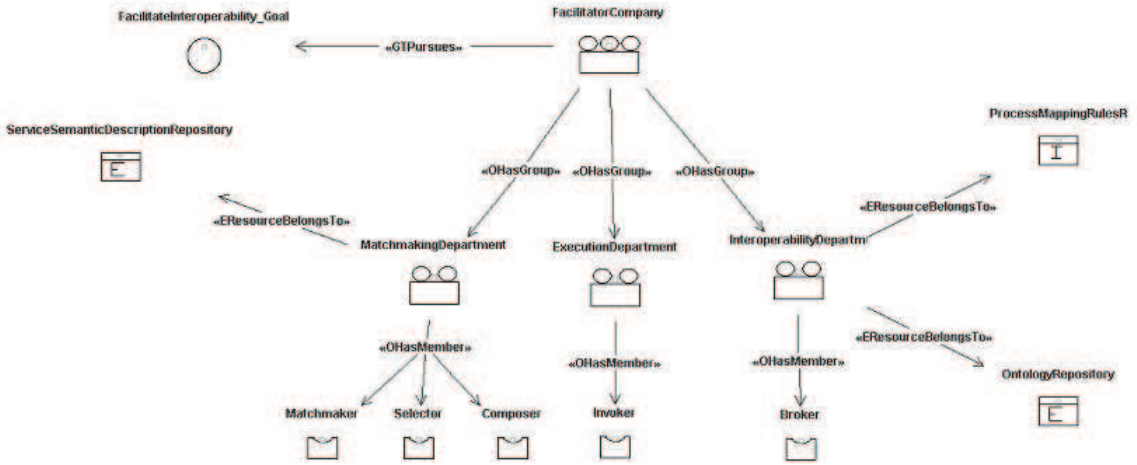

Figura 4: Modelo facilitador de organización

Este modelo es el primero en generarse, como se aconseja en INGENIAS.

2.3. Modelo de agentes

Este modelo describe cómo son los agentes, elementos fundamentales del sistema. En particular describe su funcionalidad, asigna los roles que juega cada uno y las tareas que deberán ejecutar. Lo normal es que este modelo se componga de una recolección de información de otros modelos, por lo que INGENIAS no aconseja tomarlo como punto de partida. La Figura 5 muestra el modelo de agentes obtenido.

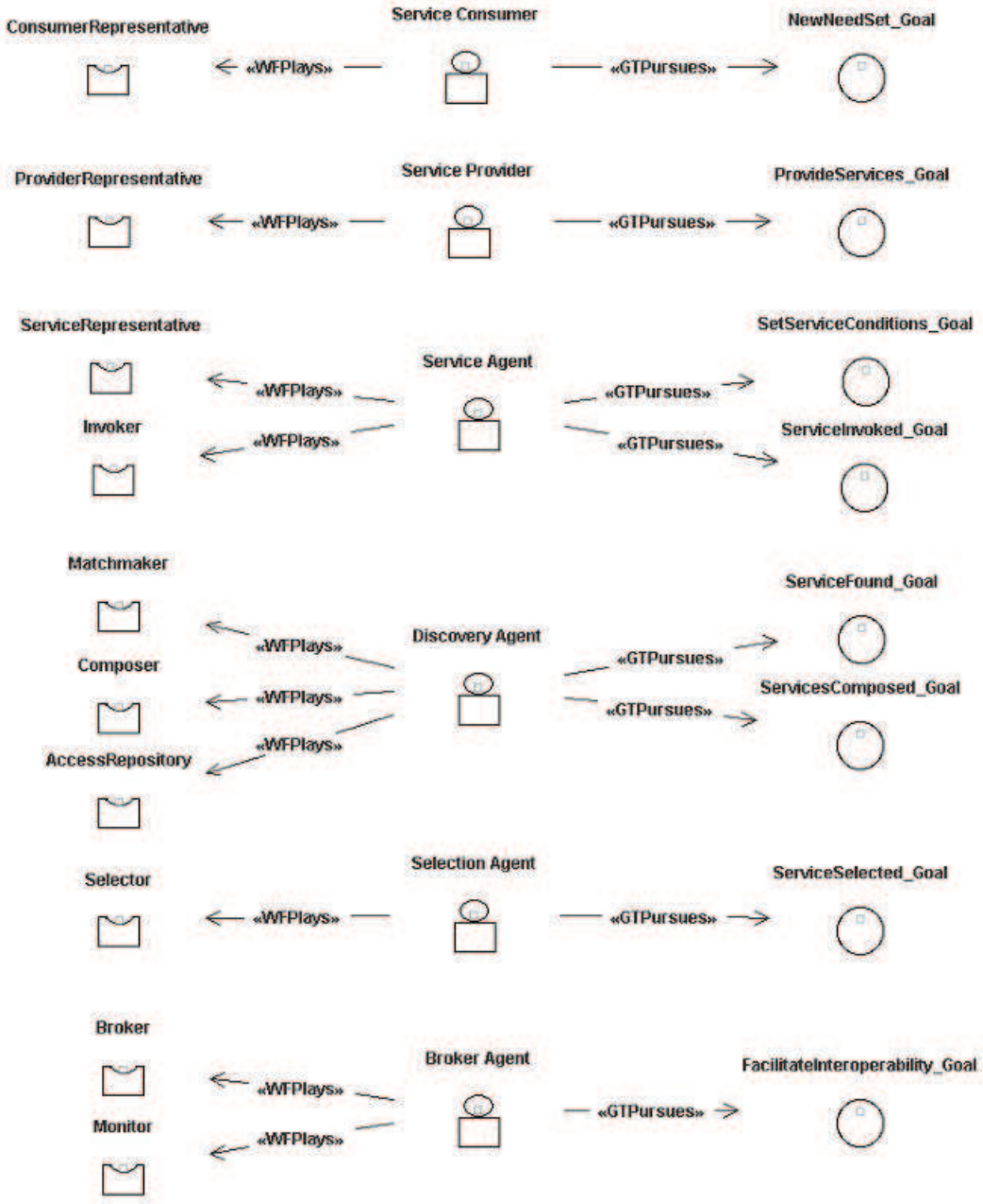

Figura 5: Modelo de Agentes

2.4. Modelo de objetivos y tareas

Este modelo tiene como propósito definir los subobjetivos surgidos al descomponer el objetivo general. Puede verse en la Figura 6

El objetivo principal es *Accomplish Activities Goal*. Este se divide en cuatro subobjetivos que a su vez se dividen en más subobjetivos. Los subobjetivos en los que se divide el objetivo principal son *Administration Goal*, *Obtain What Is Needed*, *Facilitate Interoperability*

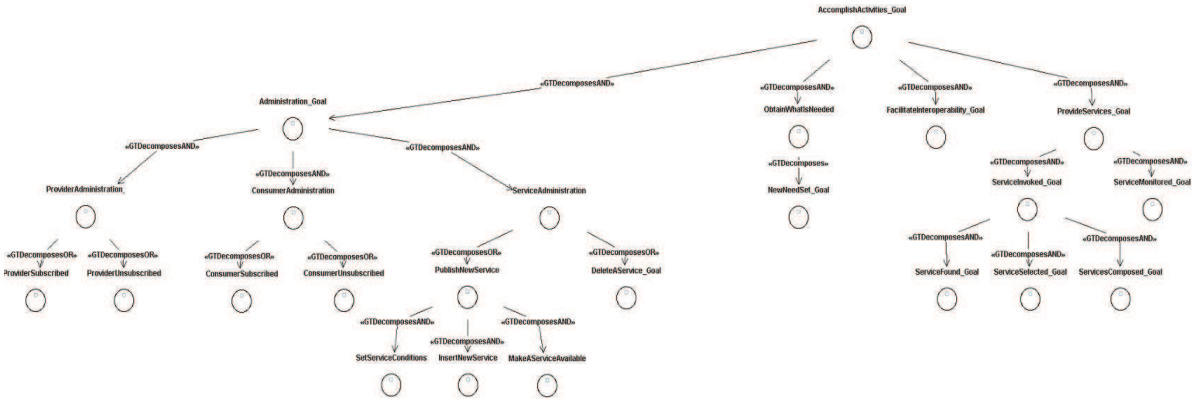

Figura 6: Modelo de Objetivos

y *Provide Service*, y deben llevarse a cabos todos por completo para alcanzar dicho objetivo general.

A continuación se dará una breve explicación de los subobjetivos más relevantes:

- ***Administration Goal:*** garantiza que la organización del sistema se haga de forma eficaz y eficiente. Se encarga de gestionar a los proveedores, consumidores y servicios.
- ***Obtain What Is Need:*** obtiene las necesidades de los consumidores que deben ser satisfechas por el sistema.
- ***Facilitate Interoperability:*** debe garantizar la interoperabilidad entre los distintos elementos del sistema.
- ***Provide Service:*** su labor es vital en el sistema, ya que es el encargado de proporcionar los servicios que satisfagan los requerimientos de los clientes. Se descompone en *Service Monitored* y *Service Invoked*.
- ***Service Monitored:*** garantiza la supervisión de los servicios.
- ***Service Invoked:*** se encarga de hacer posible la invocación de los servicios. Se descompone en *Service Found*, *ServiceSelected* y *Services Composed*.

2.5. Modelo de interacción

El modelo de interacción refleja el comportamiento que deben adoptar los agentes al recibir instrucciones de otro agente o de un humano. En la Figura 7 se muestra el diagrama de interacción general.

Para cada una de las interacciones que aparecen en la figura anterior se debe generar su diagrama de interacción específico. En la Figura 8 se puede ver el diagrama específico

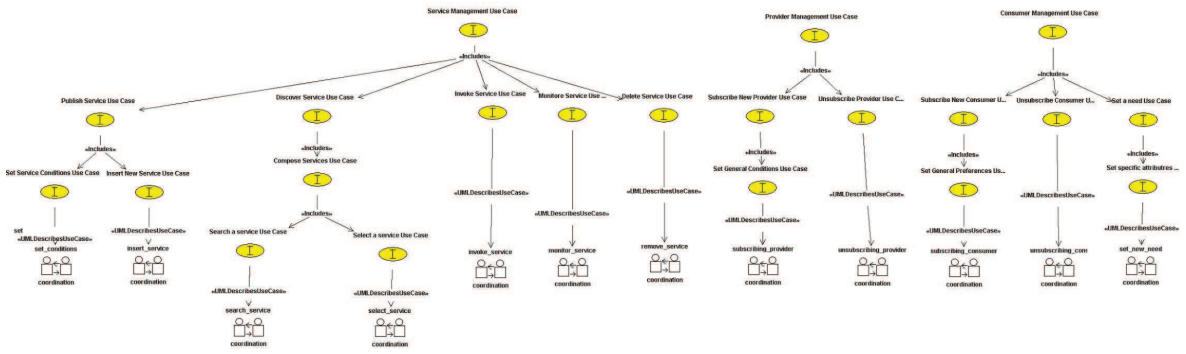

Figura 7: Modelo de Interacción general

de la interacción *insert_service*. En él se identifica el objetivo general de la interacción y se muestran las interacciones con la aplicación, agentes (*Service Provider*) y roles (*Service Representative*).

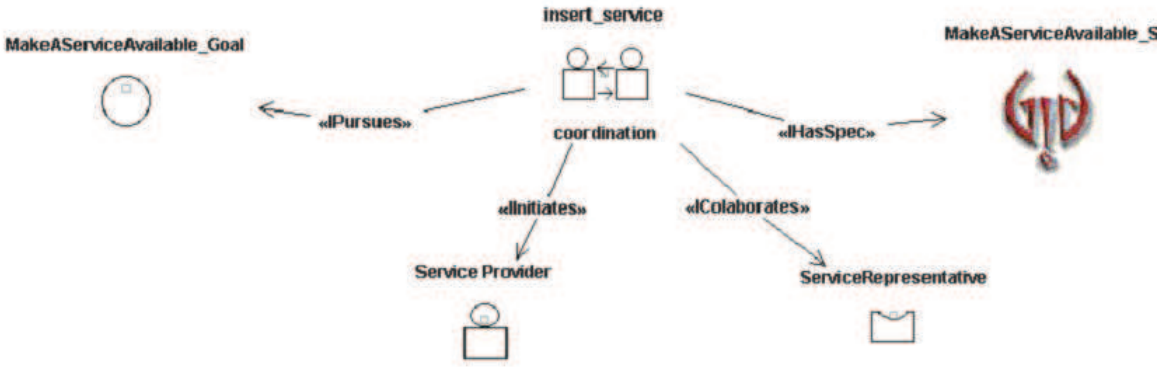

Figura 8: Modelo de Interacción de *Nuevo Servicio*

A cada diagrama de interacción le corresponde un diagrama de colaboración que defina el protocolo de comunicación que se utilice para las comunicaciones que se establecen durante la interacción. A continuación, en la Figura 9, se puede ver el diagrama de colaboración asociado a la interacción *insert_service*.

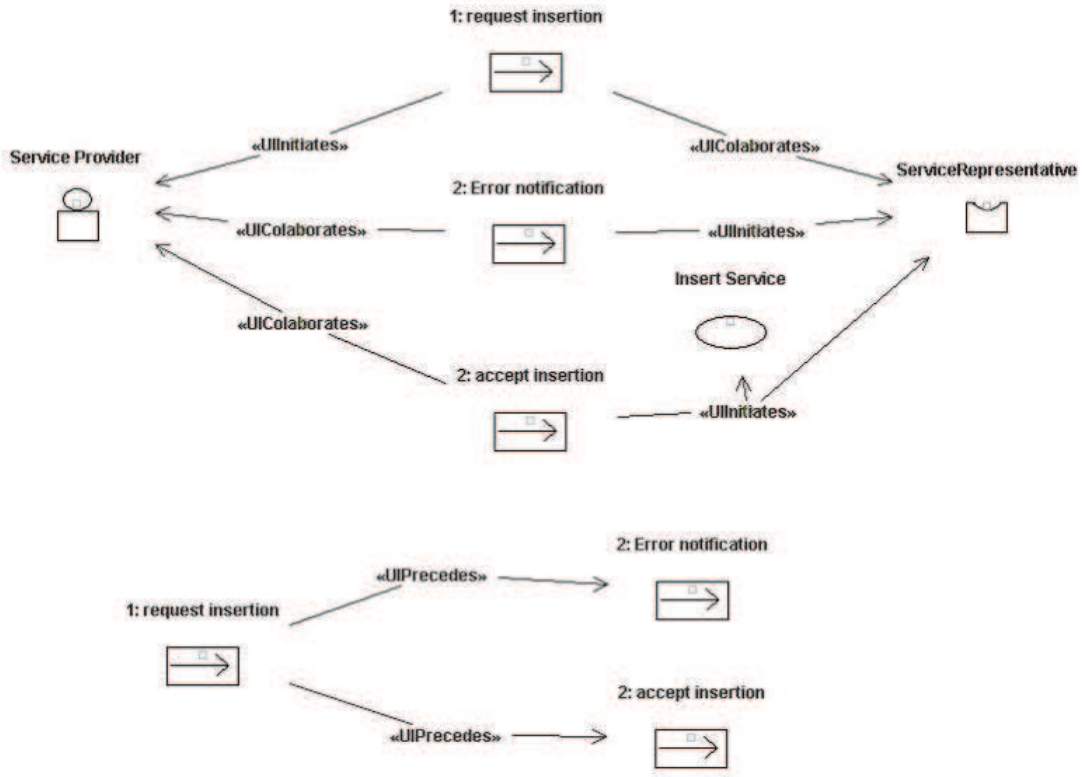

Figura 9: Modelo de Interacción

2.6. Modelo de entorno

Este modelo especifica las entidades del entorno en el que se encuentra el sistema de agentes, ya sean recursos consumibles o no consumibles, aplicaciones, o agentes. En la Figura 10 se muestra el modelo de entorno, en él aparecen todos los agentes del sistema y los recursos que se utilizan a lo largo de su ejecución. Según INGENIAS, la mayoría de información utilizada para la generación de este modelo proviene de la captura de requisitos.

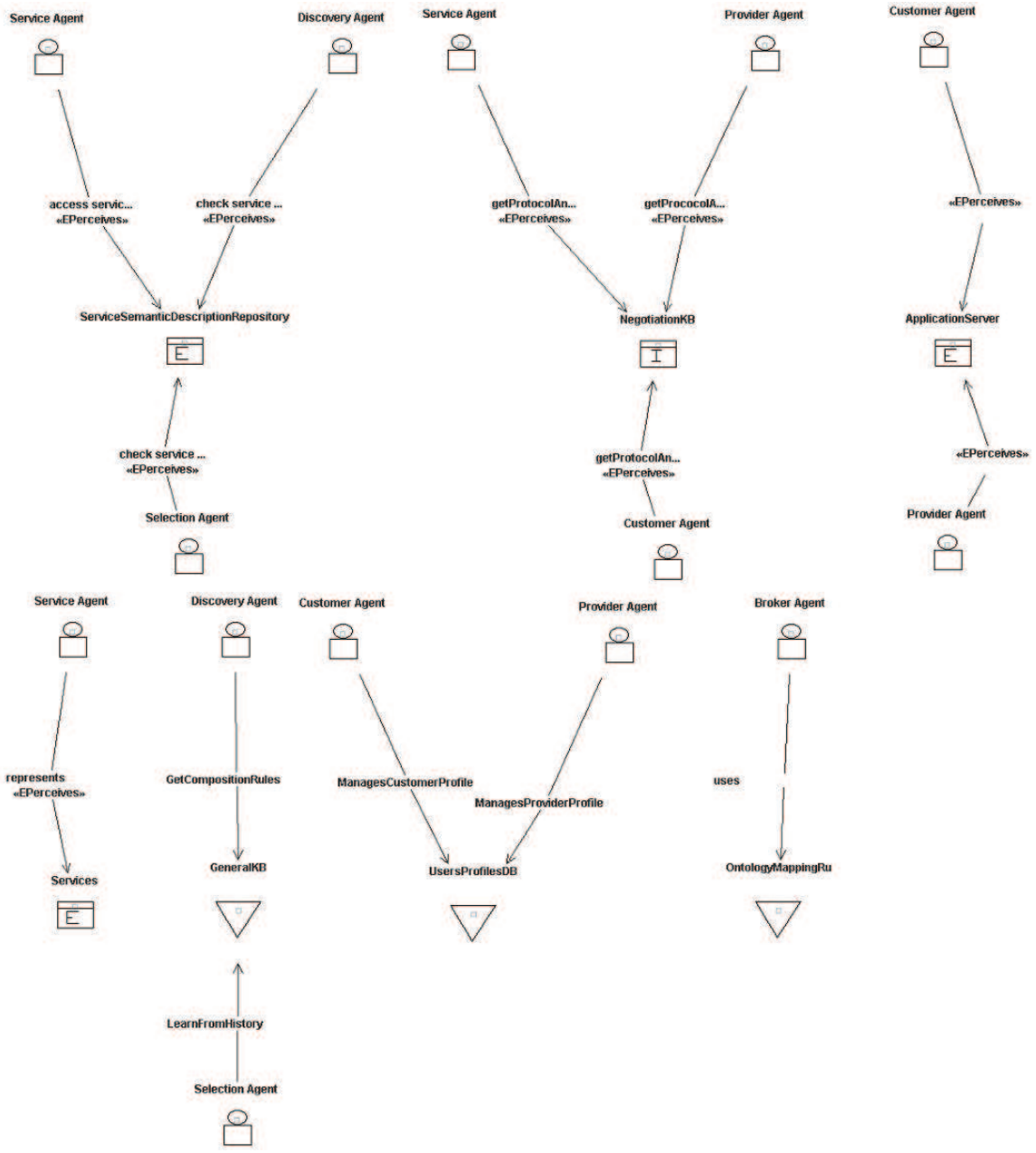

Figura 10: Modelo de Entorno

Referencias

[1] *“INGENIAS”*. 2005. Universidad Complutense de Madrid (GRASIA). `http://grasia.fdi.ucm.es/ingenias`.