

AGENTE USUARIO PARA EL PORTAL UNIVERSIDAD DE LOS ANDES

MERILIN OSPINO REALES

**UNIVERSIDAD DE LOS ANDES
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
BOGOTÁ D.C.**



AGENTE USUARIO PARA EL PORTAL UNIVERSIDAD DE LOS ANDES

MERILIN OSPINO REALES

**Proyecto de grado para optar al título de magíster en ingeniería de
sistemas y computación**

**ASESOR
ING. FRANCISCO RUEDA
PROFESOR FACULTAD DE INGENIERÍA
UNIVERSIDAD DE LOS ANDES**

**UNIVERSIDAD DE LOS ANDES
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
BOGOTÁ D.C.**



AGRADECIMIENTOS

En cada momento de la vida mi vida es solo agradecimientos a Dios por permitirme tener estas bellas oportunidades, a mis padres por ser el pilar de mi caminar, a mis hermanos por brindarme su apoyo incondicional, a todos mis amigos que creen en mi y que saben que amo lo que hago y a Jose por que su hombro y su amor fueron testigos de los sacrificios y alegrías que trajeron a mi vida estos años, me lo goce como todo en mi vida y que Dios me bendiga para continuar en este camino que apenas empieza...

TABLA DE CONTENIDO

1	INTRODUCCIÓN	7
1.1	Definición del problema	8
1.2	Objetivos	9
1.1.1	Objetivo General	9
1.1.2	Objetivos Específicos	9
1.3	Solución Aportada	9
1.4	Limitaciones	10
1.5	Terminología empleada	10
2	ESTADO DEL ARTE	11
2.1	Introducción	11
2.2	Agente Usuario	11
2.3	Web Semántica	15
2.4	Generación Dinámica de Interfaces a partir del Conocimiento	18
2.5	Sistemas Multimodales	20
2.5.1	Proxy Multimodal. Herramienta de Interacción Multimodal.	30
2.5.2	Interfaces de Usuario Multimodales	33
2.6	Sistemas Basados y Sensibles en Contexto	37
2.7	OSGI	40
3	SISTEMA DE COMUNICACIÓN INTERMODULOS PARA UN PORTAL DE SERVICIOS PARA SISTEMAS MÓVILES SENSIBLE AL CONTEXTO. CASO UNIVERSIDAD DE LOS ANDES.	42
3.1	Introducción	42
3.2	Descripción Del Sistema	43
3.3	Arquitectura Del Sistema	45
3.3.1	Servicios ofrecidos por el Portal	48
3.3.1.1	PULL. Consulta del Usuario.	49
3.3.1.2	PUSH. Promoción de Ofertas de Servicios.	50
4	MODULO AGENTE USUARIO INTERFAZ	51
4.1	Introducción	51
4.2	Descripción Del Modulo.	52
4.2.1	Funciones	52
4.3	Modelo Del Modulo	54
4.4	Arquitectura Del Modulo	55
4.5	Definición de los Componentes de la Arquitectura	57
4.5.1	Multimodal.	57
4.5.2	Conocimiento. Manejador de Ontologías	58
4.5.2.1	Características del Módulo.	58
4.5.2.2	Funciones del módulo	59
4.5.3	Composición. Generador de Documentos Web.	59
4.5.3.1	Características del Módulo.	60
4.5.3.2	Funciones del módulo	61
4.5.4	Visualización. Generador de Vistas dinámicas a Partir de Conocimiento.	62
4.5.4.1	Características del Módulo.	62

4.5.4.2	Funciones del módulo	65
4.5.4.3	Ontologías	66
5	IMPLEMENTACIÓN MODULO GENTE USUARIO INTERFAZ	71
5.1	Lenguaje de Implementación	71
5.2	Descripción de la Implementación realizada	72
5.3	Servicios Implementados	73
	CONCLUSIONES Y TRABAJO FUTURO	76
	ANEXOS	78
	BIBLIOGRAFIA	83

TABLA DE FIGURAS Y TABLAS

Figura 1. Interacciones a través de WAP o HTTP por el Proxy	32
Figura 3. Modelo Portal de Servicios para Sistemas Móviles.	43
Figura 4. Arquitectura de Sistema.	46
Figura 5. Modelo Agente de Usuario para Portal Uniandes.....	54
Figura 6. Arquitectura del sistema	55
Figura 7. Interacción WAP o HTTP	58
Figura 8. Grafo Ontología de Visualización	68
Figura 9. Grafo Ontología de Servicio	70
Tabla 1. Principales tipos de Servicios. Adaptado de [Chen, 2005b].....	49

1 INTRODUCCIÓN

Actualmente uno de los principales ejes alrededor de los cuales gira la nueva economía mundial es el mercado de las Tecnologías de la Información y las Comunicaciones (TIC). Este mercado sustenta las bases de un nuevo entorno social y económico caracterizado por el destacado papel del conocimiento y la información como fuente de riqueza.

Hoy en día son innumerables los dispositivos que usamos y que, sin darnos cuenta, de una manera u otra se comunican por medio de ondas de radio, o por infrarrojo. Difícilmente contemplaríamos nuestra vida sin estos dispositivos; todo lo contrario, vemos la acogida de la telefonía celular, que es un fenómeno donde crecen los usuarios, las aplicaciones y los tipos de conectividad. Esto se debe a que los dispositivos móviles, proporcionan conectividad, utilidad y movilidad; factores que agilizan nuestras actividades.

En un futuro muy cercano, todos nos conectaremos a Internet inalámbricamente desde un portátil, un PDA, o desde un celular; y la constante necesidad de estar conectados con nuestro negocio, nuestra universidad, o con nuestra familia, moldeará nuestra vida y cambiará nuestra forma de comunicarnos, como ha venido ocurriendo hasta ahora.

El presente trabajo hace parte del macroproyecto de maestría titulado “Portal de Servicios para Sistemas Móviles, un enfoque desde la computación Context-aware”, la cual realiza un análisis conceptual y describe una arquitectura de referencia y modelos de interacción de un sistema orientado a la prestación de servicios móviles delimitado por el contexto del usuario.

En el transcurso del libro se abordará en el capítulo 1 la definición del problema a resolver, cuáles son los objetivos de la investigación y la solución planteada como acercamiento para resolver el problema, en el capítulo 2, se hace un acercamiento a todas las tecnologías que se utilizan en el marco de esta tesis,

como base de la investigación, en el capítulo 3 se describe de manera general el macroproyecto que enmarca esta tesis, Portal Universidad de los Andes, para luego pasar a detallar en el capítulo 4 el resultado de la investigación propósito, Modulo Agente Usuario, diseño, estructura y componentes, en el capítulo 5 se muestra la manera como se implementó y se termina con un caso ejemplo que muestra de una manera más clara la funcionalidad del componente.

1.1 Definición del problema

En los sistemas actuales, que mantienen una interacción con el usuario, se empieza a presentar una inquietud, sobre qué usuario lo utiliza y cómo, a través de esta interacción, se puede aprender más sobre él, la importancia en las interfaces ha empezado a reconocerse de tal manera que muchos estudios ya se encuentran en este campo. Con esta visión, se piensa en los agentes como medio para conseguir este fin, el trabajo de un agente es el de actuar autónomamente para satisfacer deseos que el usuario tiene, además, de llegar a conocerlo que llegue a proponerle servicios o simplemente poder llegar a asumir lo que le gustaría hacer o tener, permiten dar una mayor sensación de productividad y reducir el trabajo del usuario.

En esta misma tendencia, el portal contextual Uniandes, pretende lograr que la interacción con sus usuarios sea constructiva, personalizada y más intuitiva. El trabajo de asistencia al usuario pretende ser abarcado en esta tesis, buscando el apoyo de las nuevas tecnologías y un trabajo colaborativo con los otros módulos que componen el portal.

1.2. Objetivos

1.1.1 Objetivo General

Diseñar y construir un módulo que haga parte del portal contextual de la Universidad de los Andes, que sea el “asistente” del usuario, realizando las tareas solicitadas, proponiendo servicios que se acerquen al conocimiento que tiene del usuario, que lo represente frente al sistema y que permita el acceso multimodal al portal, desde diversos dispositivos, y mediante varias tecnologías, en especial, las involucradas con la telefonía celular.

1.1.2 Objetivos Específicos

Diseñar un asistente que representa al usuario frente al sistema y le permita el acceso multimodal a recursos de la WWW y que tenga las siguientes características:

- Generación de interfaces dinámicas y contextuales a partir del contexto del usuario y de la especificación de servicios.
- Generación de interfaces multimodales a partir del dispositivo de interacción utilizado por el usuario.
- Comunicación transparente con los otros módulos que componen el portal para cumplir todas las funcionalidades.
- Administración de las políticas de privacidad del usuario que representa frente al sistema.
- Administrar todos los agentes de usuario que se creen a medida que cada usuario ingrese al sistema y que se encargan de recibir las propuestas de información.

1.3. Solución Aportada

Bajo un trabajo colaborativo entre los módulos que componen el portal, se participó en el diseño general del portal, logrando un diseño detallado y cuidadoso, teniendo en cuenta todos los aspectos que se quieren involucrar en

este portal, con el ánimo de diseñar un portal contextual, se decidió usar tecnologías como la Web semántica, que permite el compartir el conocimiento entre módulos del portal y un Framework como OSGI que permite la interacción entre módulos de manera transparente e independiente de la ubicación del mismo.

Como parte de la solución general del portal, el módulo objetivo de esta tesis, sigue esta solución, está sustentado en ontologías, bajo el Framework OSGI, con un sistema multimodal y multiagentes.

1.4. Limitaciones

Una de las principales limitaciones es que el período de desarrollo de algunos módulos involucrados en el portal, no fue paralela al desarrollo de este módulo, determinando que la interacción con otros módulos se manejará a niveles de demo. De igual manera, todas las interacciones quedaron implementadas.

1.5. Terminología empleada

Web semántica, ontologías, interfaces de usuario, recuperación de información, visualización, contexto, OSGI, agentes, multimodalidad.

2 ESTADO DEL ARTE

2.1 Introducción

En este capítulo se describe el estado actual de las tecnologías que forman el núcleo de esta tesis, a saber, ingeniería Ontológica, Agentes Inteligentes en específico el de usuario, sistemas basados en usuarios, servicios y contextos.

En los sistemas actuales, que mantienen una interacción con el usuario, se empieza a presentar una inquietud, sobre qué usuario lo utiliza y cómo, a través de esta interacción, se puede aprender más sobre él, la importancia en las interfaces ha empezado a reconocerse de tal manera que muchos estudios ya se encuentran en este campo. El trabajo de un agente es el de actuar autónomamente para satisfacer deseos que el usuario tiene, dar una mayor sensación de productividad y reducir el trabajo del usuario.

Para este trabajo es importante la congruencia de varios tipos de sistemas; como el centrado en el usuario y su contexto, dado que es un sistema basado en los servicios que ofrece el portal, hay que abarcar este tipo de sistemas, en la investigación fue reconocido otro tipo de tecnologías como el uso de ontologías para aprovechar el conocimiento que el sistema crea para poder hacer inferencias que permitan una interacción con el usuario más confiable y acercada a la realidad, con el ánimo de aprovechar esta tecnología se hace necesario a la vez el estudio de cómo esta se utiliza a la hora de interactuar directamente con el usuario bajo la presencia de una interfaz. El objetivo es seguir la tendencia de los sistemas actuales que pretenden lograr que la interacción con sus usuarios sea constructiva y más intuitiva.

2.2 Agente Usuario

El concepto “Agente” es un término que en las investigaciones tiene variadas definiciones, pero un término que se acerca a los objetivos del portal, es el de

“Agente de Interfaz Inteligente”, se puede decir que las características de este agente son: Adaptación a las preferencias del usuario, interacción multimodal y entradas abiertas. Los agentes son los procesos autónomos o semiautónomos que realizan una misión bien definida.

En el caso de agentes de usuario, también comúnmente referidos como agentes de interfaz, su misión es ejecutar en nombre del usuario que representa (es su delegado). El considerar a agentes de usuario como la base para un estilo alternativo de interacción hombre-máquina brinda una nueva oportunidad para el diseño de los sistemas actuales, pero a la vez, el potencial de los agentes para contribuir a un ambiente más animado, personalizado y cooperativo para el usuario no puede ser sobre enfatizado.

Para permitir la adaptación a las preferencias del usuario el sistema de interfaz debe estar capacitado para asociar condiciones variadas. Así, incorporando facilidades de adaptación en el sistema agente se permite tener en cuenta las preferencias individuales de los usuarios. Para permitir una interacción intuitiva, el agente debe poder entender e integrar las instrucciones de usuarios en diferentes modalidades, es decir, independiente del dispositivo que se utilice para interactuar con el portal.

Para que un agente sea considerado un agente de interfaz, se requiere que se comunique con la persona a través de entradas y salidas en la interfaz del sistema. Esta clase de agente puede “observar” las acciones tomadas por el usuario y la interacción de este con el sistema (en sus solicitudes de servicio o a la vez en aceptación de servicios que el sistema como iniciativa brinde), para aprender a tomar acciones por sí mismo, el agente podría agregar elementos dependiendo del dispositivo que tenga el usuario, esto es utilizar interfaces de usuario adaptativas.

El uso de agentes en la arquitectura de sistemas de información complejos permite abordar problemas como la disponibilidad automática de información que no implica el acceso fácil para cada usuario individual o grupo de usuarios. El volumen, la complejidad y el dinamismo de la información manejada

plantean problemas significativos en el acortamiento del hueco de accesibilidad entre las necesidades de los usuarios y recursos disponibles.

Los agentes también pueden proporcionar insinuaciones al usuario basadas en su conocimiento o en el uso observado por otros usuarios, o ponerse en contacto con otros usuarios (o agentes de usuario) para obtener la información necesaria. Como lo sugiere *Kay*, los agentes de usuario bien pueden ser el salto cualitativo de usar instrumentos manipuladores a la dirección de procesos, los dos caminos en los cuales la gente, históricamente, se ha ampliado.

La puesta en práctica de agentes de usuario, comparte muchos de los desafíos afrontados para el desarrollo de sistemas complejos (como tratar con diversos medios de comunicación, el hardware y plataformas de software posiblemente heterogéneos).

Otra aplicación de los agentes de usuario es la posibilidad de enseñar al usuario que representan; el área de aplicación que tiene la historia más larga de empleo de agentes de interfaz es la de la Inteligencia artificial llamados Sistemas de Formación Inteligentes. Como los sistemas de formación tienden a tratar con usuarios inexpertos, la facilidad de empleo en el interfaz es una consideración principal.

Los sistemas de formación generalmente son puestos en práctica como los consejeros conversacionales que deciden cuándo y cómo presentar el material de formación al estudiante y ayudan al estudiante con preguntas de respuesta o problemas. El agente puede ser provechoso al usuario porque puede tener más conocimiento del material que el estudiante que comienza, y esto también puede codificar la experiencia de profesores en el reconocer problemas comunes o errores.

Igualmente importante es la relación en la que el usuario enseña al agente. Agentes que pueden ser enseñados por el usuario son lo que vamos a conocer como agentes instructibles. El tipo más simple de interfaz de agente ya tiene

sus objetivos y condiciones de prestar sus servicios para el usuario sin ninguna otra entrada explícita por parte del usuario.

El paso siguiente sería si el usuario le dice al agente explícitamente lo que quiere, por comandos o por "programación" del agente en algún tipo de lenguaje procedimental. Este es el tipo de interacción que estamos más acostumbrados en las computadoras de hoy. También estamos ya familiarizados con los problemas de este tipo de comunicación, los lenguajes son inflexibles e implacables, incapaces de tratar con información incompleta, y el proceso de programación es tedioso.

Otra clase de agente aprende simplemente por la observación. Programándolo, por ejemplo, exige la participación activa del usuario, esto puede también en muchas situaciones ser deseable para el agente para simplemente hacer inferencias a partir de la observación de las acciones del usuario sin cualquier entrada adicional. Una clase de agentes de interfaz de usuario llamados agentes de reconocimiento aprenden mirando acciones en el interfaz de usuario y compilando un perfil de usuario.

El perfil de usuario sirve para ayudar al agente a prever lo que el usuario podría querer o necesitar. Estos agentes realizan el reconocimiento, un proceso de mirada delante que prevé las necesidades del usuario e intereses y ahorra tiempo al usuario. La meta no es de ningún modo obtener una "mejor respuesta", sino hacer el mejor uso posible del recurso más limitado y valioso: la atención del usuario.

Teniendo un agente en funcionamiento continuo significa que el agente puede ser más audaz en sus recomendaciones al usuario. No necesita sacar una conclusión perfecta con respecto a qué servicio responde mejor al interés del usuario, ya que es probable que consiga algo mejor en algún momento del futuro, si el usuario continúa navegando en áreas conexas.

2.3 Web Semántica

La Web semántica, pretende convertir la información en conocimiento. Para ello, se basa fundamentalmente en el marcado semántico y descriptivo no sólo de los documentos, sino también de los datos, por medio de información estructurada y legible automáticamente, que proporciona a las máquinas una mayor capacidad para gestionar y recuperar esos datos.

Uno de los ejes principales es la noción de ontología, herramienta clave para representar la semántica del dominio, heredada de la inteligencia artificial, Una ontología se puede definir como una especificación explícita y formal sobre una conceptualización compartida, una de las definiciones más citadas es la anunciada por Gruber (1993):

“Una ontología es una especificación explícita de una conceptualización”

El autor considera que una conceptualización está compuesta por objetos, conceptos y otras entidades que existen en una determinada área y las relaciones que se dan entre ellos. De un modo más abstracto, una conceptualización se puede definir como una interpretación estructurada de una parte del mundo que usan los seres humanos para pensar y comunicar sobre ella. Por explícita, se entiende que los conceptos y las restricciones se definen de forma explícita. Es decir, proporciona un modelo explícito obtenido por consenso descrito en un lenguaje que contiene a los conceptos, propiedades y relaciones más relevantes de un dominio.

La definición de Gruber fue posteriormente refinada por Borst (1997) de la siguiente manera:

“Una ontología es una especificación formal de una conceptualización compartida”

Las ontologías permiten definir de manera compartida y consensuada conceptos sobre un área específica del conocimiento y establecer las

relaciones entre dichos conceptos. Esto permite a humanos y máquinas comunicarse de manera eficiente al proveer una definición común de un dominio. Además las relaciones entre conceptos son la base para el razonamiento semántico.

Finalmente, se puede destacar una definición extensiva de ontología indicando los componentes que la forman. En general, las ontologías proporcionan un vocabulario común de área y definen a diferentes niveles de formalismo, el significado de los términos y relaciones entre ellos. El conocimiento en ontologías se formaliza principalmente usando seis tipos de componentes: clases, atributos, relaciones, funciones, axiomas e instancias (Gruber 1993).

- Una *clase* puede ser algo sobre lo que se dice algo, como por ejemplo un tipo de objeto, la descripción de una tarea, función, acción, estrategia, proceso de razonamiento, etc.
- Los *atributos* representan la estructura interna de los conceptos. Atendiendo a su origen, los atributos se clasifican en específicos y heredados. Los específicos son aquellos que son propios del concepto al que pertenecen, mientras los heredados vienen dados por las relaciones taxonómicas en las que el concepto desempeña el rol de hijo, y por tanto, hereda los atributos del padre.
- Las *relaciones* representan un tipo de relación entre los conceptos de dominio. Entre los diferentes tipos de relaciones posibles, se encuentran las relaciones taxonómicas (“*es_un*”) y las meteorológicas o partonómicas (“*parte_de*”) como relaciones binarias más destacados.

Para la construcción de estas ontologías se han desarrollado varios metalenguajes y estándares, entre los que se tiene el RDF [39] y el OWL [27]. En orden histórico serían SHOE (1997) y después de la aparición de XML en 1998: RDF: 1999, RDF Schema (RDFS), DAML, OIL, DAML+OIL Y OWL.

Una aplicación de las ontologías en la Web Semántica es la visualización y navegación de resultados en un entorno de recuperación de información, la cual da soporte a cuatro tareas básicas: una visión general de la información, un acercamiento, el filtrado de la información y por último una visión detallada de la parte de interés.

En este contexto, la visualización de la información ayuda al usuario a comprender y navegar a través del espacio de dominio y de este modo recuperar las instancias que satisfagan su necesidad de información. Dadas las características de la búsqueda por navegación, la visualización puede ser aplicada a este modelo de búsqueda más fácilmente que en el modelo de búsqueda por consulta. La Web semántica y sus características han permitido adaptar la visualización de la información para que tome ventaja de las propiedades que ofrecen las ontologías. Es aquí donde se habla de *visualización de información basada en ontologías*.

De acuerdo con las metodologías de referencia en la web semántica, los pasos recomendados según Natasha Fridman Noy (2001) para la construcción de una ontología son:

1. Determinar el dominio y ámbito de la ontología.
2. Determinar la intención de uso de la ontología.
3. Reutilizar ontologías o vocabularios controlados existentes.
4. Enumerar los términos importantes del dominio.
5. Definir la jerarquía de clases.
6. Crear las instancias.

Existe un gran interés desde el entorno corporativo, el sector público y el mundo académico por hacer de la web semántica una realidad, ya que se piensa que puede ser una pieza importante para el progreso de la sociedad de la información. Para ello se está invirtiendo un gran esfuerzo en desarrollar a) la infraestructura necesaria para su despliegue, b) aplicaciones que demuestren la viabilidad y el beneficio de la web semántica y motiven el desarrollo y consumo de infraestructura y c) nuevas soluciones para resolver

problemas específicos, e ideas que mejoren, amplíen y/o exploten las posibilidades de la web semántica.

Si bien la mayoría de grupos y empresas que participan en este campo concuerdan en el gran potencial de estas tecnologías, los resultados alcanzados hasta ahora son muy preliminares si se mira desde la óptica más ambiciosa, la de la adopción universal de la web semántica. El desarrollo de infraestructura y tecnología para la construcción de la web semántica está hoy en un punto bastante avanzado, sin embargo el uso que se está haciendo de ella no ha alcanzado aún un grado de desarrollo comparable. El desarrollo de aplicaciones que hagan uso de esta tecnología ha sido identificado por Stefan Haustein y Jörg Pleumann (2002) como una de las realizaciones prioritarias en este punto para el progreso de la web semántica.

2.4 Generación Dinámica de Interfaces a partir del Conocimiento

Las ontologías fueron originalmente desarrolladas en la inteligencia artificial para facilitar el reuso y el compartir del conocimiento [Gruber 1993]. Luego, las ontologías, han sido usadas para la recuperación de conocimiento inteligente en la WWW como un instrumento para modelar información semántica (meta data) usada para anotar documentos web [Fensel 1999]. En esta investigación las ontologías son usadas para proveer máxima flexibilidad en la representación del conocimiento del dominio del proyecto (servicios y usuarios). Más aun, son un elemento esencial para alcanzar la separación entre la presentación y los contenidos [Macías 2001].

A pesar de las ventajas que los modelos semánticos presentan a la hora de ser implementados como recuperadores, herramientas para compartir y reusar la información, uno de los problemas de estos modelos es su presentación legible y comprensible al usuario final. En la fase de diseño y construcción de un modelo se valora su poder expresivo y no se toman en cuenta aspectos visuales o estéticos del mismo. Este también ha sido el caso de las ontologías que se presentan en este trabajo.

Este modulo de visualización muestra instancias de lo ontología mediante formatos generados dinámicamente. Cada clase de instancia se presenta de forma distinta, mostrando de forma selectiva sus datos y relaciones con otras instancias. En lugar de codificar este tratamiento en un programa, debe permitirse definir independientemente la presentación de cada clase de la ontología mediante uno o varios modelos de visualización por clase.

Para lograr lo anterior, se han desarrollado acercamientos en diferentes proyectos, una de las maneras de implementarlo se basa en la existencia de una ontología auxiliar (ontología de visualización) que permite definir vistas sobre el modelo de dominio. Estas vistas se definen de acuerdo con criterios de usabilidad y estéticos con el fin de garantizar una presentación legible del modelo semántico subyacente.

La ontología de visualización es un contenedor de entidades y atributos publicables que extraen los valores mediante lenguajes de consultas sobre RDF. El contenido de una instancia publicable puede agrupar varios conceptos del modelo semántico original, o al contrario, puede dividir un concepto complejo en varias entidades visualizables por separado.

Otro acercamiento para la presentación es definido por la creación de plantillas por cada clase de la ontología. Las plantillas son definidas usando un lenguaje textual basado en Paginas JSP, que permiten al diseñador de la presentación insertar expresiones java entre (<%= y %>) y declaraciones de control entre (<% y %>) en código HTML. Una plantilla define que partes (atributos y relaciones) de un tópico debe ser incluido en la presentación, su apariencia y ubicación. Utilizando reglas de presentación que permiten acotar el número de instancias a mostrar o el tamaño máximo de la información a mostrar de los atributos, por ejemplo, máximo 500 palabras.

En otra investigación [20] se implementó una transformación XSL genérica desde RDF/XML a HTML. Ello produce resultados consistentes siempre que los fragmentos a mostrar y el subgrafo así producido se hayan serializado como

RDF/XML abreviado. La serialización abreviada produce un flujo de XML que mantiene agrupadas todas las tripletas relacionadas.

Por consiguiente, es posible mostrarlas como una serie de tablas HTML, una por cada recurso descrito y con tablas anidadas para las descripciones de los recursos anónimos a los que éstos se refieran. La transformación XSL de RDF a HTML es capaz de tratar, de forma genérica, cualquier conjunto de metadatos RDF que se le proporcione serializado como XML. Además, es fácil de implementar ya que sólo se requiere un procesador de XSL, por lo que la transformación se puede hacer tanto en el navegador del usuario como en el servidor. Además puede ser manejada mediante AJAX para permitir un mayor grado de interactividad en el navegador.

2.5 Sistemas Multimodales

La comunicación entre humanos y máquinas siempre ha sido una materia difícil y tortuosa, probablemente debido a que la insuficiente sofisticación del comportamiento interno de las máquinas ha obligado a sus diseñadores a hacer que *el usuario se adapte a su funcionamiento en vez de ser al revés*.

El concepto de «interacción multimodal» se refiere a la capacidad de emplear varias formas o «modos» de comunicación a la hora de suministrar y obtener información de un servicio. Junto al imparable avance de las capacidades multimedia de los terminales móviles, la interacción multimodal se presenta como una pieza clave para el despliegue de una amplia gama de potentes y atractivos servicios para las redes de comunicaciones móviles.

Las principales líneas de desarrollo para la introducción de la interacción multimodal en los servicios móviles se centran en la integración de dos tendencias hasta el momento disyuntas:

1. El acceso visual a la información, cuyo exponente fundamental hoy por hoy es la navegación web.

2. El acceso vocal a la información, cuyo desarrollo se ha potenciado con la aparición de tecnologías como VoiceXML, que asimilan el desarrollo de servicios vocales a los servicios web.

La integración conseguida en el ámbito del desarrollo y prestación de los servicios se extiende mediante este nuevo concepto a la interacción del usuario, de forma que es posible acceder a los servicios no mediante una interfaz visual pura, sino empleando también las capacidades de entrada y salida de audio que son comunes en casi cualquier tipo de terminal. Así, se permite a los usuarios interactuar con las aplicaciones de múltiples formas:

- Introduciendo información (mediante su voz, usando un teclado, empleando un ratón, a través de un puntero, etc.).
- Recibiendo información (como voz sintetizada, audio grabado, texto plano, gráficos, vídeos, etc.)

Además de mejorar significativamente la capacidad de la interfaz, la multimodalidad produce una mejora importante de las prestaciones en la calidad de los productos basados en reconocimiento (por ejemplo, la combinación del reconocimiento del movimiento de los labios y la voz pueden reducir de forma significativa los errores de un reconocedor en un entorno ruidoso). Todo esto, sin olvidar que los servicios multimodales son esenciales para cubrir la necesidad de rentabilizar las nuevas redes de comunicación móvil 3G y de incrementar el tráfico de datos en las actuales redes móviles.

Ahora bien, la incorporación de nuevas pautas de interacción multimodal en las comunicaciones móviles debe contemplarse como un proceso gradual que vaya aglutinando tanto la integración de nuevas tecnologías, como las necesidades y preferencias de los usuarios. En este sentido, se prevé un progreso continuo desde los mínimos actuales, que permiten a un usuario elegir entre utilizar un modo u otro de forma excluyente (por ejemplo, enviar un SMS o hacer una llamada de voz), pasando por las posibilidades de salto entre modos en una misma sesión (como la consulta durante una llamada de una agenda de direcciones en el móvil), hasta una verdadera libertad en la

combinación y cambio de modo (hablar, teclear, marcar, visualizar, etc.) de los terminales que accedan a canales simultáneos de voz y datos.

Uno de los principales retos en innovación con que se encuentran las Tecnologías de la Información y las Comunicaciones (TIC) es proporcionar vías de acceso simples y naturales a los servicios y aplicaciones de las futuras redes de información construidas sobre la base de las tecnologías del conocimiento. Para conseguir lograr alcanzar el objetivo general anterior es necesario destinar un importante número de esfuerzos en dos líneas de trabajo complementarias, como son:

1. La definición de tecnologías de diseño, creación y publicación de contenido multimedia, para la amplia variedad de redes de comunicación, fija y móvil, y nuevos terminales (PDAs, Tablet PCs, teléfonos móviles de tercera generación, etc.).
2. El desarrollo de tecnologías de interfaces que ofrezcan modos de interacción hombre-máquina simples y naturales, que incluyan estrategias inteligentes de adaptación a los usuarios y acceso natural con capacidad multimodal de respuesta a diferentes estímulos.

Cabe destacar dos aspectos principales que muestran la relevancia del desarrollo de la tecnología de interfaces multimodales:

1. La falta de adecuación de las interfaces visuales actuales a las características de los futuros terminales (teléfonos móviles 3G, PDAs, XDAs, Tablet PCs, etc.) En lo que se refiere a este primer aspecto, las interfaces visuales actuales, basadas en texto y gráficos, son adecuadas cuando la entrada por teclado permite una recogida rápida y eficiente de información, y se dispone de pantallas amplias que permiten presentar una gran cantidad de información visual y que también permiten al usuario poder concentrar su atención en una determinada región de la pantalla.

Sin embargo, las características de los terminales de nueva generación presentan importantes obstáculos a la utilización exclusiva de interfaces visuales, pues incorporan pequeñas pantallas, no poseen teclado o incluyen uno de tamaño reducido y difícil de utilizar, y las condiciones de uso del terminal impiden muchas veces al usuario el empleo de las manos, por ejemplo en los automóviles. Debido a estas limitaciones, se viene dedicando un importante esfuerzo tecnológico al desarrollo de las interfaces vocales. Sin embargo, estas interfaces tienen todavía planteados muchos retos, como se verá en el punto siguiente.

2. La ausencia de alternativas unimodales. En este segundo aspecto, el empleo de la voz como forma de interacción con los nuevos terminales móviles es una de las vías más prometedoras a las que se está destinando un amplio esfuerzo de investigación (tómese como ejemplo los trabajos realizados dentro del proyecto Aurora sobre reconocimiento de habla distribuido). Sin embargo, el empleo exclusivo de la voz no resulta totalmente satisfactorio, mientras que el recurso a una interacción multimodal está comenzando a plantear importantes mejoras por dos motivos principales:

- El primero es la importante merma de las prestaciones de una interfaz vocal en situaciones de ruido (donde la tecnología de reconocimiento de habla no permite garantizar un nivel de funcionamiento satisfactorio), así como en aquellas situaciones donde un usuario puede encontrarse incómodo para utilizar la voz en su interacción con el terminal, por ejemplo en presencia de otras personas.
- El segundo hace referencia a la limitación intrínseca que un solo modo de interacción puede suponer a la hora de presentar al usuario determinados tipos de información (por ejemplo, la dificultad que supone la presentación de una larga lista de nombres o realizar la localización en un plano mediante una interfaz exclusivamente vocal). La multimodalidad, con la inclusión de técnicas visuales, puede

complementar de una forma eficaz aquellas funcionalidades de difícil accesibilidad desde el control por voz.

A los dos aspectos anteriores puede añadirse también el derivado de las consideraciones relacionadas con las tecnologías de seguridad multimodal. En este sentido, la integración de varios modos de autenticación biométrica (voz, huella, firma, cara, etc.) surge como una de las alternativas más prometedoras para superar las importantes limitaciones tecnológicas de cada modo aislado, y alcanzar así los elevados niveles de seguridad exigidos por el uso y acceso de futuros servicios y aplicaciones seguras.

Una vez revisadas las limitaciones, se deben destacar el gran número de consideraciones prácticas que hacen muy atractivo el uso de las interfaces multimodales. Estas consideraciones pueden agruparse en dos niveles diferentes:

- En un primer nivel se encuentra la multimodalidad, que permite disponer de formas más naturales de interacción con las máquinas. Si bien es cierto que la utilización del adjetivo *natural* empieza a quedar algo desvirtuada en la terminología empleada en la tecnología de interfaces, pues debe entenderse en su doble vertiente de:
 - Permitir, por un lado, un uso simultáneo de los diferentes modos de comunicación humana (auditiva, visual, etc.), en función de las condiciones ambientales (por ejemplo, de ruido) o de la simple preferencia del usuario (por ejemplo, para disponer de una mayor intimidad).
 - Proponer, por otro, al usuario un modo simple y eficiente de interacción con los sistemas automáticos. Estudios recientes han demostrado que para la realización de una misma tarea, una interfaz basada en la utilización de un lápiz electrónico y la voz es entre 3,2 y 8,7 veces más rápida que una interfaz visual

tradicional, y también un diez por ciento más rápida que una interfaz vocal unimodal.

Aunque también hay que indicar que cualquier tarea no tiene por qué beneficiarse necesariamente de la multimodalidad. En cualquier caso, se puede prever que las interfaces multimodales que combinen varios modos, tales como voz, gestos, punteros, lápices, pantallas táctiles, etc., darán soporte a nuevas aplicaciones en los terminales móviles (teléfonos móviles, PDAs y ordenadores portátiles) y en los denominados entornos virtuales y de computación ubicua, donde los modos actuales (teclado y ratón) son inapropiados.

- En un segundo nivel se encuentran las técnicas de diseño de interfaces multimodales, que permitirán disponer de mecanismos de interacción más robustos y con mayor capacidad de recuperación de errores que una interfaz unimodal. Nuestro lenguaje es por naturaleza ambiguo, por tanto las tecnologías del lenguaje tales como el reconocimiento del habla, el reconocimiento de la escritura o el reconocimiento de los gestos, podrán ir proporcionando mayores prestaciones en su desarrollo tecnológico, pero nunca serán perfectas.

Resultados recientes de investigación han mostrado que las interfaces multimodales que integran dos técnicas de reconocimiento imperfectas (reconocimiento de voz y de gestos) pueden proporcionar resultados más precisos que el empleo de una sola de ellas, principalmente debido a su capacidad de desambiguación a partir de la información proporcionada por los diferentes modos de interacción. Pero, además, la disponibilidad de varios modos de interacción proporciona mayores posibilidades en la siempre difícil tarea de recuperación de errores.

Las principales iniciativas en el campo de la estandarización de las interfaces multimodales se encuentran en la formalización de los lenguajes de representación de dichas interfaces. El W3C, a través de la *Multimodal*

Interaction Activity, se ha centrado en canalizar múltiples iniciativas de la industria, cuyo objetivo pretende, según su propia definición, «permitir múltiples modos de interacción (auditiva, visual y táctil) que permitan a los usuarios suministrar información empleando su voz o sus manos mediante una pantalla táctil, teclado, ratón o puntero. Para recibir la información los usuarios podrán escuchar voz sintetizada o grabada, y visualizarla mediante una representación gráfica».

Dentro de esta iniciativa compiten dos propuestas alternativas que presentan un alto interés, ya que cuentan con un amplio respaldo industrial e intentan homogeneizar las metodologías de diseño, potenciando la facilidad de configuración y abriendo las puertas a un rápido despliegue de nuevos servicios y aplicaciones.

Estas dos propuestas alternativas son:

- *La propuesta X+V (xHTML+VoiceXML)*. Está promovida por IBM, Opera y Motorola, y propone la representación de los diálogos multimodales a través de la combinación de dos lenguajes preexistentes en la interacción unimodal: XHTML, como estándar para la representación de información visual y VoiceXML, como elemento que define la interacción basada en voz. La integración de ambos modos de interacción se realiza a partir de la especificación XML-Events del W3C.
- *La propuesta SALT*. Está promovida por Microsoft, Cisco, Comverse, Intel, Philips y SpeechWorks, y propone una extensión de HTML, y otros lenguajes de marcado (cHTML, XHTML y WML), que añade una interfaz vocal a las páginas web existentes aprovechando las ventajas del modelo de desarrollo de aplicaciones web.

Las marcas definidas en este lenguaje han sido diseñadas para su empleo tanto para la navegación exclusivamente vocal (por ejemplo, el acceso telefónico) como para la navegación multimodal. SALT se compone de un pequeño conjunto de etiquetas XML, con un modelo de

objetos, propiedades, eventos y métodos asociado, que puede emplearse en conjunción con un documento basado en marcas derivadas de SGML, aplicando de esta forma la funcionalidad vocal al mismo con independencia de su naturaleza.

También se puede destacar otro de los principales retos planteados a las tecnologías de interfaz: la exigencia de una operatividad sobre los entornos hardware y software de los futuros terminales móviles. Durante los últimos años han adquirido gran popularidad los *handheld computers* (PDAs y Pocket PCs), pequeños ordenadores que requieren un nuevo paradigma de interacción hombre-máquina (*Human-Computer Interaction*, HCI), ya que los dispositivos de entrada tradicionales (teclado y ratón) son demasiado grandes para ellos.

Una primera respuesta ha sido lo que se conoce como *pen computing*, es decir, ordenadores cuyo principal mecanismo de entrada es un lápiz con el que se puede escribir y señalar en la pantalla. En la actualidad están surgiendo con fuerza nuevos ordenadores, conocidos como Tablet PC, algo más pequeños que un ordenador portátil tradicional, que suelen incluir un teclado, pero que pueden funcionar también sin él ya que incluyen un lápiz similar a los que utilizan los Pocket PCs y PDAs.

Uno de los principales intereses en la actualidad es el de tener multimodalidad a través de diferentes dispositivos (PDAs, teléfonos, PC) y medios (Html, WAP, SMS, MMS,..), permitiendo una interacción más flexible y adaptada a los gustos y necesidades de los usuarios. Las comunicaciones mediante redes informáticas, y más en concreto a través de Internet, están evolucionando desde lo que fue en un principio entendido como una comunicación entre ordenadores, a una comunicación entre personas. Esto se debe a la entrada cada vez mayor de estas aplicaciones informáticas en nuestra vida diaria, unida a la mayor oferta de servicios y especialmente a la convergencia con otras tecnologías, hasta ahora más cercanas al ser humano, como la telefonía. Con la introducción de los terminales móviles en el acceso a los servicios remotos se abre un enorme rango de nuevos usuarios a Internet, al tratarse de dispositivos muy extendidos en la sociedad actual. Pero también implica un

replanteamiento de los sistemas de adaptación de contenidos, que hasta ahora sólo estaban sometidos a los requerimientos de los ordenadores. Ahora surge una nueva gama de terminales preparados para acceder a los servicios, pero que cuentan con necesidades y capacidades totalmente distintas a las de un PC.

La adaptación de un contenido a un terminal implica dos tareas, o dos tipos de adaptación: una primera para adaptar la funcionalidad del servicio a las características del terminal (adaptación de servicios), y otra para preparar la información a servir de tal modo que su formato se ajuste correctamente a las capacidades de la interfaz de cada dispositivo (adaptación de información). En una pantalla de PC tienen cabida muchos más contenidos, menús y funciones que en la de un terminal móvil. Por ello, la información que se muestra en una sola pantalla de PC se ve dividida en varias pantallas en un móvil. Esto implica un cambio en la navegación del servicio.

Este tipo de adaptación persigue generar distintas interfaces de un mismo servicio o aplicación para conseguir la compatibilidad de la misma con diferentes terminales. La alternativa a este enfoque sería desarrollar una versión del servicio para cada dispositivo, lo cual resultaría muy costoso en cuanto a los recursos necesarios para su implementación y su mantenimiento. Por esta razón, las soluciones que se comentarán en este apartado excluyen este enfoque de replicación, centrándose todas ellas en aproximaciones que consideren un único desarrollo del servicio útil para todos los dispositivos. Para conseguir este objetivo, casi todas las alternativas consideradas comparten el concepto de separar la capa de negocio del servicio de la de presentación, que será la que tenga la lógica necesaria para adaptar la salida. Las soluciones consideradas son las siguientes:

- ***XML-XSL***. La opción más común es desarrollar la capa de presentación dividiendo las “responsabilidades” entre la generación del contenido (en formato XML) y las plantillas de generación de formato (XSL). De este modo el servicio genera el resultado de sus procesos en lenguaje XML, que será transformado en el formato que mejor soporte cada terminal

según la plantilla XSL a aplicar, que ha sido escogida en función del terminal reconocido.

- **Transcoder.** Consiste en superponer una capa de acceso a los servicios que ya han sido desarrollados sin consideraciones de multiacceso. Esta capa haría de interfaz entre el usuario y el servicio, encargándose de adaptar la salida del servicio al tipo de dispositivo que se esté accediendo. El principal inconveniente de este sistema es que resulta complicado transformar la salida, diseñada para un dispositivo concreto, a las características particulares de otros dispositivos.
- **JSP.** Esta opción parte del mismo concepto de separación entre presentación y lógica de negocio en el servicio, pero propone el uso de la tecnología JSP (*Java Server Pages*) para la capa de presentación. Los JSP son páginas Web con trozos de código Java embebidos que las dotan de una mayor potencia y dinamismo. Una página HTML no se ejecuta, sino que se interpreta por el navegador, de tal modo que no es posible producir cambios en la salida de forma dinámica en función de parámetros como el tipo de terminal que esté accediendo al servicio. En cambio, si se el introducen fragmentos de código Java se dota a la página de mayor lógica, de tal modo que puede reaccionar de forma distinta en función de diversos parámetros, como el tipo de *user agent*, tipo de navegador, etc. De este modo, estas páginas dinámicas serán capaces de generar distintos formatos de salida para el servicio en función del terminal.
- **Módulo de Metalenguaje.** Esta opción sigue la filosofía de los anteriores, pero supone un paso más allá: disponer de un módulo especializado en la generación de presentaciones. De este modo, los servicios delegarían la generación de su salida en este módulo. Para ello, estos módulos suelen definir un metalenguaje genérico en el que se basan los servicios para enviar la información, encargándose luego dichos módulos de transformar este metalenguaje según las características de los diferentes terminales.

Hasta ahora la mayor parte de los contenidos sólo debían generarse en un único formato o versión, debido a que el conjunto de dispositivos donde debían ser mostrados era reducido, y con características muy similares. Esta situación ha cambiado, y puede ser necesario que una misma imagen o video tenga que verse tanto desde un PC como desde un terminal móvil.

Las adaptaciones de información más comunes suelen ser:

- Cambios de formato.
- Cambios de tamaño.
- Cambios de calidad: se contemplan cambios como:
 - Reducción de calidad
 - Reducción de colores
 - Reducción de la frecuencia de presentación de imágenes.

2.5.1 Proxy Multimodal. Herramienta de Interacción Multimodal.

Herramienta desarrollada como Tesis de Maestría de la Universidad de los Andes, que permite que los usuarios interactúen con una aplicación Web de manera multimodal, esto es, a través de las tecnologías HTTP, SMS, WAP, MMS y SAT, y desde diversos dispositivos. La idea central es que los recursos se codifiquen en un formato neutro y la herramienta los transforme al formato apropiado para que puedan ser visualizados a través de diferentes medios, y no menos importante, que en una misma sesión el usuario tenga la opción de cambiar la modalidad de acceso, siempre y cuando tenga sentido y sea apropiado.

Para que los desarrolladores puedan utilizar esta herramienta, deben codificar las respuestas de las aplicaciones y páginas Web estáticas en un formato neutro basado en XML, que podría ser diseñado desde cero, o podría ser uno ya existente como XHTML el cual es ampliamente conocido por desarrolladores

y Web masters, y es el formato propuesto por la W3C para la realización de aplicaciones Web multidispositivos.

Esta alternativa consiste de un sistema que intermedia entre los clientes y los servidores Web de origen. El sistema mantiene un directorio de los recursos servidores de origen, y cuando el usuario consulta el sistema intermedio, éste le muestra los diversos recursos de los servidores de origen que puede acceder multimodalmente, luego el usuario selecciona el recurso y accede a éste a través del sistema. Para garantizar la interacción multimodal, el sistema intermedio incorpora un módulo con el soporte para análisis de imágenes.

Al sistema intermedio se le llama servidor Proxy multimodal, aludiendo a la tecnología de los servidores Proxy Web caché, que son una entidad en las redes que realiza las solicitudes HTTP en representación del cliente. Los servidores de origen, como se mencionó previamente, son servidores Web ordinarios, que contienen los recursos (a los que los clientes acceden multimodalmente a través del servidor Proxy multimodal) a los que se desean acceder.

Los participantes en la interacción dentro del portal son:

- El cliente: El cliente liviano o el PC accede al Portal vía WAP o HTTP respectivamente, y el agente de usuario se comunica con el Proxy para que haga la traducción necesaria dependiendo del dispositivo usado.
- El servidor Proxy multimodal: se encarga de recibir el URI del recurso por parte del usuario, estos recursos deben estar en un directorio propio y con base en éste URI, solicita el recurso (formateado en XHTML) al servidor de origen (Agente Usuario) y lo formatea en WML, XHTML, SMS, MMS, SAT, dependiendo del canal de comunicación por el cual se le enviará la respuesta al cliente.
- El servidor de origen: Genera los recursos en formato neutro (XHTML) de manera estática y éstos deben ser transformadas en el formato apropiado por el servidor Proxy multimodal.

– Las interacciones se observan en la figura y se dan en el siguiente modo:

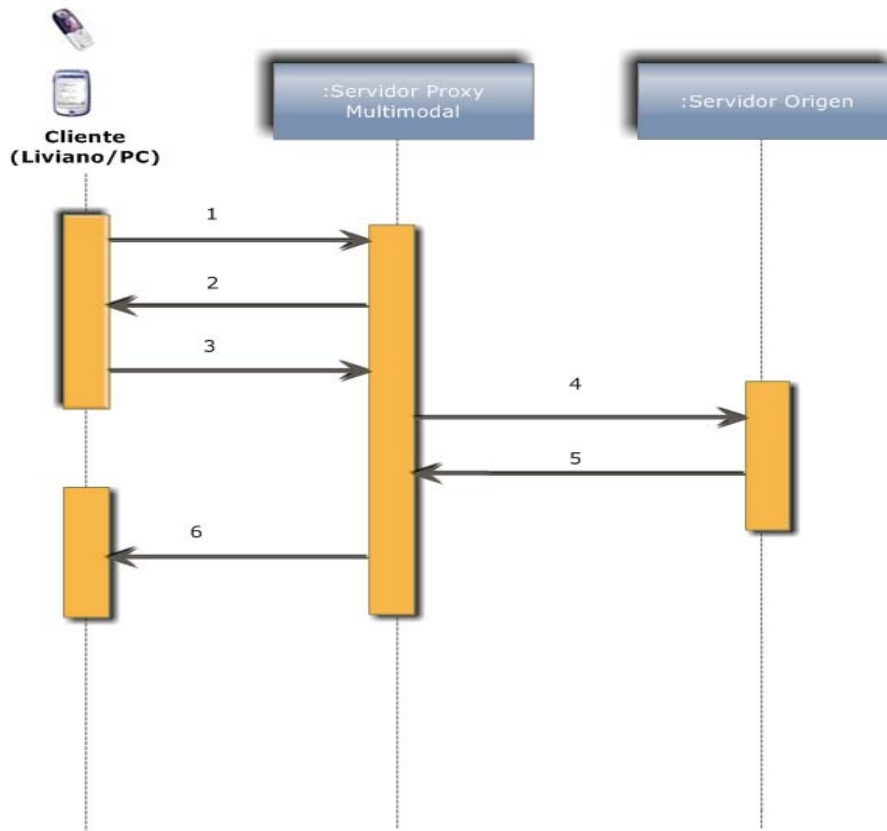


Figura 1. Interacciones a través de WAP o HTTP por el Proxy

1. El cliente solicita al servidor Proxy multimodal la página donde seleccionará el URI del servidor de origen que está suscrito y desea acceder.
2. El servidor Proxy multimodal detecta el tipo de dispositivo y con base en ello le retorna el contenido que se observa en la figura anterior.
3. El cliente le envía al servidor Proxy multimodal, mediante la selección de un enlace, el URI del recurso (en el servidor de origen) que desea acceder.
4. Con base en la información recibida de parte del cliente en el paso 3, el servidor Proxy multimodal envía una solicitud del recurso (seleccionado por el cliente) al servidor de origen.

5. El servidor de origen envía la respuesta al servidor Proxy multimodal (en el formato XHTML).
6. Por último, el servidor Proxy multimodal traduce la respuesta en el formato apropiado, o la retransmite dependiendo del dispositivo que esté atendiendo, y le envía la respuesta traducida.

La ventaja en este método de implementación consiste en que el Web master realiza las aplicaciones Web de manera convencional, únicamente debe codificar los recursos en XHTML. Por último, el uso de un nodo o sistema intermedio, es decir, el servidor Proxy multimodal, para acceder a los recursos de los servidores Web de origen, aumenta la latencia y con esto los tiempos de respuesta son más prolongados que si se pudiera acceder a la aplicación Web directamente, sin embargo, la latencia se podría disminuir significativamente a través de una caché con los recursos que no han sido modificados en los servidores de origen, tal y como lo hace un navegador Web ordinario, o un Proxy Web caché.

2.5.2 Interfaces de Usuario Multimodales

El principal objetivo del desarrollo de interfaces de usuario basado en modelos es la construcción de interfaces mediante descripciones de alto nivel de los distintos aspectos de la interfaz: estructura y comportamiento, de modo que a partir de dichos modelos declarativos se pueda generar automáticamente la interfaz de usuario final.

Al mismo tiempo, este enfoque basado en modelos declarativos proporciona un marco idóneo para la definición de interfaces de usuario independientes de contexto de uso, ya que permite utilizar descripciones de alto nivel de abstracción que pueden modelar aspectos independientes del contexto, o de las restricciones impuestas por un dispositivo o plataforma específicos

La definición de interfaces de usuario utilizando lenguajes declarativos proporciona varias ventajas, como la facilidad de aprendizaje y la posibilidad de

definir la interfaz de forma independiente de la definición de la lógica de la aplicación y el contenido.

De esta manera se permite que diferentes especialistas trabajen independientemente en el desarrollo de la aplicación. Los lenguajes basados en XML se están perfilando como serios candidatos a soportar las especificaciones gracias a la versatilidad de mantenimiento, extensión y capacidades de refinamiento que proporcionan los documentos XML.

La tecnología XML, como estándar de representación común, permite la especificación del modelo de interfaz abstracto, la descripción de las características específicas de los diferentes dispositivos, así como la especificación del proceso de transformación de los objetos de interacción abstractos en objetos de interacción concretos.

En la literatura se encuentran varios lenguajes ligados a *frameworks* de construcción de interfaces de usuario. A continuación se describen brevemente algunos lenguajes como XForms, AUIML, AAIML, UsiXML, UIML y XIML.

UsiXML. El *USer Interface eXtensible Markup Language* (UsiXML) permite la descripción de interfaces de usuario multimodales, con diferentes formas de interacción, en diferentes plataformas y dispositivos. UsiXML describe a un alto nivel de abstracción los elementos de una interfaz, pero no sólo los *widgets* o componentes gráficos, sino también los modos de interacción. Diferentes *renderers* permiten su ejecución en diferentes dispositivos y plataformas.

XForms. XForms es una especificación del W3C enfocada a formularios para una gran variedad de dispositivos, tanto electrónicos como incluso en papel. Xforms propone un modelo de interfaz de usuario que sea rica en contenido, de manera que sea válida tanto para las aplicaciones comerciales como para las pequeñas aplicaciones de usuario personal e individual. Permite una completa separación de la lógica de presentación y de la lógica de datos, así como la internacionalización y la posibilidad de tener múltiples formularios en una página, de la misma manera que múltiples páginas a través de un formulario.

Hay que destacar también que se puede integrar perfectamente con otros entornos XML y trabajar con ellos conjuntamente.

AUIML. El Abstract User Interface Markup Language (AUIML), desarrollado por IBM, es un lenguaje XML para la descripción de interfaces de usuario. El lenguaje permite representar diferentes componentes gráficos, sus propiedades, así como su distribución en paneles y el *layout*. Diferentes *renderers* convierten el documento AUIML a una interfaz para cada plataforma. IBM ha desarrollado también el AUIML Toolkit, que incorpora una herramienta de construcción de una interfaz de usuario (AUIML Visual Builder) y que genera un documento AUIML. El *toolkit* también tiene *renderers* para generar la interfaz en Java Swing o en HTML. También puede crear manejadores de eventos o Java Beans para el manejo de datos. De hecho el AUIML Visual Builder es un plugin de Eclipse, con lo que se unifica la generación de la interfaz abstracta y del código.

AAIML. El InterNational Committee for Information Technology Standards (INCITS) está desarrollando un estándar para un *Universal Remote Console* (URC) en el marco de lo que denominan *Alternate User Interface Access*. La URC ofrecería una interfaz a un gran abanico de dispositivos electrónicos como puedan ser sistemas de seguridad, termostatos, etc. Un elemento central en la definición del URC es el lenguaje *Alternate Abstract Interface MarkupLanguage* (AAIML).

AAIML permite definir de forma abstracta una interfaz de usuario. Cada URC ha de ser capaz de renderizar un documento AAIML según sus propias peculiaridades; AAIML está basado en un modelo de eventos para facilitar la interacción entre la URC y el dispositivo. El lenguaje define una serie de elementos abstractos para identificar las operaciones de entrada y salida, y no directamente los *widgets* como ocurre en otros lenguajes. Cada URC traduce un elemento abstracto a un elemento específico de la plataforma. Por ejemplo, el elemento abstracto *string-selection* se podría convertir en un *radio button* en una interfaz visual, o en una opción mediante voz en otra interfaz basada en el habla.

XIML. El *eXtensible Interface Markup Language* (XIML) tiene como objetivo dar soporte a todo el ciclo de vida del diseño de una interfaz gráfica de usuario. Esta afirmación incluye las fases de “diseño, desarrollo, operaciones sobre la interfaz, manejo de la misma, organización de los diversos componentes y la evaluación de la misma” [4]. XIML divide los componentes que pueden ser utilizados para la generación de interfaces en diversos grupos, separados según su especialización.

Hay tipos especializados como Tareas (*business processes*), Dominios (define una jerarquía de componentes dentro de un dominio), Usuario (define una jerarquía de usuarios finales), Presentación y Diálogos (definen acciones que se relacionan directamente con la interfaz). Todos estos componentes son proyectados a elementos que no son más que representaciones concretas de los *widgets* o componentes gráficos que se utilizarán para la representación de la interfaz.

UIML. (User Interface Markup Language) es un lenguaje XML para definir de forma abstracta interfaces de usuario de manera independiente de la plataforma. Básicamente, UIML describe una interfaz en tres niveles: presentación, contenido y lógica. La presentación se refiere a la apariencia de la interfaz; el contenido se refiere a los componentes de la interfaz; la lógica se refiere a la interacción usuario – interfaz (eventos del ratón, teclado).

En UIML, una interfaz de usuario es una jerarquía de elementos XML. Cada uno de los componentes de una interfaz (botones, cajas de texto, etiquetas) son una entidad XML. UIML utiliza un vocabulario abstracto. Los elementos XML de UIML son genéricos; por ejemplo, no aparecen elementos XML con la terminología menú, ventana, etc. UIML permite la descripción de interfaces de diferentes tipos. Con UIML se pueden describir tanto interfaces web como de escritorio. Un ejemplo de la variedad de interfaces son las que nuestro IDE permite crear.

2.6 Sistemas Basados y Sensibles en Contexto

El modelado de un usuario tiene sus comienzos a finales de la década del 70' cuando Allen, Cohen y Perrault presentan los primeros trabajos basados en esta idea (p. E., Perrault. 1978; Cohen y Perrault, 1979; Allen, 1979). En estos comienzos en el modelado de usuario, el modelado se realizaba en la aplicación misma y no existía una clara distinción entre los componentes que administraban la información del usuario y los componentes que realizaban otras tareas.

A partir de mediados de los años ochenta, tal separación se hizo cada vez más visible (p. e., Kobsa, 1985; Sleeman, 1985; Kass, 1988; Allgayer, 1989), pero no se puso énfasis en la modularización de los componentes de modelado de usuario para permitir que sean reutilizables en nuevos sistemas "adaptables al usuario".

Ya para fines de los 90' el valor de la personalización en la Web es altamente reconocido en el área del comercio electrónico (Hof et al., 1998; Allen et al., 1998; Cooperstein et al., 1999; Hagen et al., 1999).

La característica central de los sistemas de modelado de usuario a partir de ese momento fue la modalidad servidor, en la cual estos no se encuentran integrados en las aplicaciones mismas sino que las sirven a través de comunicación entre procesos. De esta manera pueden proveer información a más de una aplicación de manera centralizada.

La personalización, interés fundamental de la comunidad de modelado de usuario, se refiere a métodos para ganar algún conocimiento acerca de los usuarios, representados como un modelo de usuario, y explotando ese modelo para ajustar el comportamiento de los sistemas a las necesidades de los individuos. Gracias al aporte de la Web Semántica los modelos de usuario pueden disponer hoy de estándares bien definidos y ontologías, para proveer extensibilidad, flexibilidad, interoperabilidad y reusabilidad.

Modelo del usuario

Un modelo es una representación simplificada de una parte de la realidad. Si el objetivo es que las aplicaciones puedan interactuar de una manera más personalizada con los usuarios, éstas deben contar con información acerca de los usuarios para usarla como fuente en la toma de decisiones. Cuanto mayor y más depurada es la información que contiene el modelo del usuario, mejor y más ajustada es la interacción del sistema que lo utiliza con el usuario.

Un modelo de usuario mantenido en un servidor al que puedan acceder diferentes “clientes” para consultas y actualizaciones constantes es un elemento fundamental para poder centrar en el usuario las aplicaciones distribuidas en la Web. El modelo del usuario podría actualizarse y consultarse por diferentes mecanismos, y de esta manera crece el conocimiento que se tiene del usuario.

La necesidad de conseguir entornos software que se adapten cada día mejor al usuario y su perfil teniendo en cuenta, al mismo tiempo, el entorno que los rodea nos lleva a los sistemas sensibles al contexto. Dichos sistemas manejan toda la información relevante que rodea y forma parte de sus usuarios. Las preferencias de un usuario, sus gustos, su localización física, estado de ánimo, actividad, el entorno que le rodea, la temperatura de la sala, el nivel de luminosidad, etc, toda esta información es lo que llamamos Información Contextual.

Un contexto, por tanto, puede almacenar información sobre las capacidades de comunicación, almacenamiento y procesamiento del PDA del usuario, o puede contener datos sobre el modo favorito del usuario de que se le muestre información.

La tendencia actual, dada la cada vez más profusa distribución de dispositivos pequeños y portátiles tiende a la distribución en torno a sistemas heterogéneos, formados por una gran cantidad de redes de diversas características, y un conjunto de usuarios que disponen de dispositivos de muy diversa índole, tales

como PDAs, portátiles, ordenadores de sobremesa e incluso otros dispositivos tales como sensores, cámaras o electrodomésticos inteligentes.

En estos escenarios surgen diversos retos para la construcción de un sistema contextual. Primeramente debemos garantizar que la información contextual sea tratada de forma homogénea en un formato que nos permita transmitir las características semánticas propias del contexto. Seguidamente debemos tener en cuenta la componente dinámica de la información en el sentido en que está cambiando continuamente y tanto los cambios en la información como los avisos a aquellos interesados en esos cambios se deben gestionar eficientemente. Por último, se debe tener en cuenta la necesidad de garantizar la disponibilidad de la información contextual en todo momento.

El problema de la representación del contexto es abordado actualmente como un tópico abierto y un marco de investigación novedoso. De modo general, y basándose en el contexto, se han dedicado esfuerzos a definir tanto el concepto de contexto, sistemas sensible al contexto e incluso el ciclo de vida que deben seguir los sistemas sensibles al contexto como los mecanismos y métodos de captura u obtención de la información contextual. Brown y Finney coinciden en afirmar que el uso del contexto está íntimamente relacionado con el modo en como lo percibimos, y que se requiere una correcta interpretación del mismo para que resulte de utilidad.

Esta observación nos lleva al uso de ontologías para una definición común, con semántica, del modelo de información que han de usar todas las entidades implicadas en la aplicación. El uso de ontologías garantiza el uso de un marco común en el que se pueden intercambiar sin esfuerzo informaciones del contexto usando un modelo común, reusar modelos de contexto previamente realizados, interpretación semántica de la información en forma de conocimiento y permitir razonamiento sobre el contexto para integrar entidades de información de bajo nivel en otras entidades más abstractas.

2.7 OSGI

Día a día surgen nuevos y variados estándares con el fin de integrar, mejorar y volver popular el uso de nuevas tecnologías. La alianza Open Service Gateway Initiative está conformada por más de 50 empresas líderes de sectores de las comunicaciones, desarrollo de software y fabricantes de dispositivos móviles, que buscan más que todo la popularización de las nuevas tecnologías móviles.

En marzo de 2003 la alianza OSGI publicó la especificación de una plataforma abierta de servicios para la prestación y administración de múltiples aplicaciones para todo tipo de dispositivos orientados a conexión en hogares, vehículos, y ambientes móviles; agregando a su vez la ventaja de administrar el ciclo de vida de estas aplicaciones desde cualquier parte de la red.

OSGI genera una guía técnica que permite describir y proponer diferentes arquitecturas de software para aplicaciones orientadas a dispositivos móviles e inalámbricos. Esta especificación permite el diseño y construcción de una plataforma de servicios la cual contiene componentes capaces de proporcionar múltiples servicios a distintos dispositivos, plataformas de hardware y ambientes de ejecución.

Por esta razón la compatibilidad es una de las principales ventajas, la plataforma de servicios OSGI al estar desarrollada sobre la tecnología Java puede estar en cualquier ambiente de ejecución propio de Java permitiendo que se puedan desarrollar aplicaciones para cualquier dispositivo que disponga de una máquina virtual de Java como teléfonos celulares, PDA, sistemas de navegación automotrices, etc.

Otra ventaja que ofrece OSGI es una infraestructura, debido a que fue creada partiendo de una arquitectura estándar para ser utilizada por cualquier implementación, y orientada a componentes, brinda la libertad de trabajar sobre cada componente desde cualquier lugar de la red. Además cada componente se puede instalar, actualizar o eliminar sin afectar la ejecución del sistema (sin necesidad de reiniciar o detener la ejecución de la JVM), como también

permitiendo que estos componentes puedan describir dinámicamente otros componentes que ofrezcan algún tipo de servicio.

Trabajar sobre esta especificación permite un complemento y un punto de integración de varios proyectos de investigación de ingeniería de sistemas realizados en el contexto académico para brindar conocimiento y apoyo a nuevos proyectos, haciéndola un Framework deseable dentro de la arquitectura del portal.

3 SISTEMA DE COMUNICACIÓN INTERMÓDULOS PARA UN PORTAL DE SERVICIOS PARA SISTEMAS MÓVILES SENSIBLE AL CONTEXTO. CASO UNIVERSIDAD DE LOS ANDES.

3.1 Introducción

En este capítulo se describe el estado actual de las tecnologías que forman el núcleo de esta tesis, a saber, ingeniería Ontológica, Agentes Inteligentes en específico el de usuario, sistemas basados en usuarios, servicios y contextos.

En los sistemas actuales, que mantienen una interacción con el usuario, se empieza a presentar una inquietud, sobre qué usuario lo utiliza y cómo, a través de esta interacción, se puede aprender más sobre él, la importancia en las interfaces ha empezado a reconocerse de tal manera que muchos estudios ya se encuentran en este campo. El trabajo de un agente es el de actuar autónomamente para satisfacer deseos que el usuario tiene, dar una mayor sensación de productividad y reducir el trabajo del usuario.

Para este trabajo es importante la congruencia de varios tipos de sistemas; como el centrado en el usuario y su contexto, dado que es un sistema basado en los servicios que ofrece el portal, hay que abarcar este tipo de sistemas, en la investigación fue reconocido otro tipo de tecnologías como el uso de ontologías para aprovechar el conocimiento que el sistema crea para poder hacer inferencias que permitan una interacción con el usuario más confiable y acercada a la realidad, con el ánimo de aprovechar esta tecnología se hace necesario a la vez el estudio de cómo esta se utiliza a la hora de interactuar directamente con el usuario bajo la presencia de una interfaz. El objetivo es seguir la tendencia de los sistemas actuales que pretenden lograr que la interacción con sus usuarios sea constructiva y más intuitiva.

3.2 Descripción Del Sistema

El Portal de Servicios para sistemas móviles sensible al contexto se basa en la posibilidad de la interacción de un usuario, estático o en movimiento, con espacios físicos. Estos espacios físicos, denominados Espacios Activos, EA, se encuentran representados de forma lógica por los Portales, los cuales congregan una serie de servicios ofrecidos por diferentes entidades (Unidades de Espacio Activo, UEA) a los usuarios. Estos espacios físicos, EA's, pueden ser centros comerciales, parques, universidades, edificios, bibliotecas, etc. Para el caso en particular, el espacio activo es la Universidad de los Andes.

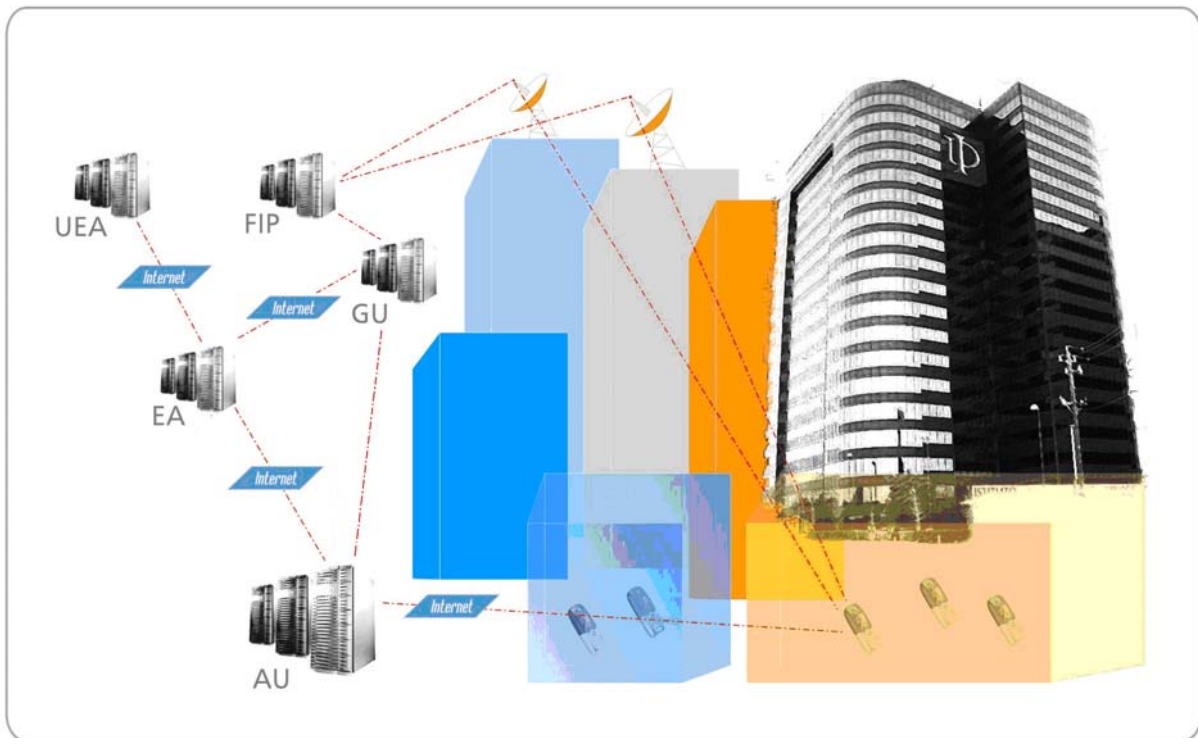


Figura 3. Modelo Portal de Servicios para Sistemas Móviles.

En la Figura 3 se describe el modelo del Portal de Servicios, el cual está conformado por un conjunto de componentes que tienen la función de gestionar todo lo relacionado con la interacción Espacio Activo, (EA), - Usuario.

El Agente de Usuario (AU) es el representante del usuario ante el sistema. Las Unidades de Espacio Activo (UEA), las cuales representan unidades de negocio o componentes del Espacio Activo, EA físico (Almacenes, secciones de bibliotecas, facultades, etc.), son las que de manera general ofrecen servicios a ser utilizados por los usuarios. Estos servicios son agrupados y ofrecidos por el Espacio Activo, EA, al Agente de Usuario para ser entregados finalmente al usuario final. Existen otros componentes, entre estos el Gestor de Usuarios (GU) y la Fuente de Información Posicional (FIP), en los cuales el Espacio Activo, EA, se apoya para lograr sus objetivos de venta de información de acuerdo con los perfiles de sus usuarios.

Según [Dey,2004], un sistema es sensible a contexto si utiliza el contexto para proveer información y/o servicios relevantes al usuario, donde la relevancia depende de la tarea del usuario. La conceptualización de un modelo de representación de un ambiente debe brindar por tanto elementos que permitan proveer servicios apropiados para cada situación particular de usuario.

En el modelo del Portal todo se fundamenta en la definición de un área geográfica restringida y claramente delimitada denominada **Espacio Activo**. Para el caso en particular el Espacio Activo es la universidad. La caracterización de un espacio activo debe permitir determinar un conjunto de lugares geográficos o **Espacios Físicos** que lo conforman así como una serie de **Servicios** representativos que allí se pueden brindar.

Servicio es toda aquella oferta de información o atención que puede estar especialmente dirigida a usuarios que en un momento dado (en su contexto actual) desempeñan una determinada **Actividad**¹ o tienen ciertos **Intereses**.

En el mundo real, la prestación de los servicios es llevada a cabo por **Objetos**. Los objetos pueden ser **Personas** o pueden ser **Virtuales**. Un objeto virtual es aquella figura prestadora de servicios diferente a una persona. Cabe anotar en

¹ Ejemplos: Estar almorzando, estar en clase, estar caminando, estar en busca de dinero, en busca de un PC disponible.

este punto que las personas siempre tienen asociada una **Ubicación** física y su posición es variable dada su característica de movilidad.

Para que un servicio pueda llegar a ser presentado a un usuario es necesario configurar una **Oferta de Servicio** que enlace todos los elementos requeridos. En ella se especifica fundamentalmente un servicio “base”, un objeto “prestador” y un espacio físico a través del cual se brinda. Adicionalmente, involucra una característica o restricción **Horaria**² de acceso al servicio y el **Rol** o roles a los cuales está enfocada la Oferta de Servicio.

Un elemento muy importante es el cómo vincular toda esta oferta de servicios con las necesidades de información de un usuario. Para esta propuesta el **Contexto**³ es el elemento que almacena toda la información biodemográfica, situacional, navegacional y social del usuario.

3.3 Arquitectura Del Sistema

La arquitectura planteada para el Portal de Servicios para Sistemas Móviles sensitivo al contexto se encuentra basada en los siguientes criterios:

- La arquitectura es una abstracción del mundo real y refleja la conformación de los Espacios físicos.
- Flexibilidad en la relación Espacio Activo - Unidad de Espacio Activo. Lo anterior mediante la publicación de los servicios por parte de las Unidades de Espacio Activo, lo cual permite que cada Unidad retire u ofrezca nuevos servicios dinámicamente sin afectar la disponibilidad del portal.
- Flexibilidad en la ubicación lógica y física de cada uno de los componentes del sistema.

² Dicha restricción horaria caracteriza el nivel de servicio (Ej. Horario de Atención).

³ Almacena una serie de variables referidas a objetos.

- Definición de las interfaces claras para la comunicación entre los componentes.
- Aislamiento entre la forma de comunicación y lo comunicado con el objeto de no generar limitantes y disminuir al máximo la evolución del sistema para enriquecer la semántica y sintaxis en el lenguaje de comunicación Usuario – Sistema.

En el marco de esta tesis se hizo un rediseño del portal tomando consideraciones para cada uno de los componentes que lo conforman, y permitiendo hacer un diseño más detallado dada la ventaja de que cada componente estaba trabajando en paralelo el desarrollo del mismo, el EA, el GU, el FIP y el AU como componentes hicieron parte de este diseño en común. A continuación, se muestra la arquitectura final establecida en ese trabajo de diseño:

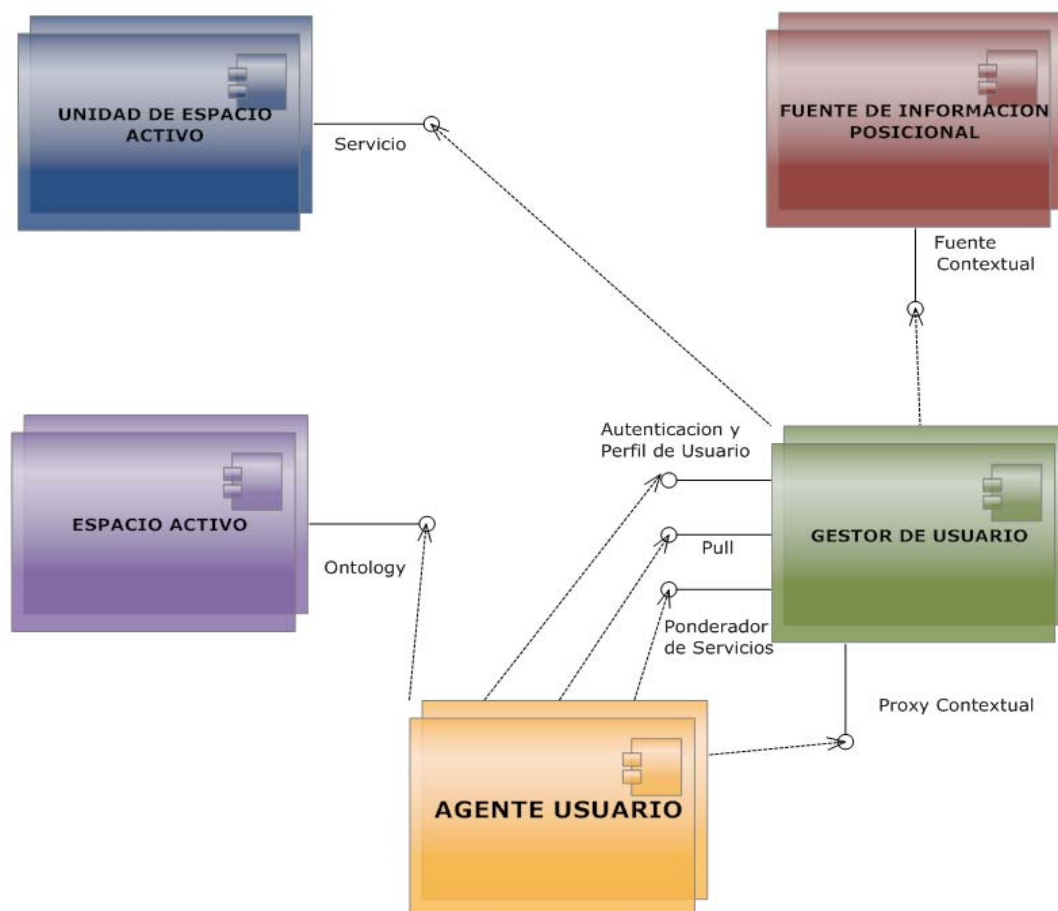


Figura 4. Arquitectura de Sistema.

Posteriormente se detalla cada uno de los componentes:

Agente Usuario: representa al usuario frente al sistema, se encarga de interactuar con el EA para solicitar el conocimiento del portal a nivel de servicios y usuario, y con el GU para solicitar los servicios a nombre del usuario o para recibir de este las ofertas que sean necesarias. Además maneja las políticas de seguridad del usuario.

Gestor de Usuarios: es el componente que se encarga de inferir el contexto del usuario, a partir de su comportamiento y de las condiciones iniciales que este plantea acerca de sus intereses, actividades y situación biodemográfica. Determina qué servicios ofrecer al usuario. Trabaja de la mano con el agente usuario para generar el contexto y lograr hacer visible al usuario el mismo de la mejor manera.

Las Unidades de Espacio Activo: (una o más) son entidades físicas o virtuales que ofrecen alguna clase de servicio, ofrecen sus servicios a través del Espacio Activo. Son la fuente de información y servicios con la cual se alimenta el Portal, y conforman la Unidad lógica de Sistema de Información. Estas Unidades de Espacio Activo, no necesariamente deben encontrarse físicamente en un Espacio Activo particular, incluso pueden prestar sus servicios e información a varios de ellos.

Espacio Activo: El espacio activo es la figura que controla y administra todos los elementos que se desenvuelven dentro de él y que tienen la característica de estar inmersos en un componente de tiempo común. De igual forma, es responsable por registrar las condiciones del sistema (Ej. condiciones climáticas, fecha, hora) que afectan a los espacios físicos que contiene y tiene conocimiento de todos los servicios que se ofrecen en él. Dirige las peticiones que realicen los usuarios a la Unidad de Espacio Activo respectiva.

Fuente Información Posicional: con el objeto de determinar la posición del usuario para determinar parte de su contexto ubicuo, este componente se

encarga de hacer seguimiento constante a los usuarios, e ir informando al GU para que pueda construir dinámicamente el contexto de los mismos.

Como se comentó anteriormente, toda la arquitectura del portal sigue el Framework OSGI, con el objetivo de aprovechar el potencial que brinda para aplicaciones móviles, la capacidad de administrar a cada uno de los componentes de manera transparente y la ventaja de permitir comunicación transparente entre ellos.

La interacción ya ha sido mencionada y visualizada en la figura 4, pero seguidamente se detallará en el diagrama de secuencia con el objetivo de aclarar y puntualizar los formatos de comunicación y los mensajes entre componentes, ver anexo.

3.3.1 Servicios ofrecidos por el Portal

En general se pueden distinguir dos tipos diferentes de servicios dependiendo si existe una interacción del usuario para la entrega de la información: Los servicios *Pull* entregan información directamente solicitada por los usuarios. Los servicios *Push* entregan información aun cuando esta no ha sido directamente solicitada por el usuario. Los servicios Push son activados por un evento, el cual puede ser generado si se ingresa a un área específica o se activa un *timer*.

En el siguiente cuadro se presenta un comparativo entre estos.

	<i>Descripción</i>	<i>Ejemplo</i>
<i>Pull</i>	El servicio es entregado cuando el cliente hala la información de los servicios. Cuando el cliente inicia la solicitud, su información de localización es agregada a la solicitud, como el contexto para definir las mejores respuestas.	Restaurantes más cercanos.
<i>Push</i>	La solicitud no es iniciada por el cliente sino por el proveedor del servicio de acuerdo con las preferencias de usuario al rastrear la posición de los usuarios móviles.	Información promocional en un centro comercial.

Tabla 1. Principales tipos de Servicios. Adaptado de [Chen, 2005b]

3.3.1.1 PULL. Consulta del Usuario.

Existen fundamentalmente tres mecanismos de interacción de un usuario con el sistema: Consulta de servicios sensible a contexto, consulta de servicios por criterios manuales y consulta de la totalidad de servicios (Catálogo) ordenadas por el contexto. La primera, para cada contexto usuario es necesario identificar qué subconjunto de ofertas de servicio de la totalidad existente es pertinente para el usuario. La segunda, un usuario siempre tiene la opción de consultar las ofertas de servicio de manera manual, omitiendo los valores de su contexto.

El objetivo de esta consulta es obtener todas las ofertas de servicio a partir de criterios de búsquedas definidos manualmente. Una búsqueda manual se puede hacer a partir de las siguientes aproximaciones:

- Servicios ofrecidos por un objeto
- Servicios que cubren un interés.
- Servicios ofrecidos en un espacio físico.

Y por último, la consulta del catálogo de servicios que permite ver la totalidad de los que presta el portal, pero organizados por el contexto del usuario, es decir mostrando los que se consideren de mas interés de primero.

3.3.1.2 PUSH. Promoción de Ofertas de Servicios.

Hay que tener en cuenta que cuando se especifica un servicio, se define hacia qué intereses/actividades está enfocado; de manera similar, cuando se configura una Oferta de Servicio, se indica hacia qué rol o roles está dirigido el servicio base. De tal forma, mediante esta característica, se busca contar con un mecanismo promocional que permita destacar la oferta⁴, para así motivar a que el usuario desee obtener detalles de la misma. De manera complementaria, se considera una característica atractiva el poder focalizar en zonas geográficas y período de tiempos la promoción.

De tal manera, se promociona la oferta de servicio aplicando la configuración de actividades, intereses, roles, tiempo y cobertura geográfica a los que está enfocado, asegurando que los ofrecimientos lleguen únicamente al público objetivo.

El mecanismo de operación se fundamenta en una especificación de oferta de servicio apoyado en un sistema de notificación promocional al usuario final, en el cual el cliente, puede observar un texto motivando el consumo de la oferta de servicio.

⁴ Mediante la presentación de un texto *banner* promocional en los clientes.

4 MODULO AGENTE USUARIO INTERFAZ

4.1 Introducción

Usualmente se usa la palabra “Agente” para describir personas que tienen una relación de asistente o colaborador para nosotros, por ejemplo, agentes de viajes, secretarias. El trabajo del agente es actuar autónomamente para satisfacer metas que podamos tener, dando una sensación de productividad y reducción de trabajo.

En esta tesis, la palabra agente será usada para describir un software que juega un rol similar, proveyendo ayuda, aconsejando y siendo un mensajero para el usuario.

Tal como con un agente humano, la naturaleza de la relación entre el agente y el cliente es destinada al éxito. Así con agentes de computador, la interfaz de usuario es a menudo la clave. La imagen de un agente o el sistema orientado por agentes, interpretando un papel de colaboración como un mayordomo, secretaria, o la organización de servicios es más viva cuando aquel papel de ayuda es reflejado directamente en la interacción con el usuario

Para permitir una interacción eficaz hombre-máquina, el sistema de interfaz debe ser capaz de encontrar condiciones que varían y determinar un contexto a la persona que representa, con el fin de conocerlo mejor. Así, incorporar facilidades de adaptación en el sistema de agentes se hace esencial.

Los agentes de software inteligentes son una tecnología base para la construcción de aplicaciones flexibles sensibles al contexto. Además de los conceptos de la programación orientada a objetos, los agentes de software también incorporan aspectos de inteligencia artificial y sistemas de comunicaciones, proporcionando un acercamiento que satisface exigencias complejas.

4.2 Descripción Del Modulo.

El objetivo principal del componente es representar al usuario ante el sistema, permitir la construcción del contexto del mismo y brindar una interacción personalizada basada en el conocimiento que el sistema tiene del usuario. Se puede decir que las características de este agente son: Adaptación a las preferencias del usuario e interacción multimodal.

4.2.1 Funciones

Las siguientes son las principales funciones de Agente de usuario:

- ♦ *Asignación de usuario:* Establece el usuario que representará el componente ante el sistema.
- ♦ *Establecimiento del estado del usuario:* Establece el estado del usuario en el sistema. Activo o no activo.
- ♦ *Modificación de la información del usuario:* Establece nuevos valores a la información que almacena el componente del usuario.
- ♦ Ayudar al Gestor de Usuario a definir el contexto en el cual se encuentra el usuario, es decir permitir mantener actualizada la información sobre las actividades que realiza, además mantiene los eventos o servicios que existen en el espacio activo en que se encuentre ubicado el usuario (si es el caso).
- ♦ Notificar al Espacio Activo (si el usuario lo permite) sobre la presencia del usuario en su espacio físico y de esta manera obtener los servicios disponibles.
- ♦ Dar a conocer al usuario los servicios que ofrece el Espacios Activo, la manera en que se da a conocer es esencial ya que los servicios siempre se mostrarán de acuerdo con el conocimiento que el portal tenga del usuario.

- ♦ Gestionar la obtención de la información solicitada por el usuario a partir de la interacción del agente con los distintos componentes que conforman el sistema.
- ♦ Gestionar los requerimientos de acciones o información de eventos ante los componentes pertinentes del sistema.
- ♦ Administrar el perfil de privacidad establecido por el usuario.
- ♦ Suministrar la información que el Espacio Activo solicite sobre el usuario, teniendo en cuenta las políticas o configuración que el usuario ha realizado para el manejo e intercambio de su información personal.
- ♦ Permitir un acceso al sistema por parte del usuario independiente del dispositivo móvil. Soportando los canales HTTP y WAP, dejando la estructura para motivar si es necesario el intercambio por mensajes de texto (SMS), mensajes multimedia (MMS), SAT, VOICE, sin que los recursos tengan que recodificarse en múltiples formatos, en el transcurso de esta tesis se desarrolla otra que permite la interacción por voz.
- ♦ Generación de una interfaz basada en el conocimiento que el sistema tenga del usuario.

4.3 Modelo Del Modulo

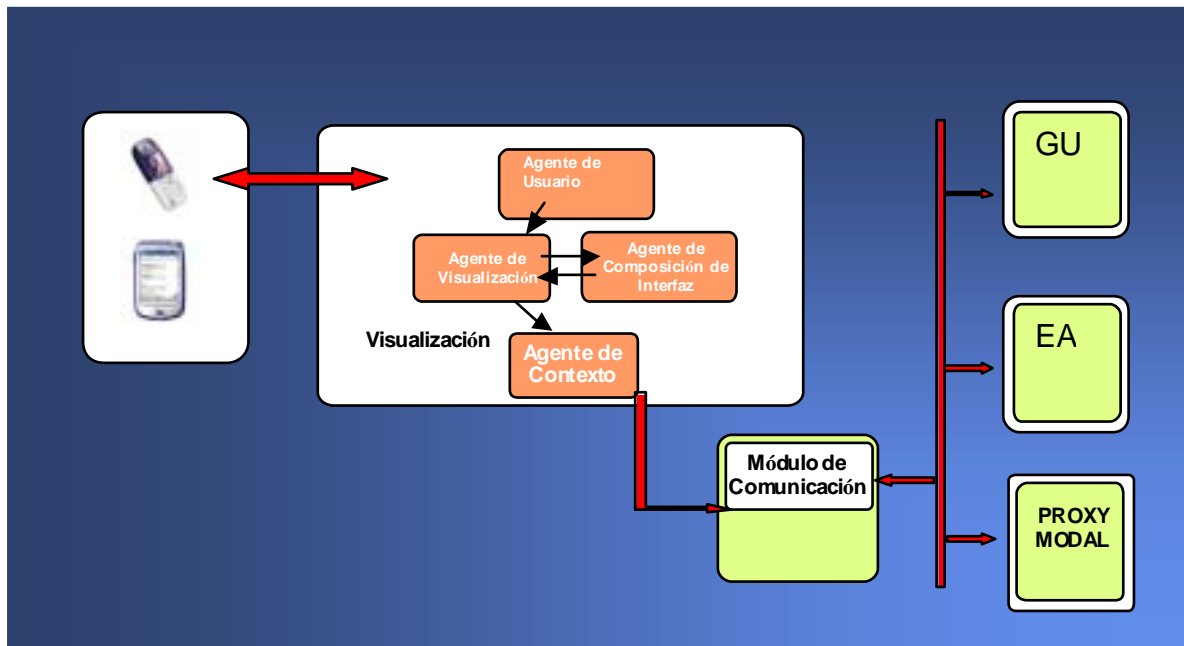


Figura 5. Modelo Agente de Usuario para Portal Uniandes.

Como se puede observar en la figura anterior se diseñó un modelo de sistema multiagentes, cuidando de diseñar un módulo de comunicación con otros módulos, que lograra transparencia y facilidad a la hora de solicitar servicios de otros módulos.

En cierto modo un sistema multiagentes es un sistema en el cual los nodos o elementos son sistemas de inteligencia artificial, o bien un sistema distribuido donde la conducta combinada de dichos elementos produce un resultado en conjunto *inteligente*.

Como casi todo enfoque informático para la solución de problemas, los *sistemas multiagentes* proponen ayudas metodológicas de ingeniería de software, en este caso metodologías de *ingeniería del software orientada a agentes*. Es decir, artefactos de desarrollo que son específicamente concebidos para crear sistemas basados en agentes.

4.4 Arquitectura Del Modulo

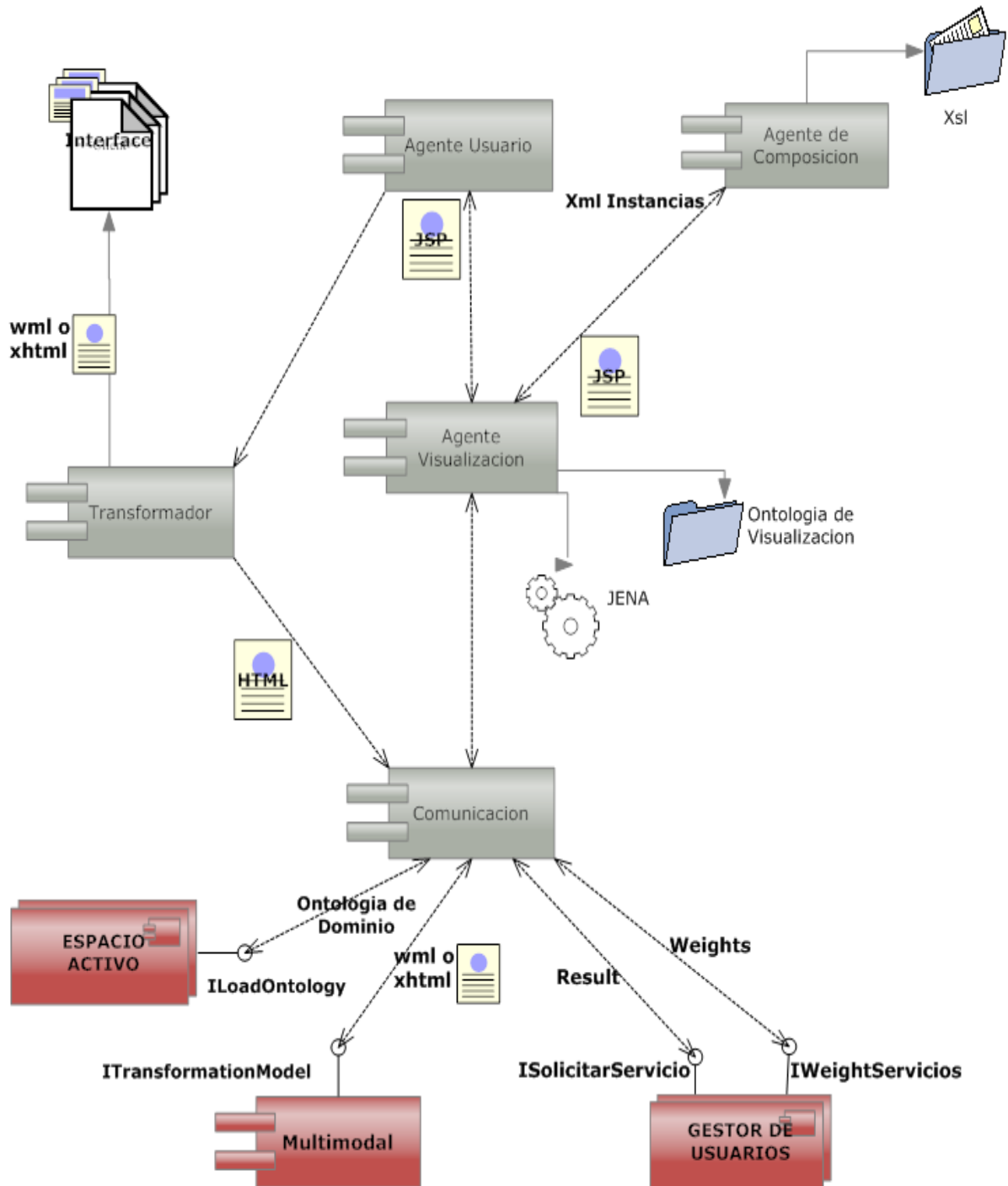


Figura 6. Arquitectura del sistema

Cuando un usuario ingresa al sistema, pide ser autenticado al componente agente usuario, el cual se comunica con el módulo autenticador del gestor de usuario que determina si es o no un usuario válido. Si el usuario es válido, el manejador de agentes de usuario le asigna un agente de usuario que será su delegado desde ese momento en adelante frente al sistema, el agente de usuario solicita al espacio activo las respectivas ontologías de servicio y usuario con el objetivo de presentar al usuario de manera contextual la información.

Frente a peticiones del usuario, el agente de usuario solicita al agente de visualización que determine a partir de la consulta del usuario cuáles son las instancias de la ontología de dominio a ser mostradas al usuario, a su vez, el agente de visualización le solicita al agente de composición que mediante una transformación convierta las instancias ontológicas en formato web (Html).

Cada vez que el usuario mantiene una interacción con el portal el agente usuario procede a pedir al servicio multimodal que haga la conversión necesaria dependiendo del dispositivo con el que el usuario mantenga la interacción, para el caso de esta tesis en formato web o wap. En alguno de los casos es necesario que el agente de visualización, se comuniquen con el gestor de usuario para ponderar los servicios y poder mostrar los servicios contextualizados.

Si por el contrario, es una propuesta de información el GU se comunica con el manejador de agentes para informarle a qué agentes de usuario le corresponde entregar la propuesta.

La comunicación con el GU y el EA se hace a partir de un módulo comunicador, que se encarga bajo la plataforma de OSGI de solicitar la referencia al servicio y acceder a los servicios pertinentes.

4.5 Definición de los Componentes de la Arquitectura

4.5.1 Multimodal.

Esta herramienta comentada anteriormente necesitó ser adaptada en el marco de esta tesis debido a que en el diseño del portal no se contemplan páginas web estáticas, sino que se construyen a medida que el usuario solicita recursos y a partir de su contexto; la facilidad de acceso al código fuente del Proxy Multimodal, permitió adaptarlo para las necesidades propias del portal, con el objetivo de aprovechar su funcionalidad y potencial, entre los cambios se encuentran, la posibilidad de que el Proxy pase de ser un intermediario a un modulo que ofrece el servicio de transformación de formatos, esto con el objetivo de no tener que mantener los recursos en esta aplicación, ya que los recursos no son estáticos, ni fijos, además, con el ánimo de permitir una interacción de páginas un poco más complejas, se adaptó la transformación de xhtml a wml para que permitiera el uso de más controles, controles más complejos, como checkbox, selects entre otros.

Es decir, la idea es que el “Proxy” pasa a ser un “servicio OSGI”⁵, en este escenario el navegador de un PC o el micro navegador de un dispositivo liviano accede a la aplicación Web vía HTTP o WAP respectivamente y la aplicación Web invoca un método de un objeto del servicio, pasándole por parámetro el contenido de la solicitud, el parámetro *tipocanal* y el campo *user-agent*, el cual realizará la transformación y retorna la respuesta generada en formato XHTML si la solicitud fue hecha por un navegador ordinario, o en WML si la solicitud proviene del micro navegador de un dispositivo liviano.

La descripción de la interacción WAP o HTTP se visualiza en la figura 2 y la explicación es la siguiente:

⁵ Open Service Gateway Initiative, genera una guía técnica que permite describir y proponer diferentes arquitecturas de software para aplicaciones orientadas a dispositivos móviles e inalámbricos.

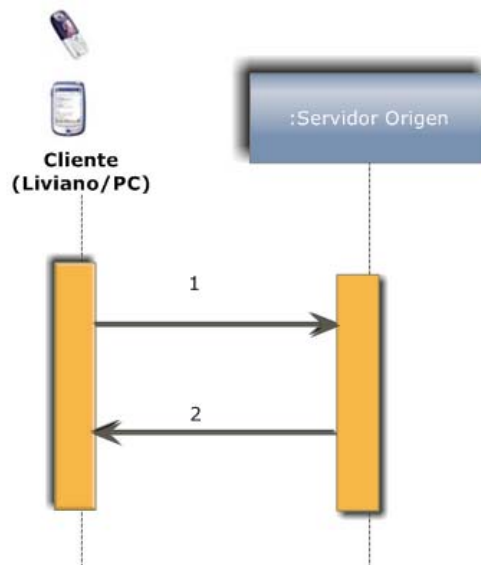


Figura 7. Interacción WAP o HTTP

1. El dispositivo cliente le realiza una solicitud a la aplicación Web, a través de WAP o HTTP, y ésta procesa la solicitud para la identificación del canal por el cual recibió la solicitud, luego, invoca el método del servicio que realice la traducción de la respuesta de la aplicación Web a WML o XHTML, dependiendo del tipo de canal.
2. Finalmente, la aplicación responde a la solicitud del cliente con lo que devuelve el método.

4.5.2 Conocimiento. Manejador de Ontologías

El manejador de ontologías es un módulo fundamental en el componente, ya que es el que se encarga de administrar y modificar si es necesario las ontologías, para ser accedidas desde cualquier módulo del componente.

4.5.2.1 Características del Módulo.

Tiene asociado un agente de contexto que se encarga de realizar la comunicación con el Espacio Activo con el objetivo de obtener las respectivas ontologías de domino y contexto que permitirán mediante su tratamiento

generar las debidas vistas que serán presentadas al usuario. El agente deberá estar disponible para aceptar cualquier cambio que se presente en los contextos, de servicio y usuario.

La comunicación con el Espacio Activo como con el Gestor de usuario se hace bajo la plataforma del Framework OSGI, sobre la cual corre el portal por ende el componente agente usuario, con el objetivo de una comunicación transparente entre los módulos, el Framework mantiene un contexto global de las aplicaciones o componentes y si un módulo desea comunicarse con otro solicita la referencia del mismo y accede a los servicios que este hizo públicos, de esta manera por ejemplo, logra acceder a las ontologías que mantiene el espacio activo y a la hora de mostrar el catálogo de servicios por ejemplo solicitar al gestor de usuario los pesos de los servicios.

4.5.2.2 Funciones del módulo

Las principales funciones del módulo son:

- Obtener las ontologías de dominio y contexto, administradas por el espacio activo.
- Modificar la ontología de dominio con el objetivo de ponderar los servicios y organizarlos con respecto a los gustos del usuario.
- Actualizar la ontología de contexto a partir de modificaciones que el usuario haga de su perfil y enviar los cambios al gestor de usuario para que a su vez actualice la información estática relacionada con el usuario.

4.5.3 Composición. Generador de Documentos Web.

El módulo composición permite a partir de una transformación obtener un objeto en formato deseado (Html) a partir de un xml, con una hoja de estilo xsl.

4.5.3.1 Características del Módulo.

Es evidente que la selección del lenguaje en el cual se definirían las interfaces dinámicas sería un factor muy importante tanto para la definición de las interfaces como para la implementación del compositor. Dicho lenguaje debería permitir que las interfaces dinámicas contaran con características como:

- Ser interpretadas por los micronavegadores.
- Permitir un diseño de interfaces con un alto grado de escalabilidad, es decir, en el caso de que una nueva herramienta o servicio se agregara al espacio personal, el lenguaje seleccionado debía proporcionar las facilidades para que sólo se agregaran las nuevas especificaciones al diseño original.

Al principio se consideraron dos lenguajes:

- UIML
- XML

Para elegir entre UIML y XML se tomaron en cuenta sus ventajas y desventajas. La principal ventaja de ambos es que, por ser lenguajes de marcado, favorecen la generación de interfaces portátiles, estos dos lenguajes en particular permiten crear interfaces genéricas que pueden convertirse a diferentes lenguajes.

En el caso de UIML, la conversión es automática; en el caso de XML la transformación la realiza un procesador XSLT. Para crear una interfaz en UIML, el desarrollador debe conocer los lenguajes de conversión (WML, VoiceXML, Java o HTML), lo que demanda asociar cada componente de la interfaz con su correspondiente componente en el lenguaje destino; con XML esto no es necesario, pues el mapeo de cada componente lo hace el procesador XSLT tomando como referencia la hoja de estilos XSL, en UIML este mapeo debe especificarse en cada interfaz; con XML las interfaces no deben modificarse: únicamente debe crearse una hoja de estilo XSL para cada lenguaje.

Hasta el momento UIML sólo contempla cuatro lenguajes de conversión: en cambio, utilizando las tecnologías XSLT un archivo XML puede transformarse en otro documento XML como WML, XHTML o VoiceXML, y también a cualquier otro formato como JSP, PDF O RDF.

Las últimas ventajas de XML sobre UIML determinaron su elección. Además en la literatura se refleja el poco conocimiento que hay sobre UIML en comparación con XML. Por lo tanto, XML y sus tecnologías asociadas fueron la base para la definición de las interfaces y el compositor.

Según lo anterior, este módulo contiene un constructor de transformadores que procesa el objeto a transformar (para este caso un documento xml; generado a partir de las instancias de la ontología de dominio) a partir de un objeto plantilla (documento xsl) y devuelve el respectivo transformador que permite generar el objeto resultante en el formato deseado (documento xhtml).

Lo que consiguen las hojas de estilo es separar la información (almacenada en un documento XML) de su presentación, usando en cada caso las transformaciones que sean necesarias para que el contenido aparezca de la forma más adecuada en el cliente. Es más, se pueden usar diferentes hojas de estilo, o incluso la misma, para presentar la información de diferentes maneras dependiendo de los deseos o de las condiciones del usuario.

La importancia de este módulo es la capacidad que le da al agente usuario de independizar la creación de las vistas del conocimiento que tiene el portal (Ontologías) y de facilitar la generación dinámica de las mismas.

4.5.3.2 Funciones del módulo

Las principales funciones del modulo son:

- A partir de la plantilla generar el transformador correspondiente.

- Con el transformador creado y el xml recibido generar el formato deseado, documento xhtml.

4.5.4 Visualización. Generador de Vistas dinámicas a Partir de Conocimiento.

Este módulo se encarga de determinar cuáles son las instancias de la ontología de dominio que serán presentados a los usuarios. Además, de engranar al agente de contexto, las vistas a generar y el agente de composición.

4.5.4.1 Características del Módulo.

El agente de usuario, luego de recibir las dos ontologías, la de dominio y contexto, debe interactuar con los agentes de visualización y de composición de vistas, dependiendo de la interacción del usuario con el portal.

En un primer paso, el agente de usuario le solicita al agente de visualización que defina qué información asociada con las ontologías (la lista de conceptos y sus relaciones, al igual que la lista de las instancias asociadas a tales conceptos), se mostrarán al usuario. Para esto, el agente de usuario le envía al agente de visualización la vista solicitada y si es necesario o la vista lo requiere parámetros para el correcto desarrollo del proceso.

Para llevar a cabo su tarea, el agente de visualización tiene asociada una ontología de visualización, que es un contenedor de entidades y atributos publicables que extraen los valores mediante lenguajes de consultas sobre RDF. Estas entidades publicables contienen aquellos atributos de la ontología de dominio que se van a presentar al usuario final.

El contenido de una instancia publicable puede agrupar varios conceptos del modelo de dominio original, o al contrario, puede dividir un concepto complejo en varias entidades visualizables por separado. Por ejemplo, el catalogo de

servicios, es una vista que en la ontología de visualización reúne a dos clases de la ontología de dominio, las categorías de servicio y a los servicios con el objetivo de mostrar un menú desplegable.

Es así como, de manera dinámica, se extrae de la ontología de visualización la entidad solicitada por el usuario (vista). Más concretamente, la ontología de visualización incluye un concepto predefinido:

- **Entidad de publicación:** concepto que encapsula objetos de la ontología de dominio tal como se verán publicados. Todo concepto definido en la ontología de dominio (Contexto o Servicios) heredará de él y deberá definir los siguientes atributos:
 - **Hoja de estilo XSL:** asociada al concepto o conceptos agrupados en la vista, que traduce sus instancias a HTML; “XSL (Extensible Stylesheet Language, lenguaje extensible de hojas de estilo) es una familia de lenguajes basados en el estándar XML que permite describir cómo la información contenida en un documento XML cualquiera debe ser transformada o formateada para su presentación en un medio”⁶.
 - **Consulta RDQL:** atributo que contiene la respectiva consulta a realizar en la ontología de dominio que permitirá obtener todas las instancias correspondientes que cumplan con la condición asociada con la vista. RDQL permite extraer información de los grafos RDF que mantiene la ontología.
 - **Tipo de vista:** atributo que permite identificar si la vista es simple o compuesta; es decir, si la vista necesita una transformación directa o más bien se necesita una transformación por partes, es útil para situaciones en que una

⁶ Tomado de <http://es.wikipedia.org/wiki/XSL>

clase de la ontología necesita ser subdividida y cada parte ser mostrada de manera o diferente, o cuando diferentes instancias de una misma clase quieren observarse de maneras distintas.

- **Has Slots:** para el caso en que la vista sea compleja deben asociarse las partes que la compongan. Este atributo contiene el nombre del slot.
 - **Has Plantilla:** para el caso en que la vista es compleja debe haber un archivo plantilla HTML, al que se le agregarán los widgets creados a partir de los slots.
- **Slot de publicación:** concepto que representa una entidad o un atributo de una clase de la ontología de dominio tal como se verán publicados. Cada parte deberá definir los siguientes atributos:
- **Has Position:** orden en el que aparecerá dentro de la plantilla el slot, permite estructurar la vista, ubicando cada widget en el orden querido.
 - **Widget Tag Html:** este atributo permite asociar qué tipo de control se presentará, combobox, textbox, checkbox, password, radiobutton, etc.
 - **Consulta RDQL:** atributo que contiene la respectiva consulta a realizar en la ontología de dominio que permitirá obtener todas las instancias correspondientes que cumplan con la condición asociada con la vista. RDQL permite extraer información de los grafos RDF que mantiene la ontología.

A partir de los atributos anteriormente mencionados, se realiza la respectiva consulta *rdql* en la ontología de visualización para determinar los atributos

necesarios de la vista, como si es simple o compuesta, determinado qué tipo de vista es la solicitada, se hace una transformación directa o para el caso de una vista compleja se construye cada uno de los controles y se agregan a la plantilla asociada, para esto se hacen a la vez las consultas rdql a la ontología de dominio, a partir de la cual se solicita que el resultado sea retornado en formato xml, Con esta información, el agente de visualización procede a solicitarle al de composición de interfaces que construya la interfaz en la que se mostrará al usuario la información correspondiente.

El módulo de visualización genera a partir de las instancias de la ontología y con ayuda del agente de composición, una página Web, esta página puede o no incluir elementos como scripts de cliente o de servidor, dado la manera en como se construyen las páginas dinámicamente esos scripts de servidor necesitan ser interpretados para generar el xhtml traducido, el cual se envía al Servicio Multimodal para que lo transforme al formato necesario para la correcta interacción modal del usuario.

La separación entre la ontología para la representación del conocimiento del dominio y la ontología de visualización facilita la independencia de estas dos labores. Otra ventaja de la aproximación seguida es que para ambas se pueden utilizar las mismas herramientas de manejo de lenguajes de ontologías, como por ejemplo Protégé.

4.5.4.2 Funciones del módulo

Las principales funciones del módulo son:

- Extraer de la ontología de visualización los atributos asociados con la vista solicitada, la consulta rdql y la hoja de estilo.
- A partir de la ontología de visualización y de la solicitud del usuario extraer las instancias que cumplieran con la consulta rdql asociada, con el objetivo de generar la vista correspondiente.

- Servir de engranaje entre las ontologías de dominio y de contexto (Agente de contexto), la composición de la vista (Agente de Composición) y el solicitante (Agente de Usuario).

4.5.4.3 Ontologías

Como se ha mencionado anteriormente el uso de ontologías es esencial en el desarrollo del portal, en la arquitectura del mismo participan una serie de ontologías, como son la de contexto, la de servicios y la de visualización. En el marco de esta tesis se participó con la creación de la ontología de visualización y de servicios, con el objetivo de seguir la metodología propuesta para la creación de las ontologías, se detallarán los pasos seguidos.

1. Determinar el dominio y ámbito de la ontología.

Aunque aparentemente la selección del dominio sería una cuestión obvia, el campo de las interfaces aborda cualquier tipo de temas y mas sin están contextualizadas, etc. Es decir, la ontología maneja las interfaces asociadas con el portal permitiendo comunicar las ontologías de dominio y al usuario.

2. Determinar la intención de uso de la ontología.

El reflejo conceptual en la ontología se limita a la definición de vistas sobre otras ontologías de dominio que permitan representar las clases e instancias de otras bases de conocimiento y lograr llevarlas del entendimiento de las maquinas al entendimiento de los usuarios.

En cuanto a la intención de uso, tras un análisis detenido, se llega a la conclusión que la gestión de las vistas que constituirán la interacción con el portal, todos los procesos que se manejen a nivel de usuario son manejados a partir de esta ontología.

3. Reutilizar ontologías o vocabularios controlados existentes.

La primera constatación, a la hora de considerar la reutilización de ontologías y estándares en este campo, es que no se ha publicado hasta la fecha una ontología de visualización propiamente dicha. En este sentido este trabajo representa una contribución al crecimiento de la web semántica y las colecciones de ontologías de dominio público.

4. Enumerar los términos importantes del dominio.

A la hora de pensar en la manera en como la interfaz manipularía la información de las ontologías de dominio se lograba ver la dificultad en mantener la información y generar de manera dinámica y fácil la interacción con los usuarios. En términos generales al ser una ontología sencilla, el término principal es entidad de publicación, que como toda publicación mantiene un tema a tratar en este caso una clase de otra ontología de dominio.

5. Definir la jerarquía de clases.

Las clases de las ontologías de dominio de las que se quieran generar vistas deben heredar de la entidad de publicación y así poder definir los atributos propios de esta clase.

6. Crear las instancias.

A medida que se avanza en la interacción se generan los documentos xsl, las consultas rdql a las ontologías de dominio, se genera la instancia de la clase de la ontología de dominio que hereda de la entidad de publicación para habilitar la vista para el usuario.

La implementación de esta ontología se realizó mediante la herramienta Protégé, utilizando owl como lenguaje, el cual se optó para guardar en un

archivo de texto en vez de una base de datos. El acceso a la base de conocimiento desde los módulos de búsqueda y visualización se lleva a cabo mediante la librería Jena para RDF desde la aplicación.

4.5.4.3.1 Ontología de Visualización.

Como se ha mencionado en apartados anteriores es esencial el uso de una ontología auxiliar que permita llevar a las ontologías de dominio a un entendimiento a nivel de usuario más que un entendimiento de las máquinas, en el apartado 4.6.3.1 se explicó cada una de las clases, atributos y relaciones que la conformaban. Es importante resaltar el significado de esta ontología por que es la pieza crucial que permite un engranaje entre los componentes del agente usuario. Y permite dar un paso adelante en el uso que hasta ahora se le ha dado a la web semántica.

A continuación se muestra el grafo de la ontología, a pesar de ser un grafo sencillo su potencial es significativo, además de permitir incluir más elementos a la hora que se quiera hacer vistas más complejas.

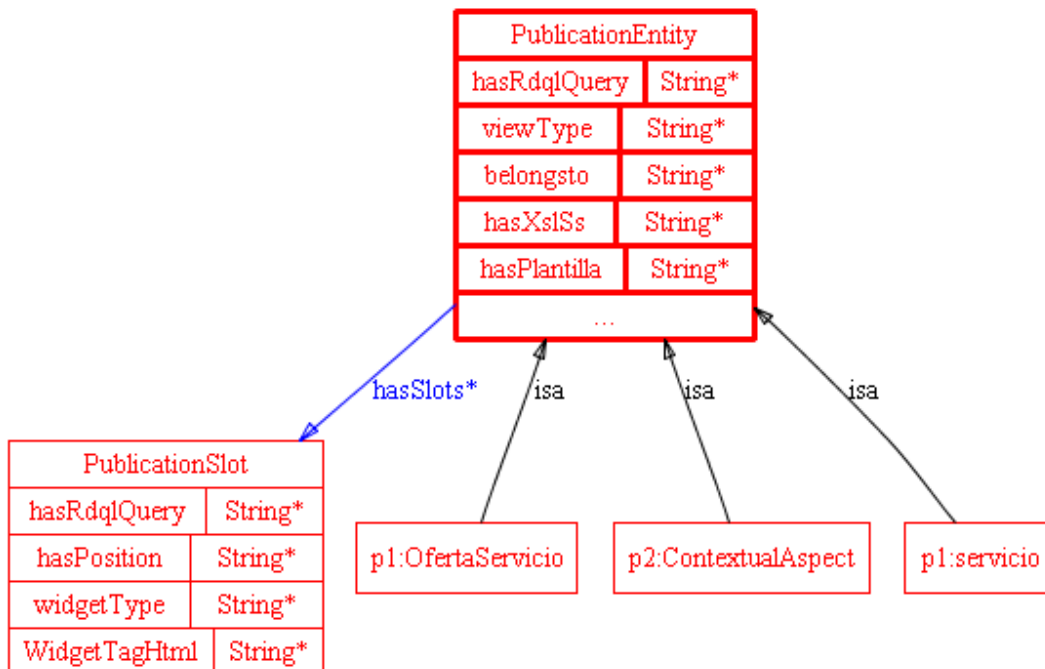


Figura 8. Grafo Ontología de Visualización

4.5.4.3.2 Ontología de Servicios.

En el apartado 3.3 se mencionó el modelo de servicios utilizado para el portal contextual, este modelo surge de la tesis *Especificación de un modelo de Sistema Sensible a Contexto (SSC) y desarrollo de prototipo. Caso Universidad de los Andes*, Desarrollada por el ingeniero *Gabriel Barrera*, contempla los elementos necesarios para permitir un amplio número de combinaciones que permiten ofrecer muchas ofertas de servicio. Los elementos principales de la ontología son los **Servicios** representativos que allí se pueden brindar, dirigidos a usuarios que en un momento dado (en su contexto actual) desempeñan una determinada **Actividad** o tienen ciertos **Intereses**. La prestación de los servicios es llevada a cabo por **Objetos**. Los objetos pueden ser **Personas** o pueden ser **Virtuales**. Cabe anotar que las personas siempre tienen asociada una **Ubicación** física y su posición es variable dada su característica de movilidad. Para que un servicio pueda llegar a ser presentado a un usuario es necesario configurar una **Oferta de Servicio** que enlace todos los elementos requeridos. En ella se especifica fundamentalmente un servicio “base”, un objeto “prestador” y un espacio físico a través del cual se brinda. Adicionalmente, involucra una característica o restricción **Horaria** de acceso al servicio y el **Rol** o roles a los cuales está enfocada la Oferta de Servicio. Para esta propuesta el **Contexto** es el elemento que almacena toda la información biodemográfica, situacional, navegacional y social del usuario. De esta parte se encarga el GU que mantiene una ontología de aspectos, con la que maneja el contexto de los usuarios y servicios.

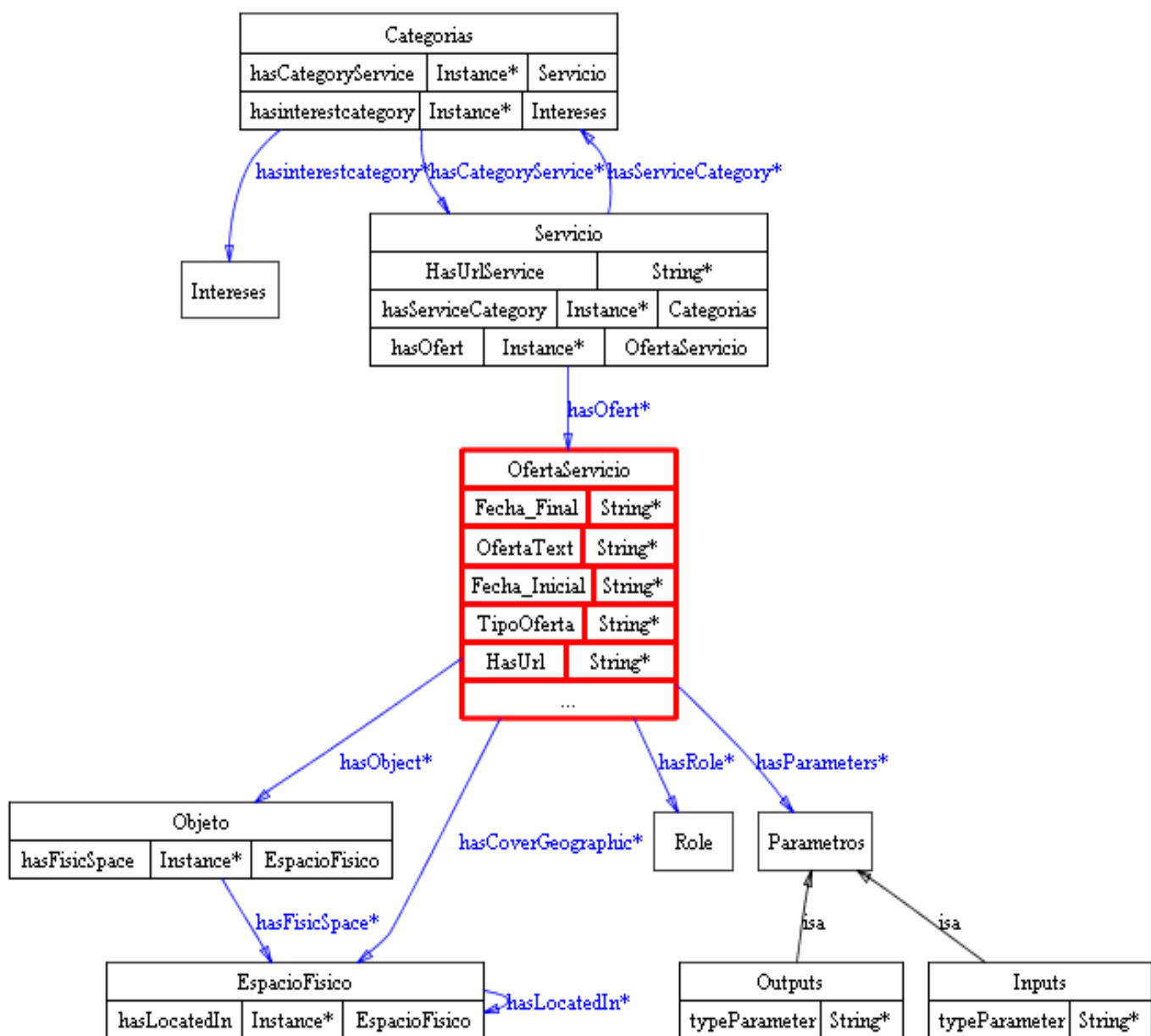


Figura 9. Grafo Ontología de Servicio

5 IMPLEMENTACIÓN MODULO GENTE USUARIO INTERFAZ

5.1 Lenguaje de Implementación

Hoy en día el avance computacional y diversidad de necesidades que se desea suplir, ha permitido el surgimiento de un sin número de lenguajes de programación, de bajo, medio y alto nivel, según el paradigma pueden ser procedimental, funcional, lógico u orientado a objetos, dependiendo de las características de las aplicaciones a desarrollar entre otras cosas, se decide escoger uno u otro.

Para el caso del portal se decidió que todo estuviera desarrollado en Java, analizando las características que brinda; *Java* ofrece toda la funcionalidad de un lenguaje potente, pero sin las características menos usadas y más confusas de éstos, su paradigma es orientado a objetos, es decir, *Java* trabaja con sus datos como objetos y con interfaces a esos objetos. Soporta las tres características propias del paradigma de la orientación a objetos: encapsulación, herencia y polimorfismo. Proporciona las librerías y herramientas para que los programas puedan ser distribuidos, es decir, que se corran en varias máquinas, interactuando. *Java* es robusto, realiza verificaciones en busca de problemas tanto en tiempo de compilación como en tiempo de ejecución.

La comprobación de tipos en Java ayuda a detectar errores, lo antes posible, en el ciclo de desarrollo. Maneja la memoria para eliminar las preocupaciones por parte del programador de la liberación o corrupción de memoria. Una de las principales características por las que Java se ha hecho muy famoso es que es un lenguaje independiente de la plataforma. Es portable, Más allá de la portabilidad básica por ser de arquitectura independiente, Java implementa otros estándares de portabilidad para facilitar el desarrollo, aunque, Java es más lento que otros lenguajes de programación, como C++, ya que debe ser interpretado y no ejecutado como sucede en cualquier programa tradicional, sus evoluciones han ido mejorando notablemente esta desventaja, hasta llegar al punto de ser competitivo. Al ser multithreaded, Java permite muchas

actividades simultáneas en un programa. Todo lo anterior, permite el desarrollo rápido de aplicaciones portable, efectivas eficientes y económicas.

Para el caso particular, además, la utilización de OSGI como Framework nos permite extender las ventajas que java como lenguaje de programación ofrece.

5.2 Descripción de la Implementación realizada

Como se mencionó el lenguaje de programación escogido fue Java para lo cual se utilizó Eclipse como plataforma de desarrollo, Eclipse es una plataforma de desarrollo open source basada en Java. En sí mismo Eclipse es un marco y un conjunto de servicios para construir un entorno de desarrollo a partir de componentes conectados (plug-in⁷). Bajo esa misma filosofía de componentes como se comentó, también, se utiliza el Framework OSGI, Eclipse nos permite extenderlo para la implementación de la especificación R4 de este Framework con un proyecto denominado *Equinox OSGi*, que básicamente es un sistema de plugins que permite implementar una aplicación o conjunto de plugins que usan servicios comunes e infraestructura, para facilidad de desarrollo se utilizó Eclipse Europa que trae incluido Equinox.

Para el caso del Portal Uniandes cada uno de los componentes que conforman su arquitectura están implementados como un plugin (EA, GU, EA, FIP) los cuales ponen a disposición entre ellos de los servicios necesarios para su correcto funcionamiento. De esta manera, se facilita la comunicación y se deja al contenedor que la maneje y administre.

Para la implementación del Agente Usuario y con el ánimo de trabajar bajo la plataforma, se implementó un espacio activo y gestor usuario Dummy que permitiera implementar la comunicación del AU de manera real con los otros

⁷ Plug-ins son paquetes estructurados de códigos y/o datos que contribuyen a la función de un sistema. Las funciones pueden contribuir en forma de librerías (clases Java classes con APIs [Interfaz de Programación de Aplicaciones] publicas), plataformas de extensión, hasta documentación. Plug-ins pueden definir puntos de extensión, sitios bien definidos donde otros Plug-ins pueden añadir funcionalidad."

componentes. De esta manera, el resultado final es apto para adaptarse al desarrollo completo de los otros componentes de manera transparente.

5.3 Servicios Implementados

Dentro de las funcionalidades del Agente Usuario se encuentran el brindar las funcionalidades Pull y Push del Portal de servicios Uniandes, a continuación se muestra cómo de manera interactiva dos usuarios a partir de la misma interacción ven dos interfaces diferentes adaptadas a su contexto:

PULL. Véase apartado 3.3.1.1

Catálogo de Servicios

Cada usuario solicita el catálogo total de los servicios, el agente de usuario lo muestra ajustándolo a la ponderación de los servicios que el gestor de usuario hizo dependiendo del contexto del usuario.

Universidad de los Andes

[Mis Preferencias](#) [Mis Servicios](#) [Catalogo de Servicios](#)

- [Deportivos](#)
 - [Organizacion Partido](#)
- [Academicos](#)
 - [Inscripción Asignaturas](#)
 - [Atencion de Estudiantes](#)
 - [Consulta de Libros Bibliotecas](#)
 - [Consulta Catalogo Biblioteca](#)
- [Culturales](#)
 - [Teatro](#)
- [Restaurantes](#)
 - [Cafeteria](#)
 - [Comidas Rapidas](#)
- [Profesionales](#)
 - [Ofertas Laborales](#)

El usuario *uno* entra y puede ver las categorías de servicios ordenados por su contexto, al igual que los servicios que agrupan cada categoría.

Universidad de los Andes

[Mis Preferencias](#) [Mis Servicios](#) [Catalogo de Servicios](#)

- [Deportivos](#)
 - [Organizacion Partido](#)
- [Restaurantes](#)
 - [Cafeteria](#)
 - [Comidas Rapidas](#)
- [Academicos](#)
 - [Atencion de Estudiantes](#)
 - [Inscripción Asignaturas](#)
 - [Consulta Catalogo Biblioteca](#)
 - [Consulta de Libros Bibliotecas](#)
- [Culturales](#)
 - [Teatro](#)
- [Profesionales](#)
 - [Ofertas Laborales](#)

Como en el caso anterior para un usuario *dos*, el catálogo de servicios es

totalmente distinto, el orden de las categorías, así como los servicios que agrupan tienen un orden que se ajusta a su contexto.

Mis Servicios

En un caso muy parecido al anterior cada usuario solicita solo los servicios que hacen parte de su contexto, el agente de usuario los muestra igualmente ajustándolo a la ponderación de los servicios que el gestor de usuario hizo dependiendo del contexto del usuario, es decir se podan del árbol ontológico los servicios cuya ponderación es 0 por no tener coincidencias con el contexto.



El usuario *uno* entra y puede ver las categorías de servicios ordenados por su contexto, al igual que los servicios que agrupan cada categoría, pero solamente los que son parte de su contexto.

The screenshot displays the top section of the Universidad de los Andes portal. At the top left is the university's logo and name. Below this is a horizontal bar with a black top section and an orange bottom section. Three main navigation categories are listed: 'Mis Preferencias', 'Mis Servicios', and 'Catalogo de Servicios'. Under 'Mis Preferencias', there are sub-menus for 'Deportivos' and 'Restaurantes'. Under 'Mis Servicios', there are sub-menus for 'Academicos' and 'Culturales'. Under 'Catalogo de Servicios', there are sub-menus for 'Organizacion Partido', 'Cafeteria', 'Comidas Rapidas', 'Atencion de Estudiantes', 'Inscripción Asignaturas', 'Consulta Catalogo Biblioteca', and 'Consulta de Libros Bibliotecas'. The text 'Para un usuario dos, el mismo caso.' is located below the screenshot.

PUSH. Véase apartado 3.3.1.2

CONCLUSIONES Y TRABAJO FUTURO

Como parte de este trabajo se planea para un trabajo futuro la integración real con los otros módulos que componen el portal Contextual Universidad de los Andes, para un comportamiento total, se estima que esta integración debería ser muy natural, ya que se tuvo mucho cuidado de usar un Framework como OSGI para la interacción entre módulos, que lo hace transparente y para el caso solo se necesitaría si hay modificaciones, cambiar la interfaz del servicio y el método donde se solicita su referencia en el módulo que lo solicite.

Otro de los trabajos futuros y de mucho interés para la evolución y mayor funcionalidad de este módulo es la extensión del Proxy para que pueda permitir la interacción por voz, de manera paralela a este proyecto se desarrolló la tesis “Arquitectura de un modulo de interacción por voz para el Portal de Servicios Uniandes”.

En la actualidad los productos software tienen la capacidad de ofrecernos información, entretenernos y facilitar nuestro trabajo, pero también pueden entorpecerlo si la interfaz que nos ofrecen es limitada o difícil de utilizar. En esta tesis se ha pretendido mostrar entonces cómo los estudios se han ido agregando a lo largo de estos años para contribuir en conjunto a crear una experiencia satisfactoria para el usuario, aumentando el grado de inteligencia, sofisticación y usabilidad de las mismas. Para ilustrar las exigencias del desarrollo actual de interfaces inteligentes, en esta tesis se ha presentado también una arquitectura que permite el desarrollo de las interfaces capaces de tener en cuenta al usuario, a las tareas que tiene que realizar, el entorno en el que el usuario trabaja o estudia y el dispositivo con el que el usuario desempeña su labor. El principal objetivo de la arquitectura presentada es facilitar la comunicación teniéndose en cuenta para ello consideraciones relacionadas con la usabilidad de la interfaz finalmente proporcionada y ofreciéndose la posibilidad de que la interfaz se adapte en tiempo de ejecución considerándose los factores antes reseñados.

El trabajo desarrollado representa una experiencia de aplicación de las tecnologías de la Web semántica en un escenario real, y aporta contribuciones novedosas en varios de los aspectos abordados: definición de ontologías en un dominio concreto, exploración, servicios de consulta de catálogo, y propuestas de información, desarrollo de una interfaz de usuario para interactuar con la base de conocimiento, transición desde un sistema en funcionamiento con tecnologías tradicionales hacia una plataforma basada en semántica.

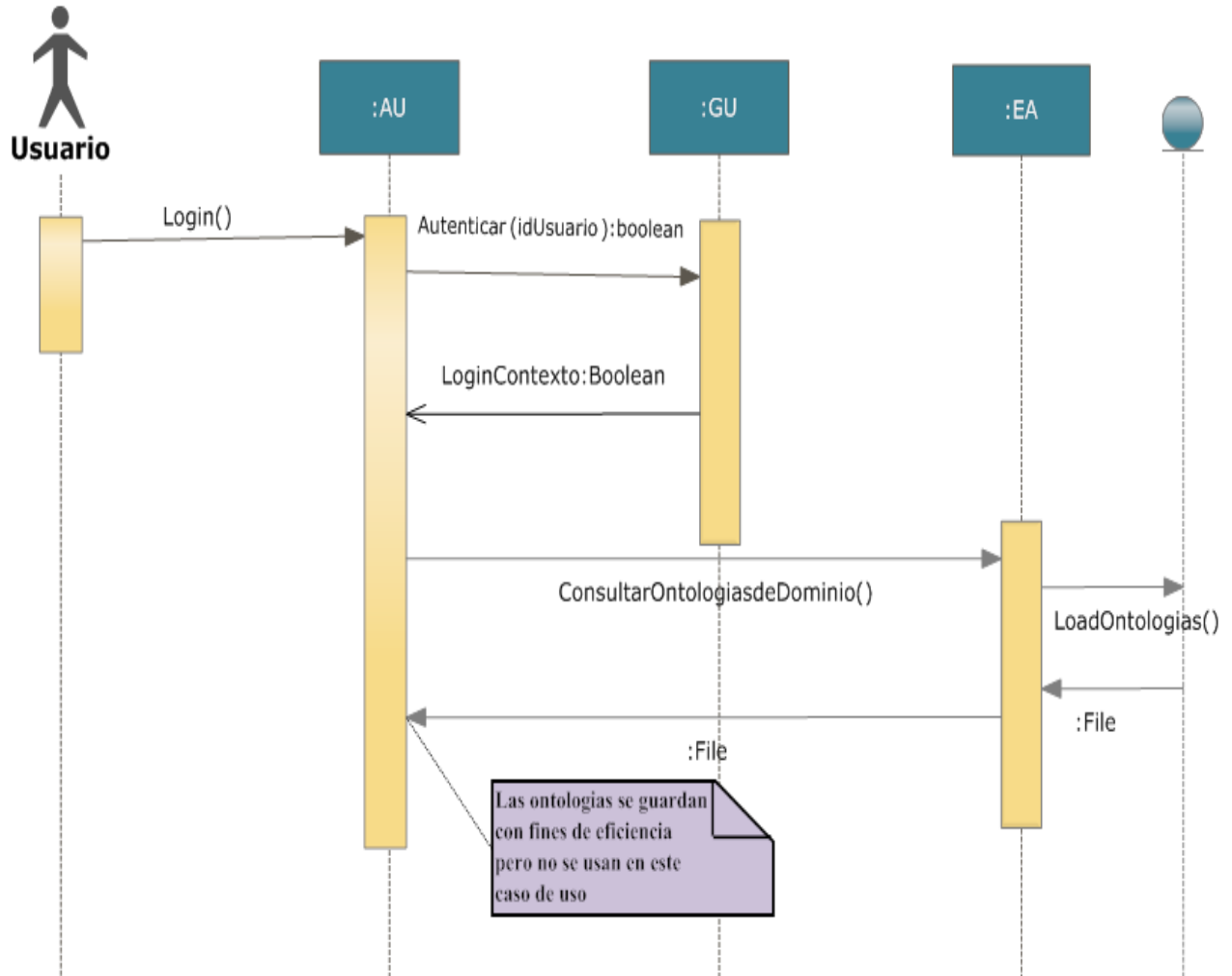
Se desarrolló un sistema basado en agentes el cual permite construir interfaces dinámicas sobre un dominio específico; el sistema desarrollado realiza consultas sobre la base de conocimiento para encontrar relaciones entre los conceptos definidas en la ontología del sistema, además, introduce la utilización de ontologías para la visualización y representación de un dominio y se señala cómo su taxonomía representa un mecanismo para explotar la información semántica subyacente.

La multimodalidad ofrecida y la arquitectura implementada para la extensión de este módulo permiten brindar valor agregado a la interacción de usuario, brindándole flexibilidad y adaptabilidad y logrando desde un mismo recurso ofrecer al usuario variedad de opciones de interacción.

ANEXOS

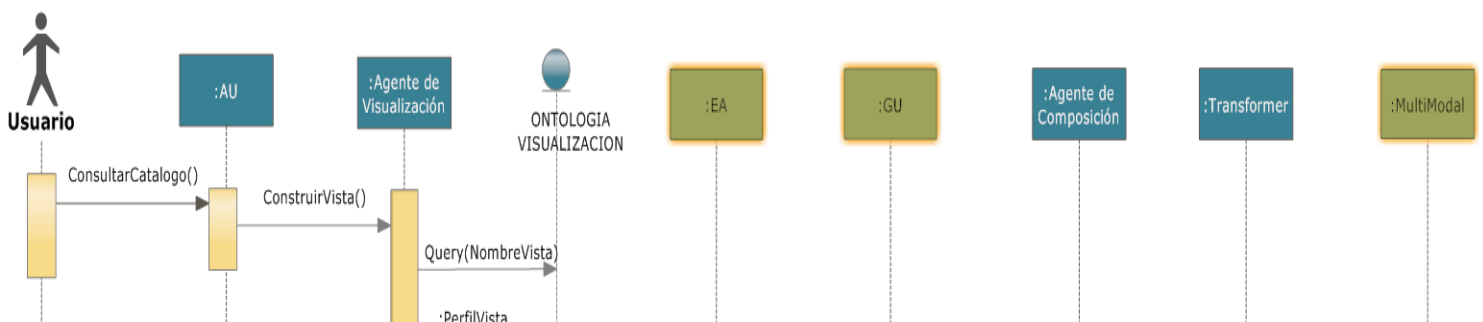
ANEXO 1. DIAGRAMA DE SECUENCIAS LOGIN

Diagrama Login

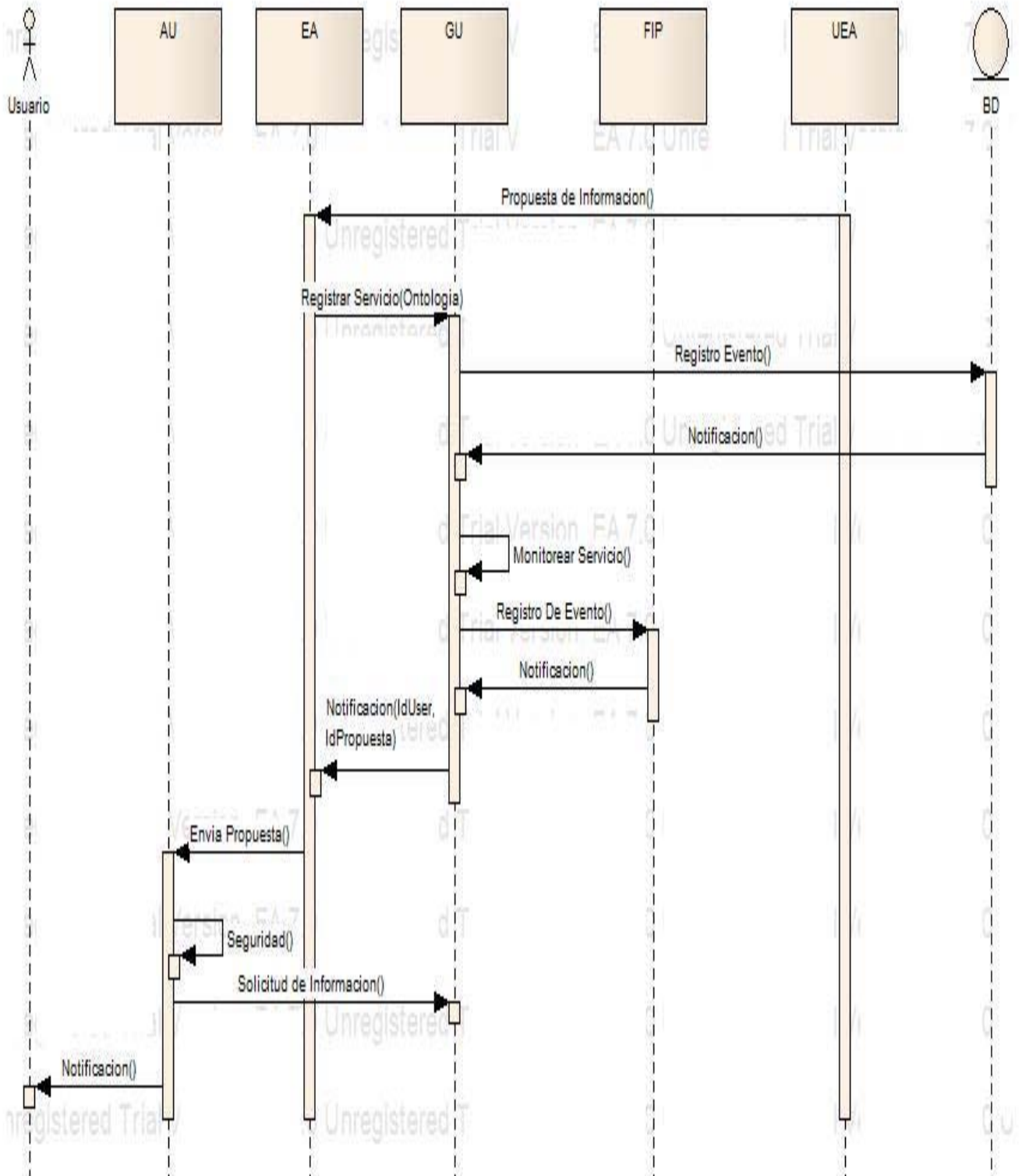


ANEXO 2. DIAGRAMA DE SECUENCIAS CONSULTA DE CATALOGO (PULL)

Diagrama Catálogo de Servicios

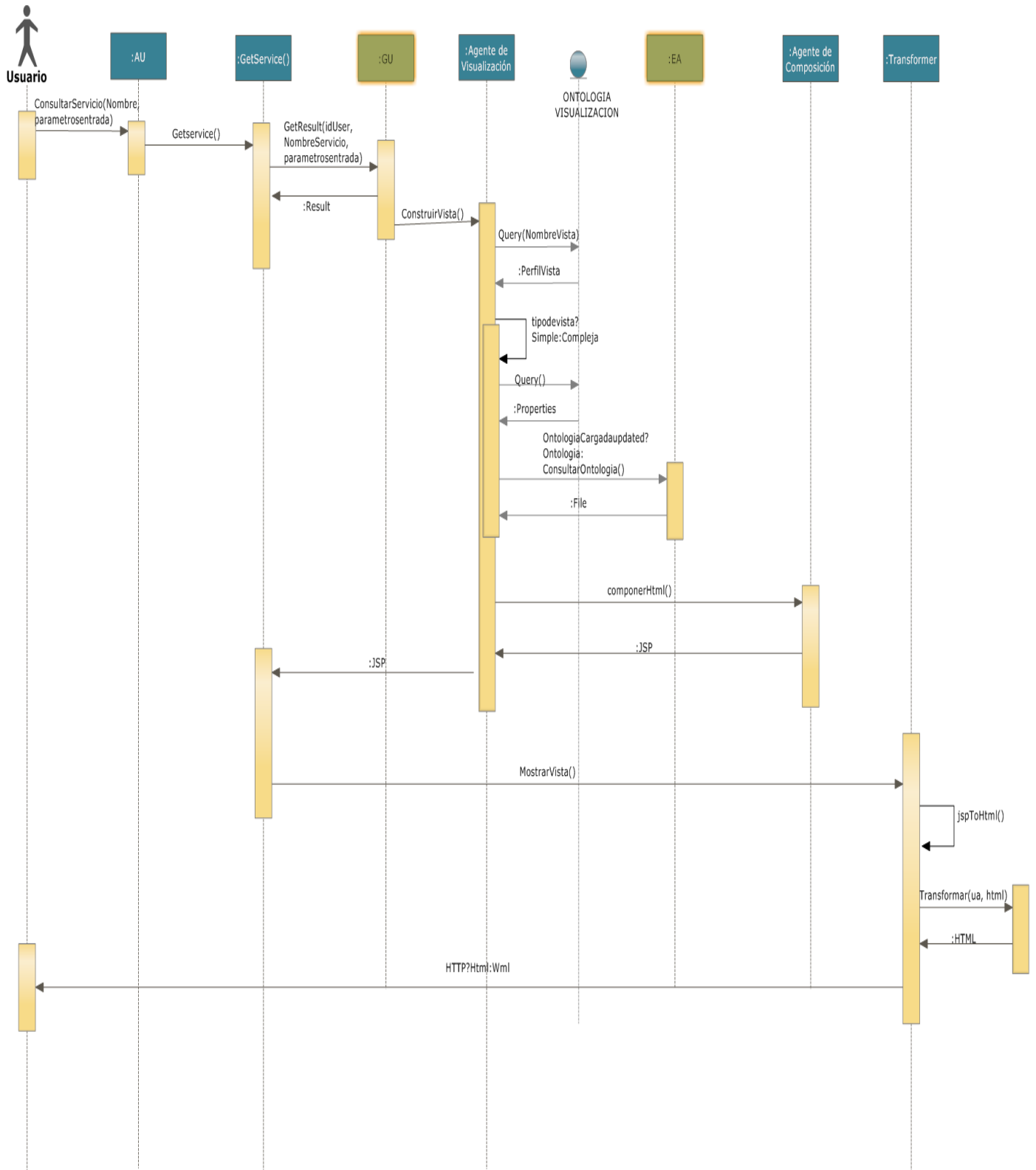


ANEXO 3. DIAGRAMA DE SECUENCIAS PUSH



ANEXO 4. DIAGRAMA DE SECUENCIAS UTILIZAR SERVICIOS

Utilizar Servicio



BIBLIOGRAFIA

- [1] KAHN, R.; Cerf, V. *The Digital Library Project. Volume 1: The World of Knowbots*. Corporation for National Research Initiatives, Reston, 1988.
- [2] JANSEN, Bernard J.; MULLEN, Tracy. *Automated Gathering of Web Information: An In-Depth Examination of Agents Interacting with Search Engines*. ACM SIGCOMM Computer Communication Review.
- [3] LIEBERMAN, Henry; SELKER, Ted. *Agents for the User Interface*. Massachusetts Institute of Technology, ACM, 2007.
- [4] KAY, A; LAUREL, B. *User interface: A personal view*. In *The Art of Human-Computer Interface Design*, Ed. Addison-Wesley, 1990, p. 191-207.
- [5] WOHLTORF, Jens, et al: *BerlinTainment: An Agent-Based Serviceware Framework for Context-Aware Services*. ACM SIGMOBILE Mobile Computing and Communications Review, DAI-Labor, Technische Universität Berlin, Berlin.
- [6] SÁNCHEZ, J. Alfredo. *User interface: User Agents in the Interface to Digital Libraries*, Department of Computer Science, Texas A & M University, College Station, Texas, USA.
- [7] LIEBERMAN, Henry. *Integrating User Interface Agents with Conventional Applications*, ACM Conference on Intelligent User Interfaces, San Francisco, 1998.
- [8] SCHOUWEN, Reinout van: *User Interface Agents: a comparison*, Media Laboratory Massachusetts Institute of Technology. Faculty of Sciences Universiteit Amsterdam.
- [9] LENZMANN B., Wachsmuth I. (1995). A User-Adaptive Interface Agency for Interaction with a Virtual Environment. In Working Notes of the IJCAI-95 Workshop on Adaptation and Learning in Multiagent Systems. 1995, p. 43-46.
- [10] Muhammed, J. *Ontology aware software service agents: Meeting ordinary user needs on the semantic web*. 1990, 191-208.
- [11] Rich, Charles. *Window Sharing with Collaborative Interface Agents*. 1996, Vol.28 No.1.
- [12] Nylander, Stina; Bylund, Markus; Waern, Annika. *Ubiquitous service access through adapted user interfaces on multiple devices*, ACM SIGCOMM Computer Communication Review. 2005, vol. 9, No. 3, p. 123 - 133.
- [13] Dryer, D. Christopher (Ph.D.). *Wizards, Guides, and Beyond: Rational and Empirical Methods for Selecting Optimal Intelligent User Interface Agents*, International Business Machines Corporation.

- [14] Simon, Rainer; Wegscheider, Florian; Tolar, Konrad. *Tool-Supported Single Authoring for Device Independence and Multimodality*. Telecommunications Research Center Vienna, Vienna.
- [15] Pucher, Michael. *Mobile Multimodal.*, 2003. Disponible en internet:
<<http://mona.ftw.at/monaed.html>>
- [16] Souchon, Nathalie; Vanderdonckt, Jean. *A Review of XML-compliant User Interface Description Languages*.
- [17] Garcia, Richard. *Diseño de una Arquitectura para un Sistema de Comunicación Inter-módulos para un portal de servicios para sistemas móviles*, UniAndes, Bogotá, 2007.
- [18] Barrera, Gabriel. *Especificación de un modelo de Sistema Sensible a Contexto (SSC) y desarrollo de prototipo. Caso Universidad de los Andes*. UniAndes, Bogotá, 2007.
- [19] Cover, Robin. *XML Markup Languages for User Interface Definition*. Disponible en Internet:
<<http://xml.coverpages.org/userInterfaceXML.html>>
- [20] Garcia, Roberto. *El punto de encuentro de la información*, Universidad de Lleida, España, 2002.