

APLICACIONES PARA DISPOSITIVOS MOVILES UTILIZANDO LA TECNOLOGIA DE AGENTES



José Fernando Barrera Campo
David Alejandro Martínez Vásquez

**Proyecto de grado presentado como requisito para optar al título de
Ingeniero en Electrónica y Telecomunicaciones**

Director: Ing. Javier Alexander Hurtado Guaca

UNIVERSIDAD DEL CAUCA
FACULTAD DE INGENIERIA ELECTRÓNICA Y TELECOMUNICACIONES
DEPARTAMENTO DE TELEMÁTICA
Popayán, junio de 2004

TABLA DE CONTENIDO

<u>INTRODUCCION</u>	<u>- X -</u>
<u>CAPITULO I. AGENTES, SISTEMAS MULTIAGENTES Y CONCEPTOS BÁSICOS.</u>	<u>- 1 -</u>
1.1 INTRODUCCIÓN.....	- 1 -
1.2 AMBIENTES	- 2 -
1.2.1 ACCESIBLE.....	- 2 -
1.2.2 DETERMINÍSTICO.....	- 2 -
1.2.3 ESTÁTICO	- 2 -
1.2.4 DISCRETO	- 2 -
1.2.5 SISTEMAS FUNCIONALES Y REACTIVOS.....	- 4 -
1.2.6 AGENTES EN TIEMPO REAL.....	- 4 -
1.3 AGENTES INTELIGENTES.....	- 5 -
1.3.1 AGENTES INTELIGENTES Y EVENTOS.....	- 6 -
1.4 CLASES DE AGENTES.....	- 6 -
1.4.1 DELEGACIÓN, INTELIGENCIA Y MOVILIDAD	- 7 -
1.4.2 ESTRATEGIAS DE PROCESAMIENTO	- 7 -
1.4.3 FUNCIONES DE PROCESAMIENTO	- 8 -
1.5. SISTEMAS MULTIAGENTE	- 9 -
1.6. AGENTES Y OBJETOS	- 10 -
<u>CAPITULO II. MODELO DE REFERENCIA PARA EL DESARROLLO DE SERVICIOS</u>	
<u>CON ORIENTACION A AGENTES VERSION 1.0</u>	<u>- 11 -</u>
2.1 INTRODUCCIÓN.....	- 11 -
2.2 OBJETIVOS Y MOTIVACIONES.....	- 13 -
2.3 CONTEXTO Y ALCANCES DEL MODELO.....	- 13 -
2.4 CONCEPTOS Y PRINCIPIOS.....	- 17 -
2.4.1 SISTEMA.....	- 17 -
2.4.2 MODELO.....	- 17 -
2.4.3 ARQUITECTURA.....	- 18 -
2.4.4 SERVICIO	- 18 -
2.4.5 RM-ODP.....	- 19 -
2.4.6 ARQUITECTURA TINA.....	- 20 -
2.4.7 FIPA - FOUNDATION FOR INTELLIGENT PHYSICAL AGENTS	- 21 -
2.5 ESTRUCTURA DEL MODELO	- 22 -

2.6 SISTEMA DE TELECOMUNICACIONES MR-DSOA	- 23 -
2.6.1 CARACTERÍSTICAS DE UN SISTEMA DE TELECOMUNICACIONES MR-DSOA	- 23 -
2.6.2 ARQUITECTURA GENÉRICA DE UN SISTEMA DE TELECOMUNICACIONES	- 24 -
2.6.3 SISTEMAS DE TELECOMUNICACIÓN COMO AMBIENTES DISTRIBUIDOS	- 27 -
2.6.4 REDES Y SERVICIOS DE TELECOMUNICACIONES	- 29 -
2.7 DESCRIPCIÓN DETALLADA DEL MODELO.....	- 30 -
2.7.1 CAPA SERVICIO	- 31 -
2.7.2 CAPA DE APLICACIÓN	- 32 -
2.8 CICLO DE VIDA	- 33 -
2.8.1 FASE DE NECESIDAD.....	- 33 -
2.8.2 FASE DE DESARROLLO	- 33 -
2.8.3 FASE DE INSTALACIÓN	- 34 -
2.8.4 FASE DE OPERACIÓN	- 34 -
2.8.5 FASE DE RETIRO	- 35 -
2.8.6 RELACIÓN CICLO DE VIDA, SERVICIOS TELEMÁTICOS Y MR-DSOA	- 35 -
2.8 MODELO EMPRESARIAL	- 36 -
2.8.1 INTRODUCCIÓN.....	- 36 -
2.8.2 LENGUAJE EMPRESARIAL.....	- 36 -
2.8.3 ENTIDADES DEL MODELO EMPRESARIAL.....	- 40 -
2.9 MODELO COMPUTACIONAL	- 55 -
2.9.1 CONVENCIONES GRÁFICAS	- 55 -
2.9.2 ESTRUCTURAS COMPUTACIONALES	- 56 -
2.10 MODELO DE INFORMACIÓN.....	- 59 -
2.10.1 INTRODUCCIÓN	- 59 -
2.10.2 INFORMACIÓN	- 59 -
2.10.3 RELACIÓN DEL MODELO DE INFORMACIÓN CON LOS RESTANTES MODELOS	- 59 -
2.10.4 ESPECIFICACIÓN DE ENTIDADES.....	- 61 -
2.10.5 TIPOS DE INFORMACIÓN	- 70 -
2.10.6 RELACIONES ENTRE AGENTES DE INFORMACIÓN	- 70 -
2.10.7 INTERCAMBIO DE INFORMACIÓN	- 73 -
2.10.8 ONTOLOGÍAS	- 74 -
2.11 MODELO DE INGENIERÍA.....	- 75 -
2.11.1 CLUSTERS, CÁPSULAS Y NODOS	- 76 -
2.11.2 CANALES	- 76 -
2.11.3 INTERFACES DE INGENIERÍA	- 77 -
2.11.4 ESTABLECIMIENTO DEL CANAL	- 77 -
2.11.5 INTERCEPTORES.....	- 78 -
2.12 MODELO DE TECNOLOGÍA.....	- 79 -

CAPITULO III. DISEÑO UN SERVICIO DE MENSAJERIA MULTIMEDIA MMS

EMPLEANDO MR-DSOA V 1.0	- 81 -
3.1 INTRODUCCIÓN	- 81 -
3.2 DESCRIPCION DEL SERVICIO	- 82 -
3.3 ARQUITECTURA DEL SISTEMA	- 83 -
3.4 MODELO EMPRESARIAL	- 85 -
3.4.1 DOMINIOS ADMINISTRATIVOS DE NEGOCIO	- 86 -
3.5 MODELO COMPUTACIONAL	- 90 -
3.6 MODELO DE INFORMACION	- 92 -
3.6.1 AGENTE DE CONTROL DE ACCESO (ACA)	- 92 -
3.6.2 AGENTE DE ENLACE (AE)	- 94 -
3.6.3 AGENTE GESTOR DE BASE DE DATOS (AGBD)	- 94 -
3.6.4 RELACIONES ENTRE DOMINIOS ADMINISTRATIVOS DE NEGOCIO	- 95 -
3.6.5. ACCESO GENÉRICO AL SERVICIO DE CONECTIVIDAD	- 97 -
3.6.6 TRANSFERENCIA DE MENSAJES	- 98 -
3.6.7 ACCESO AL SERVICIO DE MENSAJERÍA MULTIMEDIA	- 100 -
3.6.8 ACCESO AL SERVICIO MULTIMEDIA	- 101 -
3.6.9 AGENTE DE GESTIÓN REMOTA (AGR)	- 102 -
3.6.10 AGENTES GESTOR DEL SERVICIO DE CONECTIVIDAD (AGSC _{PC})	- 103 -
3.6.11 AGENTE DE PAGINAS AMARILLAS (AP _{Ai})	- 104 -
3.6.12 AGENTE DE PÁGINAS BLANCAS (AP _{Bi})	- 104 -
3.6.13 AGENTE GESTOR DEL SERVICIO DE INTERMEDIACIÓN (AGSI _i)	- 104 -
3.6.14 CONSULTA AL SERVICIO DE PÁGINAS AMARILLAS O BLANCAS	- 106 -
3.6.15 AGENTE DEL SERVICIO DE MENSAJERÍA MULTIMEDIA (ASMM _M)	- 106 -
3.6.16 AGENTE GESTOR DE SERVICIO DE MENSAJERÍA MULTIMEDIA (AGSMM _M)	- 107 -
3.6.17 AGENTE DE SERVICIO MULTIMEDIA (ASM _{PES})	- 108 -
3.6.18 AGENTE GESTOR DEL SERVICIO MULTIMEDIA (AGSM _{PES})	- 109 -
3.6.19 USO DEL SERVICIO DE MENSAJERÍA MULTIMEDIA	- 111 -
3.7 MODELO DE INGENIERÍA	- 112 -
3.7.1 NODO DE SERVICIO DE MENSAJERÍA (MINORISTA)	- 112 -
3.7.2 NODO USUARIO	- 113 -
3.7.3 NODO PROVEEDOR DE CONECTIVIDAD	- 113 -
3.7.4 NODO PROVEEDOR EXTERNO DE SERVICIOS	- 114 -
3.7.5 NODO INTERMEDIARIO	- 114 -
3.8 MODELO DE TECNOLOGIA	- 116 -
3.9 DISEÑO DE LA APLICACIÓN	- 117 -
3.9.1. ANÁLISIS INICIO	- 117 -
3.9.3. ANÁLISIS ELABORACIÓN	- 119 -
3.9.4 DISEÑO ELABORACIÓN	- 127 -

4. CONCLUSIONES Y TRABAJOS FUTUROS - 135 -

4.1 CONCLUSIONES - 135 -

4.2 TRABAJOS FUTUROS - 139 -

GLOSARIO - 141 -

BIBLIOGRAFIA - 144 -

CAPITULO I..... - 144 -

CAPITULO II..... - 144 -

INDICE DE FIGURAS

Figura 2.1.	Curva en S de la innovación tecnológica.....	12 -
Figura 2.2.	Servicios desde la perspectiva ROSA.	19 -
Figura 2.3.	Composición de la arquitectura TINA.....	21 -
Figura 2.4.	Dependencia entre arquitecturas.	21 -
Figura 2.5.	Arquitectura FIPA.....	22 -
Figura 2.6.	Composición de un sistema de telecomunicaciones.....	24 -
Figura 2.7.	Estructura básica del software de telecomunicaciones.....	25 -
Figura 2.8.	Descomposición en nodos de un sistema de telecomunicaciones.....	28 -
Figura 2.9.	Vista completa de un sistema MR-DSOA.....	29 -
Figura 2.10.	Red de transporte en el ambiente MR-DSOA.....	29 -
Figura 2.11.	Descomposición por vistas MR-DSOA.....	30 -
Figura 2.12.	Modelo para el desarrollo de servicios con orientación a agentes (MR-DSOA).....	31 -
Figura 2.13.	Ciclo de vida de los servicios MR-DSOA.	33 -
Figura 2.14.	Dominios.....	41 -
Figura 2.15.	Jerarquías en un dominio.....	42 -
Figura 2.16.	Mapeo de dominios.....	43 -
Figura 2.17.	Conceptos empresariales.	45 -
Figura 2.18.	Modelo de negocio y relaciones de negocio.....	48 -
Figura 2.19.	Representación gráfica de agentes computacionales.....	56 -
Figura 2.20.	Transferencia de mensajes.....	57 -
Figura 2.21.	Interfaz de flujo.....	58 -
Figura 2.22.	Agente computacional con múltiple interfaz.	58 -
Figura 2.23.	Arquitectura FIPA.....	61 -
Figura 2.24.	Estructura estándar de los protocolos de interacción FIPA.....	65 -
Figura 2.25.	Parámetros de un mensaje ACL.....	65 -
Figura 2.26.	Tipos de agentes.....	69 -
Figura 2.27.	Protocolo de interacción.....	72 -
Figura 2.28.	Diagrama de intercambio de información.....	73 -
Figura 2.29.	Clusters, cápsulas y nodos.....	76 -
Figura 2.30.	Canal.....	76 -
Figura 2.31.	Métodos de comunicación entre plataformas.....	78 -
Figura 2.32.	Interceptor en Línea-Frontera de tecnología.....	79 -

Figura 2.33. Interceptor dividido-Frontera administrativa	- 79 -
Figura 2.34. Interceptor dividido-Frontera combinada	- 79 -
Figura 3.1. Caso de usos	- 82 -
Figura 3.2. Arquitectura del sistema.....	- 84 -
Figura 3.3. Modelo empresarial.....	- 85 -
Figura 3.4. Modelo computacional.....	- 90 -
Figura 3.5. Agente de control de acceso.	- 92 -
Figura 3.6. Agente de enlace.....	- 94 -
Figura 3.7. Agente gestor de base de datos.	- 94 -
Figura 3.8. Acceso genérico al proveedor de conectividad.....	- 97 -
Figura 3.9. Transferencia de mensajes.....	- 99 -
Figura 3.10. Acceso al servicio de mensajería multimedia.....	- 100 -
Figura 3.11. Relación MMS-Multimedia.....	- 101 -
Figura 3.12. Agente de gestión remota.	- 102 -
Figura 3.13. Gestión del servicio de conectividad.....	- 103 -
Figura 3.14. Gestión del servicio de intermediación.	- 105 -
Figura 3.15. Servicio de intermediación.	- 106 -
Figura 3.16. Gestión del servicio de mensajería multimedia.....	- 107 -
Figura 3.17. Gestión del servicio multimedia.	- 109 -
Figura 3.18. Envío de un mensaje.	- 111 -
Figura 3.19. Nodo de servicio de mensajería.	- 112 -
Figura 3.20. Nodo de usuario.....	- 113 -
Figura 3.21. Nodo de servicio de mensajería.	- 113 -
Figura 3.22. Nodo proveedor externo de servicios.	- 114 -
Figura 3.23. Nodo de intermediación.	- 115 -
Figura 3.24. Interceptores de línea en el dominio del proveedor de conectividad.....	- 115 -
Figura 3.25. Modelo de tecnología.....	- 116 -
Figura 3.26. Diagrama de casos de uso.....	- 117 -
Figura 3.27. Modelo organizacional de la aplicación (Arquitectura inicial).	- 119 -
Figura 3.28. Diagrama de agente Usuario.....	- 121 -
Figura 3.29. Modelo de agente Registro.	- 123 -
Figura 3.30. Modelo de agente Gestor.....	- 124 -
Figura 3.31. Modelo de objetivos y tareas agente Usuario	- 126 -
Figura 3.32. Modelo de objetivos y tareas agente Gestor.	- 127 -
Figura 3.33. Interacción de Registro (Int de registro), agentes Usuario y Registro.	- 128 -
Figura 3.34. Especificación GRASIA Interacción de Registro (Int de registro).....	- 129 -
Figura 3.35. Interacción de validación (Int de validación), entre agentes Usuario y Registro.	- 130 -

Figura 3.36. Especificación GRASIA Interacción de Validación (Int de validación).....	- 130 -
Figura 3.37. Interacción Gestor-Registro, entre agentes Gestor y Registro.	- 131 -
Figura 3.38. Especificación GRASIA Interacción Gestor-Registro.....	- 132 -
Figura 3.39. Interacción Usuario-Gestor, entre agentes Usuario y Gestor.	- 132 -
Figura 3.40. Especificación GRASIA Interacción Gestor-Usuario (Rol Receptor mensajes).....	- 133 -
Figura 3.41. Especificación GRASIA Interacción Gestor-Usuario (Rol Emisor mensajes).....	- 134 -

INDICE TABLAS

Tabla 1.1. Precondiciones y acciones de un agente (Termostato).....	- 1 -
Tabla 3.1. Registro de puntos de acceso y servicios.	- 88 -
Tabla 3.2. Agentes computacionales del proveedor externo de servicios	- 90 -
Tabla 3.3. Agentes computacionales del minorista.....	- 91 -
Tabla 3.4. Agentes computacionales del intermediario.....	- 91 -
Tabla 3.5. Agentes computacionales del usuario.....	- 91 -
Tabla 3.6. Agentes computacionales del proveedor de conectividad.....	- 91 -
Tabla 3.7. Relaciones de negocio.	- 96 -
Tabla 3.8. Especificación de tecnología.....	- 116 -

INTRODUCCION

El paradigma de agentes se está convirtiendo paulatinamente en una herramienta útil para el desarrollo de servicios distribuidos, aportando nuevas capacidades que aun no han sido exploradas y que pueden contribuir valiosamente a nivel teórico y tecnológico, trazando un camino para la evolución natural que estos sistemas deben experimentar. Quizás una de las características más importantes en este sentido es el nivel de autonomía que provee el uso de agentes, tal propiedad es una característica inherente que reduce significativamente el acoplamiento entre componentes de un servicio, además las comunicaciones se realizan a través de mensajes sin necesidad de un contacto muy estrecho ente ellos, es decir, cada agente puede seguir ejecutando sus funciones mientras realiza un proceso de comunicación, estas y otras propiedades son las que se buscan explorar y explotar en el desarrollo de servicios.

Tales características entre otras proporcionadas por los agentes y su naciente aplicación en un sinnúmero de proyectos de vanguardia a nivel mundial motivan el desarrollo de una herramienta que permita el diseño de servicios basados en este paradigma y que considere a los dispositivos de cómputo personal como los terminales que intervienen y acceden a la funcionalidad provista por un sistema de telecomunicaciones. El primer paso dentro de este proceso es determinar las características que hacen de un agente una entidad idónea para el desarrollo de servicios, determinándose así como podrían formar parte de una arquitectura de servicio y que funcionalidad tendrían asociada.

Esta herramienta de diseño toma forma en el MR-DSOA o Modelo de Referencia para el Desarrollo de Servicios Orientados a Agentes, con el que se modelara un servicio prototipo con dos finalidades: servir de ejemplo y como medio de validación.

El proyecto se basó en dos estándares muy difundidos en el ámbito de las telecomunicaciones como son TINA y RM-ODP que implícitamente hablan del enfoque dado al modelo (orientado a sistemas de telecomunicación y sistemas distribuidos), la parte de agentes se incluye a través de FIPA e INGENIAS.

Como se vera más adelante TINA define la arquitectura de un sistema de telecomunicaciones y esta dividida en cuatro partes:

- Arquitectura del servicio.
- Arquitectura de red.
- Arquitectura de gestión.
- Arquitectura computacional

De las anteriores, es la arquitectura computacional en la que este trabajo se concentro ya que esta en contacto con la parte de servicios y de infraestructura, aspectos necesarios en el modelo. Lo que pretende MR-DSOA es equiparable a lo que hizo TINA pero con agentes.

El principal objetivo de este proyecto de grado es proponer un modelo para el desarrollo de servicios basados en agentes, adaptando el paradigma y proponiendo un conjunto de consideraciones a nivel de diseño para que usuarios móviles puedan utilizarlos, para ello se hace necesario estudiar y analizar el paradigma de agentes en un contexto general para su posterior aplicación.

Por último, se aplicara el modelo en el desarrollo de un prototipo con la tecnología de agentes, teniendo en cuenta los conceptos planteados y descritos en el modelo MR-DSOA.

DESCRIPCION DEL DOCUMENTO

CAPITULO I. AGENTES, SISTEMAS MULTIAGENTE Y CONCEPTOS BASICOS

En este capitulo se presenta una breve descripción de todos aquellos conceptos básicos relevantes para el proyecto, tales como: teoría de agentes, usos, tipos de agentes, sistemas, ambientes y etc. Facilitando la comprensión de los restantes capítulos.

CAPITULO II. MODELO DE REFERENCIA PARA EL DESARROLLO DE SERVICIOS CON ORIENTACION A AGENTES VERSION 1.0

El desarrollo de un servicio siempre va ligado a una especificación conceptual previa que relaciona sus objetivos o requerimientos con un arreglo de entidades software. Este capitulo describe al modelo MR-DSOA presentando cada una de sus partes, fases y consideraciones, necesarias para el desarrollo de servicios para dispositivos móviles empleando la tecnología de agentes.

CAPITULO III. DISEÑO DE UN SERVICIO DE MENSAJERIA MULTIMEDIA MMS EMPLEANDO MR-DSOA V 1.0

El desarrollo de un modelo no estaría completo hasta que se muestre como emplearlo en una situación real, o sea, en el diseño de un servicio telemático. En este capitulo se presenta el diseño de un servicio prototipo basado en el paradigma de agentes, empleado el Modelo de Referencia para el Desarrollo de Servicios Orientados a Agentes (MR-DSOA). Este diseño muestra como utilizar el modelo.

CAPITULO I. AGENTES, SISTEMAS MULTIAGENTES Y CONCEPTOS BÁSICOS.

1.1 INTRODUCCIÓN

Este capítulo presenta los conceptos básicos acerca de lo que son los agentes, los tipos existentes, sus comportamientos, ambientes sobre los que operan, la manera en que interactúan y todo lo necesario para tener una idea sobre el por qué de su existencia.

Una definición universalmente aceptada de lo que es un agente es algo difícil de encontrar. Existe un consenso general en la comunidad informática sobre la autonomía que éstos poseen, pero contrariedad cuando se tienen en cuenta otros atributos, ya que la importancia que éstos tienen depende del dominio sobre el que operan, de esta manera, la habilidad de aprender de sus propias experiencias puede jugar un papel importante en un dominio específico, mientras que para otro puede ser completamente indeseable.

Un agente es un sistema computacional capaz de actuar independientemente del usuario. En otras palabras, un agente comprende que acciones debe ejecutar para satisfacer los objetivos para los cuales fue diseñado sin necesidad de que se le especifique lo que debe hacer en un determinado momento.

Un agente no tiene un control total sino parcial del ambiente en el que se desempeña, razón por la cual el resultado de una acción ejecutada no siempre es el esperado. Generalmente por esta razón se denomina a los ambientes con mediana complejidad como no determinísticos.

Para modificar un ambiente, un agente posee un repertorio de acciones, pero cada una de ellas es aplicable solo en una situación definida, es decir, para su ejecución cada acción depende de unas precondiciones asociadas a ella. Un ejemplo muy utilizado para describir el comportamiento de un agente es un termostato. El termostato tiene un sensor para detectar la temperatura de una habitación, de manera que es capaz de producir una señal de salida cuando la temperatura es baja o está dentro del límite aceptado. El termostato tiene un conjunto de acciones para ejecutar de acuerdo al estado de la señal que el sensor esté produciendo, así, tenemos un conjunto de precondiciones y uno de acciones a ejecutar en cada caso de la siguiente manera:

Tabla 1.1. Precondiciones y acciones de un agente (Termostato).

Precondición	Acción
Temperatura baja	Subir la calefacción
Temperatura alta	Apagar la calefacción

1.2 AMBIENTES

En la literatura se pueden encontrar muchas clasificaciones de los ambientes en los que un agente puede ser ejecutado, a continuación se presentan las más relevantes:

1.2.1 Accesible

Un ambiente accesible es aquel en el cual un agente puede obtener información completa, exacta y actualizada del estado del ambiente. Muchos ambientes del mundo real no son accesibles, tal es el caso de Internet.

1.2.2 Determinístico

Un ambiente determinístico es aquel en el cual la ejecución de una acción tiene un efecto garantizado, sin incertidumbre sobre el estado que tendrá resultado debido a la acción.

1.2.3 Estático

Un ambiente estático es aquel con la propiedad de permanecer estable hasta que un agente ejecuta una acción. En contraste, en un ambiente dinámico existen otros procesos operando, por lo tanto el agente no tiene control sobre él.

1.2.4 Discreto

Un ambiente es discreto si hay un número fijo y finito de acciones y percepciones en él.

También existen sus opuesto, o sea inaccesibles, no determinísticos, dinámicos y continuos. Un buen agente es aquel que es capaz de tomar decisiones. La calidad de estas decisiones depende de que tan bien informado esté sobre su ambiente. Si el nivel de información que posee a la hora de tomar una decisión es precario, entonces el resultado muy probablemente no será el esperado.

Cuando hablamos de ambientes determinísticos decimos que el resultado de cualquier acción ejecutada es único. Esta propiedad es preferible desde el punto de vista de un diseñador de agentes ya que no existe ningún nivel de incertidumbre sobre el resultado de una acción. Desafortunadamente si un ambiente es lo suficientemente complejo, el hecho de que sea determinístico no es de mucha ayuda.

En la práctica, los ambientes deben ser asumidos como no determinísticos, ya que el efecto de una acción sobre el medio no puede ser cuantificada, este concepto se relaciona con el de dinamismo.

En los inicios de la investigación de Inteligencia Artificial se crearon algoritmos de planeación que dan una descripción del estado actual de un ambiente, las acciones disponibles dentro de este, los efectos que estas tienen y el estado esperado. En este tipo de algoritmos se genera un plan (una secuencia de acciones) de manera que cuando se ejecutan a partir del estado inicial, se alcanzará el estado objetivo. Sin embargo, este tipo de algoritmos fueron diseñados para funcionar únicamente en ambientes estáticos, es decir, aquellos que no cambian excepto cuando un agente actúa sobre ellos. Es claro que muchos ambientes del mundo real no son estáticos, tal es el caso de los sistemas operativos, en los cuales muchos procesos se están ejecutando concurrentemente en formas en las que un agente no tiene capacidad de controlar.

Desde el punto de vista de un agente, los ambientes dinámicos tienen dos importantes propiedades. La primera está relacionada con el tiempo de ejecución de una acción y el estado inicial de un ambiente, en otras palabras, un agente ejecuta una acción en un intervalo de tiempo determinado $t_1 - t_2$, en el transcurso del cual el estado inicial del ambiente puede cambiar. Esto significa que mientras un agente ejecute una acción debe recolectar información para determinar el estado actual del ambiente ya que sus condiciones iniciales pueden cambiar en el intervalo $t_1 - t_2$ y la meta lograda puede ser diferente a la esperada [Moore 1990].

La segunda propiedad está relacionada con la interferencia que otros procesos pueden producir sobre la acción que se pretende ejecutar. De esta manera un agente no puede estar seguro de que la acción que ejecutará tendrá el resultado que espera.

Las propiedades descritas para ambientes dinámicos y estáticos sugieren que a nivel de diseño son más manejables los segundos que los primeros, ya que en estos el agente necesita recolectar información sobre su ambiente una sola vez y las acciones que ejecute tendrán los efectos deseados. Además, un agente que opere en un ambiente estático no debe preocuparse por asuntos como la coordinación y sincronización de sus acciones con otros procesos ejecutándose en el mismo ambiente [Bond 1988].

Para finalizar el análisis de los ambientes, tenemos una última relación entre los discretos y los continuos. Un ambiente discreto es aquel que posee un número finito de estados; en contraste en un ambiente continuo existen infinitos estados posibles. Según lo anterior, podemos concluir que el diseño de agentes en un estado discreto es más simple que en uno continuo por varias razones, una de ellas es que los computadores son un ambiente de este tipo, aunque muchas veces pueden simular ambientes de tipo continuo realizando procesos de muestreo para transformar este tipo de ambientes a discretos, obteniendo como resultado una aproximación de ellos; de esta manera la información que un agente recolecte de un ambiente discreto producido a partir de un ambiente continuo no será completamente exacta.

En general un ambiente discreto brinda la posibilidad de enumerar todos los posibles estados y las acciones a ejecutar en cada uno de ellos, a diferencia de lo que ocurre con los ambientes continuos. En conclusión los ambientes más complejos son aquellos inaccesibles, no determinísticos, dinámicos y continuos.

1.2.5 Sistemas Funcionales y Reactivos

Originalmente, el campo de la ingeniería de software se centra en lo que se conoce como sistemas funcionales. Un sistema funcional es aquel que simplemente toma una entrada, ejecuta algún cálculo sobre ella y produce alguna salida. Formalmente un sistema de este tipo puede ser visto como una función $f: I \rightarrow O$, como un conjunto I de entradas y O un conjunto de salidas.

Uno de los principales atributos de un sistema funcional es que sus propiedades pueden ser entendidas en términos de precondiciones y poscondiciones [Hoare 1969]. La idea es que la precondición representa lo que debe ser verdad en el ambiente de un programa para que éste opere correctamente. La poscondición representa que debe ser verdad en el ambiente en el que un programa opera después de que éste termina su trabajo.

Desafortunadamente, muchos sistemas computacionales entre ellos los de agentes no son funcionales en el sentido descrito anteriormente, estos, además de operar de acuerdo a una entrada para obtener una salida, son reactivos tal como se describe a continuación:

Un sistema reactivo no puede ser descrito de manera relacional o funcional. La forma relacional considera los programas como funciones, con un estado inicial y otro final. Un sistema reactivo mantiene una interacción con su ambiente [Pnueli 1986].

Otra descripción establecida en términos de Inteligencia Artificial (AI) es: Un sistema reactivo es aquel capaz de responder rápidamente a cambios en su ambiente.

Un sistema reactivo eleva el nivel de complejidad en cuanto al diseño de agentes debido a que las decisiones que se hagan a nivel local tendrán efectos globales, es decir, un sistema reactivo es más difícil de diseñar que un sistema funcional.

1.2.6 Agentes en tiempo real

Otro aspecto relacionado con la interacción de un agente con su ambiente es el concepto de tiempo real. Este concepto juega un papel importante cuando tenemos en cuenta el tiempo de ejecución de las acciones de un agente. Se pueden identificar diferentes tipos de interacciones de tiempo real:

- Las que definen que acciones deben ser ejecutadas en un tiempo determinado.
- Aquellas en las cuales el agente debe efectuar una acción tan rápidamente como pueda.
- En las que el agente debe repetir alguna tarea a una frecuencia determinada.

1.3 AGENTES INTELIGENTES

Para poder conceptualizar más fácilmente lo que es un agente inteligente vamos a listar las capacidades que éstos deben tener: [Wooldrige 1995].

- **Reactividad:** Los agentes inteligentes son capaces de percibir su ambiente y responder de manera oportuna a los cambios que ocurren con el fin de satisfacer sus objetivos de diseño.
- **Proactividad:** Los agentes inteligentes son capaces de exhibir un comportamiento dirigido a metas, es decir, un comportamiento que busca llegar a un estado final.
- **Habilidad social:** Los agentes son capaces de interactuar con otros agentes (y posiblemente con humanos) para obtener colaboración y cooperar con el fin de satisfacer los objetivos de diseño propios y ajenos.

Consideremos cada término para explicar mejor de que se tratan:

Tal como lo mencionamos anteriormente la proactividad se refiere a un comportamiento dirigido hacia metas. No es difícil construir un sistema que exhiba este tipo de comportamiento. El procedimiento funciona como un plan que contiene los pasos necesarios para lograr la meta o estado final. Si la precondition se cumple cuando el agente invoca un proceso, entonces se logrará el estado esperado o la poscondición, que es lo mismo que alcanzar la meta.

Este modelo funciona cuando consideramos al ambiente como un sistema funcional, tal como lo discutimos anteriormente, pero si hablamos de sistemas no funcionales el modelo de comportamiento dirigido hacia metas no es aceptable, debido a que éstos asumen que el ambiente no cambia mientras se está ejecutando un proceso. Consideremos el caso de un sistema no funcional en el cual se ha establecido una precondition bajo la cual el agente ejecutará un proceso con el fin de lograr el estado meta; en este caso, debido a que se trata de un sistema no funcional la precondition puede cambiar mientras el proceso se está realizando, de igual forma el estado meta ya no va a ser el mismo a causa del continuo cambio del ambiente, siendo inútil en este caso que el agente continúe con la ejecución del proceso.

En ambientes de este tipo un agente debe ser reactivo, es decir, sensible a los eventos que ocurren en su ambiente y que afectan ya sea sus estados meta o sus estados iniciales en los que éste se basa para ejecutar sus procedimientos.

La integración de un comportamiento dirigido hacia metas y la reactividad es uno de los principales problemas que debe afrontar un diseñador de agentes. Por un lado se tiene que lograr que el agente busque llegar al estado esperado sin ejecutar un proceso ciegamente sabiendo que las condiciones iniciales en las que se basó han cambiado, o la meta no es la que el proceso en ejecución alcanzará. Por otro lado, es necesario

que la reactividad sea limitada, es decir, que el agente no se concentre solo en ella y nunca llegue al estado meta o esperado.

Como última característica está la habilidad social de los agentes. Esta tiene que ver con la actividad de intercambio de información sobre las metas que se quieren lograr, con el fin de recibir y prestar cooperación de otros agentes para cumplir con el estado objetivo.

1.3.1 Agentes Inteligentes y eventos

En el contexto de agentes inteligentes, un evento es algo que ocurre para modificar el ambiente y de lo cual el agente debe estar consciente. Cuando ocurre un evento el agente tiene que reconocerlo, evaluar cual es su significado y por último actuar ante este. Tomemos como ejemplo la llegada de un correo electrónico a través de una red a un host específico, en este caso el evento es la llegada del correo. El evento tendrá una autodescripción explicando que un nuevo correo ha llegado. El agente hará una petición al sistema de correo para saber quien ha enviado el correo, el asunto, etc. Es más, el agente puede estar diseñado para realizar una búsqueda de palabras clave dentro del texto del correo, de esta manera podría identificar si el mensaje ha sido clasificado como URGENTE y quien fue el remitente.

Los agentes pueden ejecutar acciones para hacer cosas por nosotros. Aunque, el hecho de que sistemas computacionales hagan cosas por nosotros no es nada nuevo, en los sistemas convencionales no tenemos un nivel de confianza que nos asegure que los comandos que ejecutamos produzcan los resultados que tenemos en mente. Cuando es un agente quien realiza alguna acción por nosotros, el nivel de confianza en que se ejecutará lo que pensamos aumenta debido a que en este caso el proceso tiene características de racionalidad que el agente le aporta y va de acuerdo con nuestros intereses. Como en todas las situaciones, cuando delegamos acciones a terceros corremos riesgos pero también obtenemos beneficios. En este caso el riesgo es que el agente confunda sus metas y nos tome más trabajo reparar el daño causado, pero por otro lado se encuentra la ventaja de estar libres de preocupaciones sobre la ejecución de una actividad determinada [Bigus 2001].

1.4 CLASES DE AGENTES

Aunque los agentes todavía son algo nuevo en el mundo informático comercial, muchos investigadores han trabajado en el desarrollo de éstos durante años, dando origen a diferentes formas para su clasificación. Una de ellas es ubicar a los agentes en el contexto de la Inteligencia, delegación, y movilidad. Otra forma está enfocada hacia la estrategia de procesamiento del agente. Y la última categoriza al agente según la acción que ejecuta. A continuación vamos a describir cada una de estas clasificaciones.

1.4.1 Delegación, Inteligencia y Movilidad

Cuando se habla de agentes software existen tres ramas sobre las cuales se miden sus capacidades: delegación, inteligencia y movilidad [IBM 1996].

Delegación tiene que ver con el grado de autonomía que el agente software tiene para representar al usuario ante otros agentes, aplicaciones, y sistemas computacionales, en otras palabras, el agente es un delegado del usuario. Un agente representa a un usuario, lo guía, lo ayuda y en algunos casos, toma decisiones unilaterales para beneficiarlo.

La *Inteligencia* se refiere a la habilidad del agente para capturar y aplicar un dominio específico de conocimiento para resolver problemas. De esta manera los agentes pueden ser relativamente simples cuando utilizan para este propósito lógica codificada, o pueden ser relativamente sofisticados si utilizan métodos basados en Inteligencia Artificial tales como Inferencia y Aprendizaje.

Por último está la *Movilidad*. Un agente es móvil si puede moverse entre sistemas dentro de una red ejecutando acciones en beneficio de su usuario. La movilidad implica aumento en la inteligencia que debe poseer un agente. Una de las principales ventajas de los agentes móviles es que la comunicación entre el terminal local y el remoto es reducida, permitiendo al agente ir al sistema remoto y acceder a él localmente, de esta forma es posible realizar aplicaciones que ejecuten balance de carga de manera que se pueda evitar que un agente intente ingresar a un sistema con un nivel alto de congestión y salte a otro que lo pueda atender más rápido.

1.4.2 Estrategias de procesamiento

Los dos tipos de agentes más simples son los *reactivos* y los agentes *reflejo*, los cuales responden según un modelo definido por un evento, a una condición y a una acción. Los agentes reflejo no tienen modelos internos del mundo, ellos responden únicamente a estímulos externos y a la información que perciben de el ambiente [Brooks 1986].

Cuando interactúan agentes reactivos, comparten datos de bajo nivel y no conocimiento simbólico de alto nivel. Una de las características fundamentales de los agentes reactivos es que ellos son utilizados para operar en sistemas de sensores físicos tales como los que se utilizan en robótica.

Los agentes *deliberativos* o dirigidos hacia metas tienen dominio del conocimiento y capacidades de planeación necesarias para ejecutar una secuencia de acciones con el fin de lograr una meta específica. Este tipo de agentes utilizan las técnicas de razonamiento utilizadas en Inteligencia Artificial (AI).

Los agentes *colaborativos* trabajan juntos para resolver problemas. La comunicación entre agentes es un aspecto muy importante y aunque cada agente es un ente independiente, su comportamiento colectivo y colaborativo son los que los hacen útiles. Los agentes colaborativos trabajan de manera modular, es decir, el problema es dividido en secciones cada una de las cuales es atacada por aquellos con mayor dominio de conocimiento sobre ella. En un sistema de este tipo los agentes deben estar dispuestos a intercambiar información sobre sus creencias, deseos e intenciones y posiblemente deben estar dispuestos a compartir conocimiento.

1.4.2.1 Creencias, Deseos e Intenciones

Estas tres características definen otro tipo de agentes conocido como modelo BDI por sus iniciales en inglés (Beliefs, Desires, Intentions).

Las creencias representan conocimiento sobre el estado del ambiente a través del cual es posible identificar cuales aspectos que un agente considera verídicos lo son en realidad; este paso inicial es la base para los procesos de razonamiento, planeación y las acciones subsecuentes.

Las creencias juegan un papel importante en el desempeño de un agente, ya que este no tiene un conocimiento completo sobre el mundo exterior, estas lo instruyen sobre que cosas tienen mayor posibilidad de ser ciertas, sin estar exentas de albergar incertidumbre que conduzca a errores.

En cuanto a los deseos, estos resultan de la forma en que el agente desea transformar al ambiente. Cuando un agente razona teniendo en cuenta tanto el estado de su ambiente (creencias) como los deseos, debe decidir que acciones ejecutar, esto es lo que en este modelo se conoce como intenciones.

1.4.3 Funciones de procesamiento

Existe otra forma de clasificar a los agentes inteligentes de acuerdo a la función que estos desempeñan. Se pueden clasificar de la siguiente manera:

1.4.3.2 Agentes de Información

El objetivo de este tipo de agentes es recolectar, clasificar y administrar grandes volúmenes de información. El proceso de filtrado de la información se realiza de manera autónoma y sin intervención del usuario. El tipo de información requerido es especificado por el usuario a través de unos parámetros que el agente interpreta y utiliza para realizar una recolección selectiva de los datos. Este tipo de búsqueda evita que el usuario reciba cantidades de información con contenidos indeseados tal como ocurre hoy con los buscadores de Internet y pierda tiempo seleccionando cual es relevante y cual no, ya que esa actividad le corresponde al agente.

1.4.3.1 Agentes de Interfaz

Trabajan como asistentes personales para ayudar a un usuario a ejecutar tareas. Este tipo de agentes usualmente emplean aprendizaje para adaptarse a los hábitos de trabajo y preferencias del usuario.

Se han identificado cuatro formas de aprendizaje utilizadas por este tipo de agentes:

- El agente observa el comportamiento del usuario y lo imita.
- El agente puede dar consejos y ejecutar acciones en beneficio del usuario y luego aprender de la respuesta que reciba por la ejecución de las mismas.
- El agente puede recibir instrucciones directamente del usuario.
- Un agente puede aprender recibiendo consejos desde otros agentes y aprender de las experiencias de estos.

Los agentes de Interfaz trabajan únicamente con el fin de beneficiar a su usuario, pero interactúan con otros agentes para adquirir conocimiento, de esta manera es posible que las habilidades que algunos han adquirido y que les han permitido lograr sus objetivos sean transmitidas a otros para facilitarles las cosas.

1.5. SISTEMAS MULTIAGENTE

Un sistema multiagente es aquel compuesto de múltiples agentes, los cuales desempeñan tareas encomendadas por un usuario u otro agente y deben poseer las siguientes capacidades: Primero, deben ser capaces de decidir que deben hacer para cumplir con sus objetivos de diseño (característica conocida como autonomía) y segundo, de interactuar con otros agentes no solo a través del intercambio de datos, sino actuando socialmente tal como nosotros lo hacemos en nuestras vidas, por medio de la cooperación, coordinación y negociación.

Cuando dos agentes intentan comunicarse, es necesario que hablen el mismo lenguaje y que utilicen los mismos signos y símbolos. También deben disponer de una forma para intercambiar información y no pueden hablar al mismo tiempo. Todos estos aspectos son fundamentales en la comunicación dentro de un sistema multiagente, es por esta razón que FIPA ha especificado un número definido de protocolos de interacción, lenguajes de comunicación y ontologías, a través de los cuales el intercambio de información entre agentes es posible. En cierto modo, un sistema multiagente es como un sistema distribuido donde la conducta combinada de sus elementos produce un resultado logrado en conjunto.

1.6. AGENTES Y OBJETOS

Los objetos se definen como entidades computacionales que encapsulan algún estado, están dispuestos a ejecutar acciones o métodos y se comunican transmitiendo mensajes. Aunque hay similitudes obvias, también existen diferencias significantes entre agentes y objetos.

La primera es el grado en que agentes y objetos son autónomos.

Recordemos que la característica principal de la programación orientada a objetos es el principio de encapsulación. En lenguajes de programación tales como Java y C++ podemos declarar variables y métodos como privados, esto quiere decir, que solo son accesibles dentro del objeto. También podemos declarar métodos y variables como públicos, es decir, tienen la posibilidad de ser accedidos desde cualquier lugar, dando así la opción a otros objetos de poder utilizarlos.

Según lo anterior podemos concluir que un objeto es autónomo sobre su estado, pero no ejerce control sobre su comportamiento. Una vez se ha declarado un método como público, el objeto al cual pertenece no tendrá control sobre si este será ejecutado o no.

No ocurre lo mismo en el ámbito de los agentes, en este cuando un agente Y necesita que un agente X ejecute una acción A, X puede ejecutarla o no de acuerdo a los intereses que tenga. Esto significa que en un sistema multiagente existen entidades que ejecutan acciones de manera autónoma. Cada agente tiene una meta interna y decide que hacer y cuando hacerlo para llegar a ella. Es claro entonces que el control que se tiene respecto a la decisión de ejecutar o no una acción es muy diferente para los sistemas de agentes y de objetos.

Una segunda distinción que se puede hacer entre un sistema de agentes y uno de objetos está relacionado con los conceptos de reactividad, proactividad y habilidad social, en otras palabras el comportamiento autónomo. Los objetos no poseen las características necesarias para crear un sistema que posea esas condiciones, aunque se podría darle autonomía las aplicaciones orientadas a objetos, es en sí es el paradigma el que no fue diseñado para hacerlo [Bigus 2001].

CAPITULO II. MODELO DE REFERENCIA PARA EL DESARROLLO DE SERVICIOS CON ORIENTACION A AGENTES VERSION 1.0

2.1 INTRODUCCIÓN

En un ambiente empresarial el éxito o fracaso de una organización esta dictado por su capacidad de absorción tecnológica. La asimilación de nuevas tecnologías juega un papel sumamente importante en la supervivencia de las organizaciones, especialmente en mercados y economías globalizadas. Estos factores sumados a la evolución tecnológica de los últimos años han obligan a las empresas a adoptar estrategias tendientes a optimizar su desempeño, una forma de lograrlo es a través del uso racional de la información, ya que todo proceso en algún momento involucra datos, registros, cifras y etc.

Como se ha mencionado, uno de los temas más sensibles es el relacionado con la manipulación de la información. La necesidad de aplicaciones y servicios que integren departamentos de una misma organización con otros (partners), motivan la búsqueda de alternativas que permitan aprovechar estos nuevos modelos de negocio.

Las empresas no son las únicas expuestas a estos fenómenos, en cierta medida todos lo somos, porque en una época en donde la sistematización y la computación afectan nuestras actividades, de nuevo la información juega un papel crucial.

Actualmente muchas tecnologías facilitan el procesamiento de la información, bajo distintos criterios, ofreciendo infinidad de funciones, ventajas y demás, pero no tienen punto de comparación con las capacidades teóricas de la tecnología de agentes y de ahí una de las principales motivaciones para el desarrollo del presente modelo.

Anteriormente los sistemas de telecomunicaciones se limitaban a ofrecer servicios de voz, pero esta situación ha ido cambiado paulatinamente, debido en gran medida a factores comerciales, tecnológicos y económicos, que incentivan el desarrollo de servicios cada vez mas complejos, por ejemplo: servicios multimedia, de datos entre otros. Este proyecto toma por hábitat el punto en donde convergen los sistemas de telecomunicación y los servicios, investigando entre otras cosas como los agentes se podrían utilizar en estos entornos y cuales serian sus aportes.

La evolución en los dispositivos de computo personal y de los sistemas de telecomunicación móvil, han impulsado el surgimiento de nuevas tecnologías como la de agentes, dispuestas a explorar los beneficios ofrecidos por estos equipos y sistemas.

Los agentes como tecnología apenas están surgiendo. Aunque existen muchas plataformas que permiten el

desarrollo de aplicaciones con estos, ninguna ha implementado por completo las características teóricas atribuidas a un agente. Por lo general estas se concentran en aspectos particulares, por ejemplo JADE, que se especializa en agentes peer to peer o 3APL-M, que lo hace en agentes cognoscitivos.

La curva en S es un gráfico muy simple que sirve para describir el proceso de innovación en cualquier tecnología, consta de 5 fases o etapas, figura 2.1.

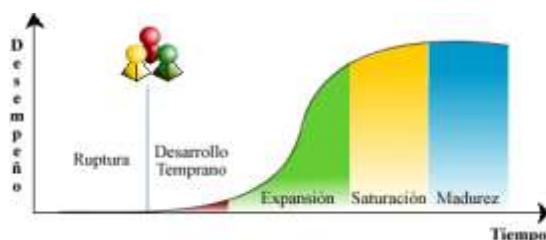


Figura 2.1. Curva en S de la innovación tecnológica.

La primera fase denominada de *ruptura*, ocurre cuando aparece un nuevo conocimiento científico en forma de teoría, método, proceso o etc, que incentiva su investigación y posterior aplicación como tecnología. La etapa de *desarrollo temprano* esta caracteriza por el refinamiento de la base conceptual que respalda a la tecnología, este perfeccionamiento permite la construcción de prototipos de prueba (innovación). En la etapa de *expansión* la comunidad de investigadores, desarrolladores, inversionistas y otros interesados, aceptan uno de los prototipos existentes como la implementación más idónea para el sistema o dispositivo, constituyéndose en el diseño dominante, por lo general este reúne las principales características de los prototipos previamente desarrollados. Al termino de esta etapa la tecnología esta lo suficientemente madura como para ofrecerse comercialmente. Cuando una tecnología se considera *madura* todas sus capacidades han sido aprovechadas y las mejoras que se le puedan realizar son de tipo parcial. La etapa de *saturación* se alcanza cuando las mejoras del sistema o dispositivo no optimizan en gran medida su desempeño, motivo por el cual podría ser considerada como obsoleta frente a otras que ofrezcan mejores prestaciones [Torres 1990].

Si se ubicara a los agentes como tecnología en la anterior figura, estos se hallarían entre el primer y segundo cuadrante, todavía en investigación. Aunque el concepto de agente es muy antiguo, es en estos últimos años en que han aparecido plataformas (prototipos) que capturan con cierto grado de exactitud las características atribuidas teóricamente a los agentes y esto gracias a los avances en inteligencia artificial, sistemas distribuidos, lenguajes de programación, Internet, entre otros tantos. Todos estos fenómenos provocaron una fisura o ruptura que permitieron al paradigma de agentes tomar forma y perfilarse como tecnología.

Debido a que la tecnología de agentes todavía se encuentra en la fase de ruptura y desarrollo temprano, no existe un consenso generalizado en la materia, por ejemplo algunos investigadores opinan que atributos como la inteligencia y la autonomía son pieza fundamental en el paradigma, mientras que otros ponderan la proactividad, delegación de tareas, reactividad, movilidad, sociabilidad o quizás otra. Esta falta de

unificación de criterios es una característica muy propia de una tecnología en desarrollo, en donde los diferentes enfoques están más orientados a nutrir y fortalecer la base conceptual que a imponer ideas. Este fenómeno es palpable en esta disciplina y de ahí la gran variedad de plataformas, definiciones de agentes, arquitecturas, propiedades, teorías, nociones, enfoques y demás. Hasta que no surja un diseño dominante que satisfaga a la comunidad de interesados el concepto de agente, sus elementos constitutivos y derivados continuaran en discusión.

Bajo este panorama se ubica nuestra propuesta de investigación, que explora la posibilidad y los mecanismos en que se puede emplear la tecnología de agentes en la prestación de servicios. La versatilidad de estos los convierte en una herramienta interesante para solventar los problemas generados a raíz de la movilidad de los dispositivos. Un aspecto clave de esta investigación es la identificación de las características propias de los agentes que favorecen el diseño e implementación de servicios.

Esta búsqueda se materializa y toma forma en el MR-DSOA o Modelo de Referencia para el Desarrollo de Servicios con Orientación a Agentes, que consideran tanto los aspectos técnicos como de ingeniería del software en el diseño de servicios para entornos móviles.

2.2 OBJETIVOS Y MOTIVACIONES

La tecnología de agentes ofrece muchas ventajas que pueden ser útiles en el diseño e implementación de servicios para dispositivos móviles. Los beneficios aportados por esta nueva tecnología permitirán potenciar y mejorar los servicios en ambientes inalámbricos, ya que las características intrínsecas de los agentes ayudarían a solventar en gran medida los problemas producidos por la movilidad de los usuarios, la marcada heterogeneidad en los dispositivos de cómputo personales y los problemas propios de una red de acceso sin hilos. El panorama planteado por esta tecnología es prometedor y ofrece un nuevo mundo de posibilidades que incentiva su exploración.

El desarrollo de un servicio siempre va ligado a una especificación conceptual que relaciona sus objetivos o requerimientos con un arreglo de entidades software, esto es lo que comúnmente se conoce como modelamiento y en el caso de los servicios con orientación a agentes se hace necesario un modelo que conjugue y concentre todas las corrientes conceptuales que convergen en el diseño, facilitando así su idealización y posterior materialización.

Como se puede apreciar las posibilidades son muchas al igual que las motivaciones pero en particular este proyecto tiene por único objetivo proponer un modelo que permita el desarrollo de servicios para dispositivos móviles empleando la tecnología de agentes.

2.3 CONTEXTO Y ALCANCES DEL MODELO

Si se analiza detalladamente los objetivos de este modelo se pueden identificar cuatro elementos básicos que deben ser considerados durante su construcción y estos son: los servicios, la orientación a agentes, la ingeniería del software y la computación móvil, todos estos tienen que incluirse de alguna forma en el modelo y cohesionarse a fin de formar una estructura homogénea y conceptualmente coherente.

Para cubrir un área temática tan extensa se hace necesario nutrir conceptualmente al modelo de muchas tendencias y especificaciones, por tal motivo se han seleccionado las investigaciones más relevantes y que en su debido momento influenciaron profundamente el desarrollo en sus respectivas disciplinas, aunque no guarden relación aparente con los objetivos del modelo ni mucho menos entre sí, en consecuencia MR-DSOA rescata las principales características de cada una de estas y las amalgama consistentemente en una unidad. El contexto del modelo está determinado por el ámbito de las investigaciones en que se sustenta, por lo tanto a continuación se mencionará las investigaciones que dieron origen a este modelo.

En cuanto a la parte de servicios el modelo reposa sobre dos arquitecturas abiertas, una de ellas es la arquitectura de telecomunicaciones TINA (Telecommunication Information Networking Architecture) enfocada en los servicios y en la estructura de los sistemas de telecomunicaciones [Chapman 1995], y RM-ODP (Reference Model of Open Distributed Processing) centrada en el desarrollo de aplicaciones distribuidas [ITU-T 1995]. El hecho de utilizar como referencia estas arquitecturas provee al modelo de un conjunto de conceptos estandarizados, ampliamente difundidos y conocidos a nivel mundial.

RM-ODP ofrece un modelo de referencia para el diseño de sistemas distribuidos, mientras que TINA está enfocado en la definición de una arquitectura abierta y genérica para sistemas de telecomunicaciones. Particularmente este modelo utiliza los conceptos de la arquitectura computacional de TINA, que entre otras cosas habla de la composición del DPE (Distributed Processing Environment) y como aprovecharlo en el diseño y desarrollo de servicios para sistemas de telecomunicaciones, nociones muy necesarias para darle al modelo MR-DSOA su orientación hacia servicios telemáticos.

Este modelo también emplea el estilo arquitectónico planteado por RM-ODP, en el que se descompone a las aplicaciones distribuidas en vistas, característica replicada por MR-DSOA en la especificación de servicios. Dado que estas arquitecturas están orientadas a objetos, este modelo solamente emplea los conceptos y principios como base para generar unos nuevos conformes a las capacidades de los agentes.

Uno de los aspectos de mayor importancia en este modelo es lo concerniente a la orientación a agentes, es por ello que se emplearon conceptos de FIPA (Foundation for Intelligent Physical Agents) combinados con los de MAFS (Mobile Agent Facility Specification) para explicar la forma en que se podría utilizar el paradigma de agentes en el diseño de servicios para sistemas de telecomunicaciones [OMG 2000].

La arquitectura abstracta propuesta por FIPA es empleada en conjunción con la de TINA para describir las características del ambiente que soporta a los servicios, adicionalmente se incluyen algunos conceptos

presentados en MAFS como soporte teórico. Estas nociones proporcionan al modelo una perspectiva útil para el diseño de servicios con orientación a agentes y allanan el camino para su integración a los sistemas de telecomunicaciones.

Fundamentar este modelo en investigaciones tan completas y serias como estas robustecen al modelo, porque aprovecha las estructuras desarrolladas por varios grupos expertos en telecomunicaciones y ciencias de la computación por más de una década.

El modelo se enfoca en tres aspectos básicos de la ingeniería del software: los métodos, las herramientas y los procedimientos. Los métodos indican como construir técnicamente el software, o sea, la planificación, el análisis de los requerimientos, el diseño de estructuras de datos, arquitectura, procedimientos algorítmicos, codificación y pruebas. Las herramientas de la ingeniería del software son aplicaciones encargadas de dar soporte a los métodos, el modelo concibe el uso de estas para facilitar el diseño y automatizar el proceso de desarrollo. Los procedimientos son las entidades encargadas de unir a los métodos con las herramientas, estos definen la secuencia en que se deben aplicar los métodos [Pressman 1993]. Aparte de conciliar todos estos elementos el modelo emplea una metodología propia del paradigma de agentes, que sirve para modelar la interacción entre los agentes, el entorno y el sistema mismo.

Teniendo en cuenta lo anteriormente expuesto, se puede afirmar que el modelo esta basado en una arquitectura estándar de telecomunicaciones e incorpora una metodología de desarrollo propia de agentes, combinado estos dos elementos con la orientación a agentes se obtiene un diseño que considera tanto el entorno en que se ejecuta los servicios como su implementación con agentes.

Los entornos móviles suponen un reto en el diseño de servicios, debido principalmente a que son medios de libre acceso en donde los diferentes dispositivos comparten un mismo espectro de frecuencias, hecho que eleva la probabilidad de perturbaciones durante las comunicaciones, a nivel de diseño se pueden desarrollar aplicaciones que minimicen el efecto de estos disturbios en la prestación de servicios, por ejemplo aplicaciones que no consuman demasiado ancho de banda o autónomas que eviten constantes interacciones entre los dispositivos y etc.

El deseo de este modelo es generar servicios con las siguientes características:

- Abiertos: Seguir el modelo fundamentalmente conduce a la especificación de un servicio telemático de arquitectura abierta.
- Flexible: Soporte a la evolución natural que debe experimentar el servicio. Flexibilidad y adaptabilidad para interactuar con nuevos componentes.
- Modularidad: Debido a que el modelo estructura los servicios a modo de componentes o módulos independientes, se puede afirmar que su aplicación conduce a una arquitectura modular, hecho que

facilita: la integración de nuevos componentes, la adición de funcionalidad entre otros.

- **Heterogeneidad:** En un sistema de telecomunicaciones en el que convergen tantas tecnologías no hay garantía de que los componentes estén desarrollados homogéneamente; por ejemplo diferentes lenguajes de programación, dispositivos, tecnologías, sistemas operativo o etc. Por tal motivo el modelo considera que pueden existir varios dominios tecnológicos que afectan el diseño de los servicios.
- **Integridad:** Como los servicios son aplicaciones que se ejecutan en sistemas de telecomunicaciones potencialmente extensos en el que muchas personas y roles de negocio intervienen, el modelo contempla esta y otras situaciones generadas por la distribución de componentes y el carácter público que deben tener estos, proponiendo alternativas y requerimientos de seguridad desde el diseño mismo.
- **Generalidad:** El modelo se puede combinar con otros conceptos de ingeniería del software para diseñar agentes genéricos, que podrían utilizarse en otros servicios, favoreciendo así la reutilización.
- **Escalabilidad:** En algún momento durante el ciclo de vida de un servicio se hace necesario realizar cambios en el funcionamiento, mejoras técnicas que eleven el desempeño, extensiones en la funcionalidad y etc. La tecnología de agentes permiten desagregar los componentes de un servicio, minimizando el efecto de modificar o incluir nuevas entidades.
- **Movilidad:** El modelo tiene en cuenta los avances en las redes de telecomunicación y especialmente en lo concierne a redes inalámbricas, donde la movilidad es piedra angular, por tal motivo se consideran dos clases de esta. A nivel físico, movilidad de usuarios, fuentes de información y nodos de procesamiento (computadoras de bolsillo, portátiles, smart phones y etc), y a nivel lógico, movilidad de procesos, aplicaciones o datos.
- **Autonomía:** Una de las razones de implementar servicios telemáticos basado en agentes es el cierto grado de autonomía ofrecido por estos, propiedad que les permite ejecutar tareas sin una constante intervención o supervisión de terceros, el modelo también considera esta propiedad en el diseño de servicios autónomos.
- **Facilidad en las pruebas:** El modelo esta diseñado considerando que las entidades software que componen un servicio pueden ejecutarse en diferentes nodos o terminales de una red, lo cual significa que los fallos generados por el servicio no dependen exclusivamente de su implementación, por esta razón el ingeniero de pruebas o arquitecto debe estimar la posibilidad de que los errores se deban a la distribución de los componentes o por la federación de recursos software o hardware, situaciones comunes en cualquier sistema de telecomunicaciones.
- **Integrable:** Facilita la incorporación de nuevos subsistemas, componentes y recursos al servicio sin grandes cambios.

Muchas de estas características no son producto directo del modelo, pero si de sus afluentes conceptuales MAFS, FIPA, RM-ODP, TINA, ANSA (Advanced Networked Systems Architecture), ROSA (RACE Open service Architecture) entre otras.

Un aspecto interesante de este modelo es que considera las tecnologías empleadas en los sistemas de telecomunicación, a primera vista esto parecería ir en contradicción con las tendencias de la ingeniería del software, que buscan todo lo contrario, mantener la separación entre el diseño y la tecnología, a nivel de servicios ignorar esta brecha puede ser un grave error, ya que en estos ambientes intervienen factores más allá de lo tecnológico, que impiden que las aplicaciones y los equipos sean interoperables y puedan establecer asociaciones (federación, internetworking y otras), actualmente existen varias tecnologías que lo hacen, pero debido a sus elevados costos no son prioridad para los operadores, por esta razón el modelo asume que pueden existir distintos tipos de tecnologías que según sea el caso afectan de una forma u otra el funcionamiento de los servicios, sin entrar en detalles sobre el funcionamiento de estas tecnologías.

2.4 CONCEPTOS Y PRINCIPIOS

2.4.1 Sistema

Un sistema es un conglomerado de elementos físicos o abstractos que interactúan entre sí por algún motivo. La complejidad de un sistema varía en función de su naturaleza, las características de sus componentes y las formas en que estos se relacionan.

2.4.2 Modelo

Un modelo es una representación simplificada de un sistema que tiene por propósito facilitar su análisis y comprensión; se presentan en distintas formas según su finalidad, por ejemplo como: arquetipo, abstracción, esquema, caso de estudio u otra, lo importante es que pueda replicarse bajo distintas circunstancias.

Un aspecto importante en la construcción de un modelo es la selección de los elementos que va a considerar, ya que los componentes clasificados como no esenciales son excluidos y su efecto considerado como nulo, como si no existieran en el micro universo creado por el modelo.

Las consideraciones iniciales de un modelo son muy importantes por que hablan de su contexto, campo de acción y enfoque, en el caso de que se consideren demasiados elementos se obtendrá un modelo muy aproximado, pero igual de complejo que la misma realidad a la cual pretende simplificar; si por el contrario, considera muy pocos, su similitud con el sistema será muy tenue. Por esta razón en la construcción de un modelo se deben analizar detenidamente sus objetivos y a partir de estos determinar las consideraciones iniciales que deben hacerse, por ejemplo si un modelo es descriptivo, este debe centrarse en capturar la forma

en que se relacionan los elementos en el sistema o si es de tipo cuantitativo, importa más la precisión en los resultados que cualquier otro factor. En resumen la cantidad de elementos considerados por un modelo definen su grado de complejidad y precisión.

Si bien un modelo responde a una necesidad particular de conocimiento, este no debe sesgarse a solucionar exclusivamente un problema, es decir; debe ser lo suficientemente genérico como para que se pueda emplear en varias situaciones siempre y cuando estén contempladas por su contexto.

2.4.3 Arquitectura

Una arquitectura software es el conjunto de reglas, conceptos, principios y directivas definidas durante la fase de diseño de un sistema y empleadas rigurosamente en su construcción. Cuando se analiza la estructura de un desarrollo software emerge un estilo arquitectónico, que esboza la forma en que estas decisiones y consideraciones se relacionan e interactúan al interior del sistema, estas pautas dependen principalmente de: las funciones, la visión de los arquitectos y de las obligaciones del sistema.

2.4.4 Servicio

En el ámbito de las telecomunicaciones han surgido varias acepciones del término servicio, debido principalmente a la falta de unificación de criterios y a los diferentes enfoques dados al tema. Motivo por el cual se discutirán varias definiciones, con el objetivo de mostrar como ha evolucionado hasta involucrar una gran cantidad de elementos.

Mucho autores definen a un servicio como un conjunto de funciones con algún valor, ofrecidas por un proveedor a un cliente. Esta concepción es muy general para los alcances del modelo, básicamente por su simpleza y ambigüedad, en ella se distinguen dos actores: los usuarios y los proveedores, con roles bien definidos: consumir y abastecer funcionalidad.

Particularmente en el ámbito de las telecomunicaciones, un servicio es visto como un conjunto de funciones ofrecidas por un proveedor de servicios a un grupo de usuarios a través de una red informática y por la que pueden obtener una retribución económica. Esta definición es un poco más completa que la anterior puesto que considera a otros actores en la prestación de un servicio e incluye la parte comercial, aspecto de gran importancia en el desarrollo del sector.

El proyecto ROSA (Arquitectura Abierta de Servicios RACE) fue una de las primeras investigaciones enfocadas al desarrollo de servicios para redes de telecomunicación, en esta el concepto de servicio es descompuesto en dos partes según su contexto, como se muestra en la figura 2.2. Uno de ellos es el contexto empresarial, bajo el cual un servicio es visto como un conjunto significativo de funcionalidades provistas por una red a todos

los que la utilizan, ya sean clientes, usuarios finales, proveedores de red y de servicios, cada uno observando una faceta distinta del servicio. El otro componente de la definición proviene del contexto computacional, en el cual un servicio no es más que la invocación de una función ofrecida por un objeto [ROSA 1992].

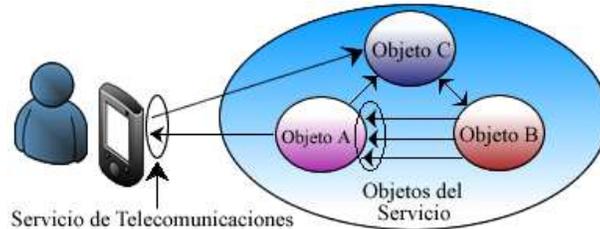


Figura 2.2. Servicios desde la perspectiva ROSA.

La definición más acorde con el contexto del modelo es la provista por TINA; “Un servicio es un conjunto apreciable de funcionalidades ofrecidas por un sistema a todos los roles de negocio involucrados, con la salvedad de que cada uno observa una perspectiva diferente del mismo”.

Ejemplos de servicios:

- Servicios de gestión: Servicios responsables de gestionar los recursos software y hardware de una red de telecomunicaciones, incluye funciones de configuración, facturación, seguridad, así como de gestión del ciclo de vida de los servicios, de sus instancias y de los usuarios.
- Servicios de telecomunicación: Servicios orientados al transporte de información (bits) entre terminales en una red de telecomunicaciones. Estos servicios son responsables del establecimiento de conexiones y del procesamiento de la información relacionada con esta. Una red de telecomunicaciones es transparente a la información que transporta.
- Servicios telemáticos: Servicios ofrecidos por una red de telecomunicaciones que involucran: componentes informáticos (ciencias de la computación), elementos de telecomunicación (redes) y por supuesto información, de ahí que sean asociados y tratados como aplicaciones software.

2.4.5 RM-ODP

El conjunto de especificaciones RM-ODP ofrece un modelo de referencia a modo de arquitectura abierta y genérica para el desarrollo de sistemas distribuidos. Uno de los conceptos más importantes y eje central de la especificación RM-ODP es la forma en que descompone a los sistemas distribuidos en partes o vistas. Cada vista describe un aspecto particular de la aplicación a partir de un conjunto de propiedades bien definidas, facilitando el diseño y reduciendo la cantidad de información a considerar en cada parte.

RM-ODP se compone de las siguientes especificaciones:

- ITU-T Rec. X.901 | ISO/IEC 10746-1: Visión general.
- ITU-T Rec. X.902 | ISO/IEC 10746-2: Marco lógico.
- ITU-T Rec. X.903 | ISO/IEC 10746-3: Arquitectura.
- ITU-T Rec. X.904 | ISO/IEC 10746-4: Significado de la arquitectura.

2.4.6 Arquitectura TINA

La arquitectura TINA reúne una serie de conceptos y principios desarrollados durante años de investigación, que involucran diversas disciplinas como: sistemas distribuidos, gestión de redes, sistemas de transmisión entre otras. Esta orientación fue la que motivó su consideración como base conceptual para la construcción de este modelo.

Debido a su dificultad y extensión, el consorcio TINA optó por dividir esta arquitectura en cuatro partes que a su vez, se subdividen en otras más pequeñas. El núcleo de la arquitectura TINA define el conjunto inicial de conceptos y principios comunes a cada sub-arquitectura y es en estas en donde se desarrollan a profundidad estas nociones. A continuación se presenta la descomposición del núcleo y el ámbito de cada una de las partes.

2.4.6.1 Descomposición de la arquitectura TINA

La arquitectura TINA se descompone en cuatro grandes subconjuntos:

- Arquitectura del servicio: Define el conjunto de conceptos y principios para el diseño, especificación, implementación, y gestión de servicios de telecomunicación [Abarca 1997a].
- Arquitectura de red: Define el conjunto de conceptos y principios para el diseño, especificación, implementación, y gestión de redes de transporte [Abarca 1997b].
- Arquitectura de gestión: Define el conjunto de conceptos y principios para el diseño, especificación e implementación de sistemas software que gestionan: servicios, recursos, aplicaciones y tecnologías base (underlying technology).
- Arquitectura computacional: Define un conjunto de conceptos y principios para diseñar y construir aplicaciones distribuidas, además, del ambiente que las soporta [TINA-C 1996].



Figura 2.3. Composición de la arquitectura TINA.

Para asegurar que cada arquitectura provee un conjunto coherente de conceptos y principios, se establecen las siguientes relaciones de dependencia entre arquitecturas. Figura 2.4.

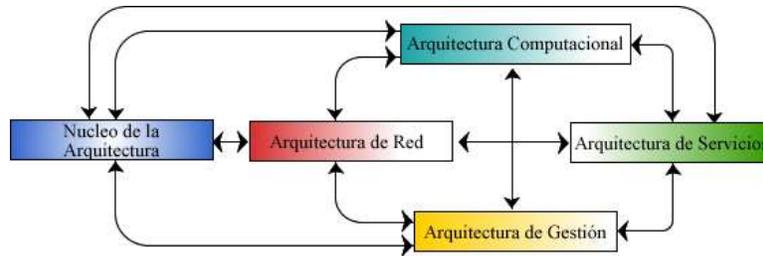


Figura 2.4. Dependencia entre arquitecturas.

2.4.7 FIPA - Foundation for intelligent physical agents

FIPA es una organización sin ánimo de lucro que desarrolla estándares para la interoperabilidad entre agentes y sistemas desarrollados heterogéneamente. Su arquitectura de referencia se descompone como se muestra en la figura 2.5.

La especificación de la arquitectura abstracta FIPA contiene una descripción formal de las características más relevantes de un sistema basado en agentes, en esta se han identificado varias entidades funcionales y se lea han asignado a través de lenguaje natural responsabilidades y roles caracterizando sus relaciones, comportamientos y demás [FIPA 2002a]. El modelo MR-DSOA utiliza esta arquitectura como base para el diseño y desarrollo de servicios.

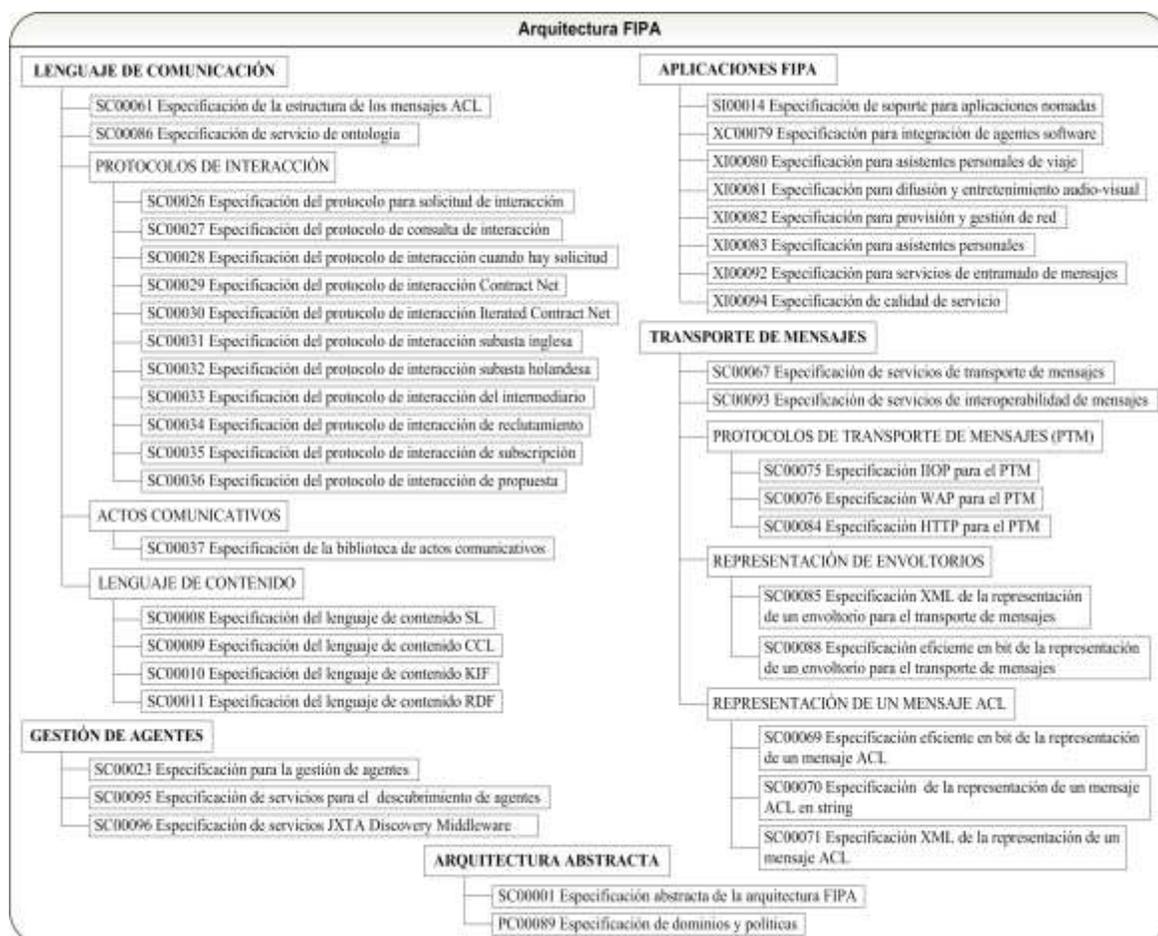


Figura 2.5. Arquitectura FIPA.

2.5 ESTRUCTURA DEL MODELO

Los conceptos y principios contenidos en este modelo se pueden clasificar en tres tipos según su finalidad (básicos internos, básicos externos y opcionales).

- Los conceptos y principios *opcionales* son todos aquellos elementos de carácter variable, empleados para describir algún aspecto particular del modelo, no es obligatorio su uso ya que pueden remplazarse por otras alternativas equivalentes, en lugar de las sugeridas por el modelo. Un claro ejemplo de esta situación son los diagramas OMT, GMDO y UML empleados aquí.
- Los *básicos externos* se refieren al conjunto adicional de conceptos y principios que deberán ser adheridos en la especificación de cualquier desarrollo (producto informático) para facilitar su diseño.
- Los conceptos y principios *básicos internos* son todos aquellos elementos provistos por el modelo para el diseño, especificación e implementación de un servicio y deberán seguirse cada vez que se aplique el modelo.

El modelo se fundamenta en dos principios o consideraciones:

- Se pueden aplicar técnicas de desarrollo propias del paradigma de agentes en la construcción de servicios.
- Se considera a los sistemas de telecomunicación como sistemas distribuidos potencialmente extensos, por lo cual es admisible emplear técnicas de computación distribuida.

Este modelo provee el conjunto de conceptos y principios básicos para el diseño y especificación de servicios telemáticos, el abanico de posibilidades es bastante amplio, ya que es lo suficientemente genérico como para emplearse en el desarrollo de diferentes tipos de servicios.

2.6 SISTEMA DE TELECOMUNICACIONES MR-DSOA

Debido a que este modelo propone soluciones para el desarrollo de servicios se hace necesario definir una entidad conceptual que los soporte, o sea una abstracción genérica de las características funcionales de un sistema de telecomunicaciones. Esta abstracción captura las principales características de un sistema de telecomunicación y se las ofrece a los servicios en forma de funciones y operaciones, ocultando de este modo los complejos mecanismos que involucran el manejo de equipos de red y software empotrado de muy bajo nivel.

Plasmar en el papel (diseño) todas las interacciones que toman parte en la prestación de un servicio resultaría muy confuso, atiborrando de información innecesaria las especificaciones, por este motivo MR-DSOA se ciñe a la división por capas propuesta por TINA, quizás uno de los beneficios mas significativos de esta decisión es que los servicios se pueden considerar como aplicaciones estándar y a estos aplicarles técnicas de modelamiento tradicionales. Esta estratificación separa a la tecnología del software, en este caso a los servicios del sistema de telecomunicaciones.

En esta sección se describirán las características del ambiente que contiene a los servicios MR-DSOA y mostrará los alcances del modelo en términos de productos software.

2.6.1 Características de un sistema de telecomunicaciones MR-DSOA

Un sistema de telecomunicaciones MR-DSOA es una colección de recursos hardware y software disponibles en forma de servicios a un conjunto de roles de negocio, ya sea directamente, sin intermediarios o indirectamente; a través de otros sistemas. En cuanto a su extensión, los sistemas de telecomunicaciones no están confinados a un dominio en particular, aislados de los demás, por el contrario, se extiende más allá de sus fronteras administrativas e interactúan con otros gracias a que su estructura se fundamenta en la federación de recursos.

En la figura 2.6 se puede observar la composición de un sistema de telecomunicaciones desde la perspectiva MR-DSOA, este consiste de dos capas, una de servicios que alberga a los agentes y otra de red, conformada por los equipos y dispositivos de telecomunicaciones, incluyendo su respectivo software, a este conglomerado se le denomina dominio administrativo de negocio y puede interactuar simultáneamente con diferentes roles de negocio, en este caso usuarios, proveedores externos de servicios, proveedores de conectividad y otros sistemas.

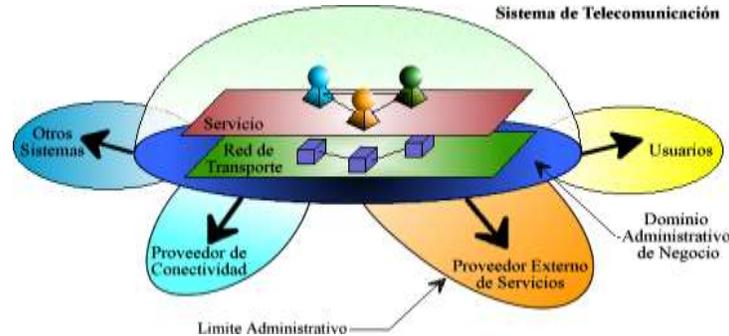


Figura 2.6. Composición de un sistema de telecomunicaciones.

2.6.2 Arquitectura genérica de un sistema de telecomunicaciones

Debido a que este modelo (MR-DSOA) está basado en las especificaciones de TINA, hereda una serie de elementos definidos por esta, entre ellos la arquitectura software para sistemas de telecomunicaciones. Esta arquitectura es aprovechada por este modelo para estructurar la composición de sus servicios y extraer las restricciones que deben tenerse en cuenta durante su diseño.

En la figura 2.7 se aprecia la estructura computacional de un sistema de telecomunicaciones. En la parte inferior de la ilustración, está la capa de recursos hardware; que incluye todos los recursos físicos, tales como: procesadores, memorias, dispositivos, equipos de comunicación entre otros. Sobre esta se encuentra el NCCE (Native Computing and Communications Environment) que incluye todo el software de comunicaciones y de computación nativa de la capa de recursos de red, por ejemplo sistemas operativos, software empujado y demás herramientas de soporte típicas de un sistema de telecomunicaciones. Luego está el PCA o Plataforma Conceptual de Agentes que provee el ambiente necesario para la ejecución de los agentes y por último está la capa de aplicación donde se encuentran los agentes que implementan los servicios.

En esta figura también se puede observar la separación existente entre los servicios, software que implementa las funcionalidades provistas por el sistema y la plataforma de conceptual de agentes, que es la encargada de soportar su ejecución.

Un sistema de telecomunicaciones puede interoperar con otros, a diferentes niveles, es decir, que la capa de aplicación, PCA, NCCE y de recursos hardware puede relacionarse con sus contrapartes en otros dominios administrativos de negocio.

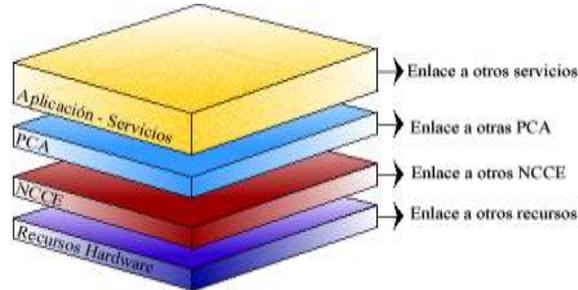


Figura 2.7 Estructura básica del software de telecomunicaciones.

La anterior figura muestra a un sistema de telecomunicaciones como una estructura homogénea, pero en realidad es todo lo contrario, ya que se encuentra segmentada en nodos, que interactúan entre sí y cada uno implementado con tecnologías hardware y software diferentes.

La plataforma conceptual de agentes (PCA) cumple la función de un aislante, ya que separa a los servicios de la tecnología que los soporta, por esta razón es uno de los componentes más importantes para el modelo, ya que permite desagregar el diseño de un servicio de la tecnología con que se implementará, permitiendo aunque sea a nivel teórico portabilidad de servicios.

Uno de los aspectos más importantes en el diseño de cualquier tipo de servicio es el manejo de los complejos mecanismos de interacción entre los componentes que integran la aplicación, aunque estos se encuentren dispersos en diferentes nodos de una red, este hecho debe pasar desapercibido al momento de diseñarse. El ocultamiento de los efectos producidos por la distribución de componentes debe ser asumido por la PCA.

En enmascaramientos básicos ofrecidos por la PCA:

- **Ubicación transparente:** Oculta la ubicación de los diferentes componentes de un servicio ante los demás, proveyendo mecanismos que facilitan su localización aunque se encuentren dispersos en diferentes terminales.
- **Reubicación transparente:** Oculta la reubicación física de un componente de otros asociados a él, este tipo de enmascaramiento permite que el servicio continúe operando aunque uno o varios componentes cambian de posición o son reemplazados.
- **Replicación transparente:** Oculta la replicación de uno o varios componentes de una aplicación sin que las entidades con las que interactúa lo noten, este mecanismo es empleado para elevar la disponibilidad y desempeño de un servicio.

-
- Acceso transparente: Oculta los detalles relacionados con las operaciones de ensamblaje y desensamblaje de mensajes, así como de las conversiones necesarias para que la representación de los datos sean consistentes en las distintas PCA, permitiendo una interacción libre y transparente entre ellas.
 - Migración transparente: En este tipo de enmascaramiento se realizan dos operaciones, una de reubicación que se encarga de ocultar los cambios de ubicación de un componente durante su ejecución y la segunda se encarga de mantener el estado interno durante su migración. El cambio de ubicación no debe afectar las comunicaciones ni la dinámica del servicio.
 - Tolerancia a fallos: En el caso de que ocurra un error inesperado durante la ejecución de algún proceso asociado a un agente, posibilita la activación de una o varias acciones de recuperación minimizando así el efecto de un error sobre la prestación de un servicio. Diferentes clases de fallos se pueden identificar; entre los mas comunes para un sistema de telecomunicaciones tenemos; fallos de comunicación, de seguridad, en los nombres y etc. Dependiendo de la factibilidad o probabilidad de ocurrencia de estos, el diseñador debe incluir cierto nivel de tolerancia a fallos en sus productos mejorando así el desempeño de los servicios.
 - Transparencia en las transacciones: Oculta los procesos desencadenados al realizar un transacción entre componentes, con el fin de mantener la consistencia de los datos o el estado interno de los agentes.
 - Persistencia transparente: Le oculta a un componente la activación y desactivación de otras entidades o de si mismo, es usualmente empleado cuando el sistema es incapaz de proveer continuamente capacidades de procesamiento, almacenamiento y recursos de comunicación.
 - Gestionable: El PCA debe permitir el monitoreo, control y gestión sobre los recursos del sistema, garantizando por ejemplo un determinada calidad de servicio (QoS), definición de políticas de acceso, configuración de equipos y componentes entre otras.
 - Enmascaramiento: El ocultamiento de los detalles y mecanismos empleados para solventar los problemas causados por la distribución, es uno de los principales requerimientos que surgen al desarrollar servicios para sistemas de telecomunicación.
 - Seguridad: El PCA tiene que garantizar la protección de los componentes del sistema contra accesos no autorizados. Los requerimientos de seguridad para este tipo de aplicaciones son de vital importancia debido a la gran cantidad de equipos o nodos que pueden acceder a una red inalámbrica.
 - Federación: Propiedad que permite combinar sistemas de diferentes administraciones o dominios para alcanzar un objetivo.

Características de un sistema de telecomunicaciones:

- **Dispersión:** Los componentes de un sistema de telecomunicaciones pueden encontrarse dispersos y las interacciones entre ellos ser de dos tipos: locales o remotas.
- **Concurrencia:** Cualquier componente de un sistema de telecomunicaciones puede ejecutarse simultáneamente con otro (paralelo).
- **Carencia de estado global:** El estado total del sistema no puede determinarse con precisión.
- **Fallos parciales:** En un sistema de telecomunicaciones un componente puede fallar en forma independiente de los demás.
- **Asincronismo:** Las operaciones involucradas en el procesamiento de la información y comunicación no están sincronizadas por un reloj maestro o único, por lo tanto no se puede asumir cambios instantáneos en el estado del sistema.
- **Heterogeneidad:** Diferencias en dispositivos móviles, equipos de telecomunicaciones y aplicaciones.

El diseño e implementación del NCCE está fuera del alcance de este modelo pero vale la pena notar tres puntos sobre su relación con la PCA.

- Si la plataforma conceptual de agentes requiere el uso de protocolos especializados, tales deben ser proporcionados por el software en el NCCE, de no ser así, deberán diseñarse e implementarse en el PCA.
- El software basado en agentes puede coexistir en un mismo nodo con otro tipo de aplicaciones.
- Algunos nodos pueden contar con recursos hardware especiales, tales como tarjetas procesadoras de video, dispositivos de grabación y etc. El software para estos dispositivos no es considerado parte del NCCE por la simple razón de que no ejecutan operaciones computacionales de propósito general o funciones de comunicaciones. Este software debe ser considerado como parte de la capa de servicios.

2.6.3 Sistemas de telecomunicación como ambientes distribuidos

Una descripción más detallada de la composición de un sistema de telecomunicaciones se obtiene cuando se le considera como un sistema distribuido, formado por un conjunto de nodos enlazados y cooperantes en la prestación de los servicios. Figura 2.8.

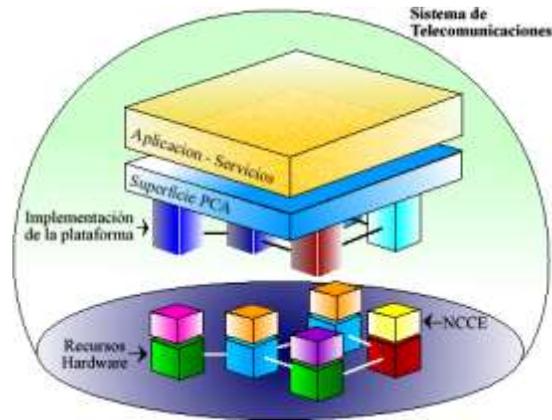


Figura 2.8. Descomposición en nodos de un sistema de telecomunicaciones

En la anterior figura se puede observar que el NCCE y capa de recursos hardware en un sistema de telecomunicaciones se encuentran divididas en nodos, cada uno con sus propios equipos de comunicación y sus respectivas aplicaciones de computación nativa (NCCE), por lo tanto unas implementaciones serán más completas y complejas que otras, suponiendo variaciones en la funcionalidad. Esta diversidad en equipos (recursos hardware) y aplicaciones (NCCE), provoca que algunos nodos posean mayores capacidades que otros, debido principalmente a dos razones: de tipo técnico (diversidad de tecnologías) o comerciales (diferentes proveedores).

La figura 2.8 también muestra como las diferentes implementaciones de la Plataforma Conceptual de Agentes pueden coexistir en diferentes nodos. Todas esas implementaciones, en principio, deberán presentarse ante los servicios como un “superficie PCA”, funcionalmente homogénea.

Los nodos que componen el sistema de telecomunicaciones pueden pertenecer a diferentes administraciones o roles de negocio, por ejemplo clientes, proveedores de servicio, operadores de red y etc.

En un mercado donde podría haber diferentes implementaciones de la PCA, se requieren interfaces inter-PCA que permitan interoperabilidad e internetworking entre estas, a fin de que los servicios vean a la plataforma conceptual de agentes como una superficie homogénea.

Los nodos en un sistema de telecomunicaciones están clasificados en dos categorías: *nodos PCA* y *no PCA*. Los nodos pertenecientes a la primera categoría hospedan alguna implementación software de la PCA, mientras que los otros no lo hacen. En general se asume que los servicios se ejecutan sobre una implementación de la Plataforma Conceptual de Agentes. Sin embargo se pueden desarrollar servicios sin agentes que accedan a la funcionalidad provista por estos a través de interfaces inter-PCA. Los nodos no PCA incluyen tanto a dispositivos sin capacidades de procesamiento, tales como teléfonos celulares, hasta terminales multipropósito, en estos el software PCA no se ha instalado por infinidad de motivos, por ejemplo: decisiones de diseño de la red, sistemas heredados, capacidades físicas y etc.

Para finalizar, la figura 2.9. Muestra la estructura completa de un sistema de telecomunicaciones desde la perspectiva MR-DSOA, en esta se pueden observar las diferentes configuraciones de los nodos, sus formas de interconexión, las diversas implementaciones de la Plataforma Conceptual de Agentes y la posible distribución de los componentes de un servicio.

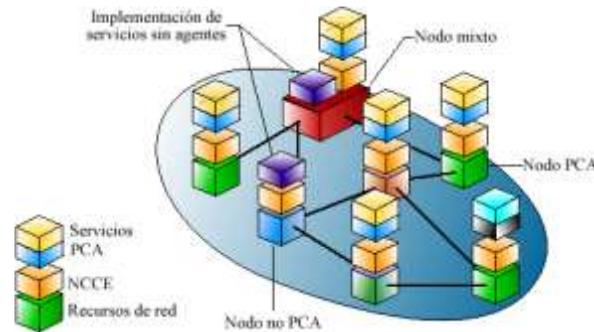


Figura 2.9. Vista completa de un sistema MR-DSOA

2.6.4 Redes y servicios de Telecomunicaciones

La anterior discusión se enfocó en la descripción del ambiente que soporta la ejecución de los servicios, pero a nivel de telecomunicaciones esta descomposición en capas estaría incompleta si no considera a las redes de transporte.

Las redes de transporte son las entidades encargadas de proporcionar los mecanismos para el transporte de la información y su respectivo enrutamiento hasta su destino, para un sistema de telecomunicación no son mas que otro recurso, el cual pueden ser usado y gestionado por los servicios en la capa de aplicación. La figura 2.10 muestra la adición de la red de transporte al ambiente básico.

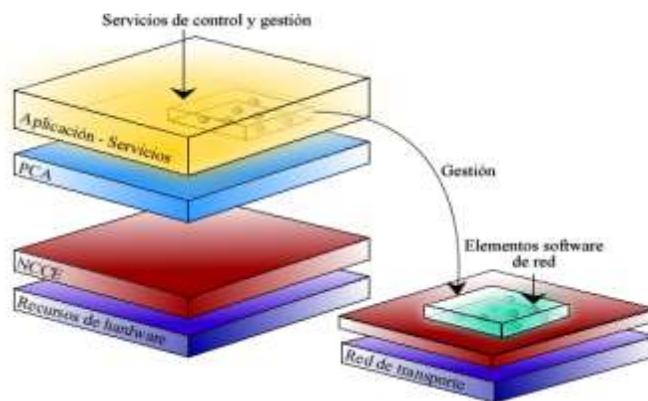


Figura 2.10. Red de transporte en el ambiente MR-DSOA

Las aplicaciones de gestión y administración del sistema deben ser ofrecidas como servicios y estas afectan directamente el funcionamiento de los equipos, recursos hardware, software y demás, en general pueden cambiar el comportamiento de todo el sistema de telecomunicaciones.

2.7 DESCRIPCIÓN DETALLADA DEL MODELO

Un modelo tiene su origen en una necesidad de conocimiento, cuando se trata de un objeto físico, como por ejemplo un edificio, un avión o una maquina, se puede construir un modelo que muestre con cierto grado de detalle las características del original pero a menor escala. Sin embargo, si lo que se quiere construir es algo intangible como un servicio o una aplicación, la situación cambia, ya que el modelo debe ser capaz de capturar la información que transforma el software, las funciones y comportamientos que permiten estas transformaciones.

En cada una de las fases del desarrollo de un servicio se crean modelos que describen distintos aspectos de la misma aplicación, por ejemplo: el intercambio de información entre agentes, sus relaciones y etc. Para ello se puede utilizar notaciones grafica basada en iconos o lenguaje natural. El modelo MR-DSOA tiene por finalidad ayudar a entender la información, las funciones y comportamientos del sistema, además de servir de guía en el diseño de servicios, proporcionando una representación del software a implementar.

La especificación completa de un sistema no trivial implica el análisis de una gran cantidad de información, intentar capturar todos los aspectos del diseño en una sola especificación es poco práctico y complicado, es por ello que MR-DSOA establece una serie de modelos que capturan distintas porciones del sistema.

Los problemas planteados durante el diseño de un servicio pueden ser tan complejos que solucionarlos de una vez resultaría realmente difícil. Por esta razón este modelo emplea el concepto de descomposición planteado por RM-ODP, en el que un sistema distribuido y todos sus componentes (servicios, PCA, NCCE y recursos hardware) pueden dividirse en partes más pequeñas y por ende manejables, favoreciendo su comprensión y análisis, la suma de cada una de estas vistas resultan en la especificación completa del servicio.



Figura 2.11. Descomposición por vistas MR-DSOA.

Cada vista presenta un conjunto de conceptos que definen los aspectos a considerar en el diseño del servicio, lo que se pretende con esta descomposición es equiparable a lo que ocurre en el dibujo técnico en donde un conjunto de perspectivas describen al mismo objeto pero desde diferentes puntos de vista o en la ingeniería civil en donde un estructura es el resultado de la superposición de muchos planos: eléctrico, hidráulico, estructural y etc.

El modelo está dividido en dos grandes partes, una de ellas se denominada *capa servicio* y es donde se diseña el servicio considerando el efecto producido por el ambiente en el software. La otra parte es la *capa aplicación*, en la que se reúnen todas las reglas, conceptos y principios que sirven de base para estructurar el servicio, con la salvedad, de que no se considera al ambiente, es decir, que se obtiene un diseño independiente del sistema de telecomunicaciones, la unión de estas dos especificaciones da como resultado la arquitectura MR-DSOA del servicio. Ver figura 2.12.

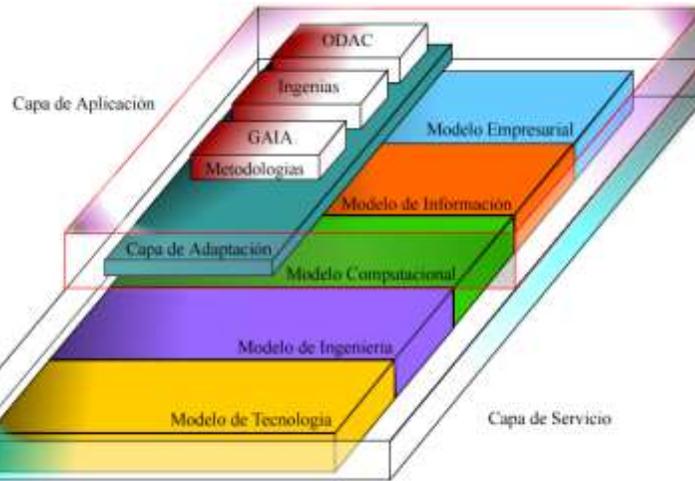


Figura 2.12. Modelo para el desarrollo de servicios con orientación a agentes (MR-DSOA).

2.7.1 Capa servicio

2.7.1.1 Modelo empresarial

En este modelo se realiza una abstracción a muy alto nivel de las operaciones internas y externas que debe realizar el servicio, especificando sus actividades y responsabilidades. Puesto que un sistema de telecomunicaciones puede ser bastante extenso y heterogéneo, la especificación empresarial identifica al servicio, su ambiente y las comunicaciones necesarias entre estos. Este modelo responde a los siguientes cuestionamientos: ¿Cuál es el propósito del sistema?, ¿Qué servicios son responsabilidad del sistema? y ¿Quién necesita los servicios?

2.7.1.2 Modelo computacional

Este modelo captura el comportamiento del servicio, en contraste al modelo empresarial que busca identificar actividades y responsabilidades. La especificación de este modelo da una idea del funcionamiento del servicio, mostrándolo como un conglomerado de agentes que satisfacen los requerimientos del diseño. El modelo debe responder: ¿Qué operaciones son válidas?, ¿Quién o que entidad desempeña las operaciones? y ¿Cómo toman lugar?

2.7.1.3 Modelo de información

Identifica a los agentes que son responsables de los flujos de información, el contenido de esta y los lugares donde se almacena. Este modelo debe dar respuesta a las siguientes preguntas: ¿Qué información se necesita en el servicio?, ¿Dónde está?, de ¿Dónde proviene?, ¿Para dónde va?, y si es necesario almacenarla. Esta especificación no debe describir la estructura de los datos, pero si su semántica.

2.7.1.4 Modelo de ingeniería

El modelo de ingeniería identifica la infraestructura necesaria para la ejecución de los servicios, durante su especificación se tienen en cuenta la forma en que se agrupan las entidades con el fin de modularizar el sistema, también incluye aspectos relacionados con la comunicación pero a bajo nivel protocolos, tramas y etc. Este modelo responde a la pregunta: ¿Cómo el sistema de telecomunicaciones soporta a los servicios?

2.7.1.5 Modelo de tecnología

En este se especifica al sistema en términos de los componentes hardware y software necesarios para implementar los aspectos identificados en los modelos anteriores, este modelo responde: ¿Cómo es la infraestructura hardware y software que soporta al servicio?

2.7.2 Capa de aplicación

2.7.2.1 Capa de Adaptación

La capa de adaptación es la encargada de separar la especificación del servicio de la de aplicación, su función es la de rescatar la mayor cantidad de información generada en los modelos empresarial, de información y computacional, para su posterior reutilización en los niveles superiores.

2.7.2.2 Capa de metodologías

Finalmente podemos ver que sobre la capa de adaptación se encuentra una serie de metodologías propias del paradigma de agentes que pueden utilizarse para elaborar la especificación de la aplicación. El modelo concibe la utilización de una o varias metodologías y no se ciñe a una en particular, mas sin embargo, en la aplicación del modelo (capítulo III) se utilizara INGENIAS, ya que esta es la que mejor se ajusta a las características del servicio a modelar.

2.8 CICLO DE VIDA

La descripción de un proceso de desarrollo, no estaría completa hasta que se considere a los diseñadores, desarrolladores o arquitectos como parte del sistema, una forma de integrarlos al entorno y por ende al modelo, es a través de la selección del ciclo de vida que regirá a los servicios. El ciclo de vida empleado por MR-DSOA esta basado en el de TINA y en el se identifican cinco fases, como se puede observar en la siguiente figura.



Figura 2.13. Ciclo de vida de los servicios MR-DSOA.

2.8.1 Fase de necesidad

Cubre la identificación y análisis de las razones que motivan la introducción del nuevo servicio.

2.8.2 Fase de Desarrollo

Cubre el diseño, implementación y prueba del servicio. Esta fase se puede descomponer en dos grandes partes: etapa de definición y de desarrollo.

8.2.2.1 Fase de definición

Esta fase se centra en la identificación de la información necesaria para obtener un software de calidad y en ella se establecen las restricciones en el diseño, los criterios de validación, la recolección y análisis de los requisitos del sistema y el servicio. En esta fase se distinguen las siguientes etapas:

- Ingeniería y análisis del sistema: Puesto que un servicio es parte de un sistema mayor, se deben establecer todos los requisitos que el servicio debe satisfacer a cada uno de los componentes del sistema, esto es muy importante ya que en un sistema de telecomunicaciones el software a desarrollar debe interactuar con otros elementos, tales como: hardware, base de datos, software y etc.

- **Análisis de requisitos:** Proceso en el que se recopilan y analizan los requisitos que debe satisfacer el servicio, en la etapa anterior eran con respecto al sistema, en esta es frente a los clientes o usuarios del servicio. El “analista” debe comprender la naturaleza del servicio, el ámbito de la información, así como las funciones para determinar el rumbo a seguir y que el software desarrollado cumpla con las expectativas.

8.2.2.2 Fase de desarrollo

Durante esta fase el desarrollador determina como ha de implementarse las estructuras de datos, la arquitectura del servicio, la implementación de los detalles procedimentales, la traducción del diseño a un lenguaje de programación y como ha de realizarse las pruebas. Esta fase contempla:

- **Diseño:** El diseño software en un proceso con múltiples etapas en las que se traducen los requisitos en una representación conceptual que cumpla con ciertos parámetros, por ejemplo: calidad, funcionalidad, sencillez y demás. En esta etapa se pueden emplear representaciones o lenguajes de modelamiento, que describen la estructura de los datos, la arquitectura, el procedimiento algorítmico y etc.
- **Codificación:** El diseño debe materializarse en una representación física que pueda ser ejecutada por una maquina, la traducción de las representaciones conceptuales del software arroja como resultado un conjunto de instrucciones ejecutables por una computadora.
- **Pruebas:** Una vez se tiene la implementación del diseño, se procede a probar los ejecutables o código en una maquina, con el fin de descubrir y corregir los defectos funcionales, en la lógica y en la implementación.

2.8.3 Fase de Instalación

Cubre la introducción del servicio al sistema de telecomunicaciones, debido a que un sistema de telecomunicaciones puede involucrar gran cantidad de nodos y equipos, se debe considerar el tiempo y esfuerzo necesario para la configuración e instalación del servicio y del software necesario: plataforma de agentes, NCCE y etc. Dependiendo de la complejidad del servicio y los elementos que este involucre (recursos software o hardware especiales) esta operación puede tardar y generar inconvenientes.

2.8.4 Fase de Operación

Cubre la ejecución, uso y gestión del servicio.

2.8.5 Fase de Retiro

Extracción del servicio.

2.8.6 Relación ciclo de vida, servicios telemáticos y MR-DSOA

Este modelo de ciclo de vida no debe seguirse secuencialmente, es decir dependiendo de la naturaleza y características del servicio a desarrollar se hará necesario darle un ordenamiento especial, aparte de esto es lo suficientemente genérico como para adaptarse a modelos de ciclo de vida más complejos.

La connotación dada al termino servicio en el marco de este modelo plantea una profunda discrepancia con el desarrollo tradicional de aplicaciones, ya que las motivaciones, necesidades, requerimientos y de mas características influyentes en el diseño cambian sustancialmente de una a otra.

Por ejemplo los usuarios finales de un servicio telemático pueden ser muchos (miles o millones), por que podría ir dirigido a varios roles de negocio, ya sean usuarios, intermediarios, proveedores de servicios, minoristas, proveedores de conectividad, otros sistemas y etc. Este grupo de consumidores debe estar en capacidad de pagar por esta funcionalidad, justificando los elevados costos incurridos en el diseño y especialmente en la puesta en marcha y operación del servicio, por esta razón los servicios deben exhibir unas características especiales que les permita captar la mayor cantidad de usuarios, solventando así los costos de desarrollo, mantenimiento de infraestructura, pago de personal, adquisición de equipos, compra de licencias software, asistencia y demás gastos presentes en un sistema de telecomunicaciones

Según la magnitud del servicio que se desee desarrollar y la complejidad de los dominios administrativos de negocio que componen el sistema se requerirán uno u otros instrumentos adicionales para cumplir con los objetivos de la primera fase del ciclo de vida. En el caso de servicios telemáticos de gran escala donde los usuarios finales se pueden contar por miles o millones por ejemplo: servicios multimedia, servicios de datos, servicios de mensajes, servicios de ringtones y etc. En ningún momento los usuarios solicitan directamente el desarrollo de estos sino que a través de herramientas de gestión tecnológica y de proyectos algunos inversionistas pueden determinar el impacto de uno u otro servicio a través de técnicas de análisis y estudios mercadotécnicos, para poder predecir el éxito o fracaso de un determinado servicio. Después de que se ha determinado la viabilidad del desarrollo del servicio como si se tratase de un proyecto cualquiera, se procede a su diseño y posterior implementación, esta fase de estudio previa a la formulación de un proyecto es muy importante por que la inversión en tiempo y dinero se ve retribuida en la medida de que los usuarios se suscriban y utilicen el servicio, en cuanto a los requerimientos y características de estos los diseñadores no pueden hallar consenso en la población objetivo por las cuestiones anteriormente expuestas, por ello deben desarrollarlos lo mas genérico posible, una especialización del servicio puede provocar que no sea atractivo para un grupo grande de usuarios por lo tanto el servicio debe ser sencillo, practico, sin lugar a confusiones y

especialmente genérico. Una estrategia para tratar con la complejidad en los requerimientos es definir perfiles de usuarios, servicios muy complejos exigirán dispositivos móviles tecnológicamente avanzados de elevados costos y por ende en manos de pocas personas, hecho que reduciría sustancialmente la población objetivo.

Durante las fases tempranas del ciclo de vida el diseñador puede tener en mente a un grupo de usuarios objetivo y caracterizarlos en perfiles, por ejemplo por indicadores económicos: estrato social, nivel de ingresos y etc. Por el tipo de contrato: prepago, pospago y otro, esta discriminación facilita la identificación de las necesidades y expectativas de los usuarios, en servicios de gran escala.

2.8 MODELO EMPRESARIAL

2.8.1 Introducción

Simplifica las condiciones imperantes en el sector de las telecomunicaciones considerando el comportamiento de los actores que intervienen en la prestación de un servicio.

Los principales objetivos del modelo empresarial son:

- Definir los mecanismos que gobiernan las interacciones entre los actores de un servicio.
- Identificar el conjunto básico de puntos de referencia.
- Establecimiento de normas claras tendientes a regular las interacciones entre entidades pertenecientes a diferentes dominios administrativos.
- Definición de un marco de referencia común a todos los actores involucrados en el modelo.
- Establecimiento de un conjunto de roles y relaciones comerciales acordes con las tendencias actuales del mercado de las telecomunicaciones.
- Identificación de los requerimientos e imposiciones aplicables a un conjunto de servicios en particular.

2.8.2 Lenguaje empresarial

El lenguaje empresarial introduce los conceptos básicos necesarios para representar el funcionamiento de un sistema en su contexto de operación. La finalidad de especificar un sistema en estos términos es mostrar como los objetivos y políticas restrictivas se interrelacionan y actúan sobre el. Para lograrlo, el sistema es descompuesto en entidades empresariales que según el grado de abstracción deseado muestran con más o menos detalle los diversos aspectos operativos del sistema. Los agentes empresariales se pueden agrupar en comunidades según los roles que desempeñan, representando así a: organizaciones, usuarios, proveedores de conectividad, de información y etc.

Uno de los conceptos más importantes de la especificación empresarial es la noción de contrato de servicio (“contract”), que vincula al ejecutante de un rol; ya sea: agente, objeto u otro componente con sus obligaciones, detallando los objetivos en común y responsabilidades al interior de una comunidad. Por ejemplo: considere el caso de una corporación de ahorros y sus clientes o una organización gubernamental y sus beneficiarios, en estos ejemplos las relaciones corporación-clientes y organización gubernamental-beneficiarios detallan implícitamente objetivos en común y responsabilidades de unos con otros.

Como se ha mencionado, el sistema esta constituido por un conglomerado de entidades empresariales, distribuidas en diferentes dominios, por lo tanto el lenguaje de especificación debería permitir asociar a cada componente etiquetas de pertenencia; que permitan establecer relaciones de propiedad y de responsabilidad ya sean de tipo administrativas, comerciales, tecnológicas, de gestión entre otras. Por ejemplo en un servicio de venta de libros por Internet, el usuario transparentemente involucra en una transacción (compra) a varias organizaciones, la casa editora que provee el bien en este caso el libro, el servicio de mensajería que lo transporta hasta su domicilio, la entidad bancaria que cobra por la venta y etc. Al especificar un sistema como el anterior se debe tener en cuenta la interacción con terceros, identificando las restricciones en los mecanismos de pago y de seguridad, por citar algunos.

La federación es un tipo particular de comunidad y es define como: la reunión de dos o más grupos administrados por diferentes autoridades (o simplemente dominios), con el objetivo de cooperar y alcanzar metas en común.

La principal ventaja de utilizar sistemas basados en agentes es la posibilidad de integrar subsistemas aislados con otros, facilitando el intercambio información sin que ello implique perdida de potestad sobre los recursos federados. Los nuevos modelos de mercado como: business to business o business to consumer (B2B o B2C) exigen cooperación entre los participantes [Lucking 2000]. Estas alianzas estratégicas a nivel empresarial son posibles gracias a la federación. Al emplear este instrumento las partes deben definir las reglas que van a regular las asociaciones, estas según el caso podrían incluir tanto aspectos técnicos como administrativos.

Los dominios involucrados en una federación pueden ser de dos tipos:

- Administrativos: Cuando los componentes que conforman el dominio comparten un mismo sistema de gestión.
- Tecnológico: Cuando los componentes que conforman el dominio están soportados por una misma tecnología (hardware o software).

Cuando la administración de un dominio decide federar alguno de sus recursos debe definir políticas que regulen las interacciones o interworking entre sus componentes y los foráneos, la descripción empresarial considerar esto y una forma de hacerlo es a través contratos, puntos de referencia, interceptores y etc.

Una organización puede utilizar la federación en dos sentidos, una, como un mecanismo para la integración de sistemas y recursos pertenecientes a diferentes dominios, o con el propósito de compartir, integrar o segmentar sus propios recursos y aplicaciones, un ejemplo de esto son las empresas multinacionales con representación en varios países, cada una de las partes cuenta con sus propios sistemas pero pertenecen al mismo núcleo.

Cualquiera de las administraciones que componen un sistema de telecomunicaciones pueden unilateralmente imponer controles sobre sus recursos (hardware y software) sin consultar a los demás, entre los mas comunes están: políticas de seguridad, tasación, monitoreo, gestión y etc.

En un contexto empresarial cada administración debe gestionar sus propios recursos, imponiendo políticas claras sobre el buen uso dado a estos, especialmente por parte de terceros (usuarios, clientes, socios y demás). En un dominio se pueden aplicar infinidad de normas producto de requerimientos particulares de su funcionamiento, una de las tantas restricciones aplicables son:

- Mecanismos de control de acceso: Definición de categorías o perfiles que permitan establecer niveles de seguridad y de control de acceso a los recursos federados.
- Monitoreo: Entidades que permiten monitorear el funcionamiento del servicio y a los usuarios.
- Tasación o tarificación por uso o alquiler de recursos.

Dependiendo del tipo de relación existente entre dos o más dominios, la especificación empresarial debe tener en cuenta el efecto causado por el cambio de administración en la operación del servicio, esta variación en la legislación es notoria en los puntos de intersección de los dominios (fronteras o límites administrativos). Los límites administrativos son el lugar donde cambian las responsabilidades en la gestión, la asignación de recursos y etc.

Como se puede deducir, la federación requiere la especificación de una entidad especial en la vista de ingeniería ya que los dominios involucrados pueden diferir ampliamente en cuanto a tecnología, aplicaciones y etc.

Las políticas se definen como el conjunto de reglas que regulan variados aspectos del funcionamiento de un sistema, por ejemplo: comportamientos, acciones y demás. La evaluación de una regla produce como respuesta tres tipos de estados: permitido, no permitido (prohibido) o el desempeño de algún tipo de acción de carácter obligatorio (obligaciones).

Las políticas son muy importantes para la vista empresarial, es por ello que a continuación se presenta una lista de los aspectos mas relevantes que deben ser cubiertos por esta:

- La estructura interna de una comunidad de agentes, por ejemplo, en una comunidad cualquiera podría ser necesario definir políticas que regulen:
 - La asignación de roles y responsabilidades a los agentes que conforman la comunidad.
 - La organización interna de los miembros que integran la comunidad (definición de jerarquías).
- También puede incluir reglas de negocio, como por ejemplo:
 - Una empresa u organización como una entidad de negocio.
 - Requerimientos en la facturación (accounting).
 - Planeamiento estratégico que ayude a cumplir los objetivos comerciales.
- Interacciones permitidas entre agentes, por ejemplo se pueden incluir políticas de seguridad que:
 - Regulen las relaciones rol-actividad-agente, definición de requerimientos de integridad, confiabilidad, confidencialidad y etc.
 - Políticas de acceso al servicio.
- Responsabilidades delegadas o confiadas a los agentes:
 - Niveles de confianza o privilegios (“trust”).
 - Permisos y/o prohibiciones para ejecución de tareas por parte de agentes (obligaciones).
- Uso de entidades informáticas o hardware para la facturación:
 - Entidades reguladoras.
 - Leyes gubernamentales.
- Tipo de recuso:
 - Publico.
 - Privado.
- Niveles de membresías:
 - Reglas de membresía de un dominio.
 - Reglas de interacción entre dominios del mismo tipo.
 - Reglas sintácticas para el nombrado de recursos (dominio de nombres).
- Propiedad de los recursos, por ejemplo, reglas de transferencia, que definan las normas empleadas para el intercambio de propiedad y/o obligaciones entre recursos.

El deseo de una descripción empresarial es identificar las acciones necesarias para el cumplimiento de las políticas impuestas a una comunidad en particular. En tales acciones se deben considerar aspectos como:

- Asignación y cumplimiento de obligaciones, por ejemplo: en una solicitud de compra, el sistema debe asignar tareas particulares a cada uno de sus componentes con el fin de procesar la transacción lo más pronto posible y así entregar el artículo.
- Acciones permitidas o prohibidas, por ejemplo en el control de acceso de un sistema el usuario puede validarse y estar autorizado a utilizar cierta funcionalidad o por el contrario no ser autorizado y rechazado del sistema.

Algunos elementos visibles desde la perspectiva empresarial también los podrían ser desde otras vistas y viceversa. Por ejemplo una actividad presente en la descripción empresarial puede aparecer en la vista de información como la especificación de algún proceso que causa la transición de estado en una entidad de información.

2.8.3 Entidades del modelo empresarial

2.8.3.1 Grupo o comunidad

Se denomina grupo al conjunto de agentes que por consideraciones de diseño se han dispuesto estructuralmente juntos, no es necesario que todas las entidades que integran el conglomerado sean idénticas, pero si deben exhibir un tipo de comportamiento que justifique su agrupación. El arquitecto tiene la absoluta libertad para seleccionar los criterios que empleará para crear y conformar grupos, estos deben ser consistentes y no dar pie a confusiones. Entre los criterios de composición más frecuentes están los funcionales; grupos de agentes con funciones similares, por objetivos; conjunto de agentes con objetivos en común, estructurales y etc.

La creación de grupos facilita el diseño, ya que considerar el comportamiento de una colectividad es mucho más sencillo que hacerlo individualmente, de ahí la importancia de su correcto aprovechamiento, especialmente para el diseño de servicios en sistemas de telecomunicación que por sus características involucran grandes cantidades de información, aplicaciones y etc.

2.8.3.2 Dominio

Conglomerado de recursos informáticos y electrónicos, entendiéndose por estos: dispositivos de telecomunicación, servidores, terminales, tecnologías, equipos, aplicaciones y etc. gestionados por una misma autoridad.

Un sistema debería descomponerse en varios dominios y estos a su vez en grupos y sub-dominios con el objetivo de reducir la complejidad en el diseño y su posterior desarrollo.

En esta vista no se consideran las tecnologías como tal pero si como esta afecta la prestación de los servicios. Descomponer un sistema en dominios y grupos permite introducir al modelo nociones de autonomía, autoridad y control, ya que cada dominio se considera como una estructura independiente de las demás, con sus propios mecanismos de gestión y control.

En la medida que se diseña un sistema se hace necesario definir varios tipos de dominios que cubran necesidades o requerimientos particulares de este. En la figura 2.14 se puede apreciar la descomposición de un sistema.

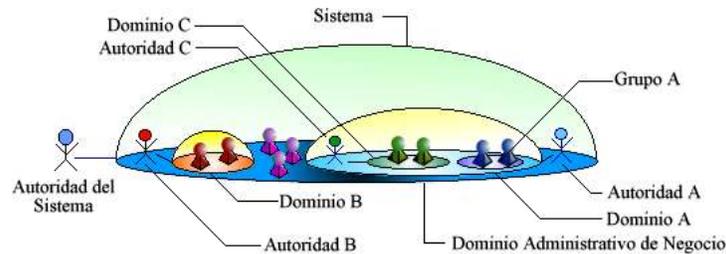


Figura 2.14. Dominios

Por ejemplo, las políticas restrictivas y de control de acceso a una red de telecomunicaciones podría delegarse a un dominio, que sería el encargado de fijar y aplicar las políticas de seguridad a todos los roles de negocio involucrados.

2.8.3.3 Autoridad

La autoridad es la entidad encargada de tomar decisiones que afectan el comportamiento de un dominio. Estas funciones son desempeñadas en algunos casos por agentes o personas según las condiciones del sistema y la naturaleza de las decisiones (técnicas u administrativas). Cada dominio debe asociarse exclusivamente a una autoridad.

En el caso de que un dominio contenga a otros, cada autoridad debe preocuparse por gestionar y regular el comportamiento de las entidades que están bajo su control, con el fin de impedir que un mismo agente sea gestionado simultáneamente por dos o más autoridades, hecho que provoca inconsistencias en el sistema. Para evitar estas situaciones se hace necesario el establecimiento de jerarquías y líneas de mando entre autoridades. En la figura 2.15 la autoridad más externa controla solamente a los agentes y autoridades que están dentro de su ámbito de influencia y así sucesivamente.

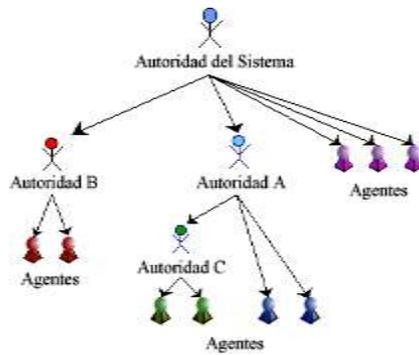


Figura 2.15. Jerarquías en un dominio.

2.8.3.4 Dominio administrativo

Se habla de dominio administrativo cuando los componentes software y hardware de un sistema están bajo el control de un grupo de personas que se dedica a gestionarlos, mantenerlos operativos y etc. Por ejemplo la red de datos de la Universidad del Cauca se consideraría como un dominio administrativo, ya que existe un grupo identificable de personas que controlan y emiten políticas de seguridad, de acceso y etc.

2.8.3.5 Dominio tecnológico

El concepto de dominio tecnológico hace referencias a las tecnologías que pueden coexistir en un sistema, en el caso de la red de datos de la Universidad del Cauca, se la puede considerar como un dominio tecnológico ya que utiliza una misma tecnología de red (ethernet), en el caso de que esta ofreciera acceso inalámbrico a sus usuarios, existirían dos dominios tecnológicos diferentes, gestionados por la misma autoridad, en este caso la red de datos; para identificar un dominio tecnológico no solo se deben emplear como criterio las redes, también las aplicaciones; en general el software es un criterio válido. En la especificación empresarial de un servicio solo se deben considerar los dominios tecnológicos que lo afectan.

2.8.3.6 Dominio administrativo de negocio

Los dominios administrativos de negocio son entidades encargadas de abstraer el funcionamiento real de un sistema de telecomunicaciones, claro esta, visto desde una perspectiva empresarial, a nivel de diseño sus requerimientos están estrechamente ligados a los roles de negocio con los que interactúa. Un dominio administrativo de negocio interactúa con otros a través de uno o varios puntos de referencia que son implementaciones de las relaciones de negocio. Los puntos de referencia deben contar con un punto de acceso (contact) que reciba estas solicitudes. Por ejemplo se podría emplear a un agente no autónomo y persistente que haga las veces de servidor recibiendo las solicitudes de servicio provenientes de otros roles de negocio, por lo general los intermediarios son los encargados de registrar y gestionar la información de estos puntos de referencia (o acceso).

El concepto de dominio administrativo de negocio esta basado en la noción de pertenencia y propiedad del lenguaje empresarial (sección 2.8.2). La propiedad implica el privilegio universal de gestionar y controlar las acciones de un grupo de entidades pertenecientes a un dominio, estos privilegios se pueden delegar a otros dominios y así dividir un gran problema en pequeñas partes, facilitando el diseño y desarrollo.

Los dominios administrativos de negocio se pueden descomponer o fusionarse en otros tipos de dominios no necesariamente pertenecientes al mismo modelo. La figura 2.16 muestra la descomposición de un dominio administrativo de negocio en un dominio de seguridad, gestión y nombrado.

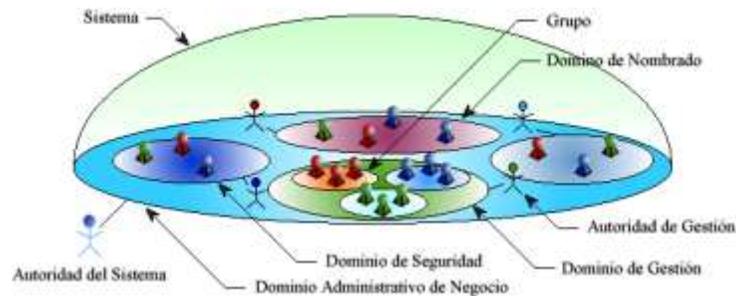


Figura 2.16. Mapeo de dominios.

2.8.3.7 Dominio de nombrado

El dominio de nombrado o servicio de nombrado se encarga de la asignación de nombres y la resolución de estos al interior de un dominio administrativo de negocio. Un nombre es una representación lingüística que sirve para distinguir unívocamente a un objeto, agente, tarea, servicio u otro de una multitud. No es obligatorio que exista una correspondencia uno a uno entre los nombres y las entidades, pero cuando esto ocurre a este se le denomina identificador.

Se puede considerar al dominio de nombrado y a sus componentes como una porción del dominio administrativo de negocio, debido a que este le cedió sus privilegios (delegación). Por esta razón no es necesario definir este dominio cada vez que se especifique un nuevo dominio administrativo de negocio.

Uno de los aspectos más importantes en un sistema distribuido es la posibilidad de establecer conexiones entre sus partes. Esto es posible gracias a las políticas de nombrado que regulan la asignación de identificadores, así como de su composición, forma de generación, espacio, contexto de los nombres y del léxico empleado.

El dominio de nombrado es uno de los tantos mecanismos empleados para ocultar o enmascarar los efectos producidos por la distribución de componentes (ubicación, reubicación, locación y demás). Por ejemplo es más fácil acceder a la funcionalidad de un agente invocando su nombre, IOR o referencia, que hacerlo por otros métodos.

Varios dominios administrativos de negocio pueden compartir un mismo dominio de nombrado o emplear cada uno el suyo. En el primer caso los nombres tienen validez en un medio mucho más amplio favoreciendo el interworking, si por el contrario cada dominio administrativo de negocio posee el suyo los nombres tienen validez en un ámbito más pequeño requiriéndose entidades que hagan conversión de nombres [Tocher 1991].

2.8.3.8 Dominio de gestión

La autoridad del dominio administrativo de red está limitada exclusivamente a controlar aspectos relacionados con la gestión de esta (infraestructura y recursos computacionales) y no tiene ninguna inferencia en otros temas. Debido a que la gestión es un aspecto fundamental en todo sistema de telecomunicaciones, el dominio administrativo de red se considera como una parte del dominio administrativo de negocio.

2.8.3.9 Contrato

Un contrato de servicio es un concepto empleado en el ambiente de las telecomunicaciones para caracterizar y regular la cooperación entre entidades, en este; las partes interesadas acuerdan las condiciones básicas para el establecimiento y mantenimiento de asociaciones. Se compone de un conjunto de restricciones destinadas a regular las interacciones entre dominios administrativos de negocio. Estas reglas controlan el funcionamiento interno del sistema y tratan diversos aspectos como por ejemplo: obligaciones, expectativas y otras tantas producto de los requerimientos particulares del servicio. No se concentran en aspectos meramente informáticos, por el contrario pueden establecer normas que trascienden de este ámbito, a cuestiones administrativas, técnicas, de equipos y etc.

Como se mencionó los contratos pueden incluir cualquier tipo de restricciones, estas dependen básicamente de las necesidades particulares de los dominios involucrados, pero a pesar de ello existe un conjunto de reglas básicas que deberían establecerse en toda relación y estas son:

- Visibilidad: Entidades visibles y por lo tanto accesibles desde el exterior de un dominio.
- Derechos: Privilegios asignados a una entidad que pertenece a un determinado dominio.
- Especificación de roles y obligaciones asignadas a los agentes involucrados en una relación de cooperación.
- Especificación de comportamientos esperados durante una relación de cooperación.
- Calidad del servicio mínima requerida para el intercambio de información.
- Tipos de comportamientos que invalida un contrato.

En la figura 2.17 se puede observar en detalle la composición de un contrato: restricciones en protocolos, restricciones administrativas, restricciones de acceso, restricciones de visibilidad y etc.

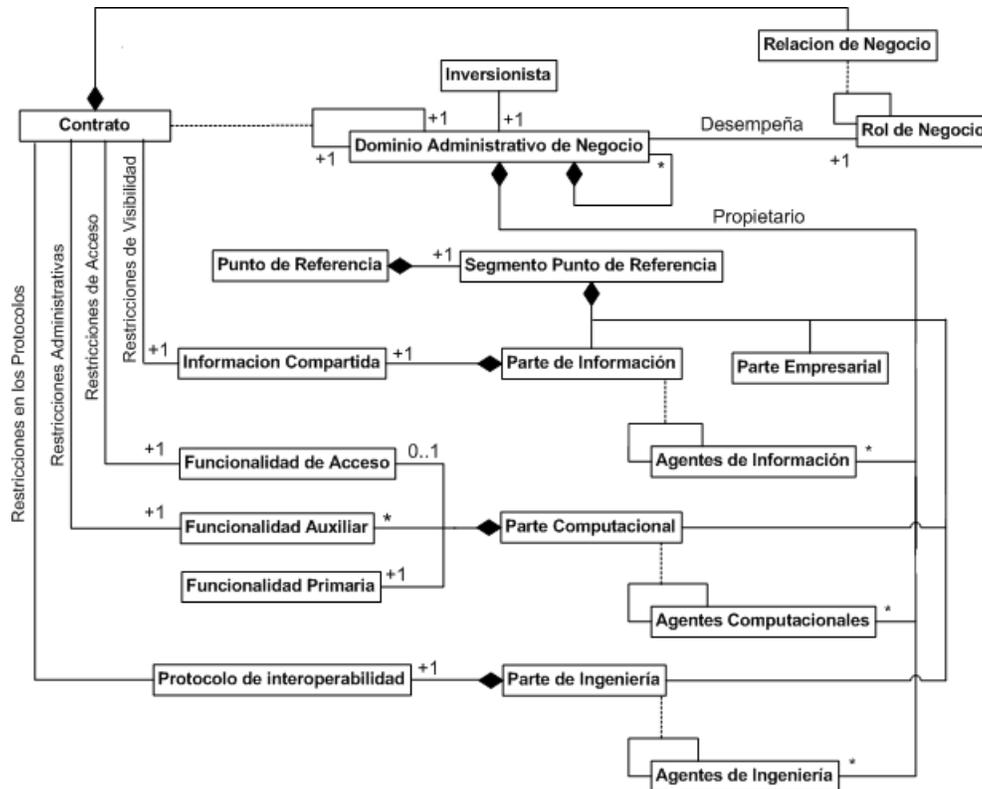


Figura 2.17. Conceptos empresariales.

La idea de diseñar un servicio con MR-DSOA es que cada agente pertenezca a un dominio administrativo de negocio y de que exista una separación entre ellos tanto para agentes como para los dominios. O sea que entre dominios administrativos de negocio debe existir un punto de referencia. Por tal motivo si se desea regular las interacciones entre estos, los contratos deben ubicarse en los puntos de referencia.

La existencia de regulaciones y restricciones en las relaciones entre agentes afectan notablemente el funcionamiento de un servicio, por tal motivo se dice que los contratos son una especificación dinámica de la configuración, por que, si bien la potencial cooperación entre agentes es constante, las reglas estipuladas en los contratos restringen estas intenciones a solo instantes o nunca según el resultado de la negociación.

No es necesario que toda interacción entre agentes esté regulada por contratos, en algunos casos, según el punto de vista y el lenguaje empleado, se podrían omitir. Se debe tener cuidado con esto ya que la exclusión de reglas y mas la supresión de contratos puede resultar en problemas de seguridad.

Un tipo particular de contrato es el celebrado entre los agentes y su ambiente, en este se establecen todos los requerimientos, obligaciones y expectativas de uno con respecto al otro y viceversa, por ejemplo ancho de

banda requerido, restricciones en la calidad del servicio (QoS) y etc. Como se dijo anteriormente no es necesario que cada agente establezca un contrato con el ambiente, este mecanismo se debe emplear solo en casos especiales donde el agente necesite alguna privilegio adicional.

Los contratos poseen otra característica adicional denominada negociación, propiedad bajo la cual los dominios administrativos de negocio pueden pactar las condiciones imperantes en una relación colaborativa. La negociación de un contrato puede darse en dos formas: on-line u off-line (en línea o fuera de línea).

En el caso de que se realice fuera de línea (off-line) las autoridades administrativas de cada uno de los dominios interesados en establecer una relación colaborativa intercambian sus contratos y llegan a un acuerdo, sobre las condiciones que regirán la asociación. Si por el contrario la negociación se realiza en línea (on-line). Existe la posibilidad de que las aplicaciones interactúen hasta lograr un convenio; siempre y cuando este sea posible.

La forma en que los dominios realizan estas operaciones depende en gran medida de su capacidad de procesamiento y de las herramientas informáticas con que cuenten. Las negociaciones se pueden realizar en distintas formas, por ejemplo: los dominio administrativos de negocio podrían intercambiar sus contratos por completo y luego decidir si los términos ahí presentados son aceptables, dependiendo del nivel de autonomía e inteligencia de los sistemas, estos podrían proponer modificaciones o incluir nuevas reglas a los contratos para que las regulaciones sean de provecho para las partes. Otra forma de negociar contratos sería que los sistemas debatieran regla por regla hasta completar el contrato y luego acordaran las regulaciones imperantes sobre la relación de cooperación (si es posible)

Cuando una organización y por ende un dominio administrativo de negocio es capaz de establecer contratos en línea, aumentan sus posibilidades de establecer nuevas relaciones comerciales, expandiendo así su área de influencia y abriendo nuevos mercados para sus servicios.

Como se ha mencionado un contrato plasma las restricciones globales del sistema, por lo tanto es el producto de la sumatoria de reglas individuales generadas en cada modelo. Un dominio administrativo de negocio nunca debe modificar reglas (contratos) sin considerar su efecto, ya que se podrían violar accidentalmente políticas esenciales para determinado modelo. Un claro ejemplo de esta situación se presenta al exponer agentes, por la modificación o adición de reglas que no consideren el modelo de información.

Si un dominio administrativo desea negociar un contrato este debe tener cuidado con quien y la forma en que realiza esta operación, por que al excluir, aceptar o modificar alguna regla podría exponer su sistemas a riesgos incalculables.

2.8.3.9 Punto de referencia

Como se muestra en la figura 2.17 un punto de referencia esta relacionado con las especificaciones de los modelos a través de una relación de agregación. Para facilitar la reutilización y la modularidad en las especificaciones de los puntos de referencia se han dividido en segmentos, unidades autocontenidas e independientes.

Los puntos de referencia deberían ser especificaciones independientes del servicio a diseñar, pero como estos están estrechamente ligados a la descripción empresarial, de información y etc. Que son especificaciones dependientes del sistema, se hace necesario desagregarlos a través de las siguientes consideraciones:

- Parte empresarial: Consideración de las restricciones, los requisitos funcionales y no funcionales expuestos en la relación de negocio. Se derivan de los requerimientos para la interacción entre roles de negocio.
- Parte de información: Define la información compartida entre dominios administrativos de negocio (ejemplo: nombres e información de gestión).
- Parte computacional: Define las interfaces computacionales o agentes computacionales accesibles desde otros dominios.
- Parte de ingeniería: Define como se descompone la plataforma conceptual de agentes en nodos, enlaces y demás elementos necesarios para el interworking entre dominios administrativos de negocio. Por definición se considera la PCA como parte integral del dominio administrativo de negocio, hecho por el cual no se considera el rol de negocio “proveedor de PCA”.

2.8.3.10 Rol de negocio y relaciones de negocio

Este conjunto inicial de roles de negocio fue identificado tras el análisis de las condiciones imperantes en el sector de las telecomunicaciones. Se emplearon tres tipos de criterios en su construcción:

- Técnicos: Áreas con diferentes capacidades de evolución tecnológica son ubicadas en roles de negocio diferentes, por ejemplo la brecha tecnológica existente entre los minoristas y los proveedores de conectividad (desarrollo rápido de servicios vs. desarrollo en redes).
- Legislativos: Debido a la existencia de ciertas restricciones legales en el ambiente de las telecomunicaciones, se generan divisiones en el mercado que dan pistas sobre la forma mas idónea para clasificar los roles de negocio. Por ejemplo, las regulaciones gubernamentales en el sector de las telecomunicaciones, separan a los intermediarios de los operadores de telefonía, por el mero hecho de nombrarlos o hacer alusión a ellos en las leyes, jurisprudencias y decretos.

- Económicos: Este criterio clasifica a los roles de negocio en dos grandes grupos según su papel en el mercado (consumidor o productor de servicios). Por ejemplo un minorista y proveedores de servicios (third party).

Basados en esto criterios se identificaron los siguientes roles de negocio:

- Usuarios: Este rol es introducido al modelo por consideraciones económicas, ya que es el único que utiliza o consume los servicios e infraestructura sin ánimo de lucro o con fines comerciales directos (a nivel de telecomunicaciones).
- Intermediario: Este rol es introducido por consideraciones legislativas ya que es el encargado de proveer libre acceso, equitativo y justo a la información que permite hallar servicios en un sistema a todos los restantes roles de negocio.
- Proveedor de conectividad: Este rol es introducido por consideraciones técnicas, dado el diferente ritmo de desarrollo tecnológico en los recursos de transporte (principalmente hardware) frente al área de servicios (básicamente aplicaciones o software).
- Minorista o comerciante al detal: Por razones económicas este rol y el de provisión de servicios se consideran por separado aunque es muy frecuente que ambos realicen las mismas tareas (provisión de servicios). Este modelo describe al minorista como la entidad encargada de gestionar usuarios y en añadir valor a los servicios, mientras que el proveedor de servicios externo está orientado a la producción y mantenimiento de servicios.

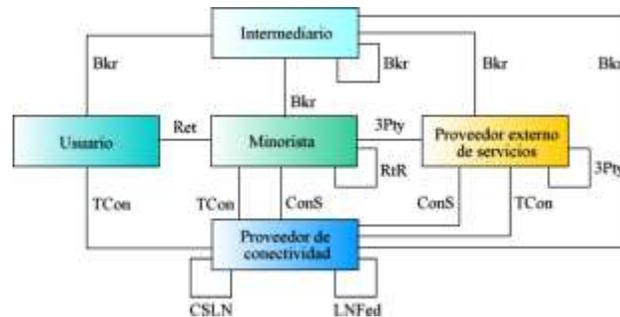


Figura 2.18. Modelo de negocio y relaciones de negocio

Los contratos que controlan las interacciones entre roles de negocio son los encargados de regular como y cuando es debe comportar un rol de negocio como consumidor o proveedor.

Los roles de negocio pueden combinarse para satisfacer las necesidades de los interesados. En este sentido, las relaciones de negocio expresan los requerimientos de las interacciones, siendo posteriormente materializadas en los puntos de referencia.

2.8.3.10.1 Roles de negocio

2.8.3.10.1.1 Inversionista

Persona o compañía que invierte de una manera u otra en un sistema de telecomunicaciones, ya sea con el fin de obtener una retribución económica o simplemente por acceder y utilizar la funcionalidad ofrecida por esta (servicios).

2.8.3.10.1.2 Usuario

El rol de usuario en un sistema de telecomunicaciones puede ser desempeñado por cualquier inversionista que no pretenda obtener un beneficio económico por ello, mientras el sistema ofrezca servicios que satisfagan las necesidades de estos, ellos serán considerados como parte de él, son la base económica del sistema, por que pagan por el uso de los servicios y su número puede ser grande, miles o millones, variando desde grandes compañías hasta individuos.

El rol de usuario también puede ser desempeñado por otros roles de negocio. Por ejemplo un minorista que desea probar sus servicios antes de hacerlos públicos, podría desempeñar el rol de usuario en su propio dominio.

Los requerimientos de alto nivel para este rol de negocio son:

- Obtener la ubicación de los minoristas, proveedores de servicio y otros usuarios.
- Registro y desregistro en minoristas.
- Iniciar relaciones de servicio que incluyan proveedores externos de servicios y otros usuarios.
- Aceptar invitaciones para unirse a sesiones de otros usuarios o minoristas.
- Aceptar descargas de aplicaciones de los minoristas.

2.8.3.10.1.3 Minorista

La principal función de un inversionista en el rol de minorista es la servir a los usuarios. El número de minoristas que pueden encontrarse en un sistema no es muy grande. Un minorista puede ser de cualquier tipo, desde una corporación hasta una compañía de “garaje”.

Un usuario puede usar uno o más minorista. Esto indica que el ciclo de vida de una relación usuario-minorista puede variar instantáneamente o en periodos largos. Por ejemplo un usuario podría usar un servicio ofrecido por un minorista solamente una vez, o por el contrario ser un cliente fiel, y utilizarlos frecuentemente.

Un minorista puede ofrecer un nuevo servicio para que sea utilizado de inmediato por sus usuarios sin consultar normas o estandarización, facilitando así la rápida puesta en marcha de estos.

El minorista esta en el eje focal del flujo de efectivo, este necesariamente deberá tener mecanismos contables que simplifique la facturación, permitiéndole determinar los costos y beneficios que debe asumir cada rol de negocio.

Un minorista seguramente requerirá de otros roles de negocio que le ofrezcan soporte para ofrecer servicios, en este modelo se han caracterizado tres tipos: intermediarios, proveedor externo de servicios y proveedor de conectividad.

Un inversionista desempeñando el rol de minorista puede a su vez tener más de un rol de negocio. Por ejemplo puede ejecutar el rol de minorista, intermediario y proveedor de conectividad al mismo tiempo, este es el caso de los operadores que son dueños de su propia red de transporte.

Los requerimientos de alto nivel para este rol son:

- Gestión de usuarios.
- Autorización para el uso de servicios.
- Mantenimiento de niveles de sesión, perfiles de usuarios y políticas del servicio.
- Gestión sobre las comunicaciones: sesiones, establecimiento y mantenimiento de asociaciones.
- Control y gestión de los flujos de información soportados por el proveedor de conectividad.
- Provisión de aplicaciones a los usuarios y demás roles de negocio.
- Acopio de información contable para la facturación y cobro por la prestación o consumo de servicio.

2.8.3.10.1.4 Intermediario

Un inversionista desempeñando el rol de intermediario tiene la misión específica de proveer a los demás roles de negocio de información que les permita encontrar otros dominios administrativos de negocio y servicios en el sistema.

Se asume que un intermediario provee la siguiente información:

- En respuesta a un nombre provee referencias a dominios administrativos de negocio (instancias).
- En respuesta a un conjunto de criterios provistos por un inversionista, busca en los registros uno o varios servicios que se ajusten con los criterios suministrados.

En el primer caso provee la función de paginas blancas donde dado el un nombre de un dominio administrativo de negocio, se retorna una referencia a una interfaz computacional del punto de acceso (contacto).

En el segundo caso provee la función de paginas amarillas donde dado un conjunto de atributos de un servicio se busca el nombre del proveedor, posteriormente este nombre puede ser resuelto en una referencia a un punto de contacto usando la función de páginas blancas.

De nuevo, un inversionista en el rol de intermediario también puede desempeñar más de un rol. Los requerimientos de alto nivel para este rol de negocio son:

- En respuesta a un identificador, proveer una única dirección de punto final o un conjunto de direcciones.
- En respuesta a una categoría de servicio, provee una lista de identificadores asociados con la categoría de servicios.
- Introduce, actualiza y remueve información de los dominios administrativos de negocio (instancias) y servicios ofrecidos.

2.8.3.10.1.5 Proveedor externo de servicios

El deseo de un inversionista en el rol de proveedor externo de servicios es proveer a los minoristas y a otros roles de negocio con servicios. El proveedor externo de servicios puede ofrecer la lógica del servicio, el contenido o las dos.

La diferencia entre un proveedor externo de servicios y un minorista es que el proveedor externo de servicios no tiene relaciones con los usuarios.

Los requerimientos de alto nivel para este rol son:

- Registro y obtención de la ubicación de los minoristas.
- Indicar la disponibilidad de los minoristas.
- Permitir almacenamiento de información a los minoristas.
- Establecimiento de relaciones para la provisión de servicios.
- Acopio de información de contabilidad para propósitos de facturación.
- Provisión y gestión de servicios.

2.8.3.10.1.6 Proveedor de contenidos

Una especialización del anterior rol de negocio es el proveedor de contenidos, este se concentra en la generación de contenidos. Por ejemplo una estudio de música o video que ofrece el servicio de distribución multimedia bajo demanda.

El proveedor de contenidos tiene las siguientes funciones y obligaciones:

- Propiedad intelectual de los contenidos.
- Entrega de contenidos.
- Gestión de los contenidos: control de versión, derechos de copia, distribución, licencias y etc.
- Cobro por el acceso a los contenidos.

2.8.3.10.1.7 Proveedor de conectividad

Un inversionista en el rol de proveedor de conectividad es propietario o gestiona una red de transporte que soportar flujos de información. El proveedor de conectividad ofrece una interfaz a cada uno de los roles de negocio permitiendo que ellos se comuniquen.

Los proveedores de conectividad tiene equipos hardware y software federados con todos los roles de negocio con los que interactúan, tarjetas de red, puntos de conexión y etc.

La red de transporte de un proveedor de conectividad no es una red global que conecta a todos los usuarios, minoristas y proveedores externos de servicios en un sistema. Las redes de transporte globales están segmentada en un numero de sub-redes controladas por diferentes inversionista (en el rol de negocio proveedor de conectividad).

Requerimientos de alto nivel para este rol:

- Configuración y gestión de asociaciones entre puntos terminales de una red.
- Adición y modificación de estas asociaciones (por ejemplo adición de nuevas rutas).
- Gestionar estas asociaciones (seguridad, tolerancia a fallos o etc).
- Acopiar información para propósitos de facturación.

2.8.3.10.2 Conjunto inicial de relaciones de negocio

Los requerimientos de cada relación de negocio pueden dividirse en dos partes, tratadas a continuación.

2.8.3.10.2.1 Acceso genérico entre dominios administrativos de negocio

Estas interacciones son genéricas para todas las relaciones entre dominios administrativos de negocio y son ejecutadas antes de cualquier interacción.

- Iniciar comunicaciones entre los dominios administrativos de negocio.
- Identificación de los dominios administrativos de negocio.
- Establecimiento, liberación y gestión de asociaciones de seguridad.
- Establecimiento de condiciones de facturación y tarificación.
- Servicio de descubrimiento e inicio.
- Establecimiento del contexto inicial de gestión (fijación de políticas por ejemplo disponibilidad, fiabilidad y manejo de fallos).
- Negociación de las condiciones iniciales en las interacciones (por ejemplo exposición de agentes de información para la provisión de servicios).

2.8.3.10.2.2 Relación de negocio minorista (Ret)

Esta relación es utilizada por usuarios y minoristas para el intercambio de información. Las siguientes interacciones son permitidas en esta relación:

- Acceso genérico.
- Descubrimiento e inicio de operaciones (gestión del servicio, configuración, gestión de perfiles de suscriptor, servicios, ofertas y etc).
- Control y gestión de sesiones por ejemplo anuncio, parada, suspensión, invitación, cambios en la notificación y etc.
- Control y gestión de flujos de información en las asociaciones.
- Control y gestión en los contenidos de los flujos de información.

2.8.3.10.2.3 Broker Business relationship (Bkr)

Esta relación provee acceso a la información controlada por el intermediario y a la gestión de esta por parte de cualquier otro rol de negocio. El intermediario puede proveer diferentes tipos de información a diferentes roles de negocio con distintos propósitos. El usuario puede interactuar con el intermediario por ejemplo para obtener referencias de minoristas disponibles. El minorista puede interactuar con el intermediario por

ejemplo para obtener la referencia de un grupo de usuarios o de los servicios ofrecidos por algún proveedor externo de servicios.

Las siguientes interacciones son ejecutadas en esta relación:

- Interacciones genéricas de acceso.
- Registro de nombres o instancias de dominios administrativos de negocio.
- Registro de servicios ofrecidos y sus atributos.
- Gestión de interacciones (por ejemplo actualización, adición, borrado) de la información anterior.

2.8.3.10.2.4 Third party business relationship (3Pty)

Un minorista puede interactuar con un proveedor de externo servicios para proveer de servicios a sus usuarios.

Las siguientes interacciones son desempeñadas en esta relación:

- Acceso genérico de interacción.
- Interacciones definidas por la relación de negocio Ret.
- Control y gestión de interacciones para servicios y/o contenidos de servicios.
- Gestión de interacciones para los servicios ofrecidos en el dominio del minorista.

2.8.3.10.2.5 Servicio de conectividad (ConS)

El servicio de conectividad es provisto por el proveedor de conectividad a los restantes roles de negocio, permitiendo el transporte de información punto a punto o multipunto. Las conexiones son establecidas entre los puntos de acceso en la red (capa de recursos de red).

Las siguientes interacciones son permitidas en esta relación:

- Interacciones de acceso genéricas.
- Control y gestión de conexiones.
- Gestión de los contextos para el servicio de conectividad (métodos de facturación, reportes y etc).

2.8.3.10.2.6 Retailer-to-Retailer business relationship (RtR)

La relación RtR reutiliza la funcionalidad del 3Pty y la relación de negocio Ret y considera el hecho de que la información que pasa sobre los puntos de referencia puede ser diferente.

2.8.3.10.2.7 Relación de negocio Terminal connection (TCon)

El TCon provee el enlace gestión entre el proveedor de conectividad y los roles de negocio involucrados en el grafico de conexión física.

Las siguientes interacciones son permitidas en esta relación:

- Interacciones de acceso genéricas.
- Negociación y control de la capa de conexión, configuración de concentradores, selección de canales y etc.
- Control conexiones.

2.8.3.10.2.7 Otras relaciones

Relaciones como: Layer network federation business relationship (LNFed) que permite la provisión de un servicio de conectividad mas allá de múltiples dominios administrativos de negocio están fuera del alcance del modelo al igual que la relación Cliente-servidor layer network relationship (CSLN) que provee el uso de la capa de red entre dominios administrativos

2.9 MODELO COMPUTACIONAL

2.9.1 Convenciones gráficas

Se han adoptado las siguientes convenciones gráficas para representar la especificación computacional:

- Los rectángulos sólidos representan agentes.
- Los cuadros con líneas inclinadas representan interfaces computacionales.
- Los cuadros blancos representan interfaces de flujo, los círculos blancos pequeños representan fuentes de flujo, y los círculos oscuros pequeños representan sumideros de flujo.
- Una línea fina que va desde un agente hasta una interfaz significa que el agente es un cliente de la interfaz.
- Una línea gruesa entre una fuente de flujo y un sumidero indica un flujo de información.
- Una interfaz que toca/cruza un rectángulo de grupo de agentes, representa un contrato.

La siguiente figura muestra las anteriores convenciones:

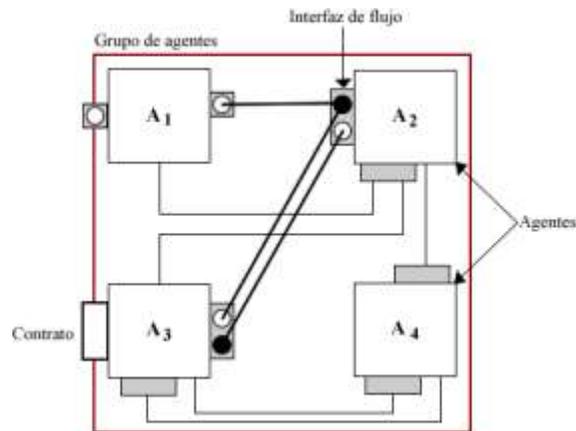


Figura 2.19. Representación gráfica de agentes computacionales.

2.9.2 Estructuras computacionales

En esta sección se identifican las estructuras computacionales básicas que componen un servicio telemático. Todas las interacciones entre entidades computacionales ocurren a través de las interfaces ofrecidas por éstas.

2.9.2.1 Agentes computacionales

Desde el punto de vista de este modelo, un servicio es una colección de agentes computacionales, esta entidad es la que proporciona interfaces de comunicación para el intercambio de información con otros dominios.

La Plataforma Conceptual de Agentes provee los mecanismos necesarios para transportar la información entre dos o más agentes que establecen una conversación. A continuación se describirán cada uno de ellos para comprender el papel que desempeñan dentro del enlace:

- *Protocolo de transporte de mensajes (MTP)*: Utilizado para la transferencia física de mensajes entre ACCs. Un ACC (Canal de Comunicación de Agentes) es la unión directa entre dos agentes que se comunican.
- *Servicio de transporte de mensajes (MTS)*: Es un servicio proporcionado por la plataforma conceptual de agentes y realiza el transporte de los mensajes ACL entre agentes dentro de una misma plataforma o plataformas externas.
- Un mensaje ACL representa la carga útil de los mensajes transportados tanto por el MTP y el MTS.



Figura 2.20. Transferencia de mensajes

A pesar de que las funcionalidades de las entidades mencionadas están implícitas en la Plataforma Conceptual de Agentes, serán tenidas en cuenta ya que a nivel de modelamiento aclaran aspectos de la configuración y constitución del servicio.

2.9.2.2 Agentes e interfaces

Una interfaz es el medio a través del cual dos o más agentes computacionales intercambian información (mensajes ACL), a diferencia de lo que ocurre con la orientación a objetos en donde estos publican los métodos ofrecidos por un objeto. Un agente encapsula datos, estados y comportamientos, proporcionando un conjunto de capacidades (o funciones) que pueden ser utilizadas por sus semejantes. Esta colección de capacidades está agrupada dentro de uno o varios subconjuntos, cada uno de los cuales es llamado servicio. De esta manera, un agente proporciona uno o más servicios que son accedidos por otros a través de las interfaces computacionales.

En una plataforma existen dos agentes que no pueden faltar en la implementación de un servicio telemático, a continuación se mencionaran cada uno de ellos:

El primer caso se presenta en el elemento conocido como DF (Facilitador de Directorio), quien es un agente encargado de servir como un directorio de páginas amarillas en donde cada agente publica el servicio que ofrece. En otras palabras, este agente ofrece una interfaz computacional proporcionada por el MTS que permite a otros agentes acceder a su servicio de publicación.

El otro caso se presenta en el AMS (Sistema de Gestión de Agente), quien tiene como tarea principal registrar a cada agente creado dentro de la plataforma, asignándole un identificador único (AID); para ello necesita ofrecer una interfaz provista también por el MTS que permite acceder a todos los agentes que requieran registrarse u obtener información sobre otro agente en particular (servicio de paginas blancas).

Al igual que el DF y el AMS, cada agente ofrece interfaces computacionales que permiten a otros agentes utilizar los servicios que ofrecen, siempre y cuando los necesiten y estén autorizados. Además de las interfaces necesarias para comunicar agentes existen otras que comunican a estos con entidades ajenas a la plataforma denominadas interfaces externas, que permiten conexiones con bases de datos, servidores http, y en general aplicaciones que no soportan mensajería ACL.

2.9.2.3 Interfaces de flujo

Una interfaz de flujo es un tipo de interfaz externa que proporciona un punto de comunicación para la recepción y transmisión de flujos de información. Estos pueden ser de diversos tipos, tales como audio, video o flujos de bytes.

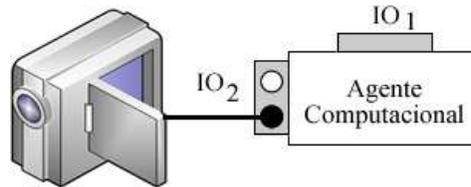


Figura 2.21. Interfaz de flujo.

Los flujos son unidireccionales y finalizan en los terminales ubicados dentro de las interfaces, cada terminal puede ser una fuente o sumidero de flujos de información dependiendo de las necesidades. Una interfaz puede albergar una mezcla de tipos, por ejemplo un punto terminal de video, figura 2.21.

2.9.2.4 Agentes con interfaz múltiple

Un agente puede proporcionar múltiples interfaces. Considere, por ejemplo, un agente que se comunica con otros agentes y además debe acceder a información almacenada en una base de datos. Las interfaces de comunicación con otros agentes son proporcionadas por el MTS tal como se mencionó anteriormente y la comunicación con la base de datos se realizará a través de una interfaz externa, en este caso peticiones SQL.

Como otro ejemplo consideremos el caso de un agente de suscripción que se encarga de llevar un registro de todos los agentes que están participando en un servicio (cada uno representando a un cliente), con el fin de gestionar el ingreso y salida de estos. En este caso el agente gestor solo tiene una interfaz que es la provista por el MTS.

Por otra parte, se puede tener un agente que maneja flujos multimedia en un servicio de videoconferencia, en este caso este debe poseer una o varias interfaces de flujo que reciban y envíen señales de video y audio como en la figura 2.21. La siguiente figura muestra un ejemplo de un agente que ofrece dos interfaces: IO1 es una interfaz MTS, mientras que IO2 es externa.

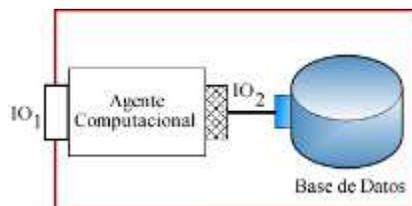


Figura 2.22. Agente computacional con múltiple interfaz.

2.10 MODELO DE INFORMACIÓN

2.10.1 Introducción

Una especificación de información MR-DSOA define:

- Las cualidades informativas del servicio, es decir, el significado de los datos almacenados o que son intercambiados entre componentes.
- Los requisitos para manipular la información al interior del servicio.

Generalmente se ve a la información como los datos que están presentes en el sistema y que se intercambian durante la ejecución de un servicio. Pero también, se les puede atribuir características cognitivas, es decir, tratarlos como la representación del conocimiento que el sistema tiene de sí mismo y de su ambiente. El modelo de información fusiona estos dos puntos de vista y trata a los datos como una parte del conocimiento general del sistema, estableciendo una relación recíproca, entre el significado físico de los datos y el conocimiento [Christensen 1995].

2.10.2 Información

Considerando el contexto de este modelo se entiende por “información” a todo aquello dato presente en el sistema. Para comprender el funcionamiento, comportamiento y estructura de esta base de conocimiento, no es necesario entender como está implementado el servicio, ni el tipo de tecnología que utiliza, la disposición de los recursos hardware y etc. Esos aspectos no tienen nada que ver con la especificación de la información.

Este modelo describe la estructura de la información en un servicio identificando los agentes de información, sus relaciones, restricciones, comportamientos, ciclo de vida entre otros.

El concepto de interfaz es de gran relevancia en una especificación de este tipo ya que es la entidad que permite la interacción entre componentes del sistema y otros dominios. En una especificación de información el concepto de interfaz es reemplazado por el de “relación”, ya que los agentes de información, como su nombre lo sugiere se encargan de manipular la información, a diferencia de los agentes computacionales que proveen las interfaces de acceso. Un sistema se debería descomponer en subsistemas y a su vez en agentes de información y computacionales.

2.10.3 Relación del modelo de información con los restantes modelos

La especificación de información está muy relacionada con la computacional, aunque sus objetivos son un tanto diferentes. Una especificación de información provee el conocimiento necesario para interactuar con un

sistema o subsistema, describiendo los procesos relacionados con la manipulación y transformación de la información. En cambio la especificación computacional, describe al servicio como un conjunto de agentes que ofrecen funcionalidad a otros. El desarrollo del modelo computacional depende del de información, puesto que los agentes computacionales son los que ofrecen las interfaces de comunicación a los de información, de ahí la fuerte dependencia entre estas dos especificaciones.

Desde un punto de vista metodológico, se pueden presentar dos casos en el desarrollo de los modelos de información y computacional: en el primer caso, la especificación de información se puede ubicar en una etapa temprana del diseño, de manera que se convierta en la base para la especificación computacional. En el segundo caso, las dos especificaciones se realizan simultáneamente y el modelo de información se utilizaría para caracterizar el comportamiento de las entidades computacionales.

Por otra parte, el nivel de abstracción de este modelo puede variar desde uno muy alto, en el cual no se consideran los aspectos relacionados con el sistema de telecomunicaciones (por ejemplo la red de transporte) y otro muy bajo y detallado, en el que algunos de estos aspectos son minuciosamente especificados.

A continuación presentamos algunas características de las especificaciones de información y computacional:

- No necesariamente existe una correspondencia uno a uno entre un agente computacional y uno de información, sin embargo, un agente computacional puede representar a un agente de información proporcionándole interfaces para que recupere o modifique la información.
- Los agentes computacionales interactúan únicamente con sus similares. Esto significa que un agente computacional no puede modificar el estado mental de un agente de información a través de una interacción computacional. Lo que si puede ocurrir es que el estado mental de un agente de información sea modificado y esta información sea transmitida entre agentes computacionales a través de una interacción.
- Los agentes computacionales no actúan sobre los agentes de información, aunque estos últimos representan los datos identificados en la especificación computacional.
- Las interacciones en una especificación de información son comúnmente modeladas como relaciones. Una relación puede, por ejemplo, establecer que la creación de un agente de información depende de la existencia de otro agente.
- Cualquier correspondencia entre las dos especificaciones debe ser aclarada para cada caso, de manera que se pueda verificar la consistencia entre ellas. La forma general para especificar la correspondencia entre ellas es tener en cuenta la especificación de información cuando se están documentando las interfaces computacionales. En la mayoría de los casos la especificación computacional es un refinamiento de la especificación de información.

2.10.4 Especificación de entidades

El primer paso dentro de esta especificación, es identificar las entidades de información. Para esto es importante definir la estructura del sistema, la cual como es lógico esta compuesta por agentes que interactúan entre sí e intercambian información con el fin de alcanzar un objetivo en común. La arquitectura FIPA (The Foundation for Intelligent Physical Agents) indirectamente define el conjunto inicial de relaciones o proceso de comunicación entre las entidades básicas que constituyen cualquier sistema de agentes. Por lo tanto a continuación se presentan los aspectos más importantes de esta arquitectura, claro esta desde la perspectiva del modelo de información. Cabe aclarar, que además de las entidades básicas que se describen aquí, en el diseño de un servicio aparecen unas nuevas que también se deben especificar.

2.10.4.1 Arquitectura FIPA

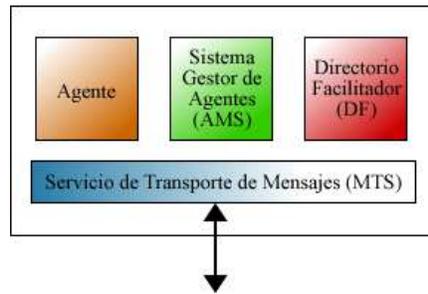


Figura 2.23. Arquitectura FIPA

Cada una de las entidades presentes en la arquitectura FIPA, figura 2.23, posee un conjunto de capacidades que son parte importante en el proceso de intercambio de información en un servicio. A continuación se describirán cada una de ellas:

Agente: En los procesos de intercambio de información los agentes se comunican utilizando un lenguaje estandarizado por FIPA llamado ACL (Agent Communication Language). Un agente puede ofrecer uno o varios servicios que son publicados de manera que sus semejantes puedan conocer las capacidades que ofrecen. Además posee un identificador llamado AID (Agent Identifier) que lo hace único y distinguible dentro del sistema en el que se encuentre. Por último un agente puede tener un número de direcciones de transporte con las cuales puede tener contacto.

Facilitador de Directorio (DF): Es un componente que se encarga de proporcionar el servicio de páginas amarillas en la plataforma de agentes. En él los agentes pueden publicar los servicios que ofrecen y también consultar los que están disponibles.

Servicio de Transporte de mensajes (MTS): Es el medio de comunicación por defecto utilizado por los agentes que están en diferentes plataformas.

Sistema gestor de agentes (Agent Management System - AMS): Es un componente obligatorio en una plataforma de agentes. Este se encarga de registrar a cada agente creado, asignándole su respectivo AID, además ofrece el servicio de páginas blancas. El AMS almacena entre otras cosas, la información de las direcciones con las cuales tiene contacto cada AID registrado.

Plataforma de Agentes (AP): Proporciona la infraestructura física sobre la cual los agentes son ejecutados. La plataforma provee soporte software a los agentes y está compuesta por los componentes de gestión FIPA (DF, AMS y MTS) y los agentes.

El segundo paso dentro de la especificación de las entidades de información, consiste en identificar las relaciones que existen entre ellas y las reglas que gobiernan su comportamiento. Tales relaciones están definidas por un estándar de comunicación conocido como mensajes ACL.

Para proporcionar facilidad en el intercambio de mensajes de información, FIPA define un conjunto de sentencias que identifican el tipo de información que se está intercambiando [FIPA 2002c]. A continuación se describirán cada una de ellas:

- **Accept-proposal:** Esta sentencia es utilizada por un agente para indicar que acepta una propuesta (request) hecha por otro agente.
- **Agree:** Es usada por un agente para indicar que ha aceptado una propuesta (request) hecha por otro agente. Indica que el remitente de la sentencia agree pretende ejecutar la acción solicitada.
- **Cancel:** Es utilizada por un agente para expresar que ya no desea ejecutar la acción solicitada en una petición anterior.
- **Cfp (call for proposals):** Utilizada para iniciar una negociación entre agentes. Este tipo de sentencia posee tanto una acción como una condición. Esto significa que se expresa la acción que el agente quiere ejecutar pero bajo unas condiciones que establece y deben cumplir aquellos agentes que le envíen sus propuestas. Por ejemplo un agente que desea recibir información sobre destinos turísticos de una ciudad cuyo costo de ingreso sea nulo. En este caso la acción es: recibir información sobre destinos turísticos, y la condición: que su costo de ingreso sea nulo. Aquellos agentes que posean información que esté de acuerdo con la acción y su condición pueden enviar sus propuestas, de las cuales el agente emisor de la sentencia cpf seleccionará la mejor.
- **Propagate:** El contenido de una sentencia de este tipo está compuesta de dos partes: Un mensaje y una expresión que denota un conjunto de agentes. La idea en este caso consiste en que el agente receptor reenvíe el mensaje a los agentes pertenecientes al conjunto.
- **Disconfirm:** Esta indica a un receptor que el contenido sobre el que está inseguro es falso.

- Inform-if: Se utiliza cuando se desea comunicar si una afirmación determinada es verdadera o falsa. Generalmente una sentencia de este tipo forma parte de un mensaje, por ejemplo cuando un agente envía una sentencia request a otro agente, una parte del contenido puede ser un mensaje inform-if, con el cual este pretende averiguar si cierta información es verdadera o falsa.
- Confirm: Esta sentencia es utilizada por el emisor de un mensaje para confirmar la veracidad del contenido de un mensaje que ha enviado anteriormente.
- Failure: Permite indicar a un agente que una petición solicitada previamente no se pudo ejecutar. El mensaje es enviado por el agente al cual se hizo la petición.
- Inform: Junto con la sentencia request es la más importante de las definidas en ACL. Es el mecanismo básico para la comunicación de información. El contenido de una sentencia de este tipo es una afirmación que el emisor desea que el receptor crea.
- Not-understood: Se utiliza cuando un agente desea indicar a otro agente que ha realizado una acción pero no entiende por que razón la ejecutó. El uso más común de la sentencia not-understood es cuando un agente desea informarle a otro que el mensaje que ha recibido no fue entendido. Un mensaje de este tipo consta de dos partes, la primera es la acción (aquella cuyo propósito no fue entendido) y la segunda es la afirmación (statement), la cual da alguna explicación sobre por qué el mensaje no fue entendido. Esta sentencia es la más importante en el manejo de errores del lenguaje ACL.
- Propose: Esta sentencia permite a un agente hacerle una propuesta a otro, por ejemplo en respuesta a un mensaje cfp enviado previamente.
- Proxy: Este tipo de mensaje permite a su emisor solicitarle al receptor que opere como un proxy para un conjunto de agentes. El contenido de una sentencia de este tipo contendrá un mensaje (el que se enviará a otros agentes) y una especificación del tipo de agentes que el emisor desea que sean los receptores.
- Query-if: Esta sentencia permite a un agente preguntar a otro si alguna afirmación específica es verdadera o falsa. El contenido del mensaje será la afirmación sobre la cual el emisor desea averiguar.
- Query-ref: Esta sentencia es utilizada por un agente para determinar el valor específico de una expresión.
- Refuse: Una sentencia refuse es utilizada por un agente para informarle a otro agente que no ejecutará una acción. El contenido del mensaje indicará tanto la acción que el agente no ejecutará como la razón de tal decisión.
- Request: Es la segunda sentencia más importante ya que permite a un agente solicitar a otro la ejecución de una acción.

- **Reject-proposal:** Utilizada cuando un agente informa que no acepta una propuesta hecha en un proceso de negociación. El contenido del mensaje consta tanto de la acción como de la razón por la cual no se ejecuta.
- **Request-when:** El contenido de una sentencia de este tipo consta de una acción y una expresión. La acción será ejecutada cuando la expresión sea verdadera.
- **Request whenever:** La acción se ejecutará en el momento en que la expresión sea verdadera.
- **Subcribe:** El contenido de esta sentencia será una expresión y el emisor será notificado cuando algo en ella cambie.

Ya hemos hablado del lenguaje por medio del cual los agentes se comunican. Pero además de esto, los agentes necesitan utilizar un vocabulario común que les permita entender el significado de cada palabra que intercambian dentro del ambiente en el que operan. Por ejemplo, supongamos que tenemos una aplicación de búsqueda de información y un agente envía dentro de una sentencia cfp como acción la búsqueda de un “restaurante” y como condición que sea “italiano” (como ya lo mencionamos anteriormente una sentencia cfp se compone de una acción y una condición), en este caso el agente receptor de la sentencia debe entender la información que recibe, es decir, a que se refiere el ítem “restaurante” y el tipo específico requerido “italiano” y deben tener para él, el mismo significado que tienen para el agente emisor. De esta manera es como surge el término ontología, que define un vocabulario y un significado para cada palabra que pertenece a este y que son comunes para todos los agentes que pertenecen a un mismo dominio.

En un proceso de comunicación un agente emisor envía información a un receptor por medio de un mensaje ACL, la información que va dentro de este es representada por medio de un lenguaje interno (el sugerido por FIPA es el lenguaje SL) y un formato específico (en el caso del lenguaje SL se utiliza Strings). El uso del lenguaje interno va de acuerdo a las necesidades de cada desarrollador, muchas plataformas de agentes incluyen lenguajes internos propios, incluso se pueden definir nuevos según se requieran. Según el modelo que aquí se propone el lenguaje interno no se tiene en cuenta por estar más relacionado con la implementación. Lo mismo ocurre para las ontologías, generalmente las plataformas proporcionan recursos para utilizar una o varias por defecto o crear una nueva. No obstante, es indispensable tenerlas en cuenta dentro de la especificación del modelo de información.

2.10.4.2 Protocolos de Interacción

La finalidad de un protocolo de interacción es permitir al agente cliente verificar si el efecto esperado debido a una petición enviada dentro de un mensaje ha sido alcanzada o no [FIPA 2002d].

La siguiente figura muestra la estructura general que tienen los protocolos de interacción. El agente iniciador envía un mensaje ACL que puede representar una petición, una solicitud de información u otro tipo (sección 2.10.4.1). El receptor de tal mensaje puede enviar una respuesta del tipo not-understood, refuse o agree, de acuerdo a las condiciones en las que se encuentre, tal como se muestra en la segunda fila de la figura 2.24. Finalmente, el receptor ejecuta la acción solicitada y envía una nueva respuesta que puede ser del tipo inform, para notificar al agente cliente (remitente de la petición) que la acción ha sido ejecutada, o un mensaje failure si algo salió mal.

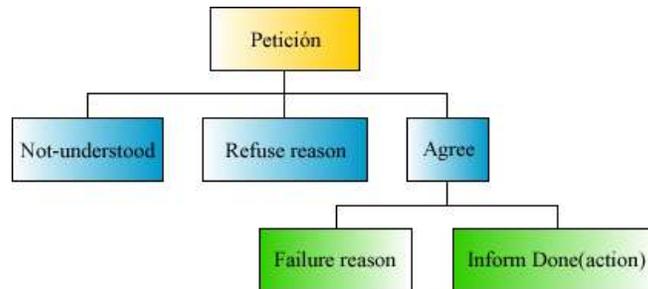


Figura 2.24. Estructura estándar de los protocolos de interacción FIPA

2.10.4.3 Estructura de un mensaje ACL

Un mensaje ACL contiene los parámetros necesarios para el establecimiento de una comunicación entre dos o más agentes sin importar las circunstancias. Si un agente no reconoce alguno de los parámetros del mensaje tiene que contestar con un mensaje not-understood. Algunos de los parámetros del mensaje son opcionales, en el caso de que se omitan su valor debe poderse deducir del contexto de la conversación. En la figura 2.25 están relacionados todos los parámetros de un mensaje ACL [FIPA 2002b].

Parámetro	Categoría del parámetro
Performative	Tipo de acto comunicativo
Sender Receiver Reply-to	Participantes en la comunicación
Content	Contenido del mensaje
Language Encoding Ontology	Descripción del contenido
Protocol Conversation-id Reply-with In-reply-to Reply-by	Control de la conversación

Figura 2.25. Parámetros de un mensaje ACL.

2.10.4.3.1 Tipo de acto comunicativo

2.10.4.3.1.1 Performative

Indica el tipo de acto comunicativo del mensaje, este parámetro es obligatorio para todos los mensajes ACL.

2.10.4.3.2 Participantes en la comunicación

2.10.4.3.2.1 Sender (remitente)

Indica quien es el remitente del mensaje, o sea el nombre del agente que envió el mensaje, es posible que este parámetro se deje intencionalmente en blanco si el remitente desea mantener en reserva su identidad.

2.10.4.3.2.2 Receiver (destinatario)

Indica quien es el destinatario del mensaje ACL, este parámetro es de tipo opcional. Si se omite su valor debe poderse inferir del contexto de la conversación, no es obligatorio que siempre se relacione a un solo agente puede ocurrir que el destinatario sea un grupo y en este caso los nombres de cada uno de estos son colocados a modo de lista de distribución (mensaje multicast).

2.10.4.3.2.3 Reply to (responda a)

Este parámetro indica que el siguiente mensaje de la conversación debe dirigirse al agente citado en este campo.

2.10.4.3.3 Contenido del mensaje

2.10.4.3.3.1 Content (contenido)

Indica el contenido o carga útil del mensaje, el significado de este debe ser interpretado por el destinatario, es muy importante que tanto el remitente como el receptor interpreten el contenido del mensaje en forma idéntica.

2.10.4.3.4 Descripción del contenido

2.10.4.3.4.1 Language (lenguaje)

Indica el lenguaje empleado para escribir el contenido del mensaje ACL.

2.10.4.3.4.2 Encoding (codificación)

Indica el método empleado para codificar el contenido del mensaje ACL

2.10.4.3.4.3 Ontology (Ontología)

Le indica al destinatario la ontología que debe usar para interpretar los símbolos en el contenido

2.10.4.3.5 Control de la conversación**2.10.4.3.5.1 Protocol (Protocolo)**

Indica el protocolo de interacción que el agente remitente esta utilizando.

2.10.4.3.5.2 Conversation identifier (Identificado de conversación)

Expresión o número que se utiliza para identificar la secuencia de los mensajes durante un acto comunicativo, es necesario utilizar un valor único global que permita a los participantes reconocer el orden de los mensajes, debido a que un agente puede mantener varias conversaciones simultáneamente. Un mecanismo simple para asegurar la singularidad en la secuenciación de los mensajes es concatenar el AID del remitente con un número que aumente progresivamente.

2.9.4.3.5.3 Reply with (Responda con)

Introduce una expresión que permite al agente remitente identificar el mensaje, este parámetro se utiliza para seguir el hilo de una conversación cuando un agente sostiene múltiples diálogos simultáneos.

2.9.4.3.5.4 In reply to (En respuesta a)

Indica una expresión que relaciona la ejecución de una acción con el mensaje de respuesta. Por ejemplo, si un agente A envía un mensaje a un agente B con el campo Reply-with <expresión>, El agente B deberá responder con un mensaje: In-reply-to <expresión>.

2.9.4.3.5.5 Reply by

Indica el tiempo y/o fecha en el que ocurrió la ultima interacción remitente-destinatario. La fiabilidad de este parámetro depende del modo de sincronización del sistema: esclavo/maestro o independientes. En el primer caso se puede decir que todas la PCA están sincronizadas y por ende la medición del tiempo será la misma para todos los nodos en el dominio administrativo de negocio.

Si son independientes este parámetro no es fiable por que cada PCA tendría su propio reloj y la noción del tiempo cambiaría de un nodo a otro.

La respuesta a un mensaje se puede identificar en cuatro formas:

- Por la secuencia del mensaje en un protocolo de interacción.
- Por el parámetro reply-with.
- Por el parámetro conversation-id.
- Por el parámetro reply-by (solo en esclavo/maestro).

2.10.4.4 Agentes de Información

Un agente de información es el componente básico de este modelo y las interacciones entre ellos uno de los aspectos claves a describir, ya que a través del intercambio de información es que un grupo de agentes presta un servicio.

Un agente puede tener un estado inicial A_0 antes de interactuar con otro y cambiar de estado durante el transcurso de una interacción, porque las acciones en las que se ve envuelto le obligan a modificar su comportamiento. El conjunto de acciones en las que un agente podrá participar depende del estado actual en que se encuentre y de los roles que puede desempeñar.

2.10.4.5 Tipos de Agentes de Información

Un agente de información pertenece a un tipo si está caracterizado por un predicado específico. Aquellos agentes que cumplen con el mismo predicado pertenecen al mismo tipo de entidad; por ejemplo, el predicado “vendedor”, identifica a los agentes que pertenecen a una aplicación de ventas y cuya función es ofrecer un producto específico a un grupo de agentes “compradores”; de acuerdo con esto, los predicados “vendedor” y “comprador” identifican dos tipos diferentes de agentes.

Existe una distinción entre tipos y clases; una clase es la extensión de un tipo, es decir, el conjunto de todos los agentes de un determinado tipo que fueron creados y que aún no han sido eliminados. Ya que los agentes pueden ser creados y eliminados, una clase, puede variar en el tiempo. Además, pueden existir predicados que se refieren al estado de un agente, de esta forma si un agente varía su estado, entonces su tipo también cambiará (Por ejemplo un niño se puede convertir en adulto).

Mientras un agente modela una entidad, un tipo de agente modela un concepto a través de un predicado. Un tipo puede ser visto como una plantilla utilizada para instanciar agentes.

2.10.4.6 Clasificación de agentes

La clasificación de agentes en tipos o categorías es importante por que permite identificar un conjunto de características comunes aun grupo. Una vez identificadas esta propiedades el diseñador modela estos comportamientos solamente una vez y según las necesidades particulares del servicio los agrupa en plantillas, esta estrategia de trabajo ahorra tiempo y esfuerzo tanto en el diseño como en la codificación. Por ejemplo considere al agente que representa a un usuario y le ayuda a acceder a los servicios provistos por un sistema, como se vio en el modelo empresarial el numero de suscriptores que pertenecen a un dominio administrativo de negocio puede llegar a miles o millones, por lo tanto en teoría podrían existir cientos de miles de instancias del mismo agente en el sistema. Ya que no todos los dispositivos móviles y perfiles de usuario son iguales ni mucho menos todos los suscriptores tienen contratados los mismos servicios este agente en esencia podría variar ligeramente de forma, pero muchas de sus características y comportamientos serian iguales, en consecuencia se debe considerar esto, desarrollando plantillas a las que se les pueda fácilmente adicionar o retirar funcionalidad evitando el diseño y codificación de n variantes de un mismo agente.

Existen dos formas de clasificar a los agentes en un sistema de telecomunicaciones, por sus características físicas o de comportamiento ver figura 2.26.

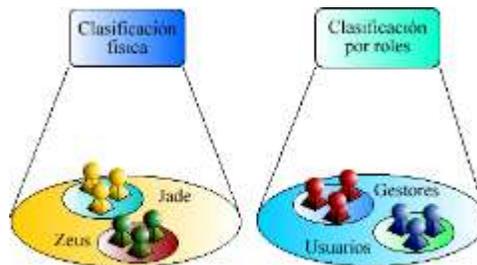


Figura 2.26. Tipos de agentes

2.10.4.7 Plantillas

Una plantilla de agentes se especifica igual que un agente convencional, la única salvedad es que estas modelan comportamientos muy generales que pueden especializarse para cumplir tareas particulares o servir de patrón para replicarse varias veces en el sistema, por ejemplo en el diseño una plantilla que controla el acceso a un dominio administrativo de negocio. El modelamiento se hace teniendo en mente un concepto tan abstracto como es el de dominio administrativo de negocio (conceptualización de las principales características de un sistema de telecomunicaciones). Este diseño y su posterior implementación sería tan general que podría utilizarse en muchos sistemas pero no aprovecharía las propiedades intrínsecas de cada uno, en consecuencia se hace necesario refinarle, aumentándole roles o redefiniendo sus comportamientos para que exploten las características particulares del sistema. La consideración de estos casos evita el esfuerzo de diseñar y codificar varias veces los mismos roles.

El uso de plantillas permite el reuso, ya que no es necesario crear agentes sencillos de un determinado tipo cada vez que se necesiten, sino que se invocan a partir de la plantilla que los define.

Las plantillas capturan las propiedades de un grupo o conglomerado y son utilizadas para instanciar nuevos agentes. Si se caracterizan por su comportamiento el agente se considera como una “caja negra” que describe el conjunto de actividades en las que puede participar. Por otra parte, el agente puede ser caracterizado en términos de sus estados y acciones, cada estado permite al agente participar en un conjunto de acciones y cada acción lleva al agente a un nuevo estado.

2.10.4.8 Subtipos

Si existen dos tipos de agentes diferentes T1 y T2 y existe un agente que pertenece al tipo T1 y también al tipo T2, se dice que es un subtipo de T1 y T2.

2.10.5 Tipos de información

En un sistema de telecomunicaciones existen básicamente dos tipos de información, una de carácter *persistente* que constituye la base de conocimiento del sistema y con la cual muchos de los procesos asociados con la Plataforma Conceptual de Agentes pueden funcionar, esta información por lo general también es necesaria en los procesos, actividad, tarea o roles desempeñados por cualquier entidad, según su naturaleza es clasificada como pública o privada, por ejemplo los registros AID en la PCA son públicos para el AMS o Sistema Gestor de Agentes pero no para las demás entidades, esta noción de pertenencia y propiedad de la información es un rasgo que debe tenerse en cuenta en ambientes abiertos y de libre acceso como los tratados por MR-DSOA.

El otro tipo de información es de carácter *volátil*, porque es generada en tiempo de ejecución por los agentes y sus interacciones. Esta información es muy importante en el diseño del servicio ya que permite descomponer la funcionalidad total en varios agentes especializados que trabajan en conjunto. El modelamiento de las relaciones y del flujo de información es lo que permite la descomposición del servicio en entidades finas y mantiene unido a los componentes del servicio en una estructura coherente. La información volátil se transfiere dinámica entre agentes y sirve de base para la ejecución de la mayoría de los procesos en el servicio

2.10.6 Relaciones entre agentes de información

Una relación es una conexión entre dos o más agentes que necesitan interactuar para alcanzar una meta u objetivo. Por ejemplo un mensaje ACL del tipo cfp en un servicio de subasta en línea establece una relación temporal o permanente según sea el caso entre un agente del tipo vendedor con uno o varios agentes del tipo comprador.

Los protocolos de interacción son el mecanismo que emplean los agentes para establecer relaciones e intercambiar información, como se menciono anteriormente los mensajes ACL son el medio en el que viaja la información. Los protocolos de interacción estandarizados por FIPA son:

- Request Interaction Protocol Specification.
- Query Interaction Protocol Specification.
- Request When Interaction Protocol Specification.
- Contract Net Interaction Protocol Specification.
- Iterated Contract Net Interaction Protocol Specification.
- English Auction Interaction Protocol Specification.
- Dutch Auction Interaction Protocol Specification.
- Brokering Interaction Protocol Specification.
- Recruiting Interaction Protocol Specification.
- Subscribe Interaction Protocol Specification.
- Propose Interaction Protocol Specification.

En la figura 2.27 se utiliza una versión modificación de los diagramas de secuencia descritos en UML 2.0, con los cuales es posible definir protocolos de interacción, ya que UML esta orientado a objetos, se hace necesario extender y modificar alguno de sus elementos de la especificación original para que sean útiles en el paradigma de agentes, por ejemplo: las flechas de mensajes asíncronos indican actos comunicativos o mensajes ACL, también contemplan la inclusión de cardinalidad en los mensajes, la posibilidad de modelar comportamientos de grupos de agentes, roles y etc. Estas modificaciones son conocidas como AUML [FIPA 2003].

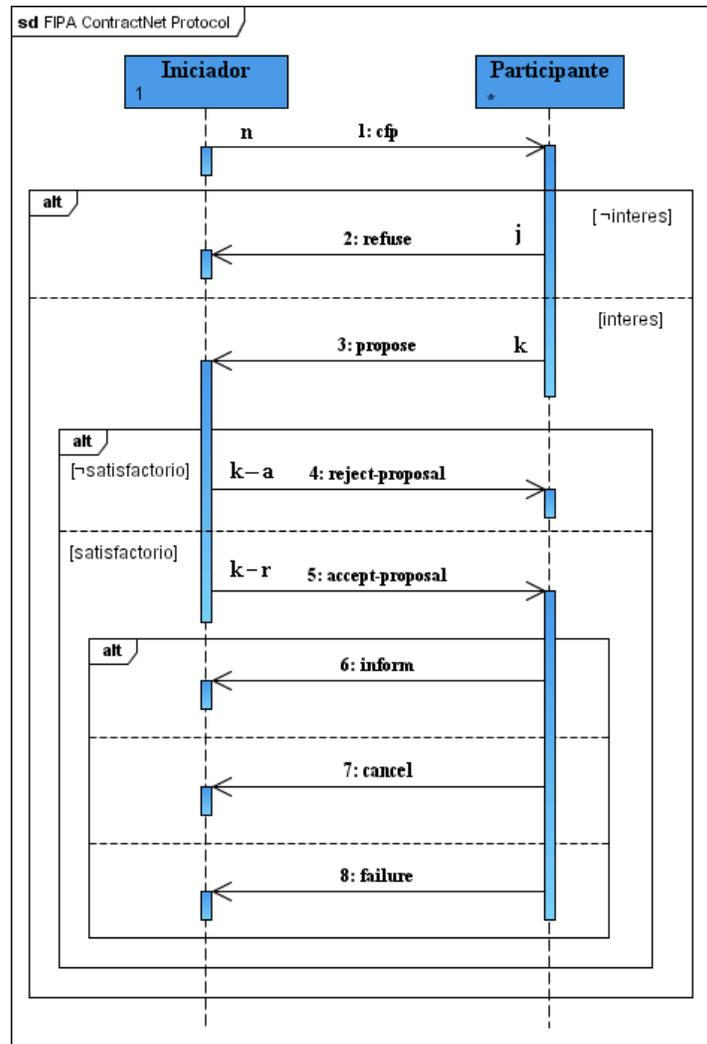


Figura 2.27. Protocolo de interacción.

En la anterior figura se pueden observar expresiones que indican cardinalidad como: n, j, k, k-a y k-r, ya que UML no concibe el envío y recepción de mensajes asíncronos multicast, se hace necesario adicionar esta notación en los diagramas de secuencia. Por ejemplo la letra n sobre el mensaje cfp (call for proposals) indica que este mensaje es de tipo multicast y va dirigido a un grupo de agentes participantes, de ese grupo j mensajes serán del tipo refuse y otros tantos (k) del tipo propose, los que rechazan la propuesta ya no participan en la interacción y para estos la relación ha terminado. Los que envían una propuesta esperan a que se les informe si su propuesta fue aceptada o rechazada (reject-proposal k-a /accept-proposal k-r). Los k-r agentes que les fue aceptada su propuesta entonces pueden emitir uno de los siguientes tres tipos de mensajes: cancel, inform o failure [Bauer 2005].

2.10.7 Intercambio de información

Un rol es una descripción abstracta de la funcionalidad esperada de una entidad, en este caso un agente de información. Durante la ejecución de un rol es probable que se requiera de información adicional a la que el agente posee internamente, ya sea de tipo persistente o volátil, cual fuera el caso es el agente el encargado de conseguirla, porque son a estos a los que se les asocian los roles, en consecuencia son los encargados de consultar la base de conocimiento del sistema o relacionarse con otros para obtenerla, el modelo identifica estas interacciones y responde a cuestionamientos como: ¿Que información necesita un agente o un rol?, ¿De donde proviene?, ¿Cuales son actos comunicativos?, ¿el tipo de mensaje?, su cardinalidad y etc.

Los diagramas de secuencia se pueden emplear en dos formas una ellas es como se muestra en la figura 2.27 en donde las columnas representan roles (iniciador y participantes) o como en la figura 2.28 en donde las columnas representan a un agente, con uno o mas roles. Este tipo de diagramas son importantes para el modelo de información ya que permiten modelar el comportamiento de la información en el sistema.

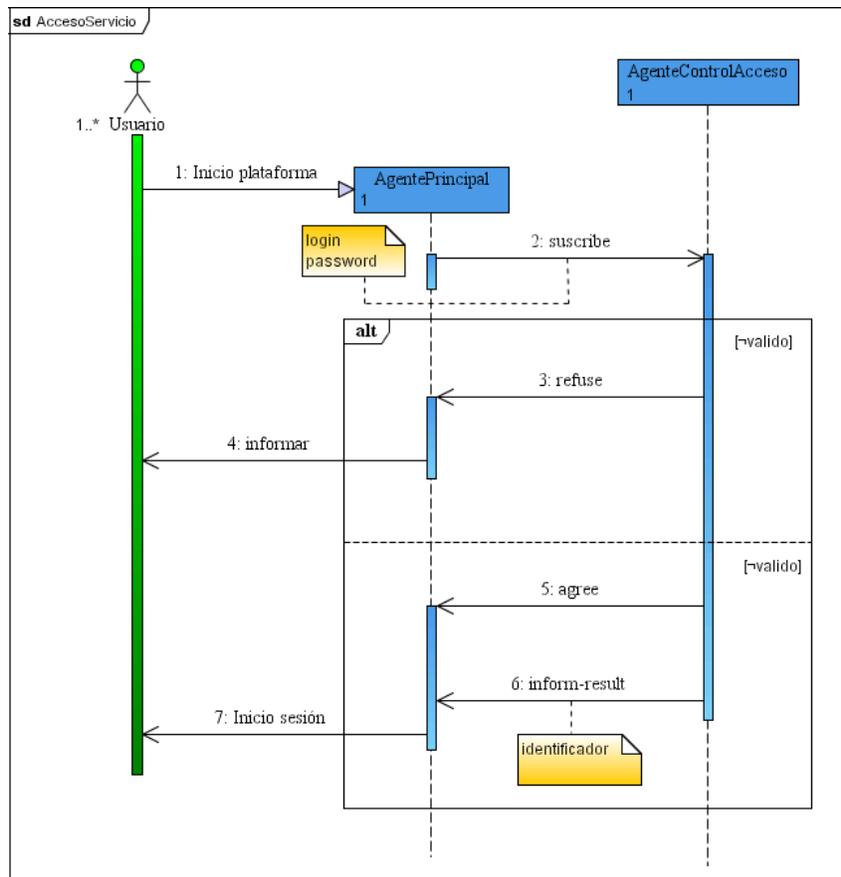


Figura 2.28. Diagrama de intercambio de información

2.10.8 Ontologías

Aunque el termino ontología se utiliza desde hace varios siglos es con el surgimiento de la inteligencia artificial que la controversia entorno a este comenzó, ya que fue empleado tradicionalmente en la filosofía para hablar de temas relacionados con la existencia, específicamente en la metafísica. Pero sucesivas investigaciones de tipo técnico contextualizaron al término como una descripción formal y explícita de los conceptos y relaciones que pueden existir en un ambiente donde es necesario compartir información [Noy 2001].

Una ontología define el vocabulario común necesario para compartir la información en un dominio e incluye la definición de intérpretes y sus relaciones. Las razones para desarrollar una ontología son:

- Facilitar el entendimiento de las estructuras de la información.
- Facilitar el reuso del conocimiento.
- Separar el conocimiento del dominio -del- conocimiento operacional.
- Análisis del conocimiento de un dominio

Uno de los principales objetivos de una ontología se permitir que varios agentes *entiendan la estructura de la información*. Por ejemplo, suponga un servicio de búsqueda y recuperación de información, si ninguno de los terminales publica una ontología los agente no podrían interpretar los datos almacenados en cada dispositivo, en el caso contrario de que todos utilicen la misma o que exista un servidor de ontologías los agentes podrían extraer y lo mas importante interpretar la información.

Permitir el *reuso del conocimiento*. Por ejemplo, FIPA define ontologías para representar diferentes conceptos relacionados con los agentes y su ambiente, estos podrían utilizarse en varias aplicaciones o servicios, permitiendo que agentes foráneos accedan y utilicen el conocimiento del dominio.

Otro uso común dado a las ontologías es el de *separar el conocimiento del dominio -del- conocimiento operacional*, considere el caso hipotético del desarrollo de agentes que implementan roles ligeramente diferentes, por ejemplo el tipo de usuarios que se suscriben a un servicio, debido a que no todos los dispositivos móviles son iguales, unos usuarios tendrán la posibilidad de acceder a servicios mas elaborados que otros, quizás no deseen invertir en determinados servicios y etc. Cada posibilidad se caracterizaría como un perfil $A_1, A_2, A_3, \dots, A_n$. En este punto hay dos posibilidades: diseñar y desarrollar roles por cada perfil o diseñar e implementar un rol que interprete una ontología y a través de esta manejar la información de cada uno de los perfiles. En este ejemplo solo se han considerado dos factores: dispositivos y disponibilidad económica pero en un sistema de telecomunicaciones pueden ser muchos mas.

A continuación se presenta una guía con las principales actividades que se deben seguir en el diseño y especificación de una ontología.

1. *Determine el dominio y contexto de la ontología:* formule y conteste las siguientes preguntas:

- ¿Cuál es el dominio que la ontología desea cubrir?
- ¿Para que se va a utilizar la ontología?
- ¿A que preguntas la información en la ontología debe proporcionar respuesta?
- ¿Quién usa y alberga la ontología?

2. *Considere el reuso de ontologías:* Con el auge de Internet y de los servicios web las ontologías han cobrado gran relevancia en estos últimos años, por tal razón hay varios organismos públicos y privados que ofrecen ontologías en formato digital con propósitos de estandarización. Esta información es muy útil ya que puede utilizarse como base para la construcción de las ontologías del servicio, otra ventaja de reutilizar ontologías es que facilitamos la integración del servicio con otros. Hay muchas librerías de ontologías disponibles en Internet por ejemplo Ontolingua ontology library, DAML y etc.

3. *Enumeración de los términos más importantes:* Durante el diseño de una ontología es de gran ayuda hacer una lista con todos los conceptos que debería relacionar, estos se pueden identificar del dominio o del contexto en que va a operar la ontología.

4. *Implementación de la ontología:* dependiendo de la forma en que se especifique la ontología su materialización puede variar (objetos o XML).

2.11 MODELO DE INGENIERÍA

El modelo de ingeniería muestra como se establecen las interacciones entre agentes y como los recursos del sistema lo permiten. En contraste al modelo computacional que dice cuando y por qué los agentes deben interactuar en el modelo de ingeniería se especifica como es que estas entidades logran comunicarse, ya que entre agentes puede haber una gran cantidad de sistemas, dominios administrativos de negocios, redes y etc. De ahí la relación entre este modelo (ingeniería) y el empresarial.

El lenguaje de ingeniería define los conceptos necesarios para describir la infraestructura requerida para soportar a los servicios y las reglas para:

- Estructurar canales de comunicación entre agentes.
- Estructurar sistemas para propósitos de gestión de recursos.

2.11.1 Clusters, cápsulas y nodos

La implementación software de los agentes, o sea el código, es instalado en *nodos*, que representan a un equipo hardware con cierta capacidad computacional y que pertenecen a un dominio administrativo de negocio, el cual puede no ser gestionado por la misma autoridad a la que pertenece el servicio; a pesar de esto el nodo debe permitir la interacción entre agentes locales y otros foráneos.

Dentro de un nodo puede haber varias *cápsulas*, cada una de las cuales está asociado a un agente especial llamado gestor de cápsula que maneja las interacciones internas y externas de esta. Una cápsula típicamente contendrá muchos agentes; tal agrupamiento reduce las interacciones.

El agrupamiento más pequeño de agentes se denomina *cluster* y está contenido dentro de una cápsula. Los agentes están agrupados en un cluster para facilitar su manipulación. Clusters, cápsulas y nodos se muestran en la siguiente figura.

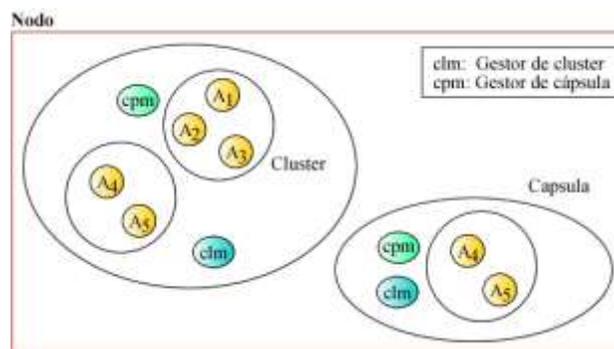


Figura 2.29. Clusters, cápsulas y nodos

Las acciones sobre un cluster son iniciadas y controladas por un agente gestor de cluster.

2.11.2 Canales

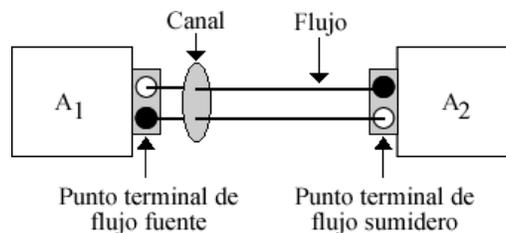


Figura 2.30. Canal

Cuando interactúan agentes ubicados en diferentes clusters surge la necesidad del establecimiento de canales para el intercambio de información, esta responsabilidad recae en algunos casos sobre la implementación de la Plataforma Conceptual de Agentes (PCA) o en los agentes figura 2.30.

La PCA controla el ciclo de vida de los agentes, su creación y eliminación. Todas estas operaciones son

transparentes al servicio. El conjunto de mecanismos que permiten la interacción entre agentes es lo que se denomina canal, el cual está constituido por un grupo de entidades que interactúan independientemente del servicio. Estas entidades son:

- Stubs: Están relacionados con la información transmitida en una interacción.
- Binders: Mantienen las asociaciones entre los agentes.
- Entidades de protocolo: Gestionan la comunicación.

2.11.3 Interfaces de ingeniería

Las interfaces de ingeniería son las que permiten la interacción entre nodos, es decir, entre equipos terminales que están ubicados dentro de un mismo dominio administrativo de negocio o en distintos.

2.11.4 Establecimiento del canal

Para establecer un enlace entre agentes, es necesario identificar y describir sus interfaces; para ello se utiliza una referencia a la interfaz de ingeniería, la cual puede ser pasada a través de una o más interacciones entre agentes. Cuando este obtiene dicha referencia, puede establecer un enlace con la interfaz sin ningún tipo de información adicional.

Es posible que sea necesaria una configuración del canal cuando las interfaces de ingeniería no son compatibles, es decir, cuando no tienen la misma representación de datos. La comunicación entre un par de agentes que están ubicados en plataformas diferentes, se establece tal como se describe a continuación:

2.11.4.1 Establecimiento de un canal entre dominios administrativos de negocio

Teniendo en cuenta que cada dominio administrativo de negocio puede contar con su propia Plataforma Conceptual de Agentes la comunicación entre estos debe estar regida por un estándar que permita una interacción compatible.

Bajo estas circunstancias un agente tiene tres opciones para enviar un mensaje a otro agente que reside en una plataforma remota (Figura 2.31); tal mensaje puede ser de establecimiento de canal o simplemente una transmisión de información normal:

1. Un agente envía el mensaje al ACC (Canal de Comunicaciones de Agente) local utilizando una interfaz propietaria o estándar. El ACC envía el mensaje a su similar remoto utilizando un MTP (Protocolo de Transporte de Mensajes) adecuado. El ACC remoto entregará el mensaje.
2. El agente A envía el mensaje directamente al ACC ubicado en la plataforma remota en la cual el agente B

reside. Para utilizar este método, el agente A debe tener acceso a una de las interfaces del ACC remoto.

3. El agente A envía el mensaje directamente al agente B utilizando un mecanismo de comunicación directo. La transferencia, el direccionamiento, almacenamiento y cualquier error en los mensajes debe ser manipulado por los agentes emisor y receptor.

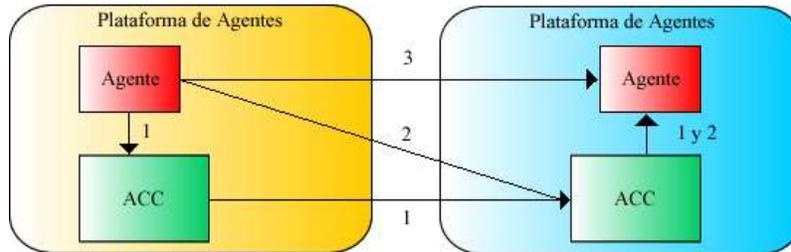


Figura 2.31. Métodos de comunicación entre plataformas

Tal como podemos apreciar, la anterior descripción muestra a bajo nivel la transmisión de mensajes ACL entre plataformas diferentes que cumplen con el estándar FIPA; con lo anterior se pretende hacer una aclaración sobre los elementos que se deben involucrar cuando se especifica el modelo de ingeniería de una aplicación basada en agentes.

2.11.5 Interceptores

Los interceptores fueron diseñados para utilizar los requerimientos identificados en la especificación empresarial y con ellos poder realizar la federación de recursos o la creación de dominios. La noción de interceptor es similar a la de “gateway”; estos se ubican en la frontera de un dominio y permiten la interacción entre dominios tecnológicos o administrativos, siempre y cuando estén permitidas por los contratos suscritos entre las administraciones. Según se definió en el modelo empresarial un dominio administrativo es aquel en el cual coexisten uno o varios sistemas de comunicación bajo la misma autoridad. Por otra parte, un dominio tecnológico es un sistema que opera bajo una tecnología particular.

Dentro de un dominio de tecnología, los agentes tendrán representaciones de datos, protocolo, nombres y direccionamiento idénticos. Entre dominios de tecnología diferentes, son necesarias la conversión de protocolos y la traducción de nombres. Cuando dos dominios de tecnológicos se relacionan, existe la posibilidad de fusionarlos, es decir, los agentes en cada dominio son habilitados para usar tanto su dominio de tecnología propio como el externo. Donde esto ocurre, la frontera de tecnología desaparece.

Los interceptores cubren el caso en el cual uno o los dos dominios son incapaces de modificar su tecnología, consecuentemente, debe ocurrir una interacción en la frontera que proporcione conversión de protocolo y traducción de nombres.

La traducción de datos y de protocolo sobre las fronteras de tecnología es ejecutada por un tipo especial de

interceptor, denominados interceptores en línea, los cuales están involucrados en todas las interacciones entre agentes sobre la frontera. Por eficiencia, solo un interceptor en línea es utilizado por cada frontera de tecnología, figura 2.32.

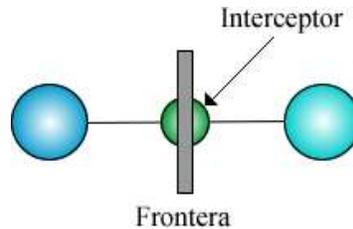


Figura 2.32. Interceptor en Línea-Frontera de tecnología

Otro tipo de interceptores son los interceptores de frontera administrativa, los cuales están dentro de una administración y son los responsables de la protección de ésta. Por ejemplo, un interceptor de frontera administrativa podría ser utilizado en la verificación de información que contiene permisos de seguridad. También puede comunicarse con un interceptor similar en el otro dominio para intercambiar información, tal como llaves criptográficas y comprobar información administrativa antes de traducirla, figura 2.33. Una vez realizada la traducción, puede ser utilizada por diferentes interacciones entre agentes.

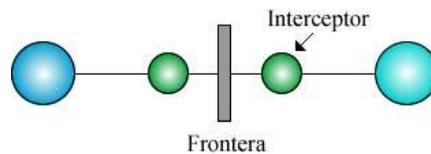


Figura 2.33. Interceptor dividido-Frontera administrativa

Donde una frontera administrativa y tecnológica coexisten, tanto el interceptor en línea como el de frontera administrativa deben ser utilizados (combinación de las dos anteriores figuras). Alternativamente, el interceptor en línea puede estar dividido, de manera que hay dos interceptores en línea, uno en cada dominio administrativo para soportar las relaciones de la unión entre administraciones.

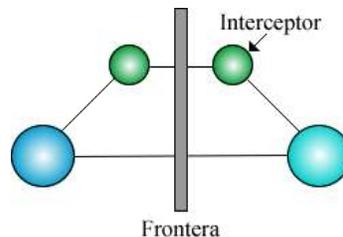


Figura 2.34. Interceptor dividido-Frontera combinada

2.12 MODELO DE TECNOLOGÍA

El modelo de tecnología describe la implementación del sistema en términos de una configuración de agentes

que representan los componentes hardware y software, siendo restringido por el costo y disponibilidad de los agentes de tecnología (productos hardware y software). De esta manera, este modelo proporciona un enlace entre los modelos empresarial, de información, computacional y de ingeniería con la implementación real.

La meta de este modelo es proporcionar información extra necesaria para la implementación. En esta se seleccionan las tecnologías que soportaran a los componentes diseñados y los mecanismos de comunicación, tal selección completa la especificación del sistema.

Las selecciones hechas determinan el desempeño de las interacciones, la calidad del servicio entre otras, por ejemplo la selección de motores de base de datos, tecnologías de transporte, selección de componentes software, tales como: plataformas de agentes y aplicaciones de apoyo.

CAPITULO III. DISEÑO UN SERVICIO DE MENSAJERIA MULTIMEDIA MMS EMPLEANDO MR-DSOA V 1.0

3.1 INTRODUCCIÓN

La implementación completa de un servicio como se describe en MR-DSOA es extremadamente larga, ya que considera aspectos reales del funcionamiento de un sistema de telecomunicaciones, que si bien se deben tener en cuenta durante el diseño e implementación de un servicio su reproducción en una aplicación prototipo resultaría extremadamente difícil demandante en tiempo, recursos y etc. Un claro ejemplo de esta situación es lo concerniente a la gestión del sistema, la cual debe hacerse extensible a equipos y aplicaciones, para ello se tienen que usar conceptos de TMN (Telecommunications Management Network) y desarrollar todo un software basado en agentes que permita a un administrador gestionar su sistema. Desarrollar aplicaciones de este tipo es muy difícil, considerando que los proveedores de estas tecnologías utilizan sistemas propietarios, impidiendo a terceros operarlos a través de aplicaciones distintas a las que ellos suministran, sin contar que las plataformas de agentes no las soportan. La calidad del servicio (QoS) es un tema que frecuentemente se menciona en el modelo y ocurre exactamente lo mismo que en el caso anterior, por ejemplo: considerar la influencia del ancho de banda en la calidad del servicio, implica el uso de aplicaciones y equipos especializados que permitan gestionarlo. El modelo habla de estos temas y otros tantos, por que son importantes para el desarrollo de servicios telemáticos pero su implementación trasciende más allá de los objetivos del proyecto ya que esta funcionalidad se supone que debe existir y estar disponible para los servicios que se vayan a implementar. Por tal motivo este diseño esta centrado en los aspectos esenciales del paradigma de agentes dejando a un lado estos temas y concentrándose en demostrar la validez del modelo.

El desarrollo de un modelo no estaría completo hasta que se muestre como emplearlo en una situación real, o sea, en el diseño de un servicio telemático, puesto que MR-DSOA fue concebido para el desarrollo de servicios para sistemas de telecomunicaciones, su utilidad aumenta en la medida que la aplicación objetivo este enfocada en este tema, por tal motivo se ha seleccionado el servicio de mensajería multimedia (MMS), ya que su diseño demanda el uso de gran parte de los conceptos tratados en el modelo, además incluye la parte móvil. Dado que este servicio es muy popular una persona cualquiera puede comprender sin mucho esfuerzo de que se trata, ¿Qué es?, ¿Para que sirve?, ¿Que debería hacer? y etc. Facilitando aun más la comprensión del diseño. El servicio es lo suficientemente complejo como para demostrar ¿Cómo emplear la tecnología de agentes en el desarrollo de servicios telemáticos?, logrando un equilibrio entre la parte técnica y pedagógica.

Más allá de la aplicación lo importante es demostrar que el modelo sirve para desarrollar servicios y de que es una herramienta realmente efectiva.

El modelo contempla la aplicación de siete grandes fases, las cuales se detallaran claramente en el transcurso del capítulo, puesto que el servicio es demostrativo no se tendrán en cuenta algunas etapas típicas del proceso de desarrollo software, solo se aplicara MR-DSOA.

3.2 DESCRIPCION DEL SERVICIO

El servicio de mensajería multimedia MMS, permitir la transferencia de mensajes entre usuarios a través de dispositivos móviles empleando la tecnología de agentes. Los usuarios del servicio pueden enviar o recibir mensajes de texto o imágenes. Si un usuario no tiene imágenes almacenadas en su dispositivo, el sistema provee un álbum de recursos multimedia que permite a un remitente observar y seleccionar alguna de ellas para posteriormente envía.

En el momento que un usuario reciba un mensaje el sistema le notificara la llegada de este y lo almacenara para que pueda consultarlo en cualquier momento, para ello el servicio provee los mecanismos de gestión que permiten eliminar, observar o conservar algún mensaje en la cuenta.

Las personas que deseen hacer uso del servicio debe estar registradas, si no es así pueden llenar un formulario de inscripción y suscribirse como usuarios, cada vez que ingresen al sistema deben validarse por medio de un login y password, el cual será inspeccionado y dependiendo de su veracidad se concederá o no el acceso al servicio.

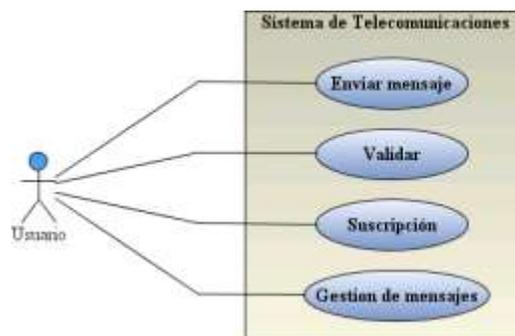


Figura 3.1. Casos de uso.

La figura 3.1 muestra los casos de uso del sistema, básicamente un usuario puede enviar un mensaje de texto o multimedia, solicitar el acceso al sistema (validarse), suscribirse al servicio de mensajería multimedia y gestionar la cola de mensajes. Dependiendo de las capacidades de su dispositivo el servicio se adaptara a estas características ofreciéndole mejor presentación (interfases graficas) y mayor funcionalidad.

3.3 ARQUITECTURA DEL SISTEMA

La arquitectura propuesta es muy similar a la de un sistema de telecomunicaciones convencional, con puntos de acceso inalámbrico en la última milla que permiten a los usuarios móviles ingresar al sistema y redes fijas para enlazar las estaciones base o nodos, aunque en la figura 3.2 se muestra solamente tres punto de acceso inalámbricos estos pueden ser muchos mas proporcionando cobertura a grandes extensiones de tierra, por ejemplo ciudades, vías y etc.

En la figura 3.2, se puede observar el diagrama simplificado de la arquitectura física del sistema y la disposición e interconexión de los recursos hardware. El proveedor de conectividad es la entidad encargada de enlazar todos los nodos y equipos terminales, permitiendo una libre interacción entre ellos, este provee los puntos acceso inalámbrico para que los dispositivos móviles puedan acceder al servicio de mensajería por medio de tres tecnologías: BLUETOOTH, WLAN y GPRS. Los usuarios que se conectan al sistema por medio de estas interfases puedan acceder a los recursos en la red fija gracias a una gateway que realiza la conversión de los protocolos. Debido a que el modelo empresarial exige caracterizar a cada segmento del sistema como una organización o empresa a esta entidad se la ha denominado *LinkAll company*,

El minorista o *MMS company*, es la organización encargada de proveer la lógica del servicio y de ofrecerle a los usuarios esta funcionalidad, como se menciona en la sección 3.2 (descripción del servicio), el minorista gestiona los usuarios, permitiendo que estos se suscriban y envíen mensajes. Esta entidad garantiza la circulación de los mensajes multimedia por el sistema hasta su destino.

El proveedor externo de servicios o *Multimedia company* se encarga de proporcionar el álbum o catalogo de recursos multimedia a los usuarios del servicio de mensajería, con el objetivo de que estos puedan incrustar en los mensajes imágenes. Para el usuario la relación minorista - proveedor externo de servicios debe ser transparente.

En un sistema de telecomunicaciones un minorista se puede relacionar con otros minoristas o con varios proveedores externos de servicios, para ampliar su portafolio de funcionalidades, en el caso de que esto ocurra, las operaciones derivadas de estas alianzas deben ser totalmente transparentes para el usuario, es decir, este tiene que percibir al servicio como una estructura homogénea, aun que en realidad esta involucrando a dos o mas empresas distintas.

El intermediario o *Broker company* se dedica exclusivamente a mantener actualizados los registros que permite ubicar los puntos de referencia o de acceso a los diferentes dominios administrativos de negocio. El intermediario debe responde a todas las consultas, ya que esta información permite encontrar a los demás actores del sistema (proveedores de servicios, de conectividad, usuarios, minoristas y etc).

Los usuarios acceden a los servicios provistos por el sistema a través de sus dispositivos de computo personales, tales como PDA's, portátiles, handheld, pocket PC, smart phones, teléfonos celulares y etc. El único requisito es que puedan ejecutar código java, ya sea MIDP 1.0 / CLDC 1.0, MIDP 2.0 / CLDC 1.1 o Personal java y por supuesto que tengan alguna interfaz para red inalámbrica.

Cada uno de los equipos en la arquitectura del sistema (figura 3.2) deben tener instalada una plataforma de agentes que implemente las funcionalidades de la PCA (sección 2.6).

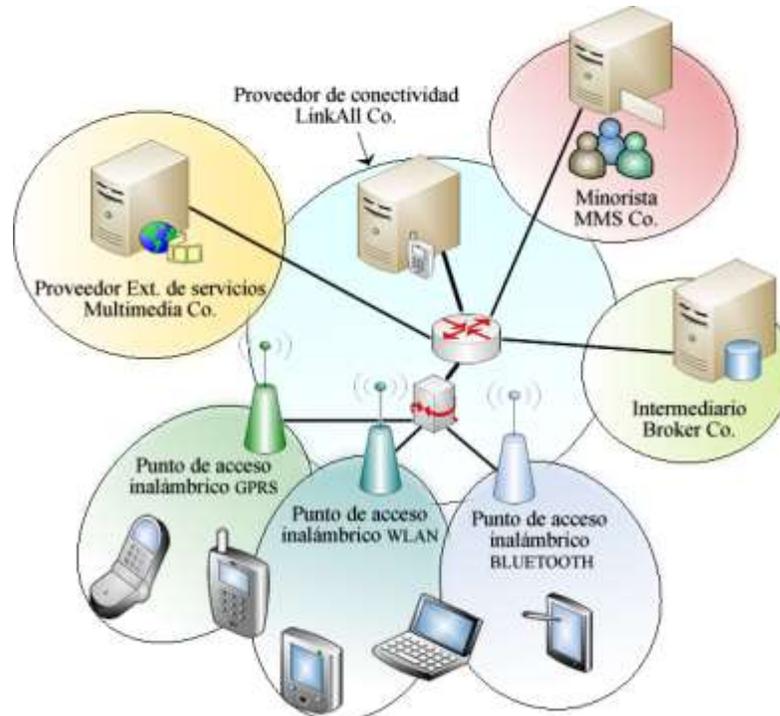


Figura 3.2. Arquitectura del sistema.

En la arquitectura del sistema, se identifican cuatro dominios tecnológicos, tres corresponden a las tecnologías de acceso inalámbricas y el otro a la red fija:

- $DT_{BLUETOOTH}$
- DT_{WLAN}
- DT_{GPRS}
- $DT_{ETHERNET}$

Los dominios administrativos de negocio son:

- Broker Co. con el rol de negocio: Intermediario.
- LinkAll Co. con el rol de negocio: Proveedor de conectividad.
- MMS Co. con el rol de negocio: Minorista.
- Multimedia Co. con el rol de negocio: Proveedor externo de servicios.
- Los usuarios con el rol de negocio: Usuarios.

3.4 MODELO EMPRESARIAL

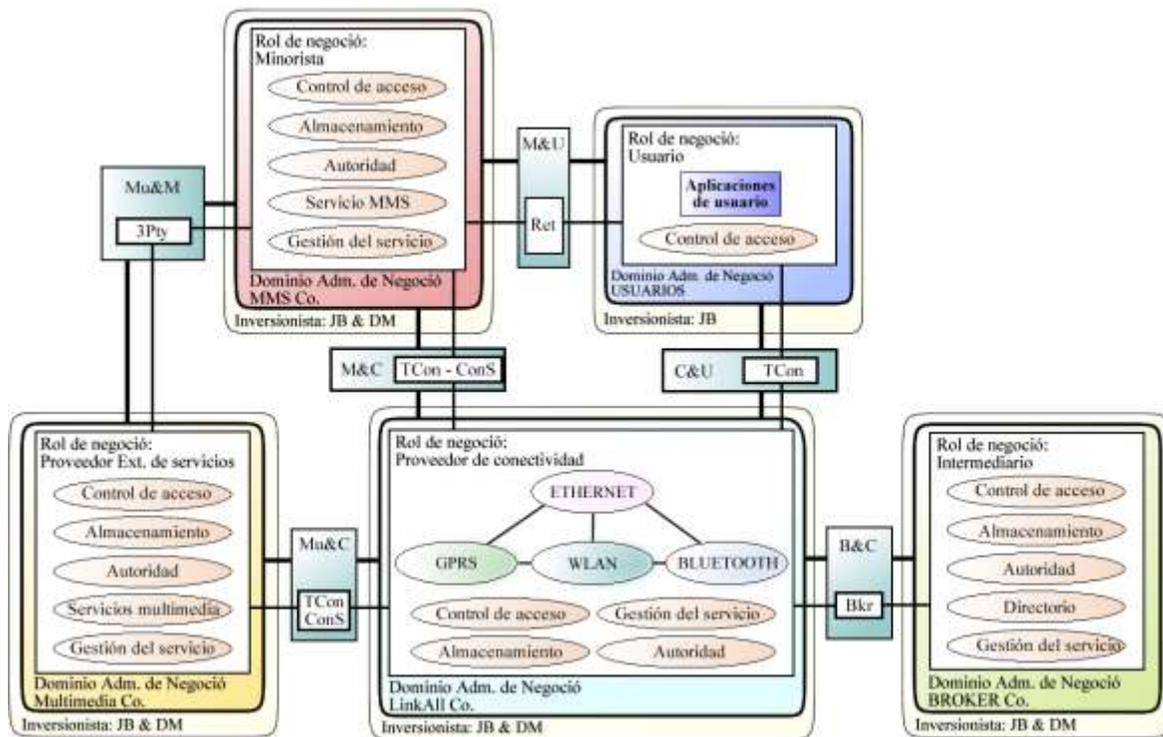


Figura 3.3. Modelo empresarial.

En la sección 2.8 se pueden encontrar los conceptos necesarios para elaborar el modelo empresarial del servicio, cada uno de los roles y relaciones de negocio están detallados en este capítulo. El modelo empresarial solo concibe cinco roles de negocio los cuales han sido caracterizados previamente y se utilizan como patrón para su elaboración, por esta razón aparte de la descripción que se le dará a cada uno de los dominios administrativos de negocio se debe tener en cuenta lo mencionado en la sección 2.8.3.10.1. La figura 3.3 corresponde al diagrama empresarial del servicio de mensajería multimedia MMS y se identificaron los siguientes dominios administrativos de negocio:

3.4.1 Dominios administrativos de negocio

3.4.1.1 Minoristas

El dominio administrativo MMS o minorista esta compuesto por los siguientes subdominios:

- *Control de acceso*: Permite solamente a los usuarios autorizados acceder al servicio y a la autoridad del dominio acceder a las funciones de gestión, este dominio actúa como un muro de fuego en el que las políticas restrictivas definidas por la autoridad del sistema se cumplen, en este caso son restricciones de acceso pero podrían ser de otro tipo.
- *Almacenamiento*: Este dominio es encarga de almacena toda la información que requieren los agentes para desempeñar sus tareas, por ejemplo: datos de usuarios, claves de acceso, mensajes y etc. El tipo de almacenamiento es persistente.
- *Autoridad*: Se encarga de tomar decisiones que afectan el comportamiento del sistema, por ejemplo ordenes de conexión y desconexión a proveedores externos de servicios y de conectividad.
- *Servicio MMS*: Provee la lógica del servicio, los agentes que pertenecen a este dominio se aseguran de que los mensajes lleguen a su destino y que los usuarios puedan enviarlos, en otras palabras implementan gran parte de la funcionalidad que los usuarios perciben del sistema.
- *Gestión del servicio*: Permite al administrador del dominio fijar ciertos parámetros del servicio, por ejemplo: gestión de usuarios, crear, eliminar o modificar registros.

Responsabilidades de alto nivel:

- Provee la lógica del servicio de MMS.
- Autorizar a los usuarios validos para que utilicen el servicio.
- Provee acceso a los administradores autorizados para hacer uso de los servicios de gestión.
- Control y gestión de los flujos de información soportados por el proveedor de conectividad.
- Registro de los servicios y puntos de referencia ante los intermediarios.

3.4.1.2 Proveedor de conectividad

El proveedor de conectividad gestiona la red de transporte que soporta los flujos de información., ofreciendo una interfaz a cada uno de los dominios administrativos de negocio para su comunicación. El proveedor de conectividad posee cuatro redes distintas las cuales corresponden a igual número de dominios tecnológicos.

Este dominio administrativo de negocio se encarga de realizar la conversión de direcciones, protocolos y tramas de una red a otra, permitiendo el flujo de la información entre puntos terminales. El proveedor de conectividad esta compuesto por los siguientes subdominios:

- *Control de acceso:* Permite solamente a los usuarios autorizados acceder a las redes de comunicación y por ende al servicio de conectividad, además revela las funciones de gestión a los administradores autorizados.
- *Almacenamiento:* Este dominio es encarga de almacena toda la información que requieren los agentes para desempeñar sus tareas.
- *Autoridad:* Se encarga de tomar decisiones que afectan el comportamiento del sistema.
- *Gestión del servicio:* Permite al administrador del dominio fijar ciertos parámetros del servicio, por ejemplo gestión de usuarios crear, eliminar o modificar registros de usuario y etc.

Los dominios tecnológicos $DT_{BLUETOOTH}$, DT_{WLAN} , DT_{GPRS} y $DT_{ETHERNET}$, son las redes que transportan la información entre los distintos actores del servicio, sus responsabilidades de alto nivel son:

- Configuración y gestión de asociaciones entre puntos terminales de una red.
- Establecimiento de enlaces.

3.4.1.3 Proveedor externo de servicios

Este dominio administrativo de negocio esta compuesto por los siguientes subdominios:

- *Control de acceso:* Permite a los minoristas autorizados acceder a los servicios provistos, lo mismo ocurre con los administradores y las funciones de gestión.
- *Almacenamiento:* Este dominio es encarga de almacena los recursos multimedia e información en general, por ejemplo: datos de minoristas, claves de acceso, imágenes y etc.
- *Autoridad:* Se encarga de tomar decisiones que afectan el comportamiento del sistema, por ejemplo ordenes de conexión y desconexión de minoristas y proveedores de conectividad.
- *Servicio multimedia:* Provee la lógica del servicio, los agentes que pertenecen a este dominio son los encargados de acceder y extraer las imágenes del dominio de almacenamiento.
- *Gestión del servicio:* Permite al administrador del dominio fijar ciertos parámetros del servicio, por ejemplo gestión de usuarios: crear, eliminar o modificar registros de usuario, gestión de recursos multimedia: adición y eliminación de imágenes.

Los requerimientos de alto nivel para este dominio administrativo de negocio son:

- Registro de los servicios y puntos de referencia ante los intermediarios.
- Establecimiento de relaciones para la provisión de servicios.
- Ubicación de minoristas.
- Provisión de recursos multimedia.
- Gestión de contenidos.

3.4.1.4 Intermediario

Todos los inversionistas que están desempeñando algún rol de negocio por ejemplo: minorista, proveedor externo de servicios, proveedor de conectividad o etc. Deben registrarse por lo menos a un intermediario, para que sus servicios puedan ubicarse fácilmente en el sistema. Es obligación de los intermediarios mantener actualizados y disponibles los registros que permiten ubicar servicios y organizaciones en el sistema (dominios administrativos de negocio), además de la información básica que permita acceder y usar su funcionalidad.

Cada agente que presta algún tipo de servicio debe suministrar la siguiente información (tabla 3.1) para que el intermediario lo pueda registrar en sus bases de datos.

Tabla 3.1. Registro de puntos de acceso y servicios.

Información	
Nombre del agente	
Ontología	
Lenguaje	
Protocolo de interacción	
Servicios del agente	Nombre
	Tipo
	Propietario
	Lenguaje
	Ontología
	Protocolo
Propiedad	
Tiempo de disponibilidad	

Este dominio administrativo de negocio esta compuesto por:

- *Control de acceso*: Aplica las políticas de acceso al servicio.
- *Almacenamiento*: Este dominio es encarga de almacenar la ubicación de los dominios administrativos de negocio.
- *Autoridad*: Se encarga de tomar decisiones que afectan el comportamiento del sistema, por ejemplo ordenes de conexión y desconexión al proveedor de conectividad.
- *Directorio*: Los agentes que pertenecen a este dominio son los encargados de acceder y extraer la información de la ubicación de los dominós administrativos de negocio en las base de datos.
- *Gestión del servicio*: Permite al administrador del dominio gestionar los registros.

Responsabilidades:

- Funciones de páginas blancas: Dado el nombre de un dominio administrativo de negocio, retorna la dirección de la interfaz computacional del punto de referencia.
- Función de páginas amarillas: Dado un conjunto de atributos busca la dirección de la interfaz del agente que provee el servicio con las características solicitadas.
- Provee listas de dominios administrativos de negocio según su rol.
- Dada la dirección de un punto de referencia se retorna el nombre del dominio administrativo de negocio.
- Introduce, actualiza y remueve información de los dominios administrativos de negocio y de los servicios ofrecidos.

3.4.1.5 Usuario

El usuario es la persona que se inscribe al servicio y accede inalámbricamente al sistema, lo puede hacer a través de un dispositivo equipado con una tarjeta WIFI, BLUETOOTH o GPRS. El dominio de control de acceso es el que le permite validarse ante el proveedor de conectividad y el minorista, haciendo las veces de punto de referencia o contacto, este dominio es empleado por las aplicaciones de usuarios para interactuar con el sistema de telecomunicaciones. Entre las aplicaciones de usuario esta la plataforma de agentes y por supuesto los agentes.

3.5 MODELO COMPUTACIONAL

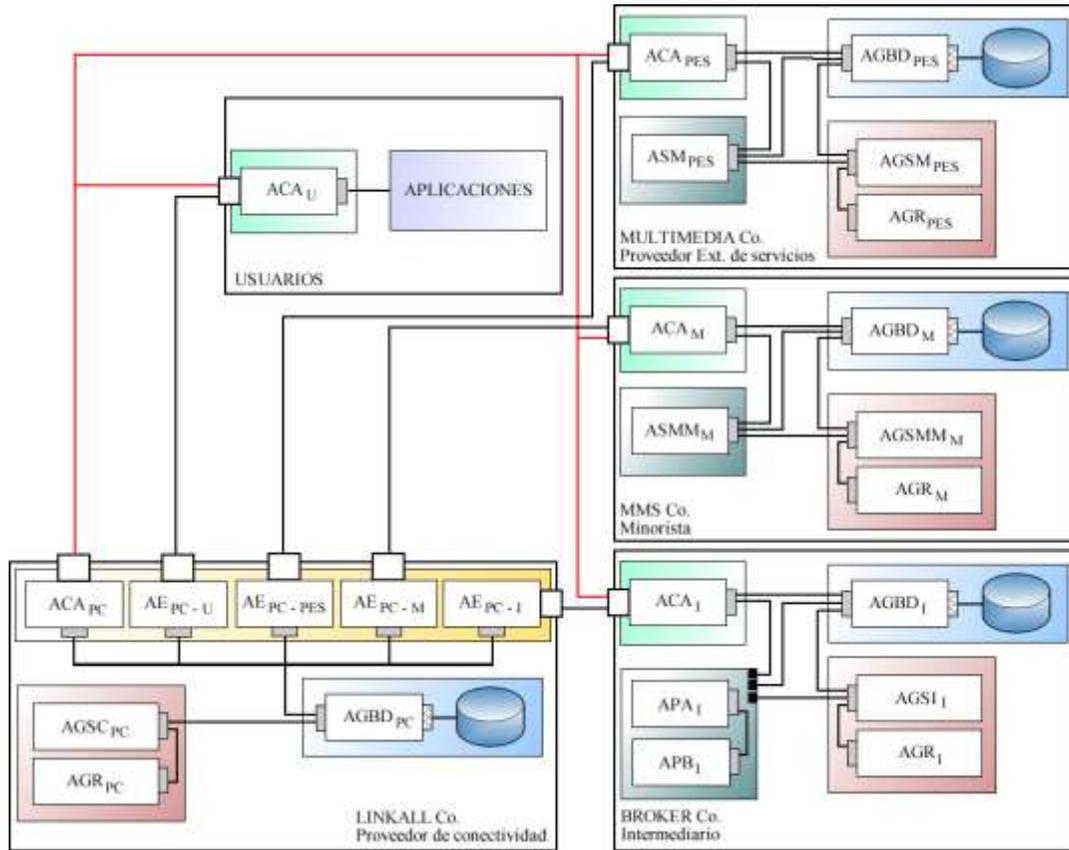


Figura 3.4. Modelo computacional.

Cada uno de los dominios y subdominios de modelo empresariales se convierten en uno o más agentes computacionales que proveen la misma funcionalidad desplazando el diseño de un plano muy conceptual a uno más tangible. En esta fase del diseño un conjunto de requerimientos descritos en lenguaje natural toman forma como agentes computacionales.

Tabla 3.2. Agentes computacionales del proveedor externo de servicios

Abreviación	Nombre
ACA _{PES}	Agente de Control de Acceso
ASM _{PES}	Agente de Servicio Multimedia
AGBD _{PES}	Agente Gestor de Base de Datos
AGSM _{PES}	Agente Gestor del Servicio Multimedia
AGR _{PES}	Agente de Gestión Remota.

Tabla 3.3. Agentes computacionales del minorista

Abreviación	Nombre
ACA _M	Agente de Control de Acceso
ASMM _M	Agente de Servicio de Mensajería Multimedia
AGBD _M	Agente Gestor de Base de Datos
AGSMM _M	Agente Gestor del Servicio de Mensajería Multimedia
AGR _M	Agente de Gestión Remota.

Tabla 3.4. Agentes computacionales del intermediario.

Abreviación	Nombre
ACA _I	Agente de Control de Acceso
APA _I	Agente de Paginas Amarillas
APB _I	Agente Paginas Blancas
AGBD _I	Agente Gestor de Base de Datos
AGSI _I	Agente Gestor del Servicio de Intermediación
AGR _{PES}	Agente de Gestión Remota.

Tabla 3.5. Agentes computacionales del usuario

Abreviación	Nombre
ACA _U	Agente de Control de Acceso
Aplicaciones de usuario	

Tabla 3.6. Agentes computacionales del proveedor de conectividad.

Abreviación	Nombre
ACA _{PC}	Agente de Control de Acceso
AE _{PC-X}	Agente de Enlace
AGBD _{PC}	Agente Gestor de Base de Datos
AGSC _{PC}	Agente Gestor del Servicio de Conectividad
AGR _{PC}	Agente de Gestión Remota.

En el modelo de información se describirá en detalle cada una de las funciones, acciones, roles y tareas de los agentes computacionales relacionados en las anteriores tablas.

3.6 MODELO DE INFORMACION

Como se puede notar en el modelo empresarial y computacional, hay muchas estructuras, que cumplen funciones similares pero en dominios administrativo de negocio distintos, por ejemplo el agente empresarial ACA o de control de acceso cumple la misma función en todos los dominios administrativos, en unos casos permite la conexión de usuarios al minorista, en otros entre el minorista y el proveedor de conectividad y etc. La identificación de estas entidades con funciones similar es lo que se va a tratar a continuación.

3.6.1 Agente de control de acceso (ACA)

El agente de control de acceso es la entidad que separa el interior de un dominio administrativo de negocio con el exterior, toda interacción se debe realizarse a través de este agente, sus funciones son:

- Proveer al dominio administrativo de un punto de referencia.
- Proteger a los agentes internos de accesos no autorizados.
- Filtrar mensajes.
- Manejar sesiones.
- Establecer niveles de confianza.
- Exponer una interfaz publica para que otros agentes puedan contactarse al dominio administrativo de negocio.
- Establece una conexión con el proveedor de conectividad.
- Interactuar con el Agente de Control de Acceso y de Enlace del proveedor de conectividad.

Este agente computacional es de tipo no autónomo ya que siempre espera el arribo de un mensaje para operar, sus funciones corresponden a la de un punto de referencia sección 2.8.3.9 y puede implementar contratos (sección 2.8.3.9) en el sentido de que solo las entidades autorizadas pueden ingresar al dominio.

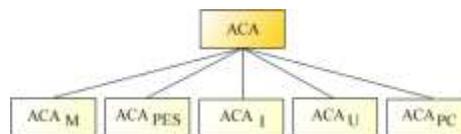


Figura 3.5. Agente de control de acceso.

Dependiendo del tipo de dominio administrativo que se considere el ACA_x puede variar su comportamiento del ACA primario creándose así un subtipo, que especializa o define nuevos comportamientos.

Subtipo ACA_M: Se encuentra en el dominio administrativo de negocio MMS (minorista) e incluye los siguientes comportamientos:

- Valida Usuarios.
- Permite el acceso al servicio de mensajería multimedia
- Rechaza mensajes provenientes de dominios administrativos no autorizados.
- Consulta al agente AGBD_M para corroborar si una dupla (login, password) es correcta.

Subtipo ACA_{PES}: Se encuentra en el dominio administrativo de negocio Multimedia (Proveedor externo de servicios) e incluye los siguientes comportamientos:

- Validar Minorista.
- Permite el acceso a los minoristas autorizados
- Rechaza mensajes provenientes de dominios administrativos no autorizados.
- Consulta al agente AGBD_{PES} para corroborar si una dupla (login, password) es correcta.

Subtipo ACA_I: Se encuentra en el dominio administrativo de negocio Broker (Intermediario) e incluye los siguientes comportamientos:

- Permite exclusivamente el acceso a los servicio de páginas amarillas y blancas.

Subtipo ACA_U: Se encuentra en el dominio administrativo de negocio Usuario e incluye los siguientes comportamientos:

- Sirve de canal para acceder al servicio de conectividad y de mensajería multimedia.

Subtipo ACA_{PC}: Se encuentra en el dominio administrativo de negocio LinkAll (proveedor de conectividad) e incluye los siguientes comportamientos:

- Recibe las solicitudes de conexión proveniente de los restantes dominios administrativos de negocio.
- Provee el servicio de conectividad a los dominios administrativos autorizados.
- Consulta al agente AGBD_{PC} para corroborar si una dupla (login, password) es correcta.

3.6.2 Agente de enlace (AE)

Los agentes computacionales de enlace como el AE_{PC-U} , AE_{PC-PES} , AE_{PC-M} y AE_{PC-I} son instanciados a partir de una plantilla como se menciona en la sección 2.10.4.7, cada uno de estos cumple la misma función, servir de enlace entre el proveedor de conectividad y el dominio administrativo de negocio que desea usar el servicio de conectividad. Las funciones del agente de enlace AE esta ligadas a las del agente de control de acceso ACA_{PC} , porque es este el que controla su ciclo de vida.

El agente de enlace en cualquiera de sus variantes opera como un canal de dos vías, todo mensaje que llega a su interfaz interna se reenviado automáticamente al agente de control de acceso asociado a el y viceversa, es así que se comunican los dominios administrativos de negocio.

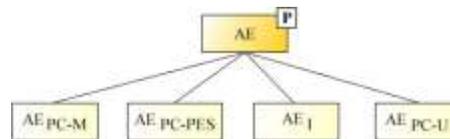


Figura 3.6. Agente de enlace.

3.6.3 Agente gestor de base de datos (AGBD)

Con este agente empresarial ocurre lo mismo que en el caso anterior, esta presente en todos los dominios administrativos y es el encargado de almacenar la información. Posee dos interfaces una computacional con la que se comunica con otros agentes y una de flujo con la que se conecta a la base de datos, sus funciones son:

- Almacenar registros.
- Eliminar registros.
- Actualizar registros.
- Consulta registros.
- Mantener un conexión constante con la base de datos

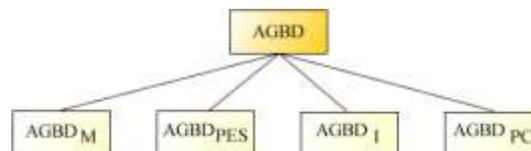


Figura 3.7. Agente gestor de base de datos.

Subtipo AGBD_M: Se encuentra en el dominio administrativo de negocio MMS (minorista) e incluye los siguientes comportamientos:

- Consulta, almacena, actualiza y elimina los registros de usuario.
- Consulta, almacena y elimina mensajes.

Subtipo AGBD_{PES}: Se encuentra en el dominio administrativo de negocio Multimedia (proveedor externo de servicios) e incluye los siguientes comportamientos:

- Consulta, almacena, actualiza y elimina los registros de los minoristas.
- Consulta, almacena y elimina recursos multimedia.
- Elimina o crea categorías.

Subtipo AGBD_I: Se encuentra en el dominio administrativo de negocio Broker (intermediario) e incluye los siguientes comportamientos:

- Consulta, Almacena, actualiza y elimina los registros que contienen los puntos de referencia de los dominios administrativos de negocio inscrito en los directorios.

Subtipo AGBD_{PC}: Se encuentra en el dominio administrativo de negocio LinkAll (proveedor de conectividad) e incluye los siguientes comportamientos:

- Consulta, almacena, actualiza y elimina los registros de los dominios administrativos de negocios autorizados para iniciar sesión en las redes.

3.6.4 Relaciones entre dominios administrativos de negocio

Cada dominio administrativo de negocio que desee establecer una asociación con otro debe hacerlo a través de los puntos de referencia o ACA, ya que hacerlo directamente entre agentes viola las políticas de seguridad, porque significa exponer interfases computacionales. En toda relación de asociación se distinguen tres fases: acceso, sesión y desconexión.

3.6.4.1 Acceso

La parte de acceso cubre las interacciones requeridas para el establecimiento de relaciones entre dominios administrativos de negocio e incluye los siguientes aspectos:

- Inicio de un dialogo entre los dominios administrativos de negocio interesados.
- Establecimiento de asociaciones seguras entre los dominios administrativos de negocio.
- Inicio de la fase de sesión entre los dominios administrativos.

3.6.4.2 Sesión

Una vez ha culminado satisfactoriamente la fase de acceso, los agentes pertenecientes a cada uno de los dominios administrativos de negocio están autorizados para interactuar a través de los ACA, solamente en esta etapa es que se pueden hacer uso de los servicios ofrecidos por un dominio administrativo. Las siguientes operaciones son permitidas:

- Control y manipulación del servicio.
- Interacción con el servicio.

3.6.4.3 Desconexión

Cuando una relación ya no es necesaria se puede dar por terminada, liberándose tanto los recursos hardware como software dispuestos, todo mensaje que llegue después son rechazados.

En la tabla 3.7 se pueden observar las relaciones de negocio existentes en el sistema, estas también aparecen en la figura 3.3 (Modelo empresarial).

Tabla 3.7. Relaciones de negocio.

Dominio administrativo de negocio		Relación
LinkAll (Proveedor de conectividad)	Usuario	C&U [TCon]
MMS (Minorista)	LinkAll (Proveedor de conectividad)	M&C [TCon ConS]
Broker (Intermediario)	LinkAll (Proveedor de conectividad)	B&C [Bkr]
Multimedia (Proveedor ext. de servicios)	LinkAll (Proveedor de conectividad)	Mu&C [TCon ConS]
MMS (Minorista)	Usuario	M&U [Ret]
Multimedia (Proveedor ext. de servicios)	MMS (Minorista)	Mu&M [3pty]
Intermediario	Todos los roles	[Bkr]

Las relaciones de negocio TCon, ConS, Bkr, Ret y 3pty se pueden encontrar especificadas en el capítulo II sección 2.8.3.10.2. Debido a que muchas relaciones de negocio se establecen en forma similar se han agrupado en los siguientes tipos

3.6.5. Acceso genérico al servicio de conectividad

Las relaciones C&U, M&C, B&C y Mu&C se establecen como se muestra en la siguiente figura:

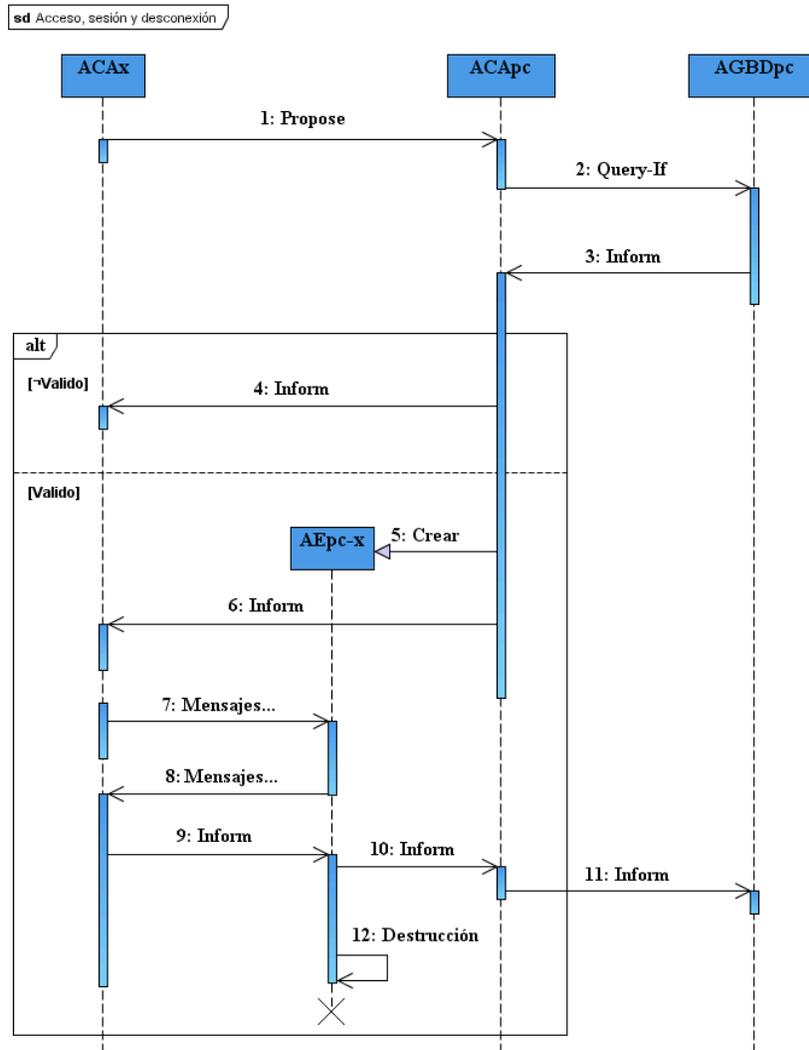


Figura 3.8. Acceso genérico al proveedor de conectividad.

Fase de acceso:

1. El agente de control de acceso ACA_X que puede ser del subtipo ACA_U , ACA_{PES} , ACA_M o ACA_I envía una propuesta de acceso al servicio de conectividad (ACA_{PC}).

2. El ACA_{PC} corrobora si los datos suministrados por el ACA_X son correctos.
3. El AGBD_{PC} busca en sus registros y le informa el resultado obtenido.
4. Si la solicitud no es válida se le informa al ACA_X que los datos suministrados no son válidos por lo tanto no se puede establecer la relación y la fase de acceso culmina.
5. Si por el contrario los datos son correctos el ACA_X crea un agente de enlace (AE_{PC-X}), el cual según sea el caso puede ser del subtipo AE_{PC-U}, AE_{PC-PES}, AE_{PC-M} o AE_{PC-I}.
6. El ACA_{PC} le informa al ACA_X que la solicitud de acceso al servicio de conectividad es válida y que el agente de enlace está activo.

Fase de sesión, solo si la interacción alcanzó el punto 6:

7. Como la fase de acceso culminó satisfactoriamente el dominio administrativo de negocio al que pertenece el agente de control de acceso ACA_X puede comunicarse con otros a través de este y del agente de enlace que el proveedor de conectividad dispuso (AE_{PC-X}).
8. Los mensajes de retorno provenientes de otros dominios administrativos de negocio se hacen a través del AE_{PC-X} y el ACA_X.

Fase de desconexión, solo si se estableció la sesión:

9. El ACA_X le informa a su agente de enlace (AE_{PC-X}) que quiere dar por terminada la sesión.
10. El AE_{PC-X} informa al ACA_X que el dominio administrativo de negocio desea desconectarse.
11. El ACA_{PC} le informa al AGBD_{PC} que el dominio administrativo de negocio cerró la sesión y que debe actualizar los registros.
12. Cierta vez después el AE_{PC-X} se destruye.

3.6.6 Transferencia de mensajes

Las restantes relaciones de negocio solo se pueden dar entre dominios administrativos de negocio que hayan iniciado sesión con el proveedor de conectividad. Ya que este es el encargado de facilitar el intercambio de información al interior del sistema de telecomunicaciones.

Cada vez que el proveedor de conectividad recibe un mensaje proveniente de algún dominio ya sea minorista, intermediario, proveedor externo de servicio o usuario debe procesar el mensaje como se muestra en la siguiente figura:

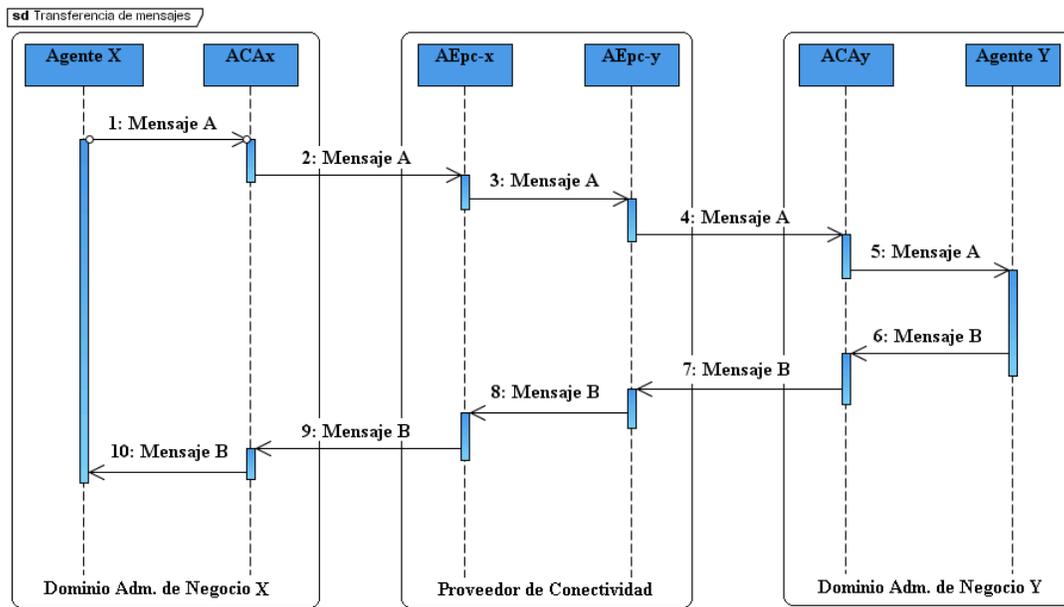


Figura 3.9. Transferencia de mensajes.

En la figura 3.9 se ha supuesto que un Agente X en un dominio administrativo de negocio X desea comunicarse con otro agente ubicado en un dominio administrativo de negocio Y, para lograr que un mensaje A llegue a su destino el proveedor de conectividad debe realizar las siguientes operaciones:

1. El agente X envía el mensaje a su punto de referencia o ACA_X.
2. El ACA_X envía el mensaje a su agente de enlace en el dominio del proveedor de conectividad (AE_{PC-X}).
3. El AE_{PC-X} reenvía el mensaje al agente de enlace Y.
4. El AE_{PC-Y} envía el mensaje al agente de control de acceso en el dominio administrativo de negocio Y (ACA_Y).
5. Por ultimo el ACA_Y envía el mensaje a su destinatario.

Lo mismo ocurre cuando el mensaje viaja en sentido contrario. El contenido y estructura de los mensajes que pasan por los agentes ACA_X, AE_{PC-X}, AE_{PC-Y} y ACA_Y no varía, lo único que sí cambia es la dirección del remitente y destinatario porque cada uno de estos agentes actúa como un proxy. Como el proveedor de conectividad es el encargado de realizar estas operaciones sobre los mensajes en forma transparente para los restantes dominios administrativos de negocio en lo sucesivo no se tendrá en cuenta el paso de los mensajes por este dominio administrativo.

3.6.7 Acceso al servicio de mensajería multimedia

En este caso el usuario del sistema desea acceder al servicio de mensajería multimedia para ello debe realizar las siguientes acciones.

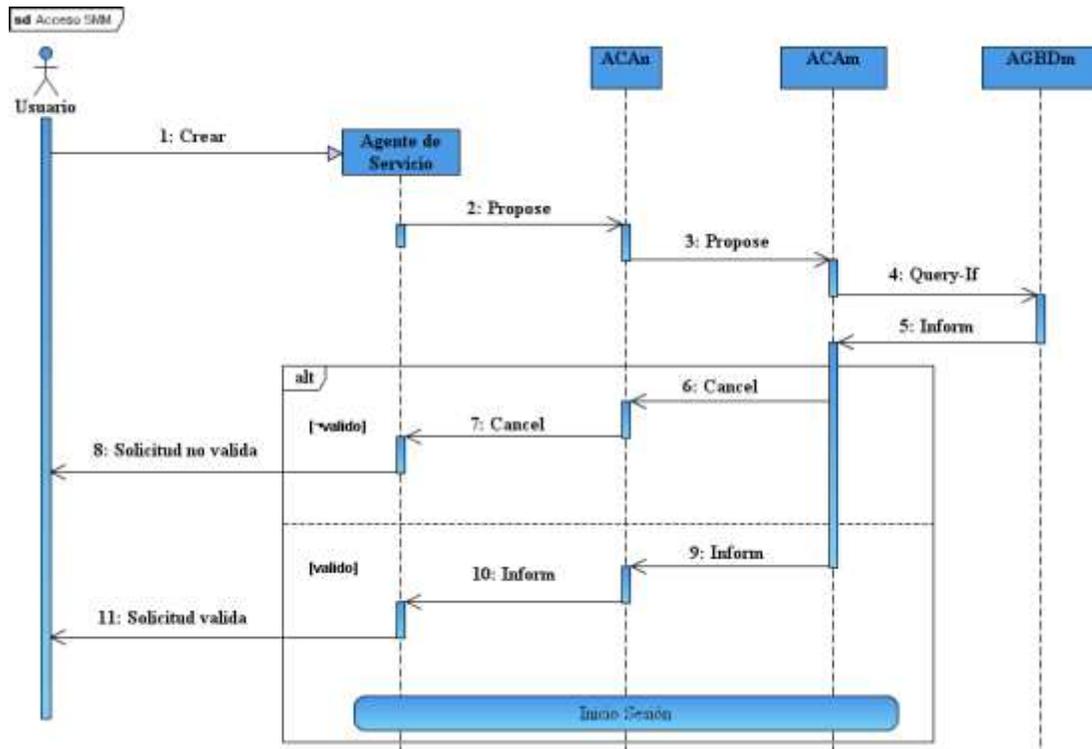


Figura 3.10. Acceso al servicio de mensajería multimedia.

1. El usuario crea el agente de servicio que le asistirá.
2. El agente de servicio intenta establecer conexión con el dominio administrativo de negocio MMS por medio del ACA_U .
3. El ACA_U le informa al ACA_M que un usuario desea acceder al servicio de mensajería multimedia.
4. El agente de control de acceso ACA_M consulta si el usuario es valido.
5. El agente gestor de base de datos $AGBD_M$ verifica si el usuario esta registrado y si esta autorizado para usar el servicio.
6. Si el usuario no esta autorizado se le informa que no puede acceder al servicio.
7. El ACA_U le informa al agente de servicio que su propuesta fue rechazada.
8. El usuario se entera que no puede acceder al servicio.
9. El ACA_M le informa al ACA_U que el usuario esta autorizado para acceder al servicio.

10. El agente de servicio ha iniciado sesión con el proveedor del servicio de mensajería multimedia.

11. El usuario se entera que ha iniciado sesión en el servicio.

3.6.8 Acceso al servicio Multimedia

El minorista se conecta con el proveedor externo de servicios para que los usuarios del servicio de mensajería multimedia puedan enviar imágenes en los mensajes. La siguiente figura muestra el establecimiento de la sesión.

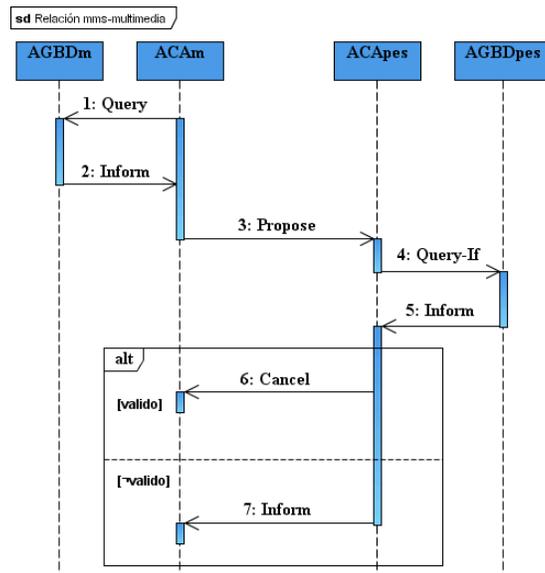


Figura 3.11. Relación MMS-Multimedia

1. El ACA_M consulta al AGBD_M los datos necesarios para el establecimiento de una relación con el proveedor externo de servicios.
2. El AGBD_M consulta los registros y responde a la solicitud anterior.
3. Con la información suministrada por AGBD_M el ACA_M envía una propuesta al ACA_{PES}.
4. EL ACA_{PES} corrobora la veracidad de los datos suministrador por el agente de control de acceso ACA_M enviando una consulta al AGBD_{PES}.
5. El AGBD_{PES} consulta los registros y responde a la anterior solicitud.
6. Si la propuesta es rechazada el ACA_{PES} envía un mensaje al ACA_M indicándole el error.
7. Si la propuesta es aceptada se envía un mensaje indicando que la sesión ha iniciado.

Cuando la sesión se ha establecido el dominio administrativo de negocio MMS puede usar y ofrecer los servicios del proveedor externo de servicios como si fueran suyos.

3.6.9 Agente de gestión remota (AGR)

Este agente computacional ofrece una interfaz grafica a través de la cual los administradores gestionan el comportamiento de todo el dominio administrativo de negocio, este agente siempre se conecta con otro que es el que realmente implementa las funciones de gestión. Su función es:

- Proveer un interfaz grafica de gestión, a través de la cual un administrador de un dominio administrativo de negocio pueda gestionar el servicio.
- Convierte acciones de gestión de alto nivel en secuencias de mensajes de bajo nivel que obligan a otros agentes ejecutar procesos.

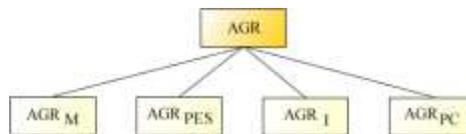


Figura 3.12. Agente de gestión remota.

Subtipo AGR_M: Se encuentra en el dominio administrativo de negocio MMS (minorista) e incluye los siguientes comportamientos:

- Intercambio de mensajes de gestión con el agente computacional AGSMM_M.

Subtipo AGR_{PES}: Se encuentra en el dominio administrativo de negocio Multimedia (proveedor externo de servicios) e incluye los siguientes comportamientos:

- Intercambio de mensajes de gestión con el agente computacional AGSM_{PES}.

Subtipo AGR_I: Se encuentra en el dominio administrativo de negocio Broker (intermediario) e incluye los siguientes comportamientos:

- Intercambio de mensajes de gestión con el agente computacional AGSI_I.

Subtipo AGR_{PC}: Se encuentra en el dominio administrativo de negocio LinkAll (proveedor de conectividad) e incluye los siguientes comportamientos:

- Intercambio de mensajes de gestión con el agente computacional AGSC_{PC}.

3.6.10 Agentes gestor del servicio de conectividad (AGSC_{PC})

El agente gestor del servicio de conectividad ofrece las funciones de gestión del servicio y permite:

- Adicionar usuarios.
- Verifica el estado de los enlaces.

La siguiente figura muestra el intercambio de información en el proceso de gestión del servicio de conectividad.

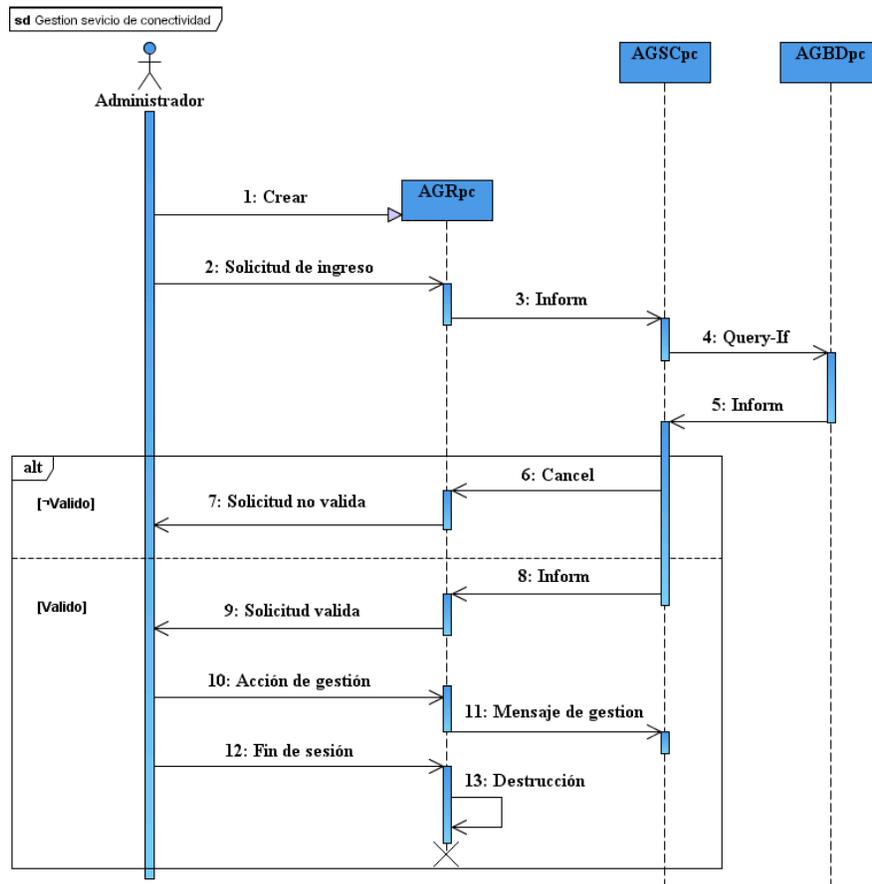


Figura 3.13. Gestión del servicio de conectividad

1. El administrador del sistema crea una instancia del agente de gestión remota AGR_{PC}.
2. El administrador solicita el ingreso al servicio de gestión.
3. El AGR_{PC} intenta suscribirse al agente gestor del servicio de conectividad AGSC_{PC}.
4. El AGSC_{PC} consulta al AGBD_{PC} sobre los datos suministrados por el administrador.
5. El AGBD_{PC} informa el resultado de la consulta.

6. Si el administrador no esta autorizado el AGSC_{PC} informa que la solicitud fue rechazada.
7. El administrador se entera de que su solicitud de ingreso fue rechazada.
8. Si el administrador esta autorizado se le informa al AGR_{PC} que puede enviar mensajes de gestión.
9. El administrador se entera que ha iniciado sesión en el servicio de gestión.
10. El administrador realiza una acción de gestión.
11. La acción de gestión de convierte en uno o varios mensaje de gestión que el AGSC_{PC} recibe y da tramite.
12. Una vez ha concluido la sesión y el administrador desea desconectarse envía un mensaje al AGR_{PC} de fin de sesión.
13. El AGR_{PC} se destruye.

3.6.11 Agente de paginas amarillas (APA_I)

Dado un conjunto de atributos busca la dirección del agente que provee el servicio con las características solicitadas.

3.6.12 Agente de páginas blancas (APB_I)

Dado el nombre de un dominio administrativo de negocio, El agente de páginas blancas retorna el punto de referencia correspondiente al dominio solicitado.

3.6.13 Agente gestor del servicio de intermediación (AGSI_I)

El agente gestor del servicio de intermediación ofrece las funciones de gestión del servicio y permite:

- Adicionar puntos de referencia.

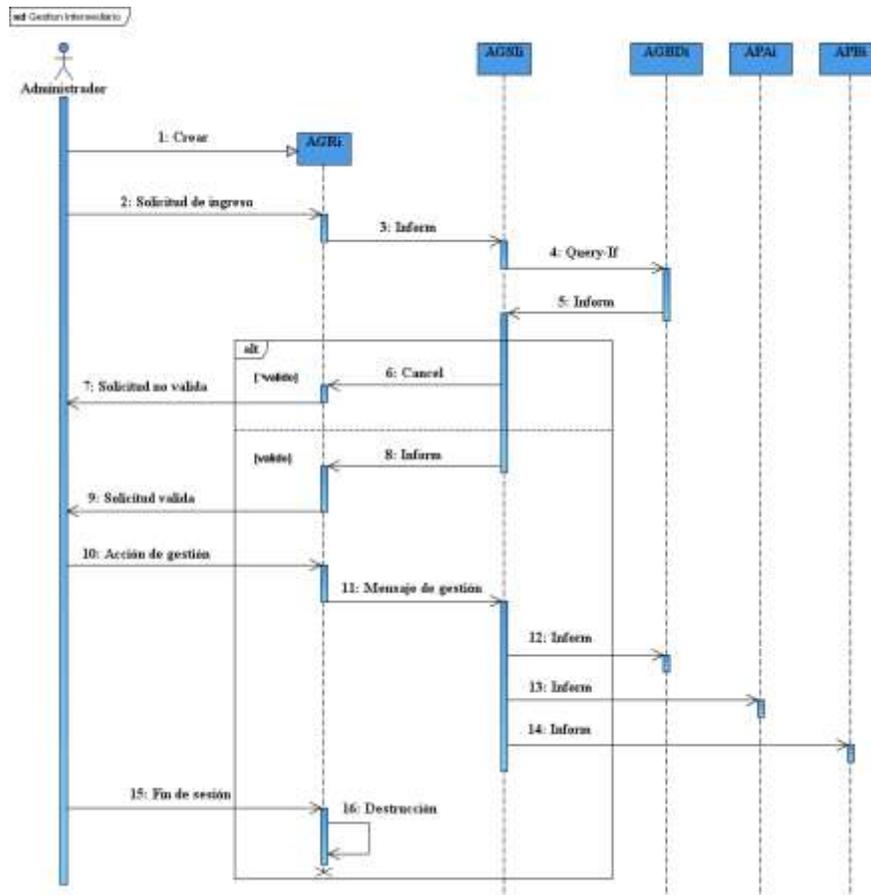


Figura 3.14. Gestión del servicio de intermediación.

1. El administrador del sistema crea una instancia del agente de gestión remota AGR_i.
2. El administrador solicita el ingreso al servicio de gestión.
3. El AGR_i intenta suscribirse al agente gestor del servicio de intermediación AGSI_i.
4. El AGSI_i consulta al AGBD_i sobre los datos suministrados por el administrador.
5. El AGBD_i informa el resultado de la consulta.
6. Si el administrador no esta autorizado el AGSI_i informa que la solicitud fue rechazada.
7. El administrador se entera de que su solicitud de ingreso fue rechazada.
8. Si el administrador esta autorizado se le informa al AGR_i que puede enviar mensajes de gestión.
9. El administrador se entera que ha iniciado sesión en el servicio de gestión.
10. El administrador realiza una acción de gestión.
11. La acción de gestión de convierte en uno o varios mensaje de gestión que el AGSI_i recibe y da tramite.

Según la naturaleza del mensaje alguno de los siguientes los mensajes: 12, 13 y 14 pueden ser enviados al $AGBD_i$, APA_i y APB_i respectivamente.

15. Una vez ha concluido la sesión y el administrador desea desconectarse envía un mensaje de fin de sesión al AGR_{PC} .

16. El AGR_{PC} se destruye.

3.6.14 Consulta al servicio de páginas amarillas o blancas

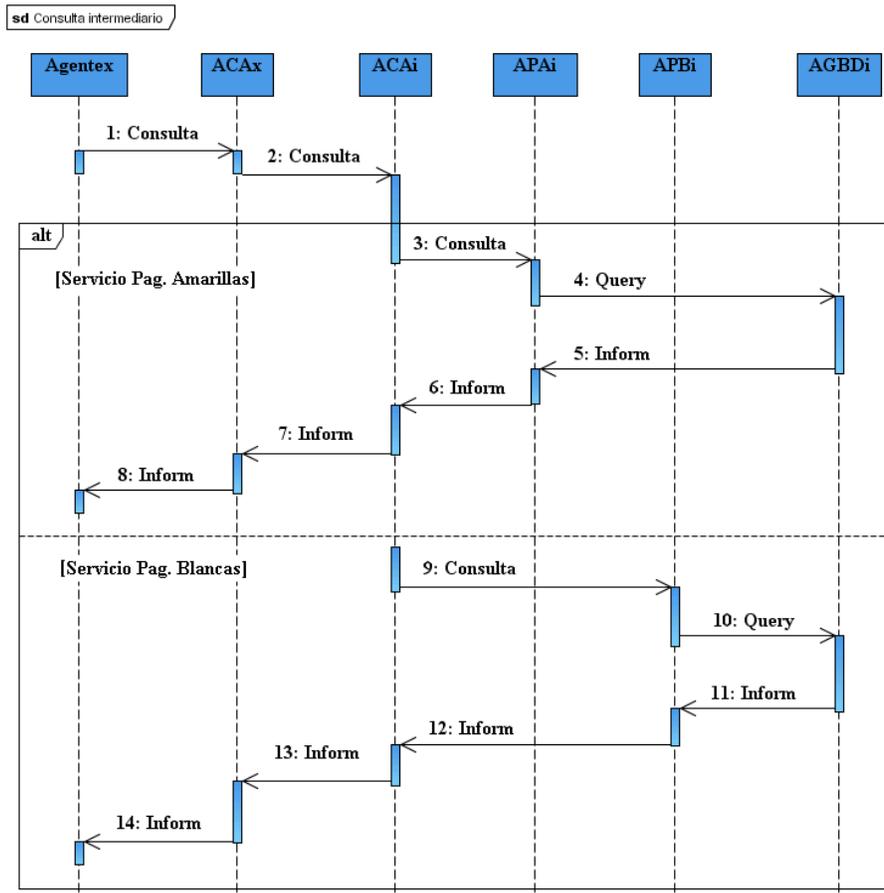


Figura 3.15. Servicio de intermediación.

3.6.15 Agente del servicio de mensajería multimedia (ASMM_M)

Este agente es el encargado de proveer la lógica del servicio de mensajería multimedia, coordina la interacción de los distintos agentes durante la prestación del servicio. Además se combina con ASM_{PES} para extender su funcionalidad. El $ASMM_M$ implementa la mayor parte de la funcionalidad que percibe el usuario (figura 3.1) y porque es muy importante para el sistema su funcionalidad se modelara con Ingenias.

3.6.16 Agente gestor de servicio de mensajería multimedia (AGSMM_M)

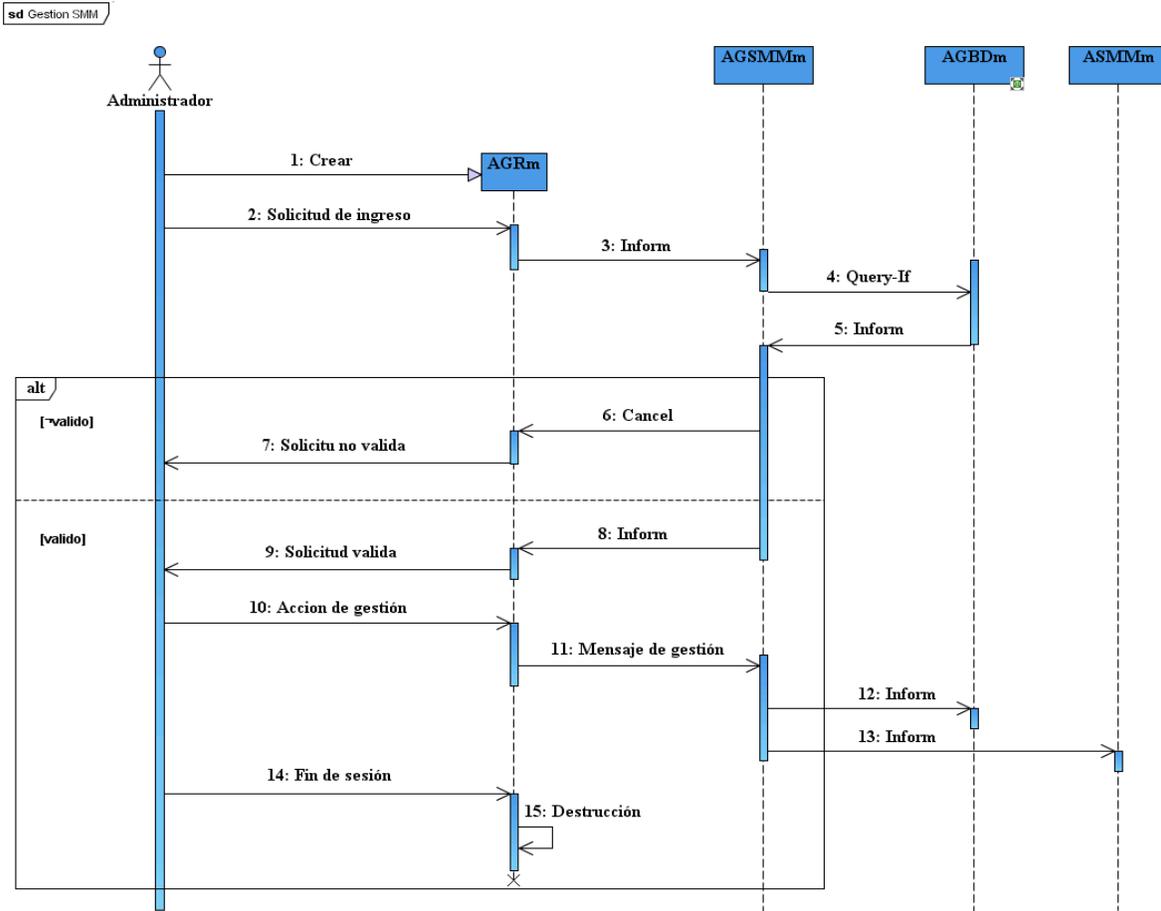


Figura 3.16. Gestión del servicio de mensajería multimedia.

1. El administrador del sistema crea una instancia del agente de gestión remota AGR_M.
2. El administrador solicita el ingreso al servicio de gestión.
3. El AGR_M intenta suscribirse al agente gestor del servicio de mensajería multimedia AGSMM_M.
4. El AGSMM_M consulta al AGBD_M sobre los datos suministrados por el administrador.
5. El AGBD_M informa el resultado de la consulta.
6. Si el administrador no esta autorizado el AGSMM_M informa que la solicitud fue rechazada.
7. El administrador se entera de que su solicitud de ingreso fue rechazada.
8. Si el administrador esta autorizado se le informa al AGR_M que puede enviar mensajes de gestión.
9. El administrador se entera que ha iniciado sesión en el servicio de gestión.
10. El administrador realiza una acción de gestión.

11. La acción de gestión de convierte en uno o varios mensaje de gestión que el AGSMM_M recibe y dará tramite.

Según la naturaleza de la acción de gestión los mensajes 12 y 13 pueden ser enviados al AGBD_M o ASMM_M respectivamente.

14. Una vez ha concluido la sesión y el administrador desea desconectarse envía un mensaje al AGR_M de fin de sesión.

15. El AGR_M se destruye.

3.6.17 Agente de servicio multimedia (ASM_{PES})

El ASM_{PES} es el encargado de ofrecerle al minorista el servicio multimedia con el cual sus usuarios pueden enviar imágenes, al igual que el ASMM_M este agente es muy importante porque aporta sustancialmente a la funcionalidad del sistema por lo tanto su comportamiento será modela con mayor detalle en la sección 3.9 (aplicación de la metodología de agentes).

3.6.18 Agente gestor del servicio multimedia (AGSM_{PES})

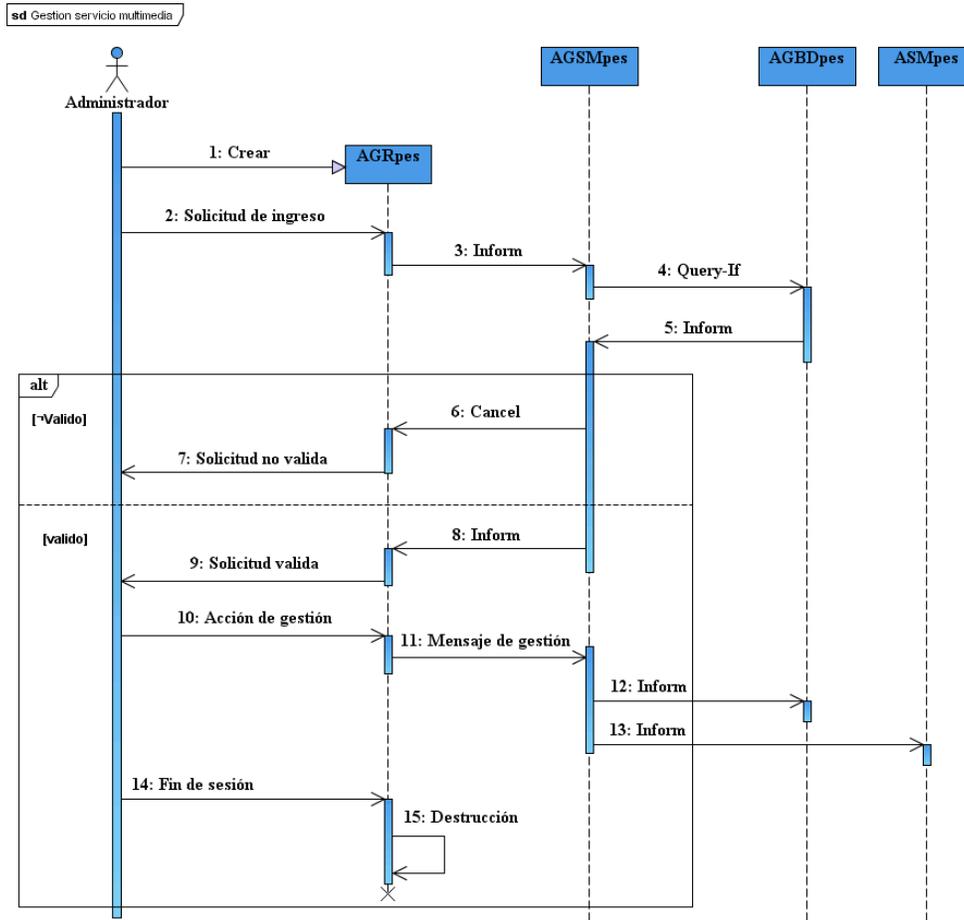


Figura 3.17. Gestión del servicio multimedia.

1. El administrador del sistema crea una instancia del agente de gestión remota AGR_{PES}.
2. El administrador solicita el ingreso al servicio de gestión.
3. El AGR_{PES} intenta suscribirse al agente gestor del servicio multimedia AGSM_{PES}.
4. El AGSM_{PES} consulta al AGBD_{PES} sobre los datos suministrados por el administrador.
5. El AGBD_{PES} informa el resultado de la consulta.
6. Si el administrador no esta autorizado el AGSM_{PES} informa que la solicitud fue rechazada.
7. El administrador se entera de que su solicitud de ingreso fue rechazada.
8. Si el administrador esta autorizado se le informa al AGR_{PES} que puede enviar mensajes de gestión.
9. El administrador se entera de que ha iniciado sesión en el servicio de gestión.
10. El administrador realiza una acción de gestión.

11. La acción de gestión se convierte en uno o varios mensajes de gestión que el $AGSM_{PES}$ recibe y da trámite.

Según la naturaleza de la acción de gestión los mensajes 12 y 13 pueden ser enviados al $AGBD_{PES}$ o al ASM_{PES} respectivamente.

14. Una vez ha concluido la sesión y el administrador desea desconectarse envía un mensaje de fin de sesión al AGR_{PES} .

16. El AGR_{PES} se destruye.

3.6.19 Uso del servicio de mensajería multimedia

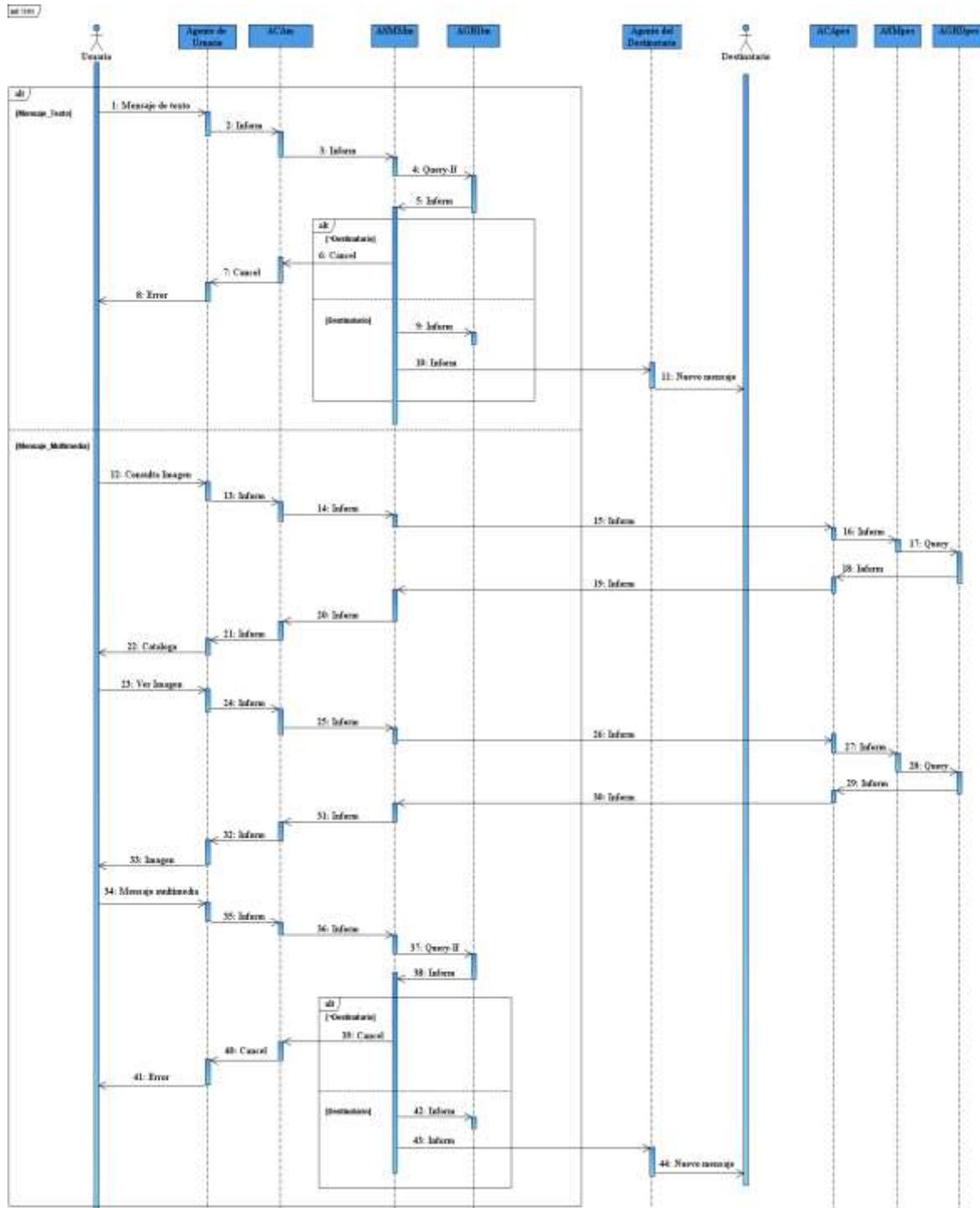


Figura 3.18. Envió de un mensaje.

3.7 MODELO DE INGENIERÍA

En este modelo se identifican a los agentes como entidades que componen los módulos Hardware de cada dominio administrativo. Para ello, se hace corresponder a cada dominio con un nodo, dentro del cual existen cápsulas que contienen agentes relacionados entre sí.

3.7.1 Nodo de Servicio de Mensajería (Minorista)



Figura 3.19. Nodo de servicio de mensajería.

Este nodo representa un equipo Terminal (PC) en el cual residen los agentes que se encargan de manejar el servicio MMS, los cuales están agrupados en las siguientes cápsulas:

Acceso: En esta se gestiona el control de acceso de usuarios al servicio y del administrador de este a las acciones de gestión. Esta cápsula está compuesta por el agente ACA.

Servicio: Esta cápsula está relacionada con la prestación del servicio (lógica del servicio).

Autoridad: Se relaciona con la gestión de las conexiones de este nodo con otros. Compuesta por el agente AGSMM que implementa la lógica de la gestión y el AGR quien provee las interfaces de usuario (administrador).

Almacenamiento: Esta cápsula contiene al agente encargado de proporcionar la funcionalidad de almacenamiento de datos de usuario, mensajes y etc, en la base de datos.

3.7.2 Nodo Usuario



Figura 3.20. Nodo de usuario.

Representa al equipo Terminal del usuario (dispositivo móvil) en el cual se encuentra el agente encargado de proporcionar las funciones necesarias para que el usuario acceda al servicio (Interfaz gráfica, funciones de envío y recepción de mensajes).

3.7.3 Nodo Proveedor de Conectividad



Figura 3.21. Nodo de proveedor de conectividad.

Este nodo representa el Hardware en el que se implementa la lógica que permite a los usuarios conectarse a la red inalámbrica. Está compuesto por las siguientes cápsulas:

Acceso: En esta se gestiona el control de acceso de los usuarios a la red y del administrador a las funciones de gestión. Esta cápsula está compuesta por el agente ACA.

Almacenamiento: Esta cápsula se encarga del almacenamiento de la información requerida por los agentes para desempeñar sus tareas y esta compuesta por el agente AGBD.

Autoridad: Se relaciona con la gestión de las conexiones de este nodo con otros. Compuesta por el agente AGSC que implementa la lógica de la gestión y el AGR quien provee las interfaces de usuario (administrador).

3.7.4 Nodo Proveedor Externo de Servicios



Figura 3.22. Nodo proveedor externo de servicios.

Este nodo representa el Hardware (PC) que funciona como un servidor de contenidos multimedia. En este se identifican las siguientes cápsulas:

Acceso: En esta se gestiona el control de acceso de usuarios al servicio y del administrador a las funciones de gestión. Esta cápsula está compuesta por el agente ACA.

Servicio: Los agentes que componen esta cápsula acceden a la información multimedia almacenada en la base de datos.

Autoridad: Se relaciona con la gestión de las conexiones de este nodo. Compuesta por el agente AGSM que implementa la lógica de la gestión y el AGR quien provee las interfaces de usuario (administrador).

Almacenamiento: Esta cápsula contiene al agente encargado de proporcionar la funcionalidad de almacenamiento de recursos multimedia, mensajes e imágenes, en la base de datos.

3.7.5 Nodo Intermediario

Es el nodo que contiene la lógica del sistema, se encargada del registro de los servicios prestados por los agentes y de los agentes en sí. Dentro de la Plataforma Conceptual de Agentes, estas labores son desempeñadas por el DF (APA) y AMS (APB) respectivamente. Contiene las siguientes cápsulas:

Registro: Esta cápsula aloja a los agentes de registro de servicios y de agentes del sistema, correspondientes a los servicios de páginas blancas desempeñado por el APB y amarillas por el APA.

Acceso: En esta se gestiona el control de acceso de usuarios al nodo. Esta cápsula está compuesta por el agente ACA.

Autoridad: Se relaciona con la gestión de las conexiones de este nodo. Compuesta por el agente AGS que implementa la lógica de la gestión y el AGR quien provee las interfaces de usuario (administrador).

Almacenamiento: Esta cápsula contiene al agente encargado de proporcionar la funcionalidad de almacenamiento de la información obtenida por los agentes APB y APA.



Figura 3.23. Nodo de intermediación.

Por ultimo identificamos los posibles interceptores presentes en los nodos. En este caso en el dominio del proveedor de conectividad existe diversidad en cuanto a las tecnologías utilizables para dar acceso inalámbrico a los usuarios. Estas tecnologías (WLAN, BLUETOOTH y GPRS) pueden interactuar entre sí y también con una red LAN (ver figura 3.3). Tales interacciones involucran interceptores de línea, quienes representan a la gateway entre los dominios tecnológicos. La siguiente figura muestra como se realiza tal interacción.

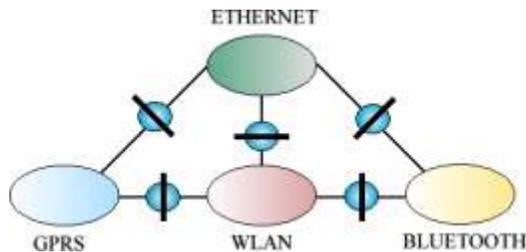


Figura 3.24. Interceptores de línea en el dominio del proveedor de conectividad.

3.8 MODELO DE TECNOLOGIA

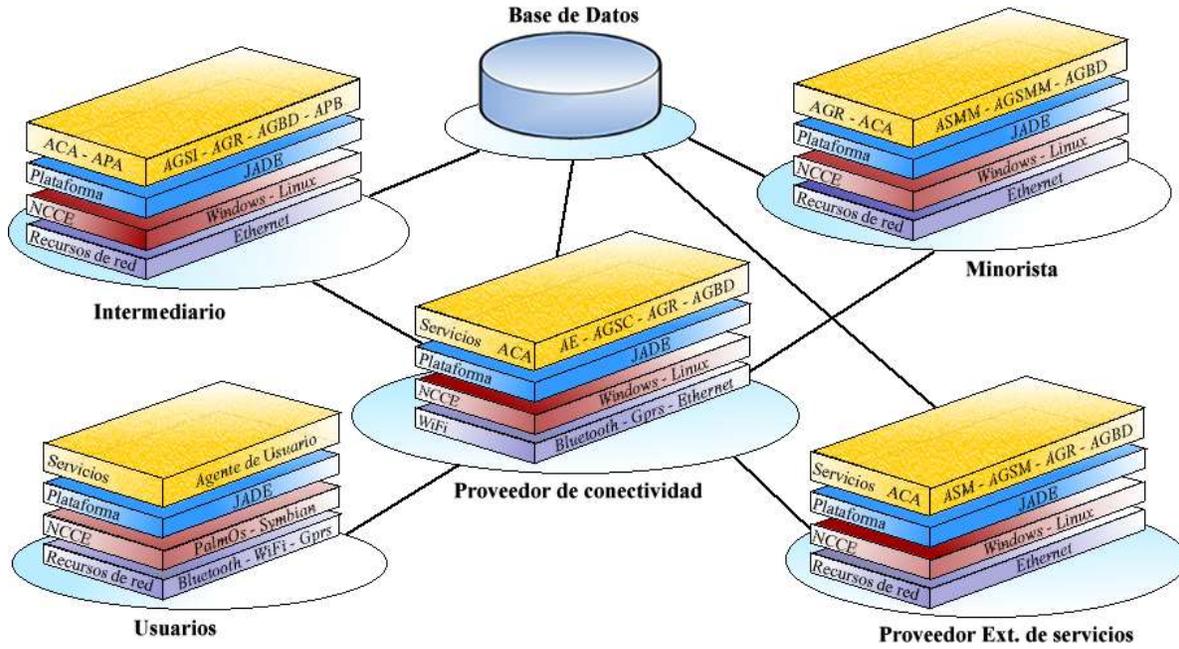


Figura 3.25. Modelo de tecnología.

Tabla 3.8. Especificación de tecnología.

Capa	Usuarios	Proveedor de conectividad	Proveedor externo de servicios	Minorista	Intermediario
Servicios	Agentes de servicio	ACA AE AGSC AGR AGBD	ACA AGSM AGR SGBD	AGR ACA ASMM AGSMM AGBD	ACA APA AGSI AGR AGBD APB
Plataforma	Jade				
NCCE	PalmOs Symbian	Windows Linux			
Recursos de red	Bluetooth Wlan Gprs	Bluetooth Wlan Gprs Ethernet		Ethernet	

Base de datos: MySql versión 5.0

3.9 DISEÑO DE LA APLICACIÓN

Para realizar el diseño de la aplicación, hacemos una integración de la metodología INGENIAS con algunas de las etapas del ciclo de vida del RUP. En primera instancia tenemos el Análisis-Inicio, en la cual se identifican los casos de uso principales que reflejen el comportamiento del sistema y otros componentes que van a tener participación en su funcionamiento. De acuerdo con esto se especifica inicialmente el siguiente diagrama de casos de uso.

3.9.1. Análisis inicio

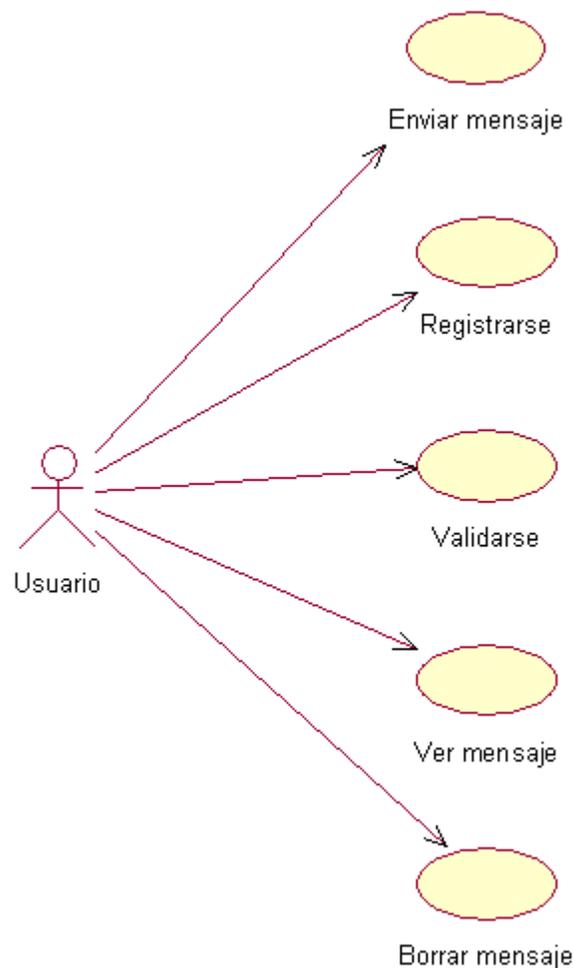


Figura 3.26. Diagrama de casos de uso.

3.9.1.1 Descripción de casos de uso

- **Enviar mensaje:** Representa la actividad de envío de un mensaje por parte de un usuario a otro que esté registrado en el servicio.
- **Ver mensaje:** Representa la actividad de desplegar un mensaje recibido, éste puede ser de error, cuando ha sido enviado por la administración del servicio o de comunicación cuando lo ha enviado otro usuario.
- **Registrarse:** Cuando un usuario no registrado en el servicio quiere acceder a éste, necesita hacer un almacenamiento de sus datos ante el sistema antes de poder utilizar sus funcionalidades.
- **Validarse:** Un usuario previamente registrado debe validarse antes de acceder al servicio, enviando su login y password para la debida autenticación en el sistema.
- **Borrar mensaje:** Una vez visto un mensaje el usuario tiene la posibilidad de borrarlo con el fin de liberar memoria.

3.9.1.2 Modelo Organizacional

Para continuar con la especificación de la aplicación, en esta etapa se identifican las entidades participantes a nivel organizacional (meta-modelo de organización) y de entorno (meta-modelo de entorno). Esto es, se ve al servicio como una organización que persigue el objetivo general “Intercambiar mensajes entre usuarios”. Para lograr tal objetivo, la organización se compone por dos grupos, Usuarios y Administración. Además, se identifican dos flujos de trabajo (Gestionar servicio y Dar acceso a servicio), los cuales se asignan a cada grupo, Administración y Usuarios respectivamente. En etapas posteriores se hace un refinamiento de la descripción de los flujos de trabajo identificando las tareas que se derivan de cada uno, los roles en las cuales están incluidas y los agentes que los desempeñan independientemente o por medio de interacciones con sus semejantes.

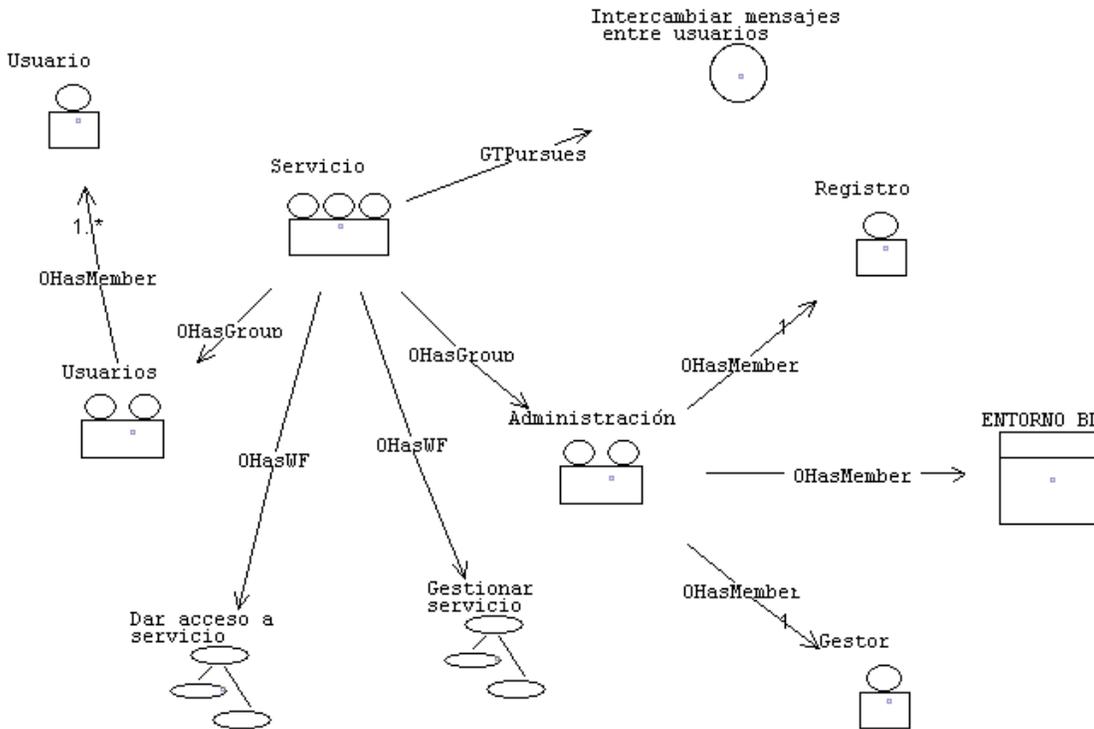


Figura 3.27. Modelo organizacional de la aplicación (Arquitectura inicial).

La figura anterior propone un esquema inicial de la arquitectura de la aplicación, en la cual se identifica el entorno con el que se va a interactuar, en este caso se trata de una Base de datos donde se almacena la información de los usuarios registrados. Obsérvese que el grupo Usuarios puede estar compuesto por n agentes *Usuario*, mientras Administración cuenta únicamente con un agente *Registro* y otro *Gestor*.

3.9.3. Análisis elaboración

En esta etapa se hace un análisis más detallado de los casos de uso identificados a través de una descripción de cada agente que entra a participar en el sistema (meta-modelo de agente), los roles que le han sido asignados y las tareas correspondientes a cada uno de éstos. Además se incluyen los objetivos que persigue cada uno, los cuales no son más que una descomposición del objetivo general perseguido por la organización (Servicio). Vamos a ver a continuación cada uno de estos diagramas de agente y su respectiva descripción.

3.9.3.1 Agente Usuario

La siguiente figura muestra los roles, tareas y objetivos relacionados con el agente Usuario, perteneciente al grupo Usuarios. En primer lugar se identifican los roles *Solicitante validación* y *Solicitante registro*, asociados a las tareas *Enviar login y password* y *Enviar datos usuario* respectivamente. Cuando una de ellas se ha ejecutado el usuario logra el objetivo “Acceder a servicio”, con lo cual se genera un nuevo objetivo para

el agente “Envío recepción de mensajes” y un nuevo conjunto de roles (*Receptor mensaje, Emisor mensaje*) asociados a tareas (*Ver, Borrar y Enviar mensaje*) que realizan el intercambio, despliegue y borrado de mensajes. Esta relación se detalla más profundamente en el diagrama de objetivos y tareas de éste agente, el cual será mostrado posteriormente.

Vamos a describir a continuación cada rol con sus tareas asociadas.

3.9.3.1.1 Rol Solicitante registro

Este rol se relaciona con la acción de registro que ejecuta un usuario nuevo que quiere acceder al sistema. Sus tareas asociadas se describen a continuación.

- Tareas
 - Enviar datos de usuario: Se encarga de enviar los datos del usuario que se va a registrar en el sistema.

3.9.3.1.2 Rol Solicitante validación

Este rol maneja el ingreso al servicio de un usuario previamente registrado.

- Tareas
 - Enviar login y password: Mediante esta tarea el usuario envía sus datos de validación para que se gestione su ingreso en la Administración.

3.9.3.1.3 Rol Emisor mensaje

Encargado de enviar los mensajes.

- Tareas
 - Enviar mensajes: Tarea mediante la cual un usuario puede enviar un mensaje a un destinatario.

3.9.3.1.4 Rol Receptor mensaje

Se encarga del manejo de los mensajes recibidos por un usuario.

- Tareas
 - Ver mensaje: Posibilita el despliegue de un mensaje enviado por otro usuario.

- Ver mensaje error: Tarea por medio de la cual es posible desplegar un mensaje de error enviado por el grupo Administración en el equipo Terminal del usuario.
- Borrar mensaje: Permite la eliminación de un mensaje que ya ha sido visualizado por el usuario.

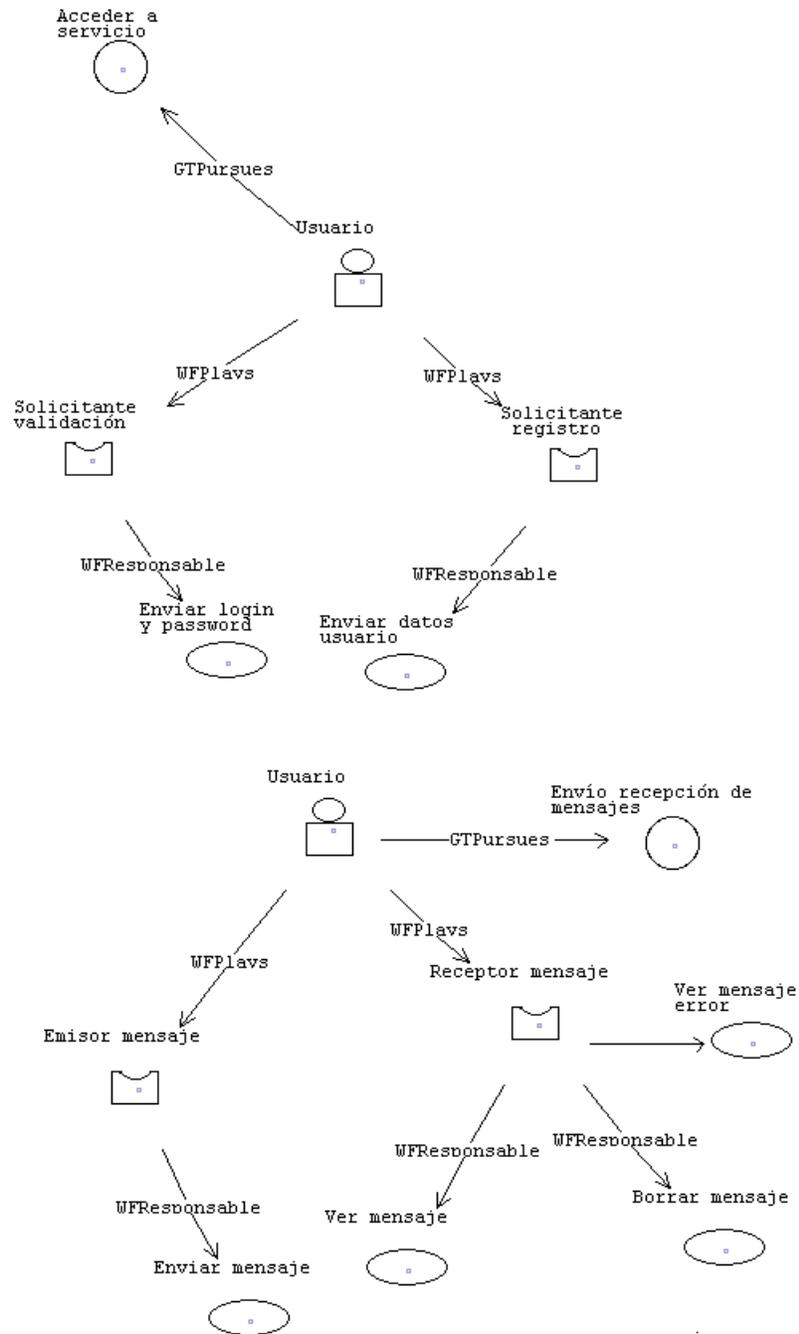


Figura 3.28. Diagrama de agente Usuario.

3.9.3.2 Agente Registro

Este agente tiene como función la comunicación con la Base de datos en la cual se almacenan los datos de los usuarios y los mensajes que no se han podido entregar debido a que el destinatario no está conectado al servicio.

El siguiente diagrama de agente corresponde a este agente, el cual pertenece al grupo Administración. En este se identifican los siguientes roles:

3.9.3.2.1 Rol Gestor de mensajes

Este rol tiene que ver con todo lo relacionado al manejo de los mensajes de usuarios a los cuales alguien les ha enviado un mensaje, pero no han sido entregados debido a que estos no están conectados al servicio. En este caso se realiza un almacenamiento en la Base de datos de los mensajes y la información de sus destinatarios.

- Tareas
 - Almacenar mensajes en BD: Se almacena en la Base de datos la información del destinatario (Usuario no conectado) y el mensaje enviado.
 - Extraer mensajes de BD: Se extrae el mensaje y se envía cuando el usuario (destinatario) se ha vuelto a conectar.
 - Enviar mensaje OK: Cuando un mensaje o un registro de usuario nuevo se han almacenado correctamente en la Base de datos, se notifica al usuario emisor sobre este hecho.
 - Enviar mensaje de error: Esta tarea maneja cualquier evento inesperado en el registro de mensajes o usuarios en la base de datos con el fin de indicarle al emisor que no se ha realizado la operación solicitada.

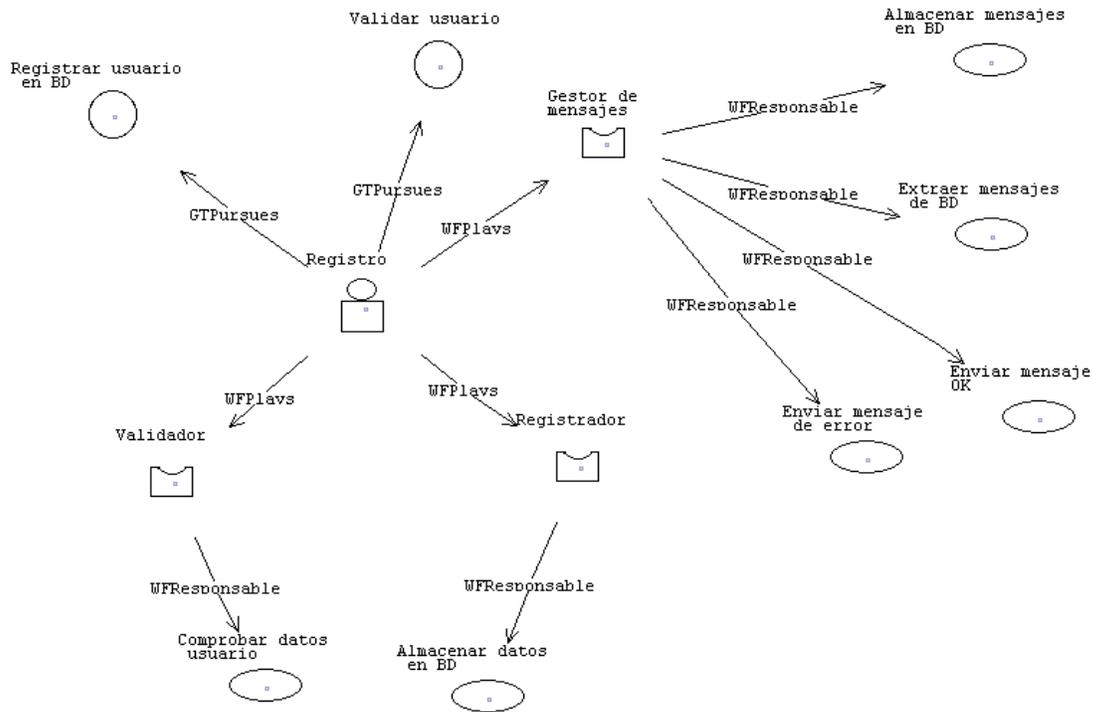


Figura 3.29. Modelo de agente Registro.

3.9.3.2.2 Rol Validador

Este rol se encarga de comprobar la existencia en el sistema de un usuario que pretende acceder a él.

- Tareas
 - Comprobar datos usuario: Se verifican el login y password enviados por el usuario, a través de una comprobación en la Base de datos.

3.9.3.2.3 Rol Registrador

- Tiene como función registrar a un nuevo usuario que quiere acceder al servicio.
- Tareas
 - Almacenar datos en BD: Toma los datos de registro enviados por el usuario nuevo y los registra en la Base de datos.

Los objetivos perseguidos por este agente son “Validar y Registrar usuarios”. El objetivo “Registrar usuarios”, no está ligado únicamente con el registro de usuarios nuevos, sino de la manipulación de los datos de estos, incluyendo el almacenamiento de mensajes de destinatario no conectado.

3.9.3.3 Agente Gestor

Este agente se encarga de registrar a los agentes que representan a los usuarios, para ello se utiliza un protocolo de interacción que permite que este trabaje como un supervisor, gestionando la creación y eliminación de agentes *Usuario*. El protocolo de interacción permite que el *Gestor* posea una lista con los identificadores de los agentes *Usuario* conectados actualmente al servicio, permitiendo de esta manera saber si es posible enviar un mensaje en determinado momento o si este debe ser almacenado en la Base de datos para su posterior envío.

La siguiente figura muestra a este agente con sus roles, objetivos y tareas.

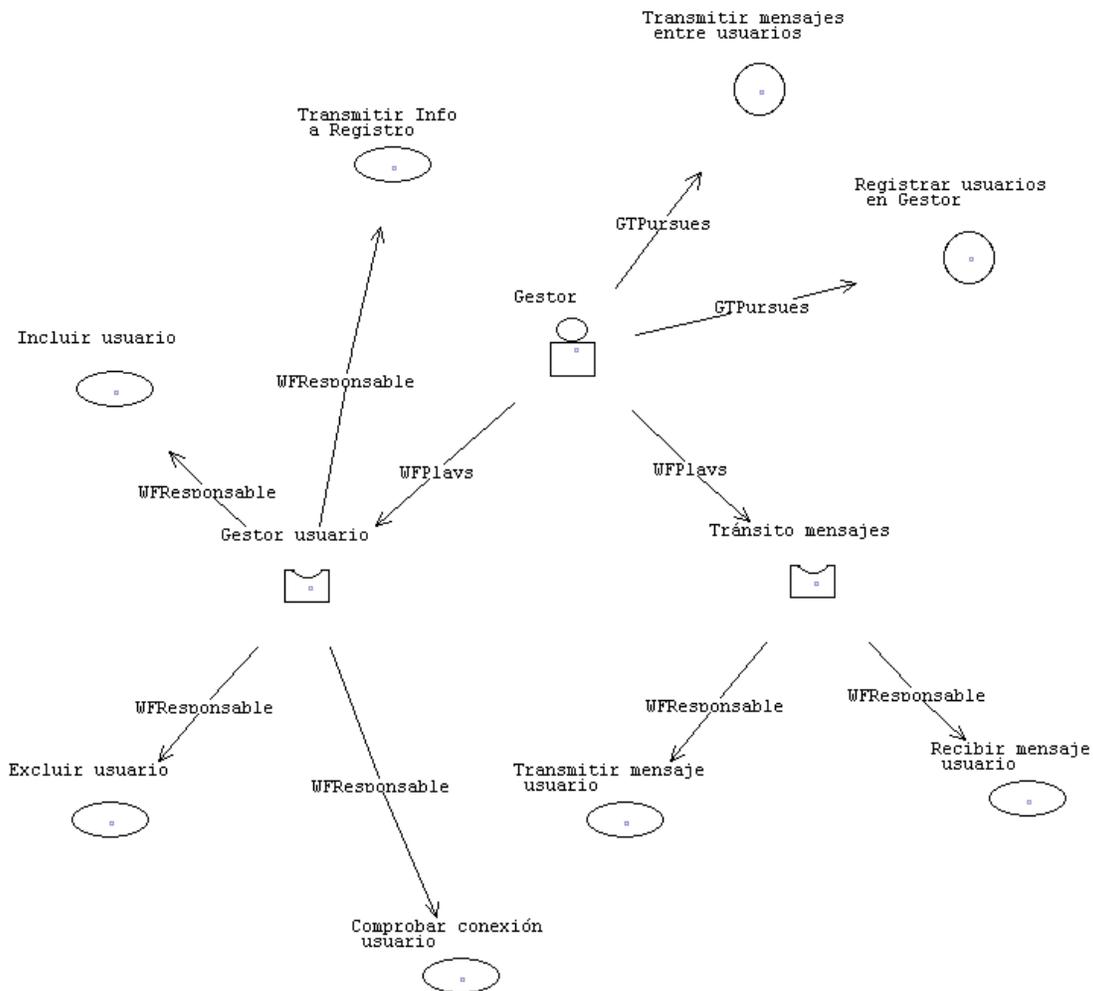


Figura 3.30. Modelo de agente Gestor

Para este agente se identifican los siguientes roles:

3.9.3.3.1 Rol Gestor usuario

Este rol tiene que ver con el registro ante el gestor de un agente que se conecta al sistema y su eliminación cuando se desconecta.

- Tareas
 - Excluir usuario: Tiene que ver con la eliminación del registro del gestor de aquellos agentes Usuario que se han desconectado del sistema.
 - Incluir usuario: Un usuario que ingresa al sistema debe registrarse ante el Gestor por medio de esta tarea.
 - Comprobar conexión usuario: Por medio de esta tarea el gestor puede saber si un usuario determinado está actualmente conectado o no.
 - Transmitir Info a Registro: Cuando un destinatario no está conectado al servicio, el mensaje debe ser almacenado en la Base de datos para su posterior entrega a través de esta tarea.

3.9.3.3.2 Rol Tránsito mensajes

Este rol está relacionado con el intercambio de mensajes entre los agentes Usuario.

- Tareas
 - Transmitir mensaje usuario: Esta tarea tiene que ver con los mensajes que un usuario desea mandar a otro. Dicho mensaje pasa a través del gestor, quien define si el destinatario está o no conectado.
 - Recibir mensaje usuario: Un agente Usuario, quien es el destinatario de un mensaje lo recibe del Gestor a través de esta tarea.

Obsérvese como los objetivos se relacionan con las tareas involucradas.

3.9.3.4 Modelo de objetivos y tareas

Cuando el logro de un objetivo se produce, en algunos casos se generan nuevas tareas y objetivos. En esta etapa se muestran los diagramas de objetivos y tareas de los agentes Usuario y Gestor, ya que sus objetivos cumplen la característica mencionada.

3.9.3.4.1 Agente Usuario

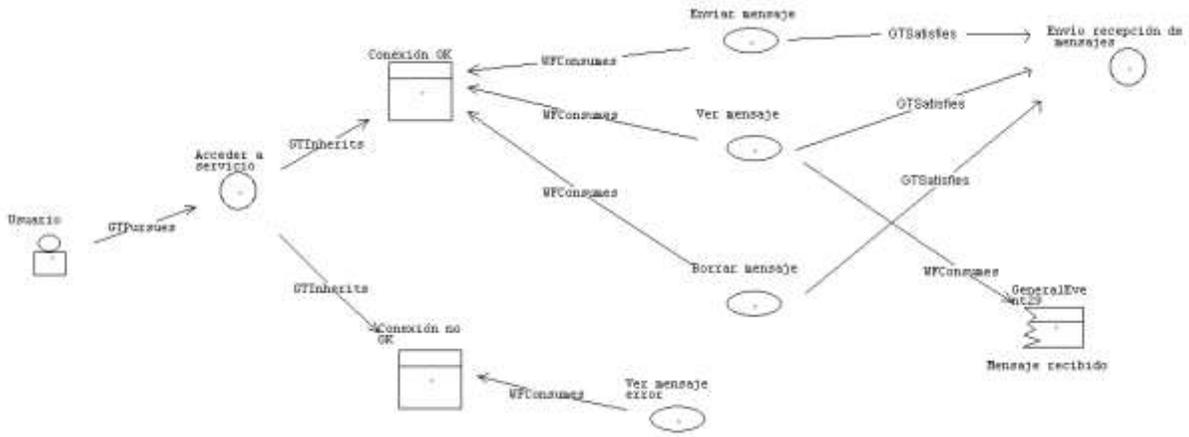


Figura 3.31. Modelo de objetivos y tareas agente Usuario

En la figura anterior es posible identificar que el cumplimiento del objetivo “Acceder a servicio” genera dos hechos posibles: *Conexión OK* y *Conexión no OK*.

En el primer caso se producen tres nuevas tareas relacionadas con el envío, despliegue y eliminación de mensajes, las cuales pueden ser ejecutadas una vez el usuario accede al servicio (validación o registro) para cumplir con un nuevo objetivo “Envío recepción de mensajes”.

Por otra parte el hecho *Conexión no OK* da paso a la tarea *Ver mensaje error*.

Obsérvese también el evento *Mensaje recibido*, el cual activa la tarea *Ver mensaje*; esto significa que cuando un mensaje arriba, se produce su despliegue en el equipo Terminal de usuario.

3.9.3.4.2 Agente Gestor

En este caso el hecho *Usuario no conectado* se produce después de la ejecución de la tarea *Comprobar conexión usuario*, después del cual se ejecutará una nueva tarea, *Transmitir info a Registro*, encargada de enviar al agente Registro el mensaje que se pretendía transmitir a un usuario que no está conectado actualmente, para ser almacenado en la Base de datos y ser retransmitido cuando éste se reconecte (Ver siguiente figura).

También se muestra en la figura el evento *Usuario se conectó*, el cual activa la tarea *Transmitir mensaje*.

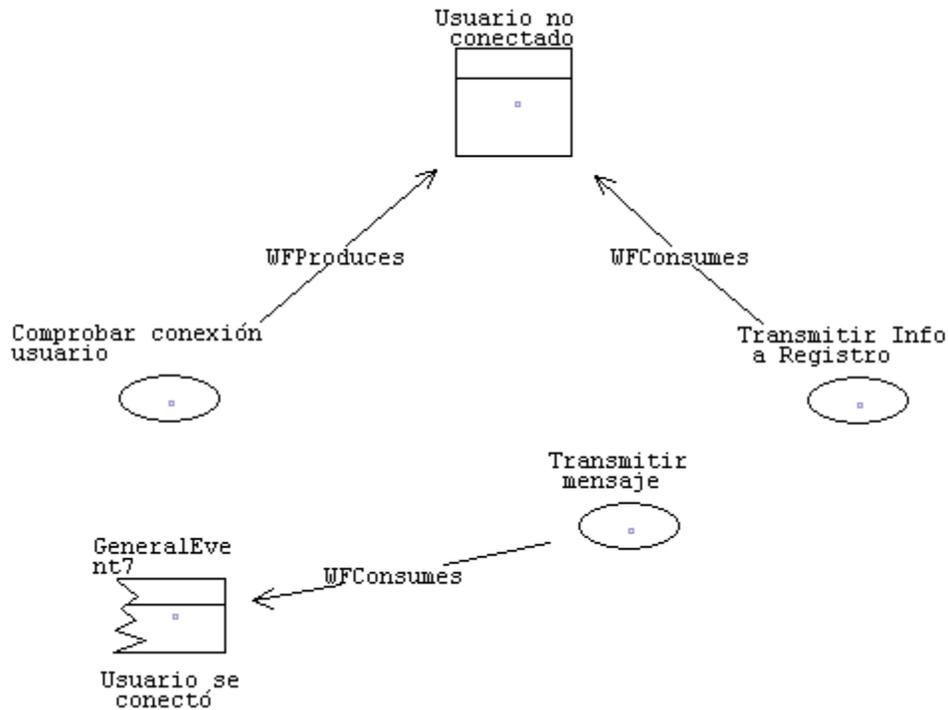


Figura 3.32. Modelo de objetivos y tareas agente Gestor.

3.9.4 Diseño elaboración

En esta etapa se refina un poco más la especificación a través de la identificación de las interacciones entre los agentes que componen el sistema. Para detallar a un nivel más bajo cada interacción se utiliza la especificación GRASIA definida en la metodología INGENIAS, de manera que es posible identificar el tipo de mensajes que se transmite entre los agentes.

Comencemos el análisis con el agente Usuario, el cual interacciona con el agente Registro para efectuar las tareas relacionadas con el registro de usuario nuevo y la validación

3.9.4.1 Interacción de Registro (Int de registro)

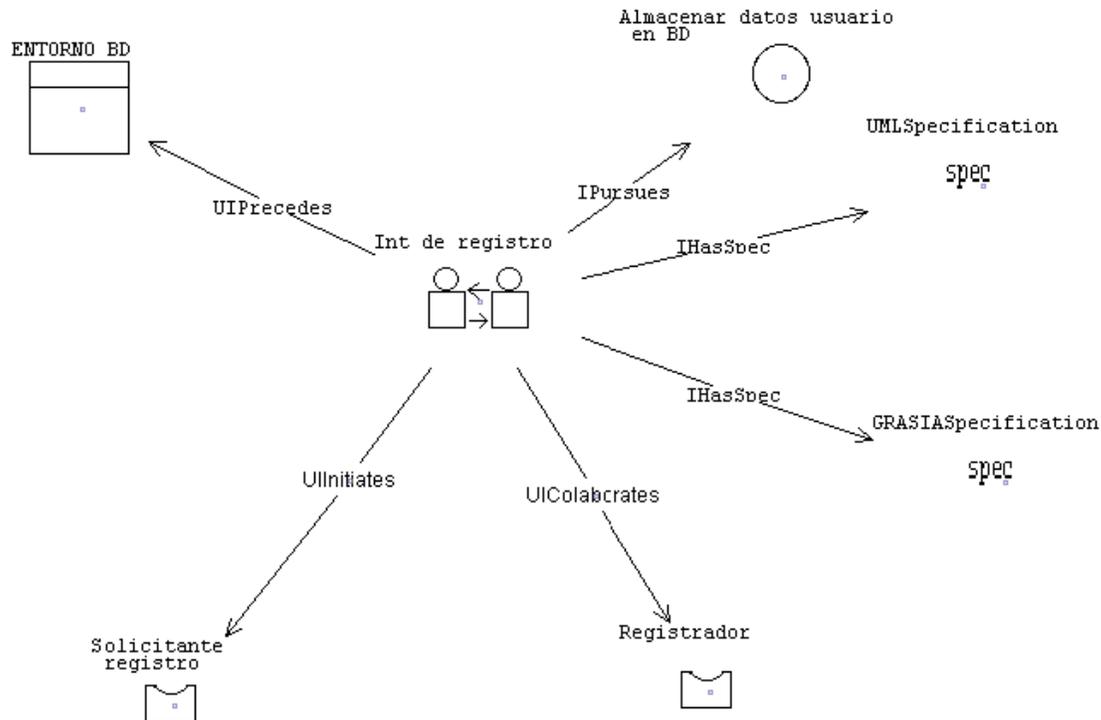


Figura 3.33. Interacción de Registro (Int de registro), agentes Usuario y Registro.

El objetivo perseguido por esta interacción es “Almacenar datos de usuario en BD”. Para lograrlo, el rol *Solicitante de registro* del agente Usuario inicia el proceso de interacción en el cual el rol *Registrador* del agente Registro colabora realizando el almacenamiento de los datos en la Base de datos. Nótese que la Base de datos aparece en el diagrama, siendo ésta parte del meta-modelo de entorno por ser una aplicación externa al sistema de agentes.

El diagrama de especificación GRASIA se incluye a continuación. En la figura se pueden apreciar los mensajes que se transmiten entre los agentes que participan en la interacción, los cuales pueden ser del tipo inform, request, reject o not understood. En las figuras mostradas aquí, la selección del tipo de mensaje queda abierto para la implementación ya que esto depende de la Ontología que se vaya a utilizar y las plantillas de mensajes ACL que se vayan a aplicar de acuerdo a lo que se plantea en JADE.

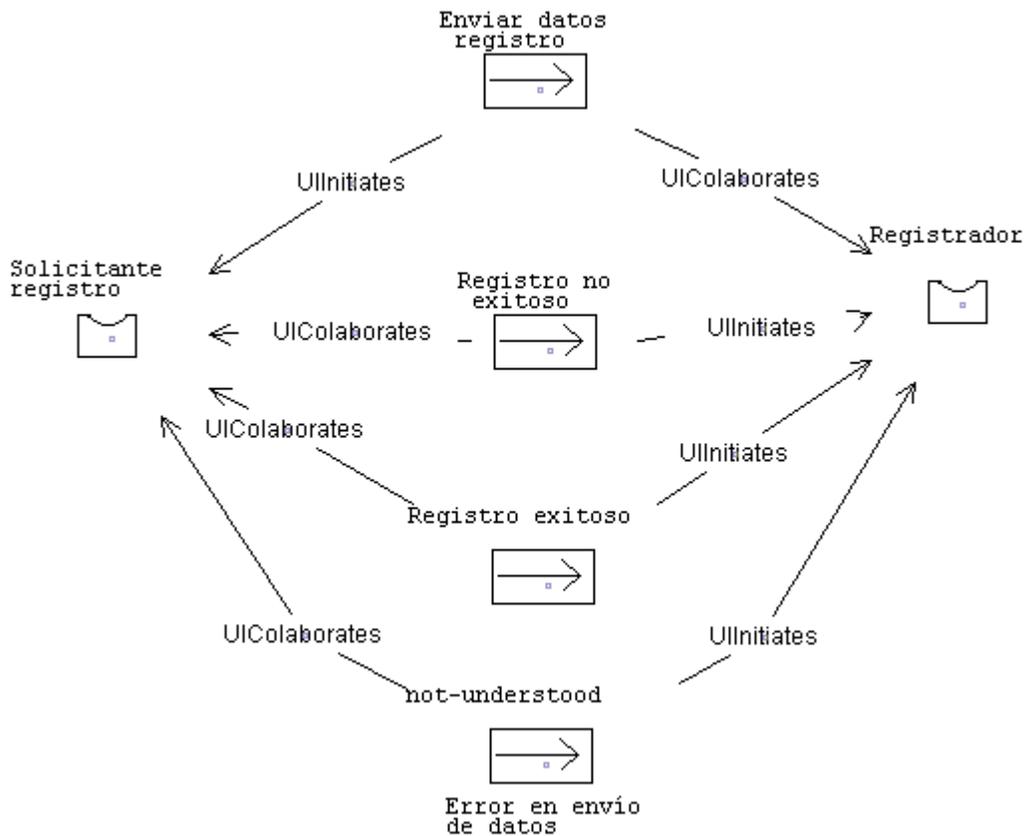


Figura 3.34. Especificación GRASIA Interacción de Registro (Int de registro).

Se ha elegido la especificación GRASIA y no UML, debido a que la primera esquematiza de manera más completa el intercambio de mensajes entre agentes, acercándose más a la implementación.

3.9.4.2 Interacción de Validación

El objetivo perseguido por esta interacción es la validación de un usuario que quiere acceder al servicio “Validar usuario”. Para lograrlo el rol *Solicitante validación* inicia el proceso en el cual el rol *Validador* del agente Registro participa como colaborador, verificando los datos enviados en la Base de datos.

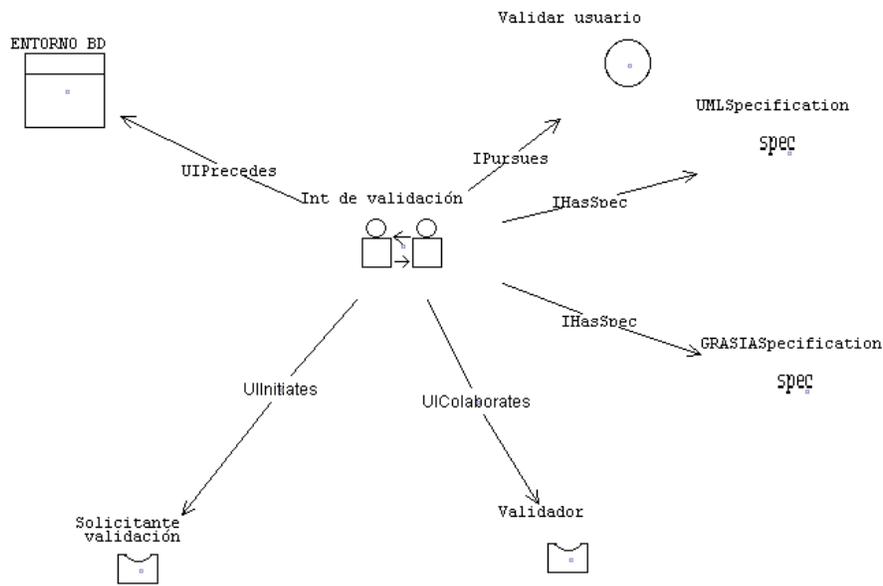


Figura 3.35. Interacción de validación (Int de validación), entre agentes Usuario y Registro.

La especificación GRASIA en este caso se muestra en la siguiente figura.

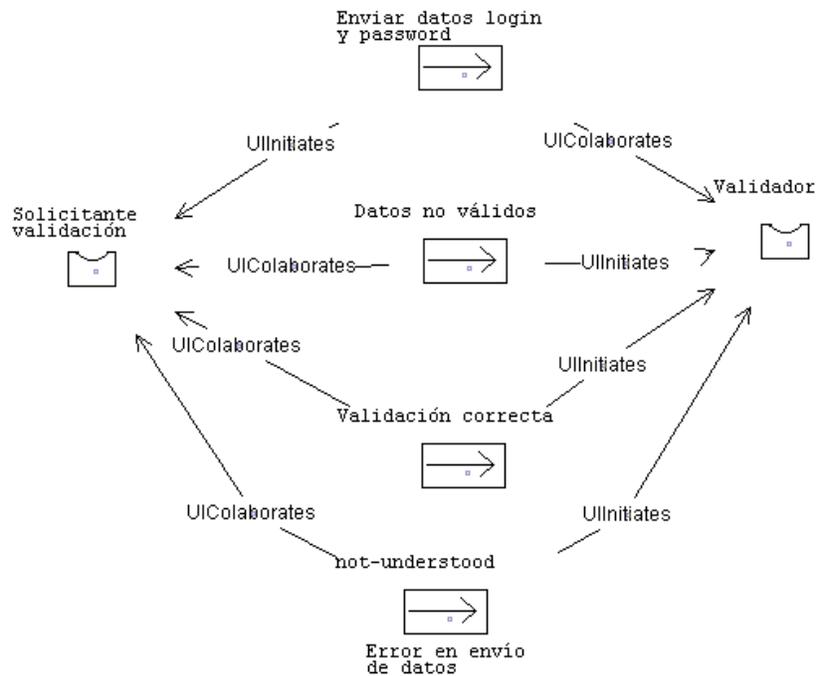


Figura 3.36. Especificación GRASIA Interacción de Validación (Int de validación).

3.9.4.3 Interacción Gestor-Registro

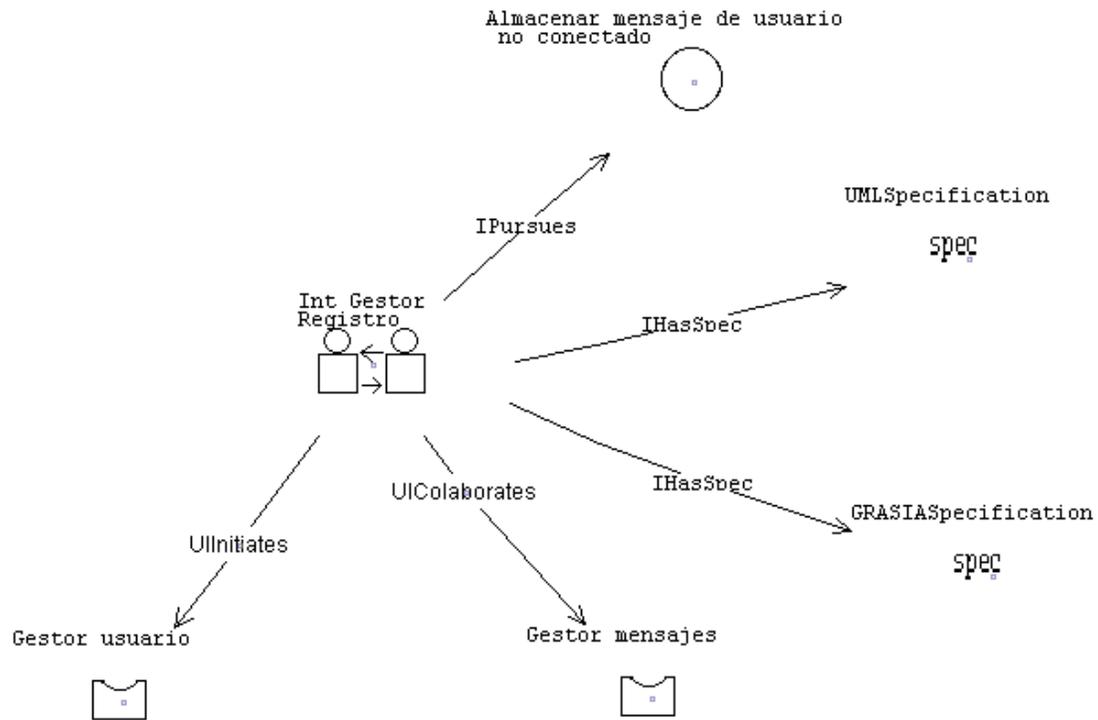


Figura 3.37. Interacción Gestor-Registro, entre agentes Gestor y Registro.

El objetivo de esta interacción es “Almacenar mensaje de usuario no conectado”. Esto significa que cuando un usuario destinatario de un mensaje no está conectado al servicio, el agente Gestor a través de su rol *Gestor usuario* inicia un proceso de interacción con el agente Registro (rol *Gestor mensajes*), con el fin de almacenar dicho mensaje en la Base de datos para su posterior entrega.

El diagrama de Especificación GRASIA se muestra a continuación.

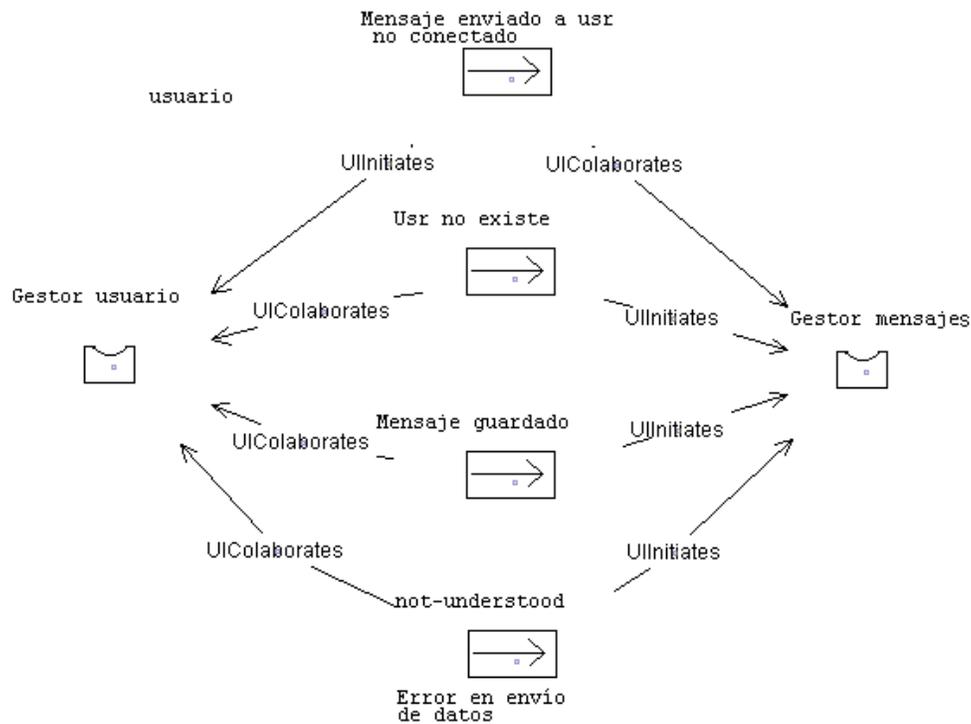


Figura 3.38. Especificación GRASIA Interacción Gestor-Registro.

3.9.4.4 Interacción Gestor-Usuario

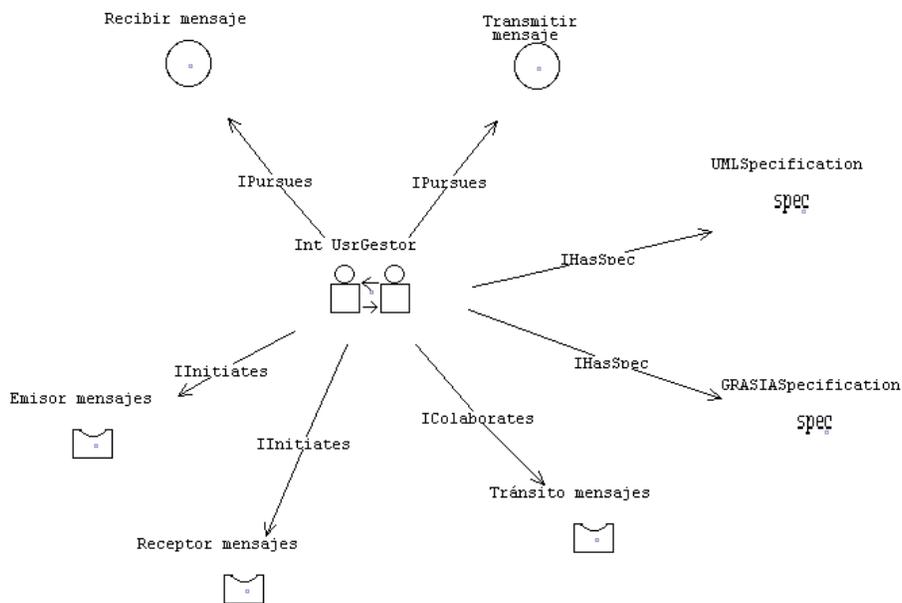


Figura 3.39. Interacción Usuario-Gestor, entre agentes Usuario y Gestor.

En esta interacción, cuyos objetivos son “Transmitir y Recibir mensajes”, el agente Usuario inicia la

interacción a través de los roles *Emisor* y *Receptor de mensajes* con los cuales emite y recibe mensajes hacia y desde otros usuarios. El rol *Tránsito mensajes* el cual es ejecutado por el agente Gestor, quien en este caso se desempeña como colaborador, se encarga de procesar el tráfico de mensajes entre los usuarios del servicio.

La siguiente figura muestra la especificación GRASIA para esta interacción. En primer lugar se muestra la interacción del rol *Tránsito mensajes* con el rol *Emisor mensajes* y posteriormente con el rol *Receptor mensajes*.

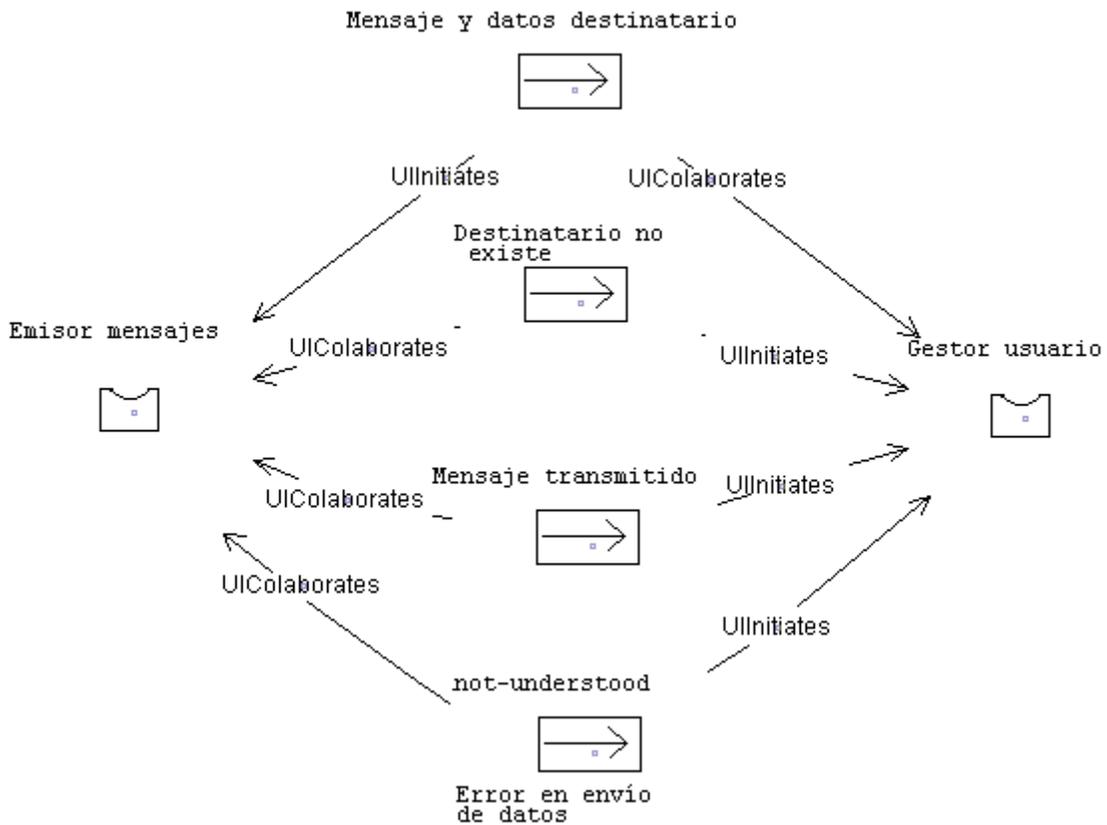


Figura 3.40. Especificación GRASIA Interacción Gestor-Usuario (Rol Receptor mensajes).

4. CONCLUSIONES Y TRABAJOS FUTUROS

4.1 CONCLUSIONES

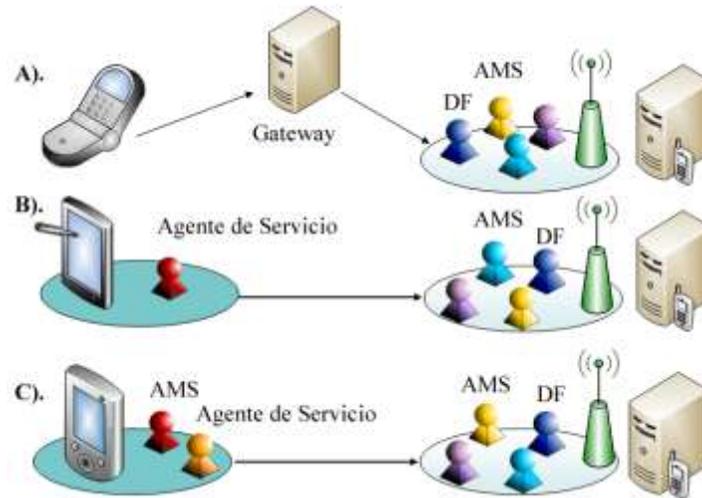


Figura 4.1. Modos de operación.

Los agentes en general requieren de un entorno especial para su ejecución, a este se lo denomina plataforma. En la figura 4.1 podemos observar los modos en que una plataforma de agentes puede operara en la prestación de un servicio. Además de aportar la capacidad para la ejecución de agentes debe brindar una serie de servicios básicos tales como:

- *Servicio de comunicación:* Conjunto de facilidades que permiten la comunicación de los agentes con el mundo exterior, principalmente, favoreciendo la interacción con otros agentes en tareas cooperativas.
- *Servicio de nombrado:* Capacidad de nombrar de forma unívoca a los agentes, a las plataformas y los nodos de manera que puedan ser identificados sin error.
- *Servicio de descubrimiento y localización:* Facilita el descubrimiento de agentes y plataformas con los que puede interactuar, o a las que se puede migrar.
- *Servicio de Movilidad:* Facilita la migración de agentes entre nodos y plataformas remotas.
- *Servicio de Seguridad:* Garantiza la protección de los agentes y de la plataforma ante posibles ataques.
- *Servicio de Transferencia de Mensajes (MTP):* Servicio de comunicación que permite el intercambio de mensajes entre agentes en distintas plataformas.

Según el número de servicios que inicialice el dispositivo móvil la plataforma se puede clasificar en:

- *Plataforma de agentes empotrada en el dispositivo móvil:* El terminal tiene instalado un completo ambiente para la ejecución de agentes que soporta la gran mayoría de los servicios básicos anteriormente mencionados.
- *Plataforma de agentes subrogada (modo vicario):* En este caso la funcionalidad total de la plataforma de agentes se encuentra dividida en dos o mas partes, es decir, que en el dispositivo móvil se encuentran algunos de los servicios básicos y el resto son soportados por otros terminales con mayor capacidad de procesamiento, actuando como delegados, representantes o vicarios del terminal móvil.
- *Plataforma de agentes de tipo portal:* La plataforma de agentes instalada en el dispositivo móvil solo se encarga de recibir y enviar información a uno o varios terminales delegados que son los encargados de albergar el ambiente para la ejecución de agentes; esta plataforma no implementa ninguno de los servicios básicos.

En la medida que una plataforma ofrezca más servicios mejores serán sus prestaciones pero el consumo de recursos ira en aumento. Particularmente en el proyecto se probaron los esquemas de operación B y C de la anterior figura, bajo la siguiente especificación.

Tabla 4.1. Configuración de los prototipos.

Perfil	Servicios	Modo	Dispositivo
MIDP 1.0	Servicio de comunicación.	subrogada	PALM TUNGSTEN T y T2
Personal Profile	Servicio de comunicación	subrogada	Emulador NOKIA serie 80
	Todos los servicios.	Empotrado	Emulador NOKIA serie 80

Como se detalla en el anexo C la aplicación desarrollada y probada sobre los dispositivos móviles arrojaron un bajo desempeño debido principalmente a los siguientes factores:

- Alto consumo de recursos por parte de la plataforma de agentes (JADE).
- Aunque el dispositivo contaba con una buena configuración física la maquina virtual no las aprovechaba al máximo.
- Si bien el protocolo de comunicación entre usuarios móviles es muy completo y provee muchas ventaja, el JICP es un protocolo de tipo peer to peer lo que implica que en un momento dado el dispositivo personal pueda comportarse como un servidor, aun que esto podría tomarse como una ventaja, en la practica la demanda de recursos que ello implica causa constantes bloqueos en el dispositivo.

Recomendaciones:

Descartar el protocolo JICP para comunicaciones entre la plataforma principal y el dispositivo móvil, en cambio emplear el protocolo HTTP por las siguientes razones:

- Permite conexiones intermitentes entre el usuario móvil y el contenedor principal.
- El protocolo HTTP es el más difundido para aplicaciones en dispositivos móviles.
- Java provee librerías optimizadas para comunicaciones HTTP.

La transferencia de mensajes ACL se debería realizar con XML y no con marcadores (token) como lo realiza JADE.

El modo de operación empotrado probado en el emulador 9200 de NOKIA funciono correctamente y logro su objetivo (portar por completo una plataforma de agentes a un dispositivo móvil) pero seguramente si se probara sobre un dispositivo real ya sea este u otro que soporte el perfil Personal Java su resultado no seria muy distinto al obtenido con la PALM TUNGSTEN.

Aun que los resultados obtenidos con el perfil PJava no son muy distintos a los conseguidos con el perfil MIDP, la capacidad física de estos dispositivos (recursos hardware) los convierte en el futuro de los agentes en ambientes móviles, ya que la maquina virtual de este perfil posee tiene mas prestaciones y algo muy importante; es reflexiva lo que permite en teoría la migración de agentes entre dispositivos.

En la figura 4.1 en el numeral (A) se encuentra el modo de operación portal el cual no fue cubierto por esta investigación pero a juzgar por el desempeño obtenido en los otros modos (subrogado y empotrado), este podría ser el esquema mas idóneo en el que un dispositivo móvil acceda a la funcionalidad provista por un sistema de agentes. En este una gateway o interfaz ejecuta por completo la plataforma de agentes evitando que el dispositivo móvil gaste recursos en ello, la gateway alberga un agente que simula estar en el dispositivo de computo personal y se encarga de comunicarse con terminal móvil, ocultándole al servicio esta operación.

En conclusión el uso de una plataforma de agentes como JADE en dispositivos móviles aun no es la mejor opción para implementar servicios, frente a otras alternativas como: wap, XHTML, aplicaciones J2ME normales y etc. Pero a pesar de ello es la mejor plataforma para realizar pruebas y su valor no esta en su posible comercialización o desempeño si no en ayudar a desarrolladores interesados en la materia a depurar o decantar conceptos del funcionamiento de los agentes, para que en el futuro este paradigma pueda perfilarse como una verdadera tecnología.

Actualmente el desempeño de los dispositivos móvil no es lo suficientemente bueno como para pensar en ofrecer servicios comerciales basados en la tecnología de agentes, pero esta situación ira cambiando paulatinamente con las mejoras tecnológicas que estos experimentan.

Teniendo en cuenta lo anteriormente expuesto y si bien los resultados técnicos obtenidos con los prototipos no son los mejores la idea de utilizar agentes en la prestación de servicios no esta errada, ya que estos aportan las siguientes ventajas:

- El uso de agentes permite ubicarlos libremente en diferentes nodos en un sistema de telecomunicaciones sin mayores cambios en el código o estructura del servicio, ya que la plataforma se encarga de encontrarlos.
- Flexibiliza el desarrollo de servicios ya que cada agente puede desarrollarse por separado y luego integrarse al sistema.
- Comunicación asíncronas basadas en mensajes.
- Disminución de la complejidad en el desarrollo de aplicaciones distribuidas, ya que el framework se encarga de hacerlo por el desarrollador.
- Cada agente es autónomo lo que disminuye la dependencia de componentes al interior de un servicio.
- El uso de ontologías para expresar y compartir el conocimiento entre agentes es un buen mecanismo para desagregar a los agentes del entorno y de ellos mismos, en el caso hipotético de que un grupo grande de desarrolladores deseen implementar un aplicación basada en agentes lo primero que deben hacer es definir las ontologías y los procesos de interacción entre agentes, después de esto cada equipo se dedica a implementar al agente sin necesidad de la intervención de los restantes miembros del grupo.

MR-DSOA transforma una representación abstracta y general de la funcionalidad de un servicio en un diseño muy completo que considera tanto a los agentes como el ambiente donde se ejecutaran (sistema de telecomunicaciones).

Los modelos definidos por MR-DSOA describen al sistema desde diferentes perspectivas a nivel empresarial, computacional, de información, de tecnología y de ingeniería, permitiendo así, considerar cada uno de los aspectos que intervienen en la prestación del servicio.

El uso de una metodología propia del paradigma de agentes, en este caso INGENIAS, ayuda a describir al sistema como una implementación netamente software, es decir, desde el punto de vista de los agentes que van a implementar la funcionalidad del servicio.

El uso de AUML como lenguaje de especificación para las interacciones entre agentes permitió cubrir aspectos no considerados por la metodología INGENIAS, aportando mayor claridad al diseño, puesto que permiten al desarrollador observar ciertos comportamientos externos del agente ante determinados mensajes.

A nivel de implementación el uso de una plataforma de agentes como JADE aporta muchas ventajas considerando la funcionalidad que esta ofrece, pero es importante para el éxito de un proyecto basado en agentes conocer cuales de las características que hacen de los agentes tan útiles se pueden implementar y cuales no; para esto es vital documentarse sobre la o las plataformas que se vayan a utilizar antes de definir las funciones que el servicio desempeñará.

JADE posee algunas falencias en cuanto a documentación y desempeño; sin embargo, es la mejor plataforma ya que posee una comunidad de investigadores que constantemente la están actualizando, publicando ejemplos, desarrollando módulos y etc.

4.2 TRABAJOS FUTUROS

Aunque la concepción de agentes y sistemas multiagentes ha estado presente desde hace muchos años es apenas ahora con los progresos en inteligencia artificial y el acelerado desarrollo en las tecnologías software: lenguajes de programación, técnicas de modelamiento y etc. Que el paradigma de agentes ha cobrado fuerza y relevancia, debe ser claro para un nuevo investigador que actualmente muchas de las principales características atribuidas a un agente no han sido implementadas por ejemplo: reactividad, proactividad, plantación, movilidad y etc. Por lo tanto desarrollos encaminados en este sentido serian un aporte realmente importante por que ninguna plataforma de agentes provee estas propiedades.

Los esfuerzos investigativos se pueden enfocar en dos aspectos:

1. Tecnológicos:

- Desarrollo de nuevos componentes software que implemente características teóricas de los agentes y que aun no tienen equivalente software como por ejemplo: proactividad, reactividad, inteligencia y etc.
- Perfeccionamiento o mejoras en las herramientas para el desarrollo de agentes.
- Desarrollo de una plataforma de agentes optimizada para dispositivos móviles.
- Descubrimiento de servicios basados en agentes.
- Integración de agentes y Web Services.
- Integración de agentes y OSA/PARLAY

2. Teórico:

- Refinamiento del MR-DSOA.
- Mejoramiento del lenguaje de especificación para agentes (AUML).

El área de agentes es un campo bastante extenso debido a que en el confluyen muchas tendencias y disciplinas por esta razón hay muchos temas por investigar, pero se debe tener cuidado con la definición de los alcances de un proyecto por que hay muchas investigaciones poco serias que publican resultados falsos y que contribuyen a la desinformación en la materia.

GLOSARIO

ACC	Canal de Comunicación de Agentes. Es la unión directa entre dos o más agentes que desean comunicarse.
ACL	<i>Agent Communication Language</i> . Representa la carga útil de los mensajes transportados tanto por el MTP y el MTS.
AID	<i>Agent Identifier</i> . Identificador único de agente.
AMS	<i>Agent Management Service</i> . De acuerdo con la arquitectura FIPA, es el agente responsable de proveer el servicio de páginas blancas.
AP	<i>Agent Platform</i> . Plataforma de agentes o implementación de la PCA.
BDI	<i>Beliefs, Desires and Intentions</i> . Modelo de agente basado en deseos, creencias e intenciones.
Bluetooth	Estándar global de comunicaciones inalámbrica, que posibilita la transmisión de voz y datos entre diferentes equipos mediante un enlace por radiofrecuencia.
DF	<i>Directory Facilitator</i> . De acuerdo con la arquitectura FIPA es el agente que provee el servicio de páginas amarillas.
DPE	<i>Distributed Processing Environment</i> . Ambiente de Procesamiento Distribuido.
FIPA	<i>Foundation for Intelligent Physical Agents</i> . Organización sin ánimo de lucro que publica y discute estándares para sistemas de agentes.
GATEWAY	Puerta en comunicaciones de red. Es una combinación de programas y hardware que comunica dos tipos diferentes de redes.
GMDO	<i>Guidelines for the Definition of Managed Objects</i> .
Gprs	<i>General Packet Radio Service</i> . Es un servicio de valor agregado que permite transmitir y recibir información a través de redes de telefonía móvil.
HTTP	<i>HyperText Transfer Protocol</i> . Es un conjunto de reglas para el intercambio de archivos (texto, gráficos, sonido, video y otros) en la Web. HTTP es un protocolo de aplicación.

HOST	En Internet, el termino "host" significa cualquier computador que tenga acceso total bidireccional a otros computadores en la red. Un host tiene un número local específico que, junto con el número de red, forma una dirección IP única. En otras palabras, un "host" es un nodo en una red.
INGENIAS	Metodología propia del paradigma de agentes que propone un lenguaje de especificación para Sistemas Multi-Agente.
IOR	<i>Interoperable Object Reference</i> . Referencia para la ubicación de un objeto en un sistema distribuido CORBA.
JAVA	Lenguaje de programación orientado a objetos diseñado para el uso en entorno distribuido como Internet. Java puede ser utilizado para crear aplicaciones completas que pueden correr en un computador o pueden ser distribuidas a través de servidores, clientes y etc.
JADE	<i>Java Agent DEvelopment Framework</i> . Ambiente para el desarrollo de agentes.
MAFS	<i>Mobile Agent Facility Specification</i> . Especificación de la OMG para agentes.
MTP	<i>Message Transport Protocol</i> . De acuerdo con la arquitectura FIPA, es el componente responsable de manejar las comunicaciones entre plataformas y agentes externos.
MR-DSOA	<i>Modelo de Referencia para el Desarrollo de Servicios Orientados a Agente</i> . Modelo para el desarrollo de servicios con orientación a agentes.
NCCE	<i>Native Computing and Communications Environment</i> . Incluye todo el software de comunicaciones y de computación nativa, por ejemplo sistemas operativos, software empotrado y demás herramientas de soporte típicas de un sistema de telecomunicaciones.
OMT	<i>Object Modeling Technique</i> . Metodología para el desarrollo aplicaciones con orientación a objetos.
PCA	<i>Plataforma Conceptual de Agente</i> . Abstracción de las funciones de una plataforma de agentes que permite la ejecución de estos en ambientes distribuidos como son los sistemas de telecomunicaciones.
QoS	<i>Quality of Service Networking</i> . La calidad de servicio se refiere a la capacidad de una red para proveer mejor servicio a un tráfico de información específico sobre diferentes tecnologías.

ROSA	<i>Arquitectura Abierta de Servicios RACE.</i> Una de las primeras arquitecturas abiertas para servicios de telecomunicación.
RM-ODP	<i>Reference Model for Open Distributed Processing.</i> Modelo de referencia para el desarrollo de aplicaciones distribuidas.
TINA	<i>Telecommunication Information Networking Architecture.</i> Especificación de una arquitectura abierta de un sistema de telecomunicaciones.
TMN	<i>Telecommunications Management Network.</i> El termino TMN es introducido por la ITU-T como un concepto que abstrae el funcionamiento de una red de telecomunicación y que permite su gestión.
UML	<i>Unified Modeling Language.</i>
Wlan	<i>Wireless Local-Area Network.</i> Sistema de comunicaciones de datos vía inalámbrica de reducido radio de cobertura y consumo de potencia. La regulación de esta tecnología esta a cargo de la IEEE (Instituto de Ingenieros Eléctricos y Electrónicos).
XML	<i>Extensible Markup Language.</i>

BIBLIOGRAFIA

CAPITULO I

- [Bigus 2001] BIGUS, J y BIGUS, J P. Constructing Intelligent Agents Using Java. Wiley Computer Publishing, 2001.
- [Brooks 1986] BROOKS, R.A. A robust layered control system for a mobile robot. IEEE journal of Robotics and Automation 2:14-23. 1986.
- [Bond 1988] BOND, H. y GASSER, L. Readings in Distributed Artificial Intelligence. Morgan Kaufmann, San Mateo, CA, 1988.
- [Hoare 1969] HOARE, C.A.R. An axiomatic basis for computer programming. Communications of the ACM, 12(10), 576-583. 1969
- [IBM 1996] IBM. Intelligent Agent Resource Manager, Open Blueprint, G325-6592-0. 1996
- [Moore 1990] MOORE, R.C. A formal theory of knowledge and action. In Readings in Planing (eds J.F. Allen, J. Hendler and A. Tate), pp. 480-519. Morgan Kaufmann, San Mateo, C.A, 1990.
- [Pnueli, 1986] PNUELI, A. Specification and development of reactive systems. In information processing 86, pp. 845-858. Elsevier, Amsterdam, 1986.
- [Wooldrige 1995] WOOLDRIGE, M. This is MY WORLD: the logic of an agent-oriented tested for DAI. In Intelligent Agents: Theories, Architectures and languages (eds M. Wooldrige and N. R. Jennings), LNAI Vol 890, pp. 160-178. Springer, Berlín, 1995.

CAPITULO II

- [Abarca 1997a] ABARCA, C; FARLEY P, FORSLÖW J. Service Architecture. TINA Consortium, junio, 1997.
- [Abarca 1997b] ABARCA Chelo; FORSLOW Jan y HAMADA Takeo. Network Resource Architecture. TINA Consortium, febrero, 1997.

- [Bauer 2005] BAUER Bernhard y ODELL James. UML 2.0 and Agents: How to Build Agent-based Systems with the new UML Standard. Journal of Engineering Applications of Artificial Intelligence Volume 18, Issue 2 , Marzo 2005, Pages 141-157.
- [Chapman 1995] CHAPMAN, Martin y MONTESI, Stefano. Overall Concepts and Principles of TINA. TINA Consortium, febrero, 1995.
- [Christensen 1995] CHRISTENSEN H y COLBAN E. Information Modelling Concepts. TINA Consortium, abril, 1995.
- [[FIPA 2002a] FIPA. Abstract Architecture Specification SC00001L. Foundation for Intelligent Physical Agents, Geneva, Switzerland, diciembre, 2002.
- [FIPA 2002b] FIPA. ACL Message Structure Specification SC00061G. Foundation for Intelligent Physical Agents, Geneva, Switzerland, marzo, 2002.
- [FIPA 2002c] FIPA. FIPA Communicative Act Library Specification SC00037J. Foundation for Intelligent Physical Agents, Geneva, Switzerland, marzo, 2002.
- [FIPA 2002d] FIPA. FIPA Request Interaction Protocol Specification SC00026H. Foundation for Intelligent Physical Agents, Geneva, Switzerland, marzo, 2002.
- [FIPA 2003] FIPA. FIPA Modeling: Interaction Diagrams. Foundation for Intelligent Physical Agents, Geneva, Switzerland, marzo, 2003.
- [ITU-T 1995] ITU-T X.901. ODP Reference Model Part 1: Overview. ISO/IEC 10746-1, Helsinki, Finland, mayo, 1995.
- [Lucking 2000] LUCKING-REILEY David y SPULBE Daniel F. Business-to-Business Electronic Commerce. Publicado en: Journal of Economic Perspectives. Vol 26, N° 10, noviembre, 2000.
- [Noy 2001] NOY, Natalya F y MCGUINNESS, Deborah L. Ontology Development 101: A Guide to Creating Your First Ontology. Stanford University, Stanford, marzo, 2001.
- [OMG 2000] OBJECT MANAGEMENT GROUP (OMG Inc). Mobile Agent Facility Specification, enero, 2000.
- [Pressman 1993] PRESSMAN, Roger S. Ingeniería del software un enfoque practico. Traducción: RUCKAUERS C, Carlos. Tercera edición, Madrid, España, McGraw-Hill, enero, 1993.

-
- [ROSA 1992] OSHISANWO, A. O; IHAPMAN, D; ULLERY A. P y SAINT-BLANCAT, J. The RACE Open Services Architecture project. Publicado en: IBM Systems Journal, Vol 31, N° 4. 1992.
- [TINA-C 1996] TINA-C Core Team Members. Computational Modelling Concepts. TINA Consortium, Mayo, 1996.
- [Tocher 1991] TOCHER, Alastair J. A Formal Model for Naming. ANSA, Advanced Networked Systems Architecture. Cambridge, United Kingdom, diciembre 1991.
- [Torres 1990] TORRES GARCIA, Arturo. Plantación estratégica y plantación tecnológica. Publicado en: Gestión tecnológica en la empresa. Colección ciencia y tecnología N° 27. CINDA. Santiago, Chile, agosto, 1990.
- [Yates 1997] YATES, Martin; TAKITA, Wataru; DEMOUEM, Laurence; JANSSON, Rickard; MULDER, Harm. TINA Business Model and Reference Points. TINA Consortium, mayo, 1997.