

Descubrimiento de procesos Web como soporte para la gestión del recurso hídrico



José Armando Ordóñez Córdoba

Maestría en Ingeniería en Ingeniería, área Telemática

Universidad del Cauca
Instituto de Postgrados en Ingeniería en electrónica
2008

Descubrimiento de procesos Web como soporte para la gestión del recurso hídrico

José Armando Ordóñez Córdoba

Trabajo de grado como requisito para optar al título de Magíster en Ingeniería,
área Telemática

Director:
Juan Carlos Corrales Muñoz

Universidad del Cauca
Instituto de Postgrados en Ingeniería en electrónica y
telecomunicaciones
2008

Resumen

La aparición de servicios Web geográficos, ofrece grandes perspectivas para la reutilización de funcionalidades, así como una prometedora reducción de gastos asociados con el reuso de servicios externos, situación que puede ser de gran ayuda en la reducción de la inversión en el área de tecnologías de la información, sobre todo en los países en vía de desarrollo. En este escenario, muchas de las funcionalidades de procesamiento geográfico necesarias por las organizaciones encargadas de la gestión ambiental, están disponibles en Internet. Sin embargo, La clara ventaja de contar con un conjunto enorme de servicios disponibles en Internet, viene acompañada de la necesidad imperiosa de realizar búsquedas rápidas y precisas de servicios entre miles de servicios candidatos, puesto que la dificultad asociada con la búsqueda de servicios, hace que la reutilización: aunque técnicamente posible, sea en la realidad una tarea demasiado dispendiosa para ser realizada. Las propuestas actuales de búsqueda o descubrimiento de servicios, se limitan a la comparación de las entradas y las salidas de los mismos, algunas veces utilizando ontologías. Recientemente se ha comprobado que estas aproximaciones no son suficientes para encontrar servicios relevantes. En este trabajo argumentamos que en muchas situaciones la búsqueda de servicios puede mejorarse teniendo en cuenta la especificación de su comportamiento (composición de servicios), puesto que difícilmente un servicio Web es utilizado de forma aislada. Por otro lado, para poder realizar búsquedas eficientes de servicios para un dominio particular, es fundamental tener en cuenta criterios de búsqueda exclusivos a dicho dominio que permitan obtener resultados apropiados, en este trabajo proponemos que la aplicación de un proceso de búsqueda de servicios al dominio de los análisis hidrológicos requiere de la consideración de criterios de calidad de servicio para la optimización las búsquedas. Finalmente, para el uso de los mecanismos de descubrimiento de servicios en este trabajo proponemos un meta-modelo del proceso de gestión ambiental del recurso hídrico que abstraiga los detalles del proceso, y que sirva de base para la generación de métodos estándares para la gestión ambiental y para la construcción de herramientas software distribuidas que permitan la reutilización de capacidades de geoprocamiento.

Tabla de contenido

Capítulo 1. Introducción	1
1.1. Contexto	1
1.2. Definición del problema	3
1.3. Nuestra Propuesta	4
1.4. Contribuciones y principales resultados	5
1.5. Contenido del documento	6
Capítulo 2. Análisis de los trabajos relacionados	8
2.1. Modelos formales para la representación de procesos	9
2.2. Técnicas de emparejamiento de servicios	13
2.3. Descubrimiento de Servicios Web Geográficos basado en Calidad del Servicio	20
2.4. Modelado de procesos de gestión ambiental del recurso hídrico	26
2.5. Resumen	27
Capítulo 3. Un metamodelo para la gestión de estudios ambientales	29
3.1. Procesos de gestión ambiental del recurso hídrico	29
3.2. El metamodelo del proceso de gestión ambiental	34
3.3. Resumen	38
Capítulo 4. Emparejamiento de comportamiento de Servicios	39
4.1. El problema de emparejamiento de grafos	40
4.2. Emparejamiento de protocolos de conversación	46
4.3. Resumen	54
Capítulo 5. Descubrimiento de WS geográficos ambientales basado en calidad del servicio	55
5.1. Perfil de QoS para los servicios geográficos ambientales.	56
5.2. Proceso de definición de la arquitectura propuesta	57
5.3. Arquitectura para el descubrimiento de servicios Web geográficos basada en QoS	61
5.4. Resumen	68
Capítulo 6. Prototipos y Experimentación	69
6.1. Plataforma para el emparejamiento de servicios basada en comportamiento.	70
6.2. Un sistema de descubrimiento de WS geográficos basado en QoS	75
6.3. Herramienta para la gestión de los indicadores ambientales basada en el metamodelo de gestión ambiental	78
6.4. Una herramienta para evaluar la efectividad del método de emparejamiento de comportamiento.	82
6.5. Evaluación experimental	85
6.6. Resumen	95
Capítulo 7. Conclusiones	97
7.1. Logros alcanzados con el trabajo	97
7.2. Principales contribuciones	98
7.3. Trabajo futuro	99
Bibliografía	100

Lista de Tablas

Tabla 2.1. Comparación de los modelos formales	13
Tabla 2.2. Comparación de las técnicas de emparejamiento	20
Tabla 2.3. Servicios de alto nivel del Modelo General de Servicios de la OGC	22
Tabla 3.1. Ejemplo de clasificación de un estudio.....	37
Tabla 4.1. Correspondencia entre elementos WSCL y elementos del grafo.....	50
Tabla 4.2 Costos para las diferencias de granularidad.....	53
Tabla 5.1. Conjunto de atributos del perfil de QoS.	56
Tabla 6.1. Conjunto de comparación	88
Tabla 6.2 Características de Calidad de los Servicios Prueba	92
Tabla 6.3 Valores para los Criterios de Calidad	92
Tabla 6.4 Atributos de Calidad Dominantes utilizados en la prueba	93

Lista de Figuras

Figura 2.1 Diagrama de Clases del Modelo General del Catálogo OGC.....	23
Figura 3.1. Descripción por paquetes del metamodelo	35
Figura 3.2. Descripción por clases del módulo de estudios.....	36
Figura 3.3. Descripción por clases del módulo de monitoreo	37
Figura 3.4. Relación entre los módulos de monitoreo y de estudios.....	38
Figura 4.1: Ejemplo de la detección de isomorfismo de subgrafos por corrección de error.	44
Figura 4.2: proceso de emparejamiento WSCL.....	47
Figura 4.3: Ejemplo de una conversación WSCL.....	48
Figura 4.4: Representación XML de la transición WSCL	49
Figura 4.5: Representación XML de una interacción ReceiveSend.....	49
Figura 5.1. Vista General de la arquitectura.	63
Figura 5.2. Modelo de información del sistema de descubrimiento.	65
Figura 5.3. Información sobre QoS y metadatos de los servicios.	66
Figura 5.4. Arquitectura de descubrimiento para servicios Web geográficos hidrológicos basada en comportamiento y en QoS.....	67
Figura 6.1. Plataforma de ranking de servicios basada en emparejamiento de comportamientos	70
Figura 6.2. Arquitectura lógica del prototipo	72
Figura 6.3. Interface de documentos WSCL	74
Figura 6.4. Interfase de opciones WSCL	74
Figura 6.5. Interfaces de resultados del emparejamiento WSCL.....	75
Figura 6.6. Arquitectura lógica de la herramienta de gestión de indicadores ambientales.....	76
Figura 6.7 Registro de información de Servicios.....	77
Figura 6.8 Interfaz del catalogo para búsqueda de Servicios	78
Figura 6.9: Arquitectura lógica de la herramienta de gestión de indicadores ambientales	80
Figura 6.10. Interfase de comparación de servicios	81
Figura 6.11: Ejemplo de análisis multiespacial de una variable.....	81
Figura 6.12: Ejemplo de análisis multitemporal de una variable.....	81
Figura 6.13. Arquitectura lógica de la herramienta	83
Figura 6.14: interfase de comparación de servicios	84
Figura 6.15: Interfase de selección de criterios	84
Figura 6.16: Interfase de comparación de ramas del servicio.....	85
Figura 6.17: Interfaz de ranking de servicios	85
Figura 6.18. Calidad del emparejamiento para diferentes funciones de costo.....	89
Figura 6.19 Tiempo de ejecución para diferentes funciones de costos	90
Figura 6.20: Tiempo de ejecución para un número creciente de nodos	90
Figura 6.21 Búsqueda de solo funcionalidad.....	94
Figura 6.22 Resultado de la búsqueda con un atributo de calidad dominante.....	94
Figura 6.23. Búsqueda con dos atributos de calidad dominantes	95
Figura 6.24 Búsqueda con tres criterios de calidad dominantes	95

Capítulo 1. Introducción

1.1. Contexto

Los procesos de gestión ambiental del recurso hídrico comprenden un amplio conjunto de análisis, que inician con un modelado de la realidad, y posteriormente, incluyen la selección de una herramienta metodológica que permite determinar el estado actual de un recurso natural; dichas herramientas emplean formulas matemáticas para el calculo de valores numéricos conocidos como indicadores. La evaluación cuantitativa de estos indicadores permite tomar decisiones sobre el estado de un recurso; por ejemplo, con indicadores como el Índice de calidad del agua ICA y el Biological monitoring working party (BMWP), se puede conocer la calidad y la escasez de agua en una zona geográfica determinada.

El cálculo de indicadores como los presentados en el párrafo anterior se encuentra definido por diversas herramientas metodológicas, las cuales agrupan una gran cantidad de variables, entre las que encontramos: las características climatológicas de la región y la topografía de la zona. En este contexto, se presentan para las instituciones encargadas de la gestión ambiental dos graves problemas [1]:

- La falta de consenso (local, regional y nacional) sobre los criterios y métodos a utilizar (dada su gran variedad) para el análisis del recurso hídrico. Por lo tanto, es muy común que aparezcan diversas perspectivas acerca de la calidad o la abundancia de agua en una región, dependiendo del tipo de indicador utilizado para el análisis. Dichas discrepancias van claramente en detrimento de la gestión eficaz de los recursos naturales y, en particular, del recurso hídrico.
- El segundo es la necesidad de las instituciones encargadas de la gestión ambiental a nivel nacional (como el Instituto de Hidrología, Meteorología y Estudios Ambientales de Colombia IDEAM) o a nivel regional (como la Corporación autónoma regional del Cauca CRC en el Cauca) de contar con un soporte informático para diferentes metodologías de análisis ambiental. Lo cual acarrea enormes gastos asociados con la compra de licencias software, y con los procesos de capacitación en cada una de las plataformas o sistemas que soportan dichas metodologías. Esta situación aparece comúnmente cuando se quiere analizar una zona geográfica nueva (con características ambientales particulares) o se intenta asumir las directrices de una institución gubernamental como el IDEAM en lo relacionado con las metodologías para el cálculo de indicadores.

En vista de la relación existente entre los análisis ambientales y su posición geográfica, las organizaciones relacionadas con la gestión ambiental requieren cada vez más de capacidades de procesamiento espacial o geoprocésamiento provistas por los sistemas de información geográfica – SIG-. Por otro lado, la complejidad y diversidad de los cálculos utilizados hace imposible mantener todas las funcionalidades necesarias en un sistema autónomo independiente y aislado, razón por la

cual, se ha comenzado a pensar en la posibilidad de compartir servicios de geoprocésamiento entre las instituciones. Hoy en día, cada una de las instituciones cuenta con un conjunto propio de recursos tecnológicos (hardware y software) para realizar ciertos análisis hidrológicos. Sin embargo, dada la heterogeneidad de plataformas informáticas, redes de comunicaciones, sistemas operativos y aplicaciones, es prácticamente imposible compartir dichos recursos de procesamiento de forma automática con las demás instituciones, debido a la falta de una infraestructura telemática que permita publicar y descubrir los servicios provistos por estos sistemas de forma transparente a los usuarios. Lo anterior repercute en la duplicidad de funcionalidades e incremento en los costos para todas las instituciones.

Los SIG requieren generalmente de datos conseguidos con grandes inversiones para su levantamiento, por parte de los gobiernos y las instituciones. Lo anterior ha impulsado la investigación acerca de estrategias de reuso de la información existente, sin embargo, los esfuerzos por compartir las capacidades de procesamiento geográfico (o geoprocésamiento) de forma estándar están aún emergiendo lentamente. Afortunadamente, la gran difusión del paradigma de la Arquitectura Orientada al Servicio (SOA) y de los Servicios Web, ha contribuido a la aparición de servicios Web geográficos (servicios de geoprocésamiento accesibles a través de Internet). Los Servicios Web son aplicaciones software modulares y auto describibles, que puede ser identificadas por una URL; estos pueden ser descritos, descubiertos y localizados a través del Internet usando estándares abiertos como SOAP (siglas de Simple Object Access Protocol) y XML (siglas de Extensible Markup Language). Este modelo permite la interoperabilidad de servicios en redes heterogéneas como Internet proporcionando bajo acoplamiento e independencia de la plataforma. Debido a lo expuesto anteriormente, el concepto de Servicios Web ha recibido mucha atención últimamente, como un medio para la integración de SIG utilizando la Internet, al proveer un marco de trabajo o framework¹ interoperable e independiente del proveedor con capacidades para el descubrimiento, acceso, integración, análisis y visualización de múltiples fuentes de datos espaciales, información de sensores y capacidades de procesamiento espacial. ([2],[3],[4]).

La aparición de servicios Web geográficos, ofrece grandes perspectivas para la reutilización de funcionalidades, así como una prometedora reducción de gastos asociados con el reuso de servicios externos, situación que puede ser de gran ayuda en la reducción de la inversión en el área de tecnologías de la información, sobre todo en los países en vía de desarrollo. Es así como muchas de las funcionalidades de procesamiento geográfico necesarias por las organizaciones encargadas de la gestión ambiental, estarán disponibles en Internet, ofertadas por parte de entidades gubernamentales y empresas privadas. Sin embargo, la clara ventaja de contar con un conjunto enorme de servicios disponibles en Internet, viene acompañada de la necesidad imperiosa de realizar búsquedas rápidas y precisas entre miles de servicios candidatos, puesto que la dificultad asociada con la búsqueda de servicios, hace que la reutilización: aunque técnicamente posible, sea en la realidad una tarea demasiado dispendiosa para ser realizada.

La mayoría de las propuestas actuales de búsqueda o descubrimiento de servicios Web genéricos (que no incluyen geoprocésamiento) son aplicables (con ciertas modificaciones) a los servicios Web geográficos. Estas propuestas se limitan a la comparación de las entradas y las salidas de los mismos, algunas veces utilizando ontologías. Recientemente se ha comprobado que estas

¹ En el desarrollo de software un Framework es una estructura de soporte definida sobre la cual otro proyecto de software puede ser organizado y desarrollado.

aproximaciones no son suficientes para encontrar servicios relevantes. En este trabajo proponemos que en muchas situaciones la búsqueda de servicios puede mejorarse teniendo en cuenta la especificación de su comportamiento (forma en que diferentes servicios se agrupan para ofrecer un servicio compuesto o proceso Web), puesto que difícilmente un servicio Web es utilizado de forma aislada; por ejemplo, siempre que se utiliza un servicio Web de validación en un sistema, también se usarán otros servicios como compra de productos, realización de pagos o generación de ordenes de envío, los cuales pertenecen al mismo Proveedor de servicios.

Por otro lado, para poder realizar búsquedas eficientes de servicios para un dominio particular, es fundamental tener en cuenta criterios de búsqueda exclusivos a dicho dominio que permitan obtener resultados apropiados, por ejemplo, cuando un cliente busca un servicio de análisis de calidad del agua, el sistema encontraría los servicios relacionados; sin embargo, si el cliente pudiera especificar además que el tipo de análisis requerido se relaciona con el análisis de calidad de agua para consumo humano y no para utilización en fábricas, los resultados del proceso de búsqueda serían mucho más precisos. Con base en este argumento, proponemos que la aplicación de un proceso de búsqueda de servicios al dominio de la gestión ambiental del recurso hídrico requiere de la consideración de criterios de calidad de servicio.

Finalmente, un elemento clave para el uso de los mecanismos de descubrimiento de servicios (mecanismos basados en comportamiento y en calidad del servicio) al dominio ambiental es el establecimiento de un consenso alrededor del proceso de gestión ambiental. Este acuerdo permite unificar elementos como nombres de los estudios, tipos de las operaciones, etc. Sin este acuerdo, el descubrimiento de servicios Web geográficos hidrológicos no sería posible. En este trabajo argumentamos que dicha estandarización, puede consolidarse partiendo de un meta-modelo de los estudios ambientales que abstraiga los detalles del proceso, y que sirva de base para la estandarización de los procesos de gestión ambiental y para la construcción de herramientas software estándares que permitan el descubrimiento y reutilización de capacidades de geoprocetamiento en entornos abiertos como la Internet.

1.2. Definición del problema

La reutilización de servicios es fundamental para la gestión eficiente de los recursos naturales y particularmente para la gestión ambiental del recurso hídrico. Sin embargo, dicha reutilización es imposible de conseguir si no existen los mecanismos de descubrimiento de servicios adecuados. Las propuestas existentes hasta el momento en el área, se basan en información básica sobre los servicios (nombres de las etiquetas, semántica). Estos mecanismos son ineficientes a la hora de realizar búsquedas en el dominio de la gestión ambiental del recurso hídrico debido a que no tienen en cuenta el comportamiento de los servicios (fundamental para la gestión ambiental, dado que difícilmente se encuentra un servicio único funcionando de forma aislada, sino por el contrario en cooperación con otros) y no incluyen en sus consultas información sobre calidad del servicio (esta información es determinante a la hora de seleccionar un servicio en un dominio en particular).

El problema a tratar en este trabajo, gira precisamente en torno de la optimización del proceso de descubrimiento de servicios Web geográficos en el dominio de la gestión ambiental del recurso hídrico. Para ello se propone el enriquecimiento de los mecanismos de consulta existentes en la actualidad con información sobre el comportamiento de los servicios (útil para servicios genéricos y

de geoprocésamiento) y con los criterios de calidad del servicio (propios del dominio hidrológico) del dominio ambiental. Esta solución conduce a definir: formalismos para el modelado de los servicios geográficos, algoritmos para la determinación de semejanzas entre instancias de estos modelos y mecanismos para la definición de las características de calidad de los servicios, y su correlación con los requerimientos del usuario. Igualmente, para la utilización de los mecanismos de búsqueda en el entorno ambiental se propone además la definición de un meta-modelo del proceso de gestión ambiental que sirva de marco conceptual para los mecanismos de búsqueda propuestos.

1.3. Nuestra Propuesta

Nuestra propuesta se enmarca dentro de dos elementos que facilitan el descubrimiento de servicios de geoprocésamiento en el entorno ambiental: el primer elemento es la aplicación de mecanismos para mejorar el descubrimiento de servicios (búsqueda de servicios basada en el comportamiento y en parámetros de calidad del servicio). El segundo elemento es el diseño de un meta-modelo del proceso de gestión de los recursos hidrológicos que permita la aplicación de estos mecanismos de búsqueda en el dominio de la gestión ambiental del recurso hídrico. A continuación se detallan estos dos elementos:

- En cuanto al primer elemento, relacionado con la aplicación de los dos mecanismos para mejorar el descubrimiento de servicios tenemos:
 - Para la definición de un mecanismo de búsqueda basado en comportamiento se usarán técnicas de emparejamiento que operen sobre modelos de comportamiento de los servicios, que permitan identificar similitudes parciales entre estos modelos, así como la evaluación de la distancia semántica entre estas similitudes con los requerimientos del usuario. En consecuencia, si no existe un servicio que satisfaga exactamente los requerimientos del usuario, se obtendrán los más similares y propuestos para su reuso o modificación. Para conseguir esto, se reduce el problema del emparejamiento de comportamiento de los servicios a un problema de emparejamiento de grafos, adaptando algoritmos existentes para este propósito. [5]
 - En cuanto al mecanismo de búsqueda de servicios geográficos para la gestión del recurso hídrico basada en parámetros de calidad del servicio, se parte de la premisa de que la calidad de servicio a nivel de aplicación puede definirse utilizando descripciones esquematizadas en forma de metadatos asociados a los servicios. De esta manera se define un perfil de calidad para servicios geográficos hidrológicos, y se extienden las especificaciones de Metadatos ISO 19115 y NTC 4611 para incluir descripciones de calidad que puedan ser confrontadas con los requerimientos de calidad de los usuarios
- El segundo elemento es el diseño de un meta-modelo del proceso de gestión de los recursos hidrológicos, que permita la aplicación de los mecanismos de búsqueda descritos anteriormente, en el dominio de la gestión ambiental del recurso hídrico. Este metamodelo ofrece una descripción global del proceso de gestión de los recursos hidrológicos, definiendo de forma genérica elementos propios del dominio (como estudios, operaciones, variables). La utilización de este metamodelo provee un marco conceptual estándar para el descubrimiento y reutilización

de servicios. De igual forma, el metamodelo propuesto sirve de base para la implementación de herramientas software y metodologías estándares de análisis ambiental (como la presentada en la sección de prototipos y experimentación).

1.4. Contribuciones y principales resultados

Las principales contribuciones de esta tesis son las siguientes:

- **Un metamodelo para la gestión de estudios ambientales.** Se desarrolló un metamodelo de la gestión de los estudios ambientales que ofrece una descripción global de dichos procesos, lo anterior representa un avance significativo, al brindar un modelo general para la implementación de herramientas software y metodologías de análisis ambiental. Este metamodelo define elementos propios del dominio de la gestión de recursos hidrológicos (como estudios, operaciones, variables); ofreciendo, de esta manera, un marco general y estándar para el descubrimiento y reutilización de servicios geográficos ambientales.
- **Un prototipo para validar el metamodelo para la gestión de estudios ambientales.** Se desarrolló un prototipo para la validación del metamodelo dentro del módulo de análisis Hidrológico de la plataforma SIRA (Sistema de Información Regional del Agua) [6]. La herramienta consiste de una Aplicación Web que permite la definición de estudios y el cálculo de algoritmos de gestión ambiental y la aplicación de los mismos sobre un conjunto de datos preestablecido. El sistema implementado sirve de base para el cálculo de todos los indicadores ambientales del sistema. Además, debido a su naturaleza abierta y flexible, el prototipo que implementa el metamodelo, permitió resolver el problema de la adaptación del sistema SIRA a los diferentes estudios ambientales hidrológicos. Lo anterior había sido una enorme limitante de las versiones anteriores de SIRA. Además, esta limitante, persiste aún en la mayoría de las herramientas software de gestión ambiental.
- **Un análisis detallado del descubrimiento de servicios Web geográficos Hidrológicos:** los principales resultados de este análisis son: (i) un análisis de las representaciones formales de los servicios, las cuales permiten describir, intercambiar y ejecutar servicios. (ii) una descripción de las técnicas existentes para el emparejamiento de servicios genéricos basadas en los formalismos para el modelado de procesos (iii) Una revisión de las propuestas específicas para el descubrimiento de Servicios Web geográficos basadas en calidad del servicio (vi) una revisión analítica del modelamiento de procesos de gestión ambiental del recurso hídrico.
- **Aplicación del emparejamiento de servicios al protocolo WSCL** Con el objetivo de validar el método de descubrimiento basado en comportamiento, se ejemplifica nuestra propuesta para los protocolos de conversación expresados utilizando el modelo WSCL (ver [[5],[7],[8],[9]). El proceso de emparejamiento de protocolos de conversación se compone de los siguientes pasos: Primero, los protocolos WSCL se convierten en grafos, a continuación, en ambos grafos se aplica el algoritmo de emparejamiento de grafos con corrección de error y finalmente la función de similitud evalúa la semejanza entre los grafos.
- **Un prototipo del descubrimiento de servicios basado en comportamiento.** Se desarrolló un prototipo que implementa la propuesta de descubrimiento basado en emparejamiento de

comportamientos. La herramienta (disponible como aplicación Web y como Servicio Web) permite la ejecución de algoritmos para emparejamiento de servicios descritos con WSCL.

- **Aplicación de criterios de calidad del servicio al descubrimiento de servicios geográficos ambientales.** Partiendo de la premisa de que la calidad de servicio a nivel de aplicación puede definirse utilizando descripciones esquematizadas en forma de metadatos asociados a servicios y datos; se presenta un perfil de calidad para servicios geográficos, y se extienden las especificaciones de Metadatos ISO 19115 y NTC 4611 para incluir descripciones de calidad que puedan ser confrontadas con los requerimientos de calidad de los usuarios, igualmente se presenta una arquitectura para el descubrimiento de servicios hidrológicos basado en calidad del servicio.

- **Un prototipo para el descubrimiento de servicios basado en Calidad del servicio.** Se desarrolló un sistema prototipo que implementa la propuesta de descubrimiento basado en calidad del servicio. La herramienta permite la definición de criterios de calidad y la realización de búsquedas de servicios basadas en esos requerimientos. Se realizó la validación de la propuesta utilizando extensiones a los estándares de Metadatos ISO 19115 y NTC 4611, en un repositorio UDDI.

- **Un prototipo para evaluar la efectividad del método de emparejamiento de comportamientos.** Se implementó una herramienta para evaluar la efectividad del método de emparejamiento de comportamiento. Esta herramienta permite crear un ranking de servicios basada en comparaciones manuales entre un servicio de consulta y los servicios en el repositorio. La herramienta permite comparar el resultado obtenido por el prototipo de búsqueda y el ranking definido por los usuarios.

1.5. Contenido del documento

El resto de esta tesis se organiza en siete capítulos:

El capítulo 2 describe los trabajos relacionados, divididos en 4 categorías: (i) Dado que uno de los aportes de nuestro trabajo es el desarrollo de técnicas de descubrimiento que operan sobre modelos de comportamiento de servicios, esta sección presenta las representaciones formales de los servicios, las cuales permiten describir, intercambiar y ejecutar procesos web. (ii) En vista de que nuestro trabajo busca mejorar el proceso de descubrimiento, cuyo proceso base es el emparejamiento, en esta sección explicamos las técnicas existentes para el emparejamiento de servicios Web genéricos (sin geoprocesamiento), las cuales son aplicables al descubrimiento de servicios geográficos, posteriormente se abordan las particularidades de los servicios geográficos ambientales. (iii) Teniendo en cuenta que uno de nuestros aportes es la definición de mecanismos de descubrimiento de servicios Web geográficos basado en calidad del servicio, en esta sección se presentan la definición de los Servicio Web Geográficos y sus características más importantes. Igualmente, se presentan las propuestas encontradas para la búsqueda de dichos servicios web geográficos basada en calidad del servicio. (iv) Atendiendo a que en este trabajo argumentamos que para poder aplicar los mecanismos de descubrimiento al dominio ambiental, es necesaria la definición de un meta-modelo del proceso de gestión ambiental del recurso hídrico, en esta sección se presenta todo el bagaje conceptual para la generación del mismo.

El capítulo 3 presenta el metamodelo de la gestión de los estudios ambientales, el cual define de forma genérica elementos propios del dominio de la gestión de recursos hidrológicos (como estudios, operaciones, variables). La utilización de este metamodelo provee un marco conceptual estándar para el descubrimiento y reutilización de servicios geográficos ambientales.

El capítulo 4 explica el emparejamiento de servicios Web basado en grafos y muestra una aplicación de este. Primero se aborda el problema de emparejamiento de grafos explicando su definición y notación. Posteriormente, se presenta un método para medir la similitud de dos grafos. Luego, se explica el emparejamiento de subgrafo por corrección de error en detalle, finalmente se expone un caso de estudio de este algoritmo al protocolo WSCL (Web Services Conversation Language) en el dominio de la gestión ambiental del recurso hídrico.

El capítulo 5 presenta el descubrimiento de Servicios Web geográficos basado en calidad del servicio, el cual busca mejorar el descubrimiento basado en comportamiento. Primero se presenta el perfil de calidad de servicios para el entorno ambiental y luego, se muestra la arquitectura para el proceso de búsqueda de servicios, la cual funciona como un filtro sobre las búsquedas basadas en comportamientos. Finalmente, se muestra la arquitectura completa del sistema de descubrimiento de servicios Web geográficos ambientales basada en comportamiento y en calidad del servicio.

Capítulo 6. Prototipos y Experimentación. Ilustra el uso práctico de nuestras propuestas para la búsqueda de servicios Web geográficos hidrológicos. Primero se muestra el prototipo del sistema de búsqueda basado en comportamiento del servicio, y un prototipo de búsqueda basado en calidad del servicio, mostrando para ambos, sus funcionalidades, arquitectura e interfaz de usuario. A continuación, el prototipo de validación del meta-modelo, que ofrece una herramienta Web para la gestión de los estudios ambientales hidrológicos, describiendo sus funcionalidades, arquitectura e interfaz de usuario; posteriormente, mostramos la herramienta para evaluar la búsqueda basada en comportamiento. Finalmente, presentamos la experimentación realizada para comprobar la efectividad de los métodos de descubrimiento propuestos, describiendo para cada uno de ellos: los escenarios de prueba, las estrategias de prueba y los resultados obtenidos.

Capítulo 7. Conclusiones. Presenta las conclusiones y algunas perspectivas de futuros trabajos.

Anexo 1. Herramientas de desarrollo de aplicaciones Web SIG de software libre: presenta una descripción de las principales herramientas de desarrollo de aplicaciones Web SIG de software libre.

Anexo 2. Ofrece una descripción de los mecanismos de integración de la plataforma de descubrimiento basada en calidad del servicio propuesta en el capítulo cinco.

Capítulo 2. Análisis de los trabajos relacionados

Los procesos de gestión ambiental pueden ser optimizados mediante la reutilización de servicios Web geográficos como se discutió en el capítulo uno. Sin embargo, si no existen mecanismos eficientes y automatizados de búsqueda de servicios, el proceso de reutilización sería imposible en la práctica. Recientemente se ha comprobado que las aproximaciones existentes para las búsquedas de servicios no son suficientes para encontrar servicios relevantes. Nuestra propuesta se centra en mejorar el proceso de búsqueda de servicios soportándose en dos mecanismos: la especificación de su comportamiento y los criterios de calidad de servicio definidos por el usuario; igualmente proponemos un metamodelo que sirva de marco de referencia para la aplicación de estos mecanismos en el dominio ambiental hidrológico.

Con base en lo anterior, el presente trabajo de investigación se aborda desde dos ejes de investigación: el primero es la creación de un modelo conceptual (metamodelo) sobre los procesos de gestión ambiental, el cual podrá ser utilizado como base de todas las aplicaciones desarrolladas en el área de la gestión ambiental del recurso hídrico, este modelo permite tener una representación del proceso de gestión de los recursos hídricos y sirve de base para la aplicación los mecanismos de descubrimiento de servicios geográficos hidrológicos.

El segundo eje de investigación, es la definición de los mecanismos que mejoren la calidad del proceso de descubrimiento de servicios basándose en la especificación de su comportamiento y los criterios de calidad de servicio definidos por el usuario. Para el primer elemento, se usarán técnicas de emparejamiento que operen sobre modelos de comportamiento (estructura) de los servicios compuestos (también conocidos como procesos Web) y permitan identificar semejanzas parciales así como la evaluación de la distancia semántica entre estas semejanzas y los requerimientos del usuario; de esta forma, si no existe un servicio que satisfaga exactamente los requerimientos del usuario, se obtendrán los más similares los cuales serán propuestos para su reuso o modificación. Para conseguir esto, se reduce el problema del emparejamiento de comportamiento de los servicios a un problema de emparejamiento de grafos, adaptando algoritmos existentes para este propósito. Para el segundo elemento, se parte de la premisa de que la calidad de servicio a nivel de aplicación puede definirse utilizando descripciones esquematizadas en forma de metadatos asociados a servicios y datos. De esta manera se define un perfil de calidad para servicios geográficos, y se extienden las especificaciones de Metadatos ISO 19115 y NTC 4611 para incluir descripciones de calidad que puedan ser confrontadas con los requerimientos de calidad de los usuarios.

Teniendo en cuenta lo descrito anteriormente, los trabajos relacionados con esta tesis de maestría han sido agrupados en cuatro categorías, la temática de las dos primeras se relaciona con el descubrimiento de servicios genéricos (que no incluyen geoprocetamiento) basado en los formalismos para el modelado de procesos. Aunque nuestro trabajo se centra en el descubrimiento de servicios Web geográficos hidrológicos, la mayor parte de los trabajos de investigación sobre descubrimiento de servicios están orientados a servicios genéricos, además, la mayor parte de estos trabajos pueden adaptarse al dominio geográfico con modificaciones mínimas. Las dos secciones restantes abordan las particularidades del descubrimiento de servicios Web geográficos hidrológicos

basados en calidad del servicio y en comportamiento enmarcadas en un metamodelo del proceso de gestión ambiental del recurso hídrico. Las secciones que componen este capítulo son las siguientes:

- (i) Como buscamos desarrollar técnicas de descubrimiento que operan sobre modelos de comportamiento de servicios, en esta sección se presentan las representaciones formales para describir, intercambiar y ejecutar servicios compuestos (o procesos Web).
- (ii) En vista de que nuestro trabajo busca mejorar el proceso de descubrimiento, y teniendo en cuenta que el proceso base de la búsqueda de servicios es el emparejamiento, en esta sección explicamos las técnicas existentes para el emparejamiento de servicios genéricos basadas en los formalismos para el modelado de procesos genéricos (posteriormente se estudiarán las características particulares de los servicios geográficos). Esta sección explica solamente las técnicas de emparejamiento de servicios basadas en interfaces, semántica y comportamiento. Por su parte, el emparejamiento basado en calidad del servicio se estudiará en la siguiente sección de descubrimiento de servicio web geográficos basado en calidad del servicio, lo anterior se debe a que el descubrimiento de servicios basado en calidad del servicio está fuertemente relacionado con el dominio de aplicación, en este caso el dominio de la gestión del recurso hídrico.
- (iii) Dado que nuestra propuesta esta dirigida al descubrimiento de servicios Web geográficos mejorada con la especificación de los criterios de calidad del servicio, en esta sección se presentan las alternativas para la búsqueda de servicios Web geográficos basadas en calidad del servicio.
- (iv) Atendiendo a que en este trabajo argumentamos que para poder aplicar los mecanismos de descubrimiento al dominio ambiental, es necesaria la definición de un meta-modelo del proceso de gestión ambiental del recurso hídrico, en esta sección se presenta una revisión de las potencialidades de los metamodelos como mecanismo de abstracción.

2.1. Modelos formales para la representación de procesos.

La computación orientada a servicio (SOC) se refiere al conjunto de conceptos, principios y métodos que representan la computación en la arquitectura orientada a servicio SOA, en la cual las aplicaciones software son construidas basadas en servicios o componentes independientes con interfaces estándares. Todos los componentes son empaquetados con interfaces abiertas estándar, estos componentes se conocen como servicios o servicios Web (si están disponibles sobre Internet) y están disponibles para los constructores de aplicaciones, de forma que un mismo servicio puede ser utilizado por muchas aplicaciones compuestas. Los servicios son autónomos e independientes de la plataforma computacional, pueden ser descritos, publicados, descubiertos, y ensamblados dinámicamente; este último aspecto lo hace propicio para el desarrollo aplicaciones distribuidas en el dominio de la gestión ambiental. Cualquier pieza de código y cualquier componente de aplicación desarrollado en un sistema pueden ser re-usado y transformado en un servicio disponible a través de la red.

Una de las características más atractivas de los servicios Web es la capacidad de integrar servicios Web individuales por medio de la composición de sus funcionalidades, para crear procesos Web [10]. Actualmente, existen diversas iniciativas como WSCL (siglas de Web Services Conversation Language), BPEL (siglas de Business Process Execution Language) y WSCDL (siglas de Web Services Choreography Description Language) para la definición de procesos Web, estas iniciativas permiten definir y gestionar actividades de procesos de negocios y protocolos de interacción de negocios incluyendo servicios colaborativos: Los términos mas utilizados son: interfase de comportamiento, orquestación y coreografía [10]. La orquestación difiere de la coreografía en que la primera expresa la lógica de procesos de negocios perteneciente a una sola organización. La coreografía por su parte, es mas colaborativa por naturaleza; esta rastrea la secuencia de mensajes

que involucran múltiples partes, donde ninguna parte “es dueño” de la conversación. La coreografía se asocia con el intercambio público de mensajes que ocurren entre múltiples servicios Web participantes que pueden asumir diferentes roles y que tienen diferentes relaciones. Por otro lado, una interfase de comportamiento expresa el comportamiento de un proveedor de servicio particular o usuario del servicio en su comunicación con otro proveedor o usuario de servicio para conseguir un objetivo particular. Ya que una interfase de comportamiento solo describe el comportamiento de un lado, solo comprende un rol en la conversación de servicios. Debido a que el presente trabajo es motivado por la idea de que un proceso inter-organizacional puede ser considerado como una cooperación de varios procesos preestablecidos de muchas organizaciones, concentramos nuestros esfuerzos en emparejar interfaces de comportamiento, considerando que esta captura las interacciones desde la perspectiva de una de las organizaciones.

A continuación se presentan cuatro formalismos para el modelado de los procesos de negocios. Estos formalismos ofrecen características de representación que las hacen propicias para describir, intercambiar y ejecutar procesos de negocios.

2.1.1. Modelos de algebra de procesos

Los procesos pueden ser modelados utilizando algebra de procesos. Ejemplos de las algebras de procesos son cálculo de sistemas comunicantes (CCS)[11], procesos secuenciales comunicantes (CSP) [12] y pi-calculus. Estos lenguajes permiten especificar y razonar sobre sistemas o procesos concurrentes. Ellos proveen un conjunto de términos, operadores y axiomas para escribir y manipular expresiones algebraicas. El comportamiento de los sistemas modelados puede analizarse basándose en los operadores definidos. Los pi-calculus son un modelo matemático de procesos cuyas interconexiones cambian mientras interactúan [13]. El paso computacional básico es la transferencia de un enlace de comunicación entre dos procesos; el receptor puede entonces utilizar el enlace para interacciones posteriores con otras partes. Esto hace que el pi-calculus sea particularmente apropiado para representar aquellos sistemas en los cuales los recursos accesibles varían con el tiempo. El pi-calculus, además de modelar sistemas concurrentes, pueden expresar procesos móviles y técnicas para analizar su comportamiento. Algunas de las operaciones soportadas por estos lenguajes son simulación y bisimulación de instancias de los procesos. Por ejemplo, dados dos descripciones de procesos en la notación de pi-calculus, es posible chequear su equivalencia utilizando operaciones de disimulación.

En [14] los autores argumentan que BPEL no esta equipada con semántica formal (como otras propuestas existentes) y ya que este incluye un gran número de aspectos (como el concepto de alcance y los mecanismo del compensación de fallas), es difícil razonar formalmente sobre el comportamiento de los procesos. Con base en esta reflexión, En [14] se propone la semántica de un fragmento BPEL. La propuesta representa una contribución importante en dos direcciones. En primer lugar, formaliza un lenguaje de orquestación novedoso que representa una simplificación de WS-BPEL incluyendo una especificación sin ambigüedades, haciendo posible razonar formalmente sobre los procesos de orquestación. En segundo lugar, permite conseguir todo lo necesario para tener el proceso ejecutable, a través de la compilación del lenguaje propuesto. Este lenguaje se compone de un conjunto pequeño de operadores que reúnen los comportamientos BPEL, ofreciendo una simplicidad razonable para los diseñadores de aplicaciones. Este lenguaje incluye: un operador paralelo que permite concurrencia explícita, un operador de restricción que permite composición y creación de recursos explícitos; una recursión o un operador de definición de procesos; un operador de secuencias permitiendo relación entre actividades; un operador de inactividad el cual es solo un término base para definición inductiva de secuencia; paso de mensajes y especialmente operadores de paso de nombre permitiendo comunicación y movilidad de enlaces.

2.1.2. Modelos de redes de Petri.

Una red de petri es un lenguaje grafico y formal para modelar sistemas o procesos [15] . Este modelo comprende lugares, transiciones, arcos y tokens. Los lugares representan estados de una red de petri y pueden contener tokens. Los arcos de entrada conectan lugares con transiciones, mientras los arcos de salida comienzan en una transición y terminan en un lugar. El estado del sistema modelado en un momento dato se llama marcaje, y esta dado por el número de tokens en cada lugar. El marcaje de sistema cambia cuando las transacciones se disparan, lo cual significa que los tokens son removidos desde los lugares de entrada y son insertados a los lugares de salida de una transición. Las transiciones solo pueden dispararse si están habilitadas, significando que existen tokens listos para dispararse en los lugares de entrada.

Las redes de Petri proveen una herramienta para la descripción de sistemas que son caracterizados como concurrentes, asíncronos, distribuidos y no determinísticos, en forma grafica, las redes de Petri pueden usarse como una ayuda de comunicación visual. En una forma similar a las notaciones de diseño estructurado de los análisis de sistemas tradicionales y metodologías de diseño. El lenguaje de las redes de Petri, provee una base matemática para la descripción y análisis de modelos matemáticos y algebraicos. Este ofrece una notación práctica para describir el comportamiento de los procesos de sistemas [16].

Un formalismo relacionado con las redes de petri es el Workflow Net (WF-Net) [17], las WF-Nets están basadas en tokens y contienen lugares y transiciones, y además, contiene un lugar inicial y final. Las WF-Nets tienen mejores propiedades computacionales que las redes de Petri, pero son usadas en modelos de comunicación asíncronos donde los mensajes pueden llegar en un orden diferente al cual ellos fueron enviados.

Los autores de [18] proponen un algebra basada en redes de Petri para componer servicios Web. La semántica formal de los operadores de composición esta expresada en términos de redes de Petri, lo cual permite tener un mapeo directo para cada operador a una construcción de redes de Petri. Así, cualquier servicio expresado utilizando los constructores de algebra pueden ser trasladados en una representación de redes de Petri. Por medio de un conjunto de propiedades de algebra, la propuesta es capaz de transformar y optimizar expresiones de servicios Web garantizando la misma semántica de las expresiones iniciales. Además el uso de un modelo formal permite la verificación de propiedades y la detección de inconsistencias dentro y entre los servicios.

2.1.3. Autómatas de estados finitos (FSA)

Un FSA esta definido por un conjunto de mensajes, estados, un conjunto de transiciones, un estado inicial y un conjunto de estados finales o permitidos [19]. Puede representarse como un grafo con un estado inicial simple, donde los nodos representan estados, los arcos representan transiciones conectando dos estados. Las transiciones están etiquetadas con mensajes dibujados desde el conjunto de mensajes. Los grafos FSA son atravesados desde el estado inicial. Los estados finales son marcados específicamente con ciclos concéntricos y representan la aceptación de una secuencia de mensajes para una FSA. Las FSAs están cerradas bajo intersecciones y tienen algoritmos de tiempo polinomial para prueba de vacío e intersección. Sin embargo las FSA en su forma original no pueden representar la semántica obligatoria de la secuencia de mensajes. Los FSA no tienen semántica para la ejecución paralela, como la provista por propuestas más expresivas como las redes de Petri.

En [20] los autores describen un modelo conceptual para representar comportamientos de e-services y restricciones temporales, basadas en FSA. FSA permite calcular una gran clase de e-services y verificar formalmente propiedades importantes de los e-services [21], tales como exactitud, seguridad (i.e., a cada punto en la ejecución de un e-service algunas invariantes lógicas aparecen), vitalidad (i.e., un e-service tiene la garantía de moverse hacia un punto donde un objetivo puede alcanzarse). Adicionalmente, esta propuesta presenta un nuevo lenguaje WSTL (web service transition language) basado en dicho modelo conceptual. WSTL provee construcciones que pueden mapearse directamente a los elementos correspondientes de FSA.

2.1.4. Representación de Grafos

Los grafos son una estructura de datos general y poderosa para la representación de objetos y conceptos. En una representación de grafos, los nodos típicamente representan objetos o partes de objetos, mientras las aristas describen relaciones entre objetos o partes de objetos. Los grafos tienen algunas propiedades interesantes. Por ejemplo, si un grafo que está dibujado en un papel, está trasladado, rotado o transformado, éste es aun el mismo grafo en el sentido matemático. Estas propiedades de invariabilidad, así como el hecho de que los grafos sean muy apropiados para modelar objetos en términos de partes y sus relaciones, los hacen atractivos para diversas aplicaciones, entre ellas el reconocimiento de modelos y la visión por computador [22]. Dada una base de datos de objetos conocidos y una consulta, la tarea es conseguir uno o muchos objetos de la base de datos que sean similares a la consulta. Si los grafos son usados para representación de objetos este problema de descubrimiento se reduce a determinar la similaridad de grafos, lo que se conoce generalmente como emparejamiento de grafos.

En [23] los autores presentan un metamodelo posible basado en los grafos conceptuales para representar procesos que llenen requerimientos de memoria corporativos (adquisición, almacenamiento, evolución y diseminación de conocimiento adquirido por una organización). El metamodelo es compuesto de tres conceptos básicos: actividad, procesos y eventos. Una actividad está definida por sus entradas y salidas, los agentes que hacen posible la actividad y las pre y post condiciones. Las precondiciones definen condiciones o estados que deben ser verificados para iniciar la ejecución de una actividad; las postcondiciones definen estados o condiciones que resultarán de la ejecución de la actividad. Un evento es un punto en el tiempo que marca el final de una actividad; este indica la realización de la postcondición de la actividad. Un proceso está definido como un conjunto de eventos que representan la ejecución de un conjunto de actividades.

En [24], los autores presentan una transformación de BPEL a EPC (Event-Driven Process Chains). Un esquema EPC plano está definido como un grafo coherente y dirigido con restricciones de tipo y cardinalidad. Construido sobre un mapeo conceptual, este acercamiento presenta un programa de transformación que es capaz de generar automáticamente modelos EPC como archivos EPML (formato de intercambio basado en XML para EPCs) a partir de definiciones de procesos BPEL. Tal transformación ayuda a comunicar procesos BPEL a los analistas del negocio que son usualmente los encargados de la aprobación de la lógica de negocios. La visualización EPC se centra en el comportamiento dinámico del modelo BPEL. Las construcciones de BPEL (i.e. actividades estructuradas y básicas) son transformadas a bloques de elementos EPC que ofrecen semántica equivalente. Los elementos EPC obtienen nombre generados a partir de los correspondientes elementos BPEL. Además, el programa puede ser usado para reingeniería de procesos BPEL. Finalmente, el concepto de transformación es general, de manera que puede ser adaptado fácilmente para generar salidas en otros lenguajes de procesos basados en grafos estén codificados en XML.

2.1.5. Comparación de modelos formales.

La tabla 2.1 resume las representaciones formales de procesos explicados esta sección. El símbolo (+) significa que el parámetro en esa columna esta soportado por el modelo formal. El símbolo (-) significa que la propiedad no esta soportada, el signo (+/-) significa que la propiedad esta soportada moderadamente. Finalmente, el símbolo N/A significa no aplicable, lo cual significa que la propiedad no aplica al modelo con el cual esta siendo asociado.

El objetivo de esta comparación es proveer criterios para el modelado de composición de servicios Web (incluyendo estado de los recursos, etc.) utilizando una notación. Se considera: integridad (un conjunto completo de semántica y madurez de notación), capacidad de composición (una propiedad que permita razonar sobre un sistema compuesto con base en de sus partes constitutivas sin ayuda adicional de información sobre la implementación de sus partes), paralelismo (un aspecto clave de formalismo el cual debe ser cumplido para modelar apropiadamente las composiciones de servicios Web) y la complejidad de la ejecutar el emparejamiento (emparejamiento polinomial)

Parámetros de comparación	Parámetros de comparación			
	Integridad	Capacidad de Composición	paralelismo	Emparejamiento polinomial
Modelos de Algebra de procesos	+	+	+	N/A
Modelos de redes de Petri	+	-	+	+
autómatas de estados finitos	+	+	+/-	+
Representación de grafos	+	+	+	+/-

Tabla 2.1. Comparación de los modelos formales

Como este trabajo se centra en el emparejamiento de las interfaces que describen el comportamiento de los servicios, y considerando que estas interfaces capturan interacciones desde la perspectiva de una de las organizaciones, consideramos que el modelo de representación de grafos provee una notación madura y simple para expresar la semántica del servicio. Por otro lado, el paralelismo soportado por el modelo de grafos es un aspecto clave del formalismo el cual es requerido para el modelado correcto de la composición de los servicios Web. A pesar de que el emparejamiento de grafos es un proceso complejo, se pueden aplicar técnicas con el fin de incrementar su rendimiento. (Ver [25], [26], [27]). Finalmente el modelo de grafos es utilizado por múltiples propuestas (Ver [24], [28], [23]) como representación formal para procesos de negocios.

2.2. Técnicas de emparejamiento de servicios

El proceso base de la búsqueda de servicios es el emparejamiento, dicho proceso consiste en encontrar una correspondencia entre dos servicios, uno de los cuales es el servicio requerido por el usuario, y el otro es un servicio tomado de un conjunto de servicios disponibles en Internet.

Aunque nuestro trabajo se centra en el descubrimiento de servicios Web geográficos hidrológicos, la mayor parte de los trabajos de investigación sobre descubrimiento de servicios trabajan con servicios genéricos (que no incluyen geoprocetamiento), por otro lado, la mayor parte de estos trabajos pueden adaptarse (con pequeñas modificaciones) al dominio geográfico. Por lo expuesto anteriormente. En esta sección explicamos las técnicas para el emparejamiento de servicios genéricos que se basan en los formalismos para el modelado de procesos, posteriormente se abordan las particularidades del descubrimiento de servicios web geográficos.

Esta sección se explica solamente las técnicas de emparejamiento de servicios basadas en interfaces, semántica y comportamiento. El emparejamiento basado en calidad del servicio se estudiará en la siguiente sección de descubrimiento de servicio web geográficos basado en calidad del servicio, lo anterior se debe a que el descubrimiento de servicios basado en calidad del servicio está fuertemente relacionado con el dominio de aplicación, en este caso el dominio geográfico.

Dado que uno de los aportes de nuestro trabajo es una propuesta de descubrimiento de servicios basada en grafos, primero introducimos los algoritmos de emparejamiento de grafos. A continuación, se presenta el estado del arte de las diferentes técnicas de emparejamiento de servicios. Finalmente, se muestra una comparación de las técnicas existentes.

2.2.1. Algoritmos de emparejamientos de grafos.

El emparejamiento de grafos se ha convertido en un campo muy activo de investigación [29]. Existe un gran espectro de algoritmos para el emparejamiento de grafos con diferentes características disponibles [22]. En su forma mas general, el emparejamiento de grafos se refiere al problema de encontrar un mapeo f entre los nodos de un grafo dado g_1 con los nodos de otro grafo g_2 , que satisfagan algunas restricciones o criterios. Por ejemplo, en la detección de isomorfismo de grafos, el mapeo f debe ser una biyección que preserva todos los nodos y las etiquetas. En la detección de isomorfismo de subgrafos, el mapeo f debe de ser inyectivo, de tal forma que todas las aristas de g_1 estén incluidas en g_2 y todas las etiquetas se preserven. Otros problemas de emparejamiento de grafos que requieren la construcción de un mapeo f con propiedades particulares son la detección del máximo subgrafo común y el cálculo de la distancia de edición de grafos. El algoritmo estándar para detección de isomorfismo de grafos y subgrafos es el de Ullman[30]. En esta propuesta, se presenta un algoritmo simple de búsqueda en árbol basado en un procedimiento de refinamiento, que consigue eficiencia por medio de la eliminación inferencial de nodos sucesores en la búsqueda en árbol. Por otro lado, la detección de subgrafos comunes ha sido trabajada en ([31],[32],[33]) y muchos métodos para el emparejamiento de grafos tolerante a errores han sido presentados en esta literatura. En esta sección presentamos una revisión de las principales propuestas de emparejamiento de grafos.

Los métodos clásicos para emparejamiento de grafos tolerante a errores pueden encontrarse en ([34],[35],[36],[37],[38]). La mayoría de estos algoritmos son versiones particulares del procedimiento de búsqueda A^* (i.e., se basan, en una extensión, a la búsqueda en árbol incorporando diversas técnicas heurísticas con el fin de reducir el espacio de búsqueda).

Adicionalmente, el emparejamiento de grafos por corrección de error es un concepto poderoso que tiene varias aplicaciones en el reconocimiento de modelos y la visión por computador, y su aplicación se centra en entradas distorsionadas. Esto constituye un nuevo acercamiento muy similar a otras técnicas de emparejamiento de grafos. En [39] este tema esta analizado y se propone una nueva medida de distancia sobre grafos que no requiere ningunas operaciones de edición. Esta medida se basa en el máximo subgrafo común entre dos grafos. Una formulación general para algoritmos de isomorfismo de subgrafos por corrección de error se presenta en [40] en términos de grafos de adyacencia, y [41] presenta un estudio sobre la influencia de la definición de funciones de adaptabilidad para el emparejamiento de grafos por corrección de error que revela unas directrices para definir funciones de adaptabilidad para los algoritmos de optimización en el emparejamiento de grafos por corrección de error. Por otro lado, en [42] se presenta un algoritmo de detección de isomorfismo de subgrafos por corrección de error entre un conjunto de grafos modelos y un grafo de entrada desconocido.

En el emparejamiento de grafos aproximado o con corrección de error, se considera un conjunto de operaciones de edición de grafos y define la distancia de edición de dos grafos g_1 y g_2 como la secuencia más corta (o de menor costo) de operaciones de edición que transforman g_1 en g_2 . Un subgrafo máximo común de dos grafos g_1 y g_2 es un subgrafo de dos grafos g_1 y g_2 que es subgrafo tanto de g_1 como de g_2 de tal forma que no existe otro subgrafo de g_1 y g_2 con más nodos. La distancia de edición de grafos y un subgrafo común máximo son conceptos bien conocidos que tienen muchas aplicaciones en reconocimiento de modelos y visión por computador. En [43] se presenta una función de costo particular para la distancia de edición de grafos, y se muestra que utilizando esta función de costos, la computación de la distancia de edición de grafos es equivalente al problema de subgrafos común máximo.

En [44] los autores explican la relación entre dos problemas importantes en el reconocimiento de patrones, utilizando grafos con atributos, el subgrafo común máximo y el subgrafo común mínimo de dos grafos. Esta relación se establece por medio de construcciones simples, que permiten obtener subgrafos comunes máximos desde del subgrafo común máximo y viceversa. Con base en esto, se propone una nueva métrica de distancia de grafos para medir las similitudes de medidas entre objetos representados por grafos con atributos. La métrica propuesta puede ser calculada por una extensión directa de cualquier algoritmo que implemente emparejamiento de grafos por corrección de error, que corra bajo una función de costos apropiada, y la extensión solo toma tiempos lineales a la medida de los grafos.

De otra parte, los grafos conceptuales han sido utilizados para modelar representaciones de conocimiento desde su introducción a principios de los 80s. El formalismo de los grafos conceptuales introducida en [45] en una representación de conocimiento consistente y flexible con unas bases teóricas bien definidas. Los grafos conceptuales simples son considerados como el núcleo de la mayoría de los formalismos de representación de conocimiento construidos sobre el modelo de Sowa [46]. Este formalismo puede capturar la semántica en la representación de datos, y ofrece algunas construcciones útiles que hacen una plataforma posible para un sistema basado en conocimiento. Una extensión de este concepto al emparejamiento de grafos se presenta en [47], donde el razonamiento sobre el modelo de Sowa puede expresarse por un homomorfismo de grafos llamado proyección. Este trabajo presenta una familia de extensiones de este modo, basado en reglas y restricciones, manteniendo homomorfismo como la operación básica. Aparte de este tipo de grafos, el emparejamiento de grafos ha sido propuesto también, desde un punto de vista práctico o teórico, en combinación con subgrafos de condición mínima [48], grafos finitos [49], pesos de un par de grafos [50], grafos medios [51] y aproximaciones de descomposición [52]. Adicionalmente, se analizan diferentes formas de modelos de representación en términos de estructuras de datos simbólicas tales como cadenas, árboles y grafos en [53].

La mayoría de los métodos presentados en esta sección garantizan encontrar la solución óptima, sin embargo, requieren tiempo y espacio exponencial debido a la complejidad NP del problema. Por otro lado, los métodos aproximados o subóptimos, son definidos polinomialmente por el número de pasos de computación, sin embargo pueden fallar en encontrar la solución óptima [25][26][27]. Otras aproximaciones se basan en redes neuronales tales como redes Hopfield [54] o el mapa Kohonen [55] y la propuesta presentada en [56]. El hecho de formular problemas de emparejamiento de grafos complejos como optimizaciones combinatoriales no es nuevo, y muchos trabajos que aplican diferentes técnicas en este campo se pueden encontrar en la literatura. Los algoritmos genéticos son solo un ejemplo de esto [57][58][59]. Sin embargo, en todos estos métodos aproximados cabe la posibilidad de ser ejecutados en un mínimo de tiempo y no encontrar la solución óptima. Las aproximaciones al problema de emparejamiento de grafos con peso utilizando raíces latentes y programación lineal, se proponen en [60] y [61], respectivamente. Como

un caso especial, el emparejamiento de árboles ha sido trabajado en una serie de artículos recientemente [62],[63],[64],[65].

Los árboles de decisiones han sido también aplicados al emparejamiento de grafos, un ejemplo de esto se encuentra en [66], en donde los árboles de decisión se utilizan para resolver el problema del subgrafo común más grande en lugar de aplicar consultas a la base de datos de modelos. Otro ejemplo puede encontrarse en [67] donde los árboles de decisión se aplican como un algoritmo rápido para el cálculo de isomorfismos de grafos por corrección de error. Los árboles de decisión también han sido aplicados en el campo del emparejamiento de grafos [68]. El árbol de decisión se crea utilizando un conjunto de grafos con modelos conocidos a priori generados a partir de la detección de isomorfismo de subgrafos.

Podemos ver que las correspondencias entre grafos pueden establecerse por una variedad de paradigmas de relación de grafos. Los paradigmas más utilizados hoy en día incluyen detección de isomorfismo de grafos y subgrafos para aproximaciones de emparejamiento de grafos exactos. Sin embargo, en nuestra aproximación de emparejamiento de servicios, los usuarios están interesados en encontrar servicios apropiados. De esta forma, el usuario formula un requerimiento como un modelo de procesos; su objetivo es utilizar este modelo como una consulta para obtener un componente cuyo modelo de proceso concuerde totalmente o con una parte de la consulta. De esta forma, es necesario incorporar los conceptos de corrección de error y emparejamiento inexacto en emparejamiento de servicios.

El término emparejamiento aproximado significa que no es posible encontrar un isomorfismo entre los dos servicios a emparejar. Este es el caso en el que el número de actividades es diferente entre la consulta y el servicio objetivo. Esto se puede deber al aspecto semántico del servicio y a la dificultad de segmentar apropiadamente el servicio en entidades significativas. De esta manera, en estos casos no se puede esperar un isomorfismo entre los dos servicios, y el problema de emparejamiento de servicios no consiste en buscar la forma exacta de emparejar actividades de un servicio con el de otro, sino encontrar el mejor emparejamiento entre ellos. En este caso, el emparejamiento ayuda a encontrar un correspondencia no biyectiva entre un servicio consulta y un servicio objetivo.

La corrección de error es la definición de los errores que deben tenerse en cuenta en el proceso de emparejamiento de servicios. Probablemente el mejor modelo de corrección de error conocido para procesos de emparejamiento es similar al modelo utilizado en la distancia de edición de cadenas. Este se basa en la idea de introducir operaciones de edición para cada tipo de error posible. Con el fin de modelar el hecho de que ciertos tipos de error son más probables que otros, se asignan funciones de costos a las operaciones de edición. Basándose en las consideraciones previas, creemos que la detección de isomorfismo de subgrafos por corrección de error es la técnica de emparejamiento más apropiada para nuestro problema de emparejamiento de servicio. En el capítulo 3 explicaremos en detalle esta técnica de emparejamiento.

2.2.2. Emparejamiento de Servicios

Esta sección describe las técnicas utilizadas para el emparejamiento de servicios. Hemos agrupado las técnicas existentes en cuatro conjuntos: emparejamiento de servicios basados en interfaces, semántica, comportamiento y calidad del servicio.

Emparejamiento de servicios basado en interfaces. Actualmente, los algoritmos para el descubrimiento de servicios Web en registros como UDDI o ebXML se basan en la búsqueda de

palabras claves o tablas de correspondencia de parejas (clave-valor). Para conseguir un proceso de descubrimiento de servicios más preciso, se han propuesto mecanismos basados en emparejamiento de interfaces (i.e. [69],[70]).

Tales propuestas de emparejamiento de servicios Web tienden a dirigir el emparejamiento semántico y/o sintáctico. Para analizar sus méritos, es útil clasificarlos en uniformes o híbridos. Los acercamientos de emparejamiento uniformes se refieren a técnicas de emparejamiento atómicas que no pueden descomponerse en técnicas de emparejamiento más pequeñas. Los acercamientos de emparejamiento híbridos de otro lado, combinan varios métodos (e.g., sintáctico y semántico) en un algoritmo compuesto. En [70] los autores discuten un conjunto de métodos complementarios para evaluar la similaridad de las especificaciones de las interfaces (WSDL), permitiendo ordenar los servicios potencialmente útiles de acuerdo a su relevancia con la consulta del desarrollador. Para calcular la similaridad entre dos especificaciones WSDL, estos métodos utilizan, por un lado, la semántica de los identificadores y de las descripciones en lenguaje natural de las especificaciones WSDL, y por otro lado, la estructura de sus operaciones, mensajes y tipos. Más precisamente, los autores describen un conjunto de métodos de descubrimiento de servicios Web que combinan la obtención de información tradicional con dos técnicas basadas en Wordnet con un algoritmo de emparejamiento de estructuras utilizando la estructura de la especificación del servicio basada en WSDL.

En [69] los autores presentan el algoritmo *WSDL-M2* que combina dos técnicas: emparejamiento léxico para calcular la similaridad lingüística entre descripciones de conceptos, y el emparejamiento estructural para evaluar la similaridad global entre conceptos compuestos. El proceso de emparejamiento global se implementa en tres pasos: Primero, todos los archivos de la colección de las especificaciones WSDL se procesan con el fin de permitir la extracción de su contenido estructurado. En el segundo paso, el documento procesado es etiquetado para permitir un análisis léxico; en el tercer paso, las especificaciones WSDL etiquetadas pueden ser analizadas adicionalmente y posteriormente catalogadas utilizando diferentes modelos de recuperación de información (V SM: Modelo Vector-Espacio y medida tf – idf). Para solucionar el mayor problema de V SM, se propone una combinación de V SM y WordNet. Finalmente, el emparejamiento estructural se trata como un emparejamiento bipartito de peso máximo para calcular la similaridad semántica entre descripciones de conceptos y para computar similaridad de conceptos WSDL complejos teniendo en cuenta sus componentes.

En otro frente, recientemente ha habido una proliferación de motores de búsqueda de servicios Web en la Internet, estos se pueden agrupar en dos tipos: el primer tipo acepta como entradas, palabras claves, que ellos utilizan para buscar dentro de las descripciones WSDL ofrecidas por los servicios. Algunos ejemplos de este tipo son: Bindingpoint, NET XMLWeb Services Repertory, WebserviceX.NET, Web Service List y SaICental (ver [71]). El segundo tipo de motores de búsquedas de servicios Web va más allá del emparejamiento simple de palabras de los contenidos WSDL ejecutando una búsqueda de similaridad sobre las operaciones WSDL de los servicios Web, considerando nombres de las operaciones y parámetros de entrada y salida. Un ejemplo que utiliza dicha técnica para encontrar servicios Web es Woogole (ver [72]). En [72] el motor de búsqueda combina múltiples fuentes de evidencia para detectar similaridad: descripciones textuales de las operaciones y de los servicios Web y similaridad entre los nombres de los parámetros de las operaciones. El algoritmo subyacente se basa en una técnica que agrupa los nombres de los parámetros de una colección de servicios Web en conceptos con significado semántico. Estos conceptos se usan en la comparación de parámetros de entrada y salida. Resumiendo, las propuestas utilizadas por los motores de búsqueda de servicios Web solo pueden hacer corresponder servicios simples, dejando de lado los aspectos de ejecución de los servicios.

2.2.3. Emparejamiento de servicios basado en semántica.

Dentro del marco de trabajo de la Web semántica, se proponen lógicas de descripción para una descripción formal precisa y más rica de los servicios. Estos lenguajes permiten la definición de ontologías, tales como OWL-S. OWL-S [73] propone una ontología para describir servicios Web basados en el lenguaje de ontologías para la Web (OWL) [74]. OWL-S está estructurado en tres tipos de conocimiento: perfiles de servicio, modelo de servicio y “grounding” de servicios. Los perfiles de servicios describen la capacidad de un servicio Web. El modelo de servicio describe los servicios en términos de entradas, salidas, precondiciones, y efectos de invocar el servicio; los procesos en OWL-S se describen en términos de sus estados, incluyendo información tales como activación inicial, ejecución y completitud.

Por medio de la descripción de las capacidades utilizando OWL-S, es posible encontrar correspondencia de servicios Web desde una perspectiva semántica. Muchos prototipos de emparejamiento de servicios Web han sido implementados utilizando esta aproximación, por ejemplo ([75],[76],[77],[78],[79],[80]). El trabajo relacionado con esto puede encontrarse en ([81],[82]), donde se describen propuestas para comentar servicios Web con información semántica y utilizar esto para el descubrimiento de servicios. En [79] y [77], un servicio publicado es correspondido con un servicio requerido cuando las entradas y las salidas del servicio requerido corresponden con las entradas salidas del servicio publicado (i.e. tienen el mismo tipo o uno es una generalización del otro). En [83], se definen filtros independientes para recuperación de servicios: el espacio de nombre, la descripción textual, el dominio de ontología que es utilizado, tipos de entradas/salidas y restricciones.

La propuesta presentada en [84] tiene en cuenta las propiedades operacionales como tiempo de ejecución, costo y confiabilidad. Los autores de [85] proveen una comparación semántica ligera de interfaces basada en métodos de evaluación de similaridad (léxico, atributos, interfaces y similaridad de QoS).

Las aproximaciones basadas en semántica mencionadas anteriormente son muy eficientes para el emparejamiento de servicios simples basados en descripciones semánticas de sus capacidades. Sin embargo, no está claro como estas aproximaciones permitirían emparejar los procesos de negocios complejos, que incluyen la información estructural de los procesos incluidos en los servicios.

2.2.4. Emparejamiento de servicios basados en comportamiento.

La recuperación de servicios basada en palabras claves o en atributos semánticos no es satisfactoria para un gran número de aplicaciones. La tendencia de los trabajos recientes es explotar más y más conocimiento sobre componentes del servicio y su comportamiento. La necesidad de tener en cuenta el comportamiento de los servicios descritos por un modelo de procesos es resaltado por muchos investigadores [[78],[80],[86],[87],[88]]. En [78], con el fin de mejorar la precisión del descubrimiento de servicios Web, se utiliza el modelo del proceso para capturar el comportamiento visible de un servicio, se define además un lenguaje de consulta para servicios que permite encontrar servicios especificando condiciones sobre las actividades que los componen, las excepciones tratadas y el flujo de los datos entre las actividades.

Recientemente, han aparecido propuestas que discuten la similaridad y compatibilidad a diferentes niveles de abstracción de una descripción de servicios (e.g., [72],[88],[89],[90]). En términos de especificación y análisis del protocolo, las aproximaciones existentes proveen modelos (e.g.,

basados en pi-calculus, redes de petri o maquinas de estados) y mecanismos para comparar especificaciones (e.g., chequeo de compatibilidad de protocolos).

En [91], los autores presentan una propuesta que permite un enlace dinámico para procesos BPEL (enlace dinámico de Servicios Web a instancias de flujos de servicios Web en tiempo de ejecución, i.e. la habilidad de intercambiar una instancia de WS participando en una instancia de flujo de WS por otra alternativa) esto se hace en tres pasos. Primero, proveyendo una descripción de alto nivel del proceso, segundo, abstrayendo el comportamiento del proceso utilizando grafos de observación simbólica (SOG) usando redes de workflow (Wf-nets) el cual es una forma específica de redes de Petri y finalmente, proveyendo un algoritmo para emparejamiento SOG usado para enlazar dinámicamente procesos de negocios.

En [92], se utiliza una representación de pi-calculus para formalizar un comportamiento de servicios y consulta. Específicamente, las operaciones simples que involucran intercambio de mensajes se expresan en pi-calculus, diferenciados por cuatro tipos de transmisiones: de una vía, notificación, solicitud-respuesta y petición-respuesta. También se expresan en pi-calculus las restricciones entre operaciones de un servicio o una consulta que definen el orden permitido de ejecución. Después de expresar el comportamiento del servicio y la consulta utilizando pi-calculus, el emparejamiento de servicios entre un servicio consulta y la descripción de un servicio es calculado utilizando las capacidades de pi-calculus. En [88], los autores ofrecen una semántica formal para el emparejamiento de procesos de negocios basada en autómatas de estados finitos extendidos por expresiones lógicas asociadas a estados.

El cálculo de la intersección es costosa computacionalmente hablando y no es aplicable para repositorios grandes de servicios. Para resolver este problema los autores de [93] presentan una aproximación de indexación para consultas cíclicas de procesos de negocios utilizando sistemas de bases de datos tradicionales. El uso de autómatas de estados finitos como formalismo de modelado limita la expresividad de los modelos, por ejemplo al representar las capacidades de ejecución paralelas, puede conducir a modelos demasiado grandes.

Un nuevo modelo de comportamiento para servicios Web se presenta en [94] que asocia los mensajes intercambiados entre los participantes con actividades ejecutadas dentro de un servicio. Los perfiles de actividad son descritos utilizando OWL-S (lenguaje de ontologías de Servicios Web). Los servicios Web son modelados como autómatas finitos no determinísticos y se desarrolla un nuevo lenguaje de consulta para expresar propiedades semánticas y temporales sobre los comportamientos del servicio.

2.2.5. Comparación de las técnicas de emparejamiento de Servicios

La tabla 2.2 resume las técnicas de emparejamiento presentadas en esta sección. El símbolo (+) significa que la propiedad en esa columna esta soportada por la técnica de emparejamiento. El símbolo (-) significa que la propiedad no está soportada. Finalmente el signo (+/-) significa que la propiedad esta soportada moderadamente.

Los parámetros evaluados para cada técnica son: mantiene estado, no mantiene estado, semántica. Mantiene o no estado, son propiedades que describen si un computador o programa de computador esta diseñado para advertir y recordar uno o mas eventos precedentes en una secuencia dada de interacciones con un usuario, otro computador o programa, un dispositivo, u otro elemento externo. Con estado significa que el computador o programa mantiene el rastro del estado de la interacción, usualmente fijando valores en un campo de almacenamiento diseñado para ese propósito. Sin estado

significa que no hay registro de las interacciones previas y cada solicitud de interacción debe ser manejada basada enteramente en la información que viene con ella.

En la tabla 2.2 se puede ver que las técnicas de emparejamiento de servicios basadas en interfaces, no mantienen estado y soportan medianamente la descripción de los servicios semánticos; dado que los motores de búsqueda solo pueden emparejar servicios simples, de igual forma no manejan los aspectos de proceso de los servicios. Finalmente, las descripciones semánticas están soportados medianamente, como en el caso del motor de búsqueda Woogole (ver [72]) que tiene un nivel de evaluación semántica.

La técnica de emparejamiento de servicios basada en semántica soporta solamente descubrimiento sin mantener el estado, así como descripciones semánticas. Debido a que tal técnica no soporta servicios con estado, es muy eficiente en encontrar servicios simples por emparejamiento de las descripciones semánticas de sus capacidades, sin embargo, no esta claro como traducir este para emparejar procesos de negocios complejos para chequear por colaboraciones bilaterales o multi laterales.

Parámetros de comparación	Técnicas de emparejamiento de servicios			
	Basado en Interfaces	Basada en Semántica	Basada en Comportamiento	Basad en Calidad del Servicio
No mantiene estado	+	+	+	+
Mantiene estado	-	-	-	-
Semántica	+/-	+	+/-	+/-
Calidad	-	-	-	+

Tabla 2.2. Comparación de las técnicas de emparejamiento

Finalmente, aunque la técnica de emparejamiento de servicios basada en comportamiento soporta medianamente la descripción semántica, algunas aproximaciones como [94] insertan semántica al emparejamiento de comportamientos. De otro lado, tal técnica soporta el mantenimiento del estado de las descripciones formales de los servicios, de forma que estas técnicas de emparejamiento tienen en cuenta el orden en el cual los servicios son ejecutados.

En vista de que nuestra propuesta esta dirigida al descubrimiento de servicios Web geográficos, a continuación se presentan las propuestas para el descubrimiento de Servicios Web geográficos basado en calidad del servicio, y se presenta una selección de los criterios de calidad de servicio más importantes en el área de los Servicios Web geográficos

2.3. Descubrimiento de Servicios Web Geográficos basado en Calidad del Servicio

Los SIG requieren generalmente de datos conseguidos con grandes inversiones para su levantamiento, por parte de los gobiernos y las instituciones. Por esta razón, la investigación en esta área se ha orientado a encontrar estrategias de reuso de la información existente[6], sin embargo, los esfuerzos por compartir las capacidades de procesamiento geográfico (o geoprocusamiento) de forma estándar están aún emergiendo lentamente. Afortunadamente, la gran difusión del paradigma de la Arquitectura Orientada al Servicio, SOA y de los Servicios Web, ha contribuido a la aparición de servicios Web geográficos (servicios de geoprocusamiento accesibles a través de Internet) [4]. Los Servicios Web son aplicaciones software modulares y auto descriptibles, que puede ser

identificadas por una URL; estos pueden ser descritos, descubiertos y localizados a través del Internet usando estándares abiertos como SOAP y XML. Este modelo permite la interoperabilidad de servicios en redes heterogéneas como Internet proporcionando bajo acoplamiento e independencia de la plataforma. Debido a lo expuesto anteriormente, el concepto de Servicios Web ha recibido mucha atención últimamente, como un medio para la integración de SIG utilizando la Internet, al proveer un marco de trabajo (framework) interoperable e independiente del proveedor con capacidades para el descubrimiento, acceso, integración, análisis y visualización de múltiples fuentes de datos espaciales, información de sensores y capacidades de procesamiento espacial. ([2],[3],[4]).

Por otro lado, también hemos presenciado la aparición de diversas propuestas para la operación en conjunto de sistemas y servicios geográficos promovidas por organizaciones de estandarización más importantes en el área: El OGC (Open Gis Consortium) [96] y el comité técnico TC211 de la ISO (International Organization for Standardization) [97]. La especificación WMS 1.0 (Web Map Server), es la primera iniciativa que tiene el objetivo de llevar el servicio de mapas a Internet de forma estandarizada. Previamente se habían desarrollado algunas especificaciones para la integración de servicios de geoprocésamiento, que funcionan en las plataformas de computación distribuida como DCOM [98] y CORBA [99].

El resto de esta subsección se organiza de la siguiente forma: primero se exponen los conceptos sobre los servicios Web geográficos (o servicios de geoprocésamiento), y posteriormente, se realiza una revisión de todas las propuestas relacionadas con descubrimiento de servicios Web geográficos basado en calidad del servicio.

2.3.1. Servicios Web Geográficos.

Con la gran difusión del paradigma de la Arquitectura Orientada al Servicio, SOA y de los Servicios Web, se popularizaron también los servicios y aplicaciones Web que posibilitan acceder a los servicios geográficos a través de Internet [100]. Los Servicios Web ofrecen una gran posibilidad de compartir geoprocésamiento de forma estándar, en vez de información “cruda” a nivel de datos. En este contexto apareció el concepto de servicios Web geográficos (también conocidos como OGC (Open Geospatial Consortium) Web Services u OWS definidos formalmente por el OGC como un servicio cuyos objetivos son: proveer acceso a metadatos que describen servicios, y descubrir metadatos que permitan acceder a dichos servicios, de manera que se faculte el desarrollo independiente de Servicios Web en el ámbito de los Sistemas de Información Geográfica. Actualmente, la OGC trabaja en recomendaciones para asegurar la operación conjunta de Servicios Web Geográficos en un proyecto denominado OWS (OGC Web Services) adaptando al dominio geográfico estándares como SOAP (Simple Object Access Protocol) y WSDL (Web Services Description Language) de la W3C (World Wide Web Consortium) [101].

El Modelo General de Servicios de la OGC [96] resume el conjunto de estándares y tecnologías para la construcción y despliegue de los servicios Web geográficos. En el modelo los niveles bajos de la torre de protocolos están orientados a facilitar la conectividad de los componentes software permitiendo establecer comunicaciones y transmitir mensajes (protocolos de comunicación como TCP/IP, HTTP, SMTP, IIOP Y FTP), mientras que los niveles altos, están orientados a la interoperabilidad por lo que definen mecanismos de publicación, búsqueda y enlace, permitiendo a los componentes software trabajar juntos de forma integrada y transparente. Los servicios de alto nivel, se resumen en la Tabla 2.3:

Nombre	Funcionalidad
Web Map Service (WMS)	Creación de mapas sobre la Web a partir de formato vector o raster. Los mapas generados pueden ser visualizados en navegadores comunes.
Web Feature Service (WFS)	Acceso de datos en formato vector basado en la Web. Su respuesta está dada en GML, y debe ser procesada por una herramienta GIS (Web o Desktop).
Web Coverage Service (WCS)	Acceso de datos en formato raster basado en la Web. La respuesta es entregada en formatos de imágenes como TIFF, GIF, JPEG, BMP, PNM.
Web Catalog Service (CSW)	Servicio de catalogo basado en la Web para administración y consulta de metadatos que describen datos y servicios geográficos. Las consultas se basan en criterios espaciales y textuales.
Web Gazetter Service (WFS-G)	Servicio que permite referenciar entidades geográficas mediante identificadores textuales (nombres de lugares).
Web Coordinate Transformation service (WCTS)	Permite realizar transformación de un sistema de coordenadas geográficas a otro a través de la Web.

Tabla 2.3. Servicios de alto nivel del Modelo General de Servicios de la OGC

Estos servicios, son la base para el desarrollo de las IDE (infraestructuras de Datos Espaciales) [102], el trabajo alrededor de las IDE ha sido acogido por organizaciones e instituciones gubernamentales y privadas, debido a que su objetivo principal es integrar a través de Internet los datos, metadatos, servicios e información de tipo geográfico que se producen a nivel nacional, regional y local, facilitando a todos los usuarios potenciales la localización, identificación, selección y acceso a tales recursos[103]. Uno de los elementos neurálgicos de la IDE, es el desarrollo de un servicio de catalogo (Catalog Services for the Web -CSW- en términos de la OGC) que permite almacenar metadatos compatibles con las especificaciones ISO 19115 (metadatos de datos geográficos) e ISO 19119 (metadatos de servicio). Esta especificación, que aun está en trámite para ser aprobada, constituye la base y el futuro del sistema de descubrimiento de los servicios geográficos. Teniendo en cuenta que nuestra propuesta se centra en la búsqueda de servicios de geoprocetamiento, el conocimiento de la especificación del catálogo de servicios geográficos es fundamental para hacernos una idea de la forma de realizar las búsquedas de forma estándar. A continuación se profundiza en esta especificación.

El servicio de catalogo descrito por la OGC [108], se fundamenta en estructuras de metadatos, relaciones y definiciones (esquemas conceptuales) con el propósito de homogenizar el intercambio de información dentro de una comunidad. Un esquema de metadatos provee un vocabulario común que soporta la búsqueda, recuperación, despliegue y asociación entre la descripción y el objeto que está siendo descrito. Este trabajo de estandarización, está a cargo del Comité Técnico 211 de la ISO e incluye:

- Un esquema formal para metadatos geográficos, el estándar de metadatos ISO 19115:2003, el cual define una propuesta de los elementos principales de uso común.
- El borrador de la especificación técnica 19139 que define una codificación formal XML
- Una estructura para intercambio de metadatos.

El servicio de Catálogo es una especificación muy reciente por lo que existen variadas propuestas de implementación, como por ejemplo: GeoNetwork [105] o Deegree [106]. Aunque cada propuesta puede variar en cuanto a técnicas, tecnologías o frameworks que utilizan, todas mantienen un esquema arquitectural básico conocido como Modelo General para la Interfaz de Catalogo.

En la figura 2.1 se presenta el diagrama de clases del Modelo General para la interfaz del Catálogo, que muestra las diferentes interfaces que el servicio de catalogo OGC debe implementar y las

relaciones que existen entre ellas. En total son seis clases, de las cuales por lo menos dos deben ser implementadas obligatoriamente, las demás permiten ampliar las funcionalidades del catálogo pero no son una camisa de fuerza, permitiendo de esta forma diferentes instancias de catálogo de acuerdo a los requerimientos del mismo.

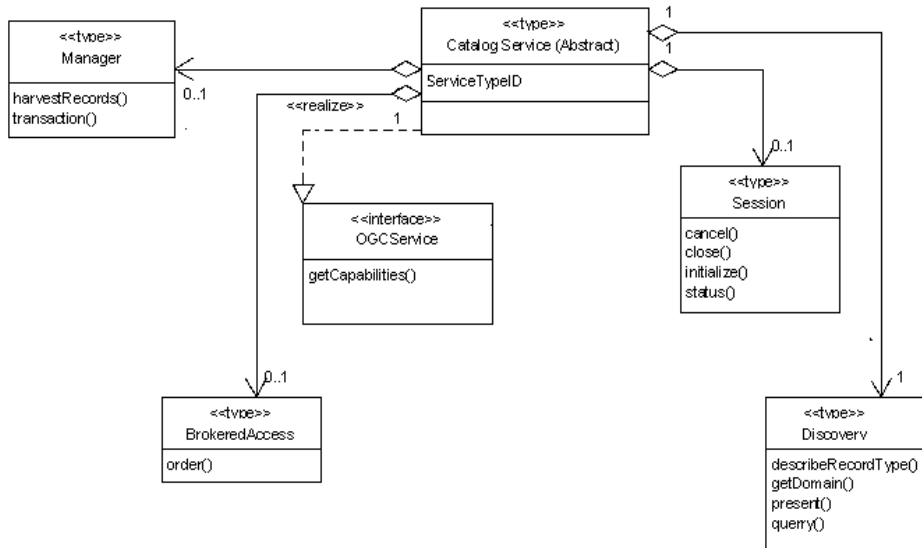


Figura 2.1 Diagrama de Clases del Modelo General del Catálogo OGC.

Una instancia de la interfaz “*Catalogue Service*” es un objeto compuesto que representa las funciones del catálogo a alto nivel. A su vez los objetos que lo componen proveen funcionalidades en áreas específicas, de manera que es posible realizar configuraciones al catálogo que sirvan para diferentes propósitos, por ejemplo: descubrimiento (solo lectura), catálogo transaccional, soporte y gestión de sesiones (stateful), etc. La clase “*Catalogue Service*” puede asociarse con:

- La clase “*OGC Service*”, que provee la operación `getCapabilities`, la cual devuelve el metadato del servicio de catálogo. Esta clase siempre se encuentra implementada debido a la importancia de ofrecer información sobre el servicio de catálogo al usuario.
- Clase “*Discovery*”, provee las operaciones para el descubrimiento de recursos registrados en un catálogo.
- La clase “*Session*” provee operaciones para el manejo de sesiones interactivas entre un cliente y el servidor.
- La clase “*Manager*” provee operaciones para insertar, actualizar y borrar metadatos de los recursos en el catálogo.
- La clase “*Brokered Access*”, provee la operación “orden”, que permite ordenar un recurso registrado en el catálogo.

Por otro lado, el modelo general para la interfaz del catálogo ofrece un conjunto de interfaces abstractas de servicio que soportan el descubrimiento, acceso, mantenimiento y organización de catálogos de información geoespacial y recursos relacionados. Las interfaces declaradas por el modelo pretenden que un usuario o aplicación software encuentre información presente en ambientes de computación distribuida, incluida la Web. Se conoce como Perfil de Aplicación a la relación del catálogo con el protocolo de enlace utilizado en la especificación base. Un perfil puede referirse a CORBA, Z39.50 [107] o HTTP de acuerdo con el protocolo de enlace utilizado. Cada Perfil de aplicación incluye un mapeo de las interfaces generales, operaciones y parámetros

específicos con una serie de restricciones o refinamientos a nivel de implementación del modelo general acordado [108].

2.3.2. Descubrimiento de OWS basado en QoS

El término QoS es usado ampliamente en varias disciplinas y tiene como premisa común la satisfacción del usuario, lo que claramente ofrece varias perspectivas entre las que se incluyen criterios de calidad cuantitativos y cualitativos relacionados con los procesos que se llevan a cabo en la operación y descubrimiento de servicios. La ITU-T [109] define la calidad de servicio como “el efecto combinado (de las características) de desempeño del servicio que determina el grado de satisfacción del usuario del mismo”. Franken [110] se centra en el usuario: “es el desempeño percibido por el usuario”, la ISO [111] es más específica y la define como “el conjunto de requerimientos de calidad en el comportamiento colectivo de uno o más objetos, que puede especificarse en un contrato o ser reportado después de un evento”. La calidad de Servicio a nivel de aplicación es el “Talón de Aquiles” de los servicios desplegados en Internet, debido a la complejidad implícita de la arquitectura multicapa y la naturaleza dinámica del modelo de composición de servicios promovido por SOA y los Servicios Web [112]. La composición dinámica requiere que consumidores de servicios descubran proveedores que cumplan apropiadamente sus requerimientos funcionales y no funcionales como costo, desempeño, confiabilidad, disponibilidad, seguridad, reputación, capacidad, veracidad entre otros. ([113],[114]).

La QoS está ligada al proceso de descubrir y componer servicios. En este sentido existe un modelo conformado por tres patrones arquitecturales, definidos según el grado de control que ejercen los usuarios sobre el proceso de ejecución y composición de servicios [115], estos son: transparente o definido por el usuario: donde el usuario define y controla el orden y la ejecución de los servicios; traslúcido o gestionado por Workflow: que permite la configuración del encadenamiento y la ejecución de servicios mediante flujos de trabajo (interfaces gráficas); opaco o servicios agregados: El usuario tiene control limitado sobre el encadenamiento y la ejecución de los servicios. En los tres patrones, la calidad de servicio tiene enfoques diferentes, así por ejemplo, mientras en los niveles transparente y traslúcido el usuario puede determinar con mayor autonomía los servicios con las características de calidad que requiera, en el nivel opaco esta responsabilidad se delega a la máquina.

Por otro lado, existen distintos niveles para la especificación de los atributos de QoS, el nivel de Usuario se refiere a necesidades en términos de atributos cuantificables, encaminadas a satisfacer el desempeño solicitado por un usuario [116]. El Nivel de Aplicación, se refiere a requerimientos no funcionales tendientes a cumplir especificaciones de calidad solicitadas [117]. El Nivel de Recurso relaciona aspectos de calidad de datos, capacidad de la arquitectura [117] y de medios de comunicación ([118], [117]). Adicionalmente en [119] se plantea la necesidad de definir mecanismos para mapear atributos desde distintos niveles de resolución con el objetivo de relacionar la capa de aplicación con las de transporte y recurso.

En el campo de los servicios geográficos, el estudio de la calidad del servicio, típicamente se ha asumido a nivel de acceso a datos ([108], [120], [121], [122]), generalmente en arquitecturas cliente servidor ([125]) sin embargo, en todos estos trabajos se menciona como trabajo futuro la identificación de atributos de calidad de los servicios para mejorar la experiencia de usuario. Este cambio de paradigma es descrito por [118] donde además se define la calidad como cualquier característica o propiedad adscrita a un Servicio Web Geográfico con el propósito de describirlo, pasando de lidiar con la calidad medida a la percibida. Según [118] la valoración de la calidad de un

servicio se realiza no sólo en términos de medidas de sus atributos, sino también por las cualidades percibidas por los usuarios.

Una de las propuestas más importantes en el área del descubrimiento de servicios geográficos basado en calidad del servicio es la especificación ISO 13236 [123], la cual ofrece un modelo de calidad que define funciones de gestión QoS y establece un marco de trabajo abstracto para QoS utilizado por [124] para definir un metamodelo de calidad en geoprocesamiento y por [125] para definir funciones de monitoreo, control y administración de atributos QoS en medio de un sistema de información denominado CITE (Compliance & Interoperability Testing & Evaluation). En [124] se propone un modelo de calidad basándose en dos niveles de abstracción. Un nivel de usuario donde se definen atributos de calidad cuantificables deseables en la interacción o necesarios para satisfacer al Usuario, y un nivel de infraestructura que es genérico, de manera que diversas abstracciones pueden ser mapeadas desde el nivel de recurso o transporte.

El modelo de Ran [126] propone crear un nuevo rol en el esquema Proveedor-Consumidor-Registro denominado Certificador. La función de este rol es la de verificar que la información de calidad de un servicios sea ingresada antes de ser registrado. Esto previene el ingreso de información inválida en la fase de registro, lo que asegura el desempeño en la fase de búsqueda. En [97] se propone un modelo de calidad para gestionar atributos de calidad y precio, el objetivo es clasificar los atributos de calidad en dos grupos denominados: estático y dinámico. Los que son considerados como estáticos (e.g. precio), son incluidos en el registro UDDI directamente, mientras que los dinámicos son direccionados en tiempo de ejecución mediante un intermediario (*broker*) de información. En [127] se propone una ontología y un agente para realizar la selección dinámica de servicios, para realizar esta selección la calidad del servicio se determina de forma colaborativa por los consumidores con ayuda de un agente. Por otro lado, en [128] se presenta una ontología DAML-QoS complementaria a DAML-S para proveer un modelo de Calidad; de esta forma, las métricas, limitaciones y requerimientos son expresados de forma muy precisa. En [129] se argumenta que para cumplir con las necesidades específicas incluidas en el encadenamiento de servicios, la calidad de servicio permite discriminar alternativas de servicios con funcionalidades idénticas o parecidas con base en Criterios de Calidad definidos, para esto se propone un componente denominado *broker* o facilitador del catálogo de servicios, utilizado ampliamente como patrón en sistemas distribuidos [130].

Resumiendo, La mayoría de las propuestas relacionadas con el emparejamiento basado en calidad del servicio, vislumbran la necesidad de una capa o módulo adicional para el proceso de descubrimiento (broker, agente, certificador), encargado de verificar el ingreso de los parámetros de calidad del servicio y complementar las consultas basándose en estos parámetros. Igualmente, las aproximaciones en este campo, proponen la adición de ciertos parámetros asociados con la calidad del servicio (e.g. precio, tiempo de respuesta) a las descripciones básicas de los servicios (e.g las especificadas utilizando WSDL) o de los procesos (e.g las especificadas utilizando BPEL). Este hecho hace que el proceso de realizar emparejamiento basado en calidad del servicio pueda implementarse como una capa sobre los demás niveles de descubrimiento (como las propuestas de emparejamiento descritas en la sección 2.1), razón por la cual, es muy común que se conjugue el emparejamiento basado en calidad del servicio con otros tipos de emparejamiento, para formar acercamientos de emparejamiento de tipo híbridos.

En el presente trabajo argumentamos, que el proceso de descubrimiento de servicios puede mejorarse teniendo en cuenta el comportamiento y que la adición de una fase de emparejamiento basado en calidad del servicio puede mejorar enormemente la calidad de los resultados en la selección de los servicios. Además, atendiendo a que el emparejamiento de servicios utilizando criterios de calidad del servicio puede especificarse tanto a nivel de servicio como de proceso, esta

descripción permite realizar un emparejamiento de servicios o de procesos comparando directamente el proceso consulta con el modelo, resultando un proceso de comparación con costos computacionales bajos.

2.4. Modelado de procesos de gestión ambiental del recurso hídrico

2.4.1. El metamodelo como mecanismo de abstracción

Los procesos de gestión ambiental del recurso hídrico comprenden un amplio conjunto de análisis, que inician con un modelado de la realidad, posteriormente, incluyen la selección de una herramienta metodológica que permite determinar el estado actual de un recurso natural, dichas herramientas emplean formulas matemáticas para el calculo de valores numéricos conocidos como indicadores. La evaluación cuantitativa de estos indicadores permite tomar decisiones sobre el estado de un recurso; por ejemplo, con indicadores como el Índice de calidad del agua ICA y el Biological monitoring working party BMWP, se puede conocer la calidad y la escasez de agua en una zona geográfica determinada. (Explicados posteriormente en esta sección).

El cálculo de indicadores como los presentados en el párrafo anterior se encuentra definido por diversas herramientas metodológicas, las cuales agrupan una gran cantidad de variables, entre las que encontramos: las características climatológicas de la región y la topografía de la zona. En este contexto, se presentan para las instituciones encargadas de la gestión ambiental dos graves problemas [1]:

- La falta de consenso (local, regional y nacional) sobre los criterios y métodos a utilizar (dada su gran variedad) para el análisis del recurso hídrico. Por lo tanto, es muy común que aparezcan diversas perspectivas acerca de la calidad o la abundancia de agua en una región, dependiendo del tipo de indicador utilizado para el análisis. Dichas discrepancias van claramente en detrimento de la gestión eficaz de los recursos naturales y, en particular, del recurso hídrico.
- El segundo es la necesidad de las instituciones encargadas de la gestión ambiental a nivel nacional (como el Instituto de Hidrología, Meteorología y Estudios Ambientales de Colombia IDEAM) o a nivel regional (como la Corporación autónoma regional del Cauca CRC en el Cauca) de contar con un soporte informático para diferentes metodologías de análisis ambiental. Lo cual acarrea enormes gastos asociados con la compra de licencias software, y con los procesos de capacitación en cada una de las plataformas o sistemas que soportan dichas metodologías. Esta situación aparece comúnmente cuando se quiere analizar una zona geográfica nueva (con características ambientales particulares) o se intenta asumir las directrices de una institución gubernamental como el IDEAM en lo relacionado con las metodologías para el cálculo de indicadores.

Por medio de la definición de un metamodelo se puede lograr una abstracción del proceso de gestión ambiental que ofrezca una correcta y eficiente relación lógica entre los estudios ambientales y la manera como estos pueden ser soportados por herramientas tecnológicas. La importancia de los metamodelos como herramienta de abstracción, han sido resaltada en diversas áreas como la educación [131][132], el desarrollo del software [133], e inclusive en el área de los negocios estratégicos [134]; ya que permiten guardar el conocimiento y reutilizar gran parte de las soluciones abstraídas. En el ámbito de la gestión ambiental el metamodelo permitirá la definición de arquitecturas software para la gestión ambiental distribuida y el desarrollo de herramientas que soporten una gestión integral de los recursos ambientales.

Un modelo es una simplificación de un sistema construido con un fin específico en mente [163], el modelo debe ser capaz de resolver algunas preguntas en lugar del sistema real. En [135] se afirma que un modelo es una descripción de un sistema (o una parte de él) escrito en un lenguaje bien definido. Una discusión mas profunda sobre todas las definiciones y las características de un modelo se puede encontrar en [136][137]. Por su parte un metamodelo es un modelo que define el lenguaje para expresar un modelo [138], y también se puede definir como un modelo de modelos [139]. Hoy en día las principales organizaciones dedicadas a la definición de estándares en el área de la telemática, como el IEEE (Institute of Electrical and Electronics Engineers) y la OMG (Object Management Group) enfatizan la necesidad de establecer las reglas adecuadas (metamodelos) para modelar las nuevas arquitecturas de software empresariales; dichas arquitecturas dependen de elementos tecnológicos que soportan tanto sus procesos productivos como administrativos.

En [132] se argumenta que el gran limitante de un modelo es la especificidad de la representación de una realidad con respecto a la visión compartida de una persona o grupo de personas, que hace a los modelos estáticos y poco modificables; por otro lado, un metamodelo define elementos esenciales de un conjunto de modelos, y no las características específicas que soporta el modelo, permitiendo definir las estructuras y elementos esenciales para armar cualquier modelo. Así se pueden generar metamodelos para un entorno de la realidad y de allí se podría definir modelos más específicos para las necesidades de los diferentes entornos posibles y los aún los no creados. El hecho de poder crear un modelo a partir de un Metamodelo, no es esencial para el desarrollo de un modelo, pero si es muy importante para la viabilidad del mismo en cuanto la dinámica de requisitos y entornos muy variables y complejos, como lo son el de la realización de los estudios ambientales. La flexibilidad que proveen los metamodelos permite capturar de manera general los elementos esenciales de los análisis ambientales hidrológicos, y además, permite definir posteriormente las características específicas de cada herramienta metodológica.

La OMG sugiere dos posibilidades a la hora de definir lenguajes específicos de dominio, y que se corresponden con las dos situaciones mencionadas antes: o bien se define un nuevo lenguaje diferente a UML(Unified Modeling Language), o bien se extiende el propio UML, especializando algunos de sus conceptos y restringiendo otros, pero respetando la semántica original de los elementos de UML (clases, asociaciones, atributos, operaciones, transiciones, etc.)[95]; la primera opción es la adoptada por lenguajes como CWM (Common Warehouse Metamodel), puesto que la semántica de algunos de sus constructores no coincide con la de UML. Para definir un nuevo lenguaje, se utiliza MOF (Meta-Object Facility), un lenguaje para describir lenguajes de modelado. Por otro lado, hay situaciones en las que es suficiente con extender el lenguaje UML utilizando una serie de mecanismos recogidos en lo que se denominan Perfiles UML (UML Profiles). En [95], se expresan las ventajas e inconvenientes de cada una de las alternativas. En el presente trabajo se realizará la descripción del metamodelo, utilizando UML, debido a que la semántica de los elementos ambientales hidrológicos se puede expresar suficientemente utilizando clases y asociaciones UML.

2.5. Resumen

En el presente capítulo los trabajos relacionados con esta tesis de maestría han sido agrupados en cuatro categorías:

- (i) Dado que desarrollamos técnicas para el mejoramiento de las búsquedas de servicios geográficos ambientales que operan en modelos de comportamiento de los servicios, en esta categoría describimos las representaciones formales que permiten describir, intercambiar y ejecutar comportamientos de servicios. Las representaciones formales presentadas fueron:

Algebra de procesos, redes de Petri, Autómatas de estados finitos y representación de grafos. Finalmente, se realizó una comparación de estas representaciones. El objetivo de esta comparación fue proveer criterios para el modelado de composiciones de servicios web. concluimos que un modelo de representación de grafos provee una notación simple y madura para expresar la semántica de servicios Web. además la representación de grafos permite razonar sobre un sistema compuesto basado en las partes constitutivas sin necesidad adicional de información sobre la implementación de aquellas partes. Además, el paralelismo soportado por el modelo de grafos es un aspecto clave de formalismo el cual debe ser llenado para el modelado exacto de la composición de servicios Web.

- (ii) Como nuestro trabajo busca mejorar el proceso de descubrimiento, y teniendo en cuenta que el proceso base de la búsqueda de servicios es el emparejamiento, en esta sección explicamos las técnicas existentes para el emparejamiento de servicios genéricos basadas en los formalismos para el modelado de procesos, posteriormente se analizan las características particulares de los servicios geográficos. En primer lugar y atendiendo a que nuestra propuesta de descubrimiento de servicios geográfica basada en comportamiento utiliza grafos para su representación, en esta categoría primero introducimos los algoritmos de emparejamiento de grafos. A continuación, se presenta un estado del arte de diferentes técnicas de emparejamiento de servicios. Las técnicas existentes se agruparon en tres conjuntos: emparejamiento de servicios basados en interfaces, semántica y comportamiento, (por su parte, el emparejamiento basado en calidad del servicio se trata en la siguiente sección debido a que el descubrimiento de servicios basado en calidad del servicio está fuertemente relacionado con el dominio de aplicación, en este caso el dominio geográfico). Finalmente una comparación de las técnicas existentes fue dada al final de la sección.
- (iii) Debido a que nos enfocamos en el descubrimiento de servicios Web geográficos, en esta sección se presentan las propuestas específicas para el descubrimiento de Servicios Web geográficos basado en calidad del servicio. La mayoría de las propuestas relacionadas con el emparejamiento basado en calidad del servicio, vislumbran la necesidad de una capa o módulo adicional para el proceso de descubrimiento (broker, agente, certificador), encargado de verificar el ingreso de los parámetros de calidad del servicio y complementar las consultas basándose en estos parámetros. Igualmente, las aproximaciones en este campo, proponen la adición de ciertos parámetros asociados con la calidad del servicio (e.g. precio, tiempo de respuesta) a las descripciones básicas de los servicios (e.g las especificadas utilizando WSDL) o de los procesos (e.g las especificadas utilizando BPEL). Este hecho hace que el proceso de realizar emparejamiento basado en calidad del servicio pueda implementarse como una capa sobre los demás niveles de descubrimiento (como las propuestas de emparejamiento descritas en la sección 2.1), razón por la cual, es muy común que se conjugue el emparejamiento basado en calidad del servicio con otros tipos de emparejamiento, para formar acercamientos de emparejamiento de tipo híbridos
- (iv) Atendiendo a que en este trabajo argumentamos que para poder aplicar los mecanismos de descubrimiento al dominio ambiental, es necesaria la definición de un meta-modelo del proceso de gestión ambiental del recurso hídrico, esta sección presenta una revisión de las propuestas que emplean metamodelos como mecanismos de abstracción.

Capítulo 3. Un metamodelo para la gestión de estudios ambientales

Gran parte del desarrollo económico del sur occidente colombiano está íntimamente ligado a la utilización sostenible de los recursos naturales y en especial del agua. Los procesos productivos existentes se caracterizan por tener una base social, buscar mejorar las condiciones de vida para una población eminentemente rural, donde existe un conocimiento tácito desarrollado autónomamente, dicho conocimiento enfrenta dificultades para articularse con los procesos de producción, abastecimiento y mantenimiento en el mercado regional y nacional [140]. En este escenario, todas las iniciativas encaminadas a conseguir la gestión eficiente de estos recursos son de inmenso valor. Para la consolidación de un proceso eficaz de gestión hídrica, es necesario un consenso alrededor del objetivo de la conservación de los recursos naturales, así como de las herramientas metodológicas para conseguirla.

A pesar de su importancia, las guías metodológicas para la gestión de los recursos hídricos, a nivel regional y nacional aún no han sido establecido claramente, siendo necesario trabajar en su definición de forma inmediata, debido a que solamente a través de convenciones comunes y acuerdos técnicos será posible fácilmente, para las personas encargadas de la toma de decisiones a nivel de comunidades locales, regionales y naciones, descubrir, adquirir, explotar y compartir información geográfica vital para la gestión ambiental [141].

En este trabajo argumentamos que uno de los pasos mas importantes para el establecimiento de un consenso alrededor de la construcción de una herramienta para la gestión ambiental, es el desarrollo de un metamodelo de los estudios ambientales que abstraiga los detalles del proceso, y que sirva de base para la generación de métodos estándares para la gestión ambiental y para la generación de herramientas software igualmente estandarizadas. Por otro lado, el metamodelo propuesto en este capítulo define un contexto general dentro del cual se definirá nuestra propuesta de descubrimiento de servicios geográficos hidrológicos. En este capítulo se describen los módulos y características generales del metamodelo para la gestión de los estudios ambientales, posteriormente, en el capítulo 6 (prototipos y experimentación), se presentará la plataforma Web que valida el metamodelo propuesto dentro del módulo de análisis Hidrológico de la plataforma SIRA)[6]. El resto de este capítulo presenta el metamodelo de gestión de estudios ambientales propuesto.

3.1. Procesos de gestión ambiental del recurso hídrico

El agua es fundamental para el desarrollo de las actividades humanas, tanto desde un punto de vista social como económico y por ende se ha constituido en uno de los pilares del progreso, sin embargo, el uso que se le ha dado al agua (ríos, lagos, lagunas, acuíferos, etc.) está irremediablemente asociado al deterioro importante de los mismos. Esto ha llevado a la elaboración de modelos de gestión del recurso hídrico que no sólo consideren la satisfacción de las demandas del recurso sino que también contemplen la calidad del mismo [142].

Los complejos procesos naturales, físicos, químicos y biológicos resultantes de la interacción entre las diferentes esferas (hidrosfera, atmósfera, litosfera, biosfera y antropósfera) y que forman parte

integral de ciclo hidrológico, son la base para avanzar en el conocimiento y producción de la información que permita interpretar estos procesos. Las dinámicas, el estado, la cuantificación y calidad del recurso hídrico en el país, teniendo en cuenta su variación tanto espacial como temporal, los cambios globales en el medio natural y la incidencia de las diferentes actividades humanas son el modelo que sirve de base para el estudio del medio ambiente desde la óptica de la hidrología [143].

Cuando un hidrólogo estudia cuencas, arroyos o acuíferos, necesita recopilar información pertinente y relevante de precipitación, caudal, niveles de agua subterránea, clima y calidad de agua. En Colombia esta información es manejada por instituciones como el IDEAM (Instituto de Hidrología, Meteorología y Estudios Ambientales de Colombia) [144], IGAC (Instituto Geográfico Agustín Codazzi), Corporaciones Autónomas Regionales, Parques Naturales, etc. Cada institución tiene su forma particular de mantener y presentar la información. Estas instituciones llevan a cabo acciones de recolección y medición de datos hidrológicos, así como el control de calidad de los mismos, basado en estándares propuestos y validados internacionalmente por organismos como la ISO (International Organization for Standardization).

Una buena captura de datos debe cumplir con características como: consistencia en las colecciones de datos con nomenclaturas variadas y complejas, unificación en el formato de los datos, simplificación y libre acceso a repositorios de información [145]. Se distinguen dos tipos de datos, los recolectados en las salidas de campo y los generados en los laboratorios. Los datos que se capturan o generan son consignados en formularios específicos para luego ser llevados a una base de datos y garantizar la confidencialidad, objetividad e integridad de los mismos.

El tratamiento general de los datos geográficos lleva a cabo las siguientes actividades [146]:

- Verificación del identificador de sitio y ubicación.
- Verificación de coherencia en magnitud y unidades de expresión de las variables medidas.
- Verificación de cálculos para generación de resultados intermedios.
- Verificación de criterios de calidad de los datos.
- Verificación de la correcta transcripción a bases de datos.
- Verificación de la seguridad en el caso de procesos informáticos.

En el caso específico de los análisis hidrológicos, existe una serie de actividades, que son:

- Evaluar históricamente el caudal y los registros climáticos.
- Coleccionar datos de caudales para evaluar la variabilidad espacial y temporal de flujos.
- Evaluar el impacto potencial de las extracciones de agua subterránea y los desvíos de agua superficial.
- Coleccionar y evaluar parámetros auxiliares de calidad de agua desde las vertientes más pequeñas.
- Coleccionar y evaluar los datos de calidad del agua.
- Desarrollar recomendaciones para mejorar las condiciones de calidad del agua en cuencas y hacer balances ante las exigencias de ecosistemas acuáticos con fuentes de agua que las poblaciones necesiten.
- Colectar, procesar, analizar y pronosticar los datos sobre la cantidad y calidad de las aguas para reconocer su estado y nivel debido a procesos naturales e impacto por actividad antropica.

Por su parte, los servicios hídricos se pueden dividir en tres grandes grupos [147]:

1. Servicios de gestión hidrológica ambiental, orientados a la toma de decisiones en el sector de abastecimiento de agua con objetivos como conservar las fuentes de agua para consumo humano ([148], [143]), aprovechamiento sostenible en actividades agrícolas [149] y urbano industriales [150].

2. Servicios de datos que entregan información de factores físicos o químicos importantes para los estudios ambientales [151], perfiles microbiológicos [152] y demás parámetros que caracterizan el estado de calidad de las fuentes de agua [153].

3. Servicios de procesamiento que van más allá de los análisis químicos y físicos del agua, y que relacionan estos datos con variables o resultados más complejos como impacto sanitario y de salud [154] o análisis ambientales [6]. El presente capítulo contiene una descripción de los análisis biológicos, físicos y químicos importantes para el estudio del agua

A continuación se detallan los métodos de análisis más importantes en el área del análisis de los recursos hídricos, esta revisión nos permitirá identificar sus características más importantes, y los elementos comunes entre dichos análisis. Estas características y elementos comunes son claves para la determinación de los componentes globales del metamodelo de gestión ambiental del recurso hídrico. Este último permitirá instanciar modelos que soporten el cálculo de cada uno de los indicadores definidos por los métodos de análisis ambiental.

3.1.1. Bioindicación o Análisis de Macroinvertebrados

El biomonitoreo, conocido también como “Monitoreo Biológico”, permite conocer la calidad de un cuerpo de agua con base en los artrópodos presentes. Se pueden utilizar organismos acuáticos como indicadores, pero se recomienda el uso de invertebrados, puesto que existe bastante conocimiento para su interpretación biológica y ecológica. Los macroinvertebrados son usados como indicadores porque su periodo de vida es lo suficientemente largo como para ser afectados por las condiciones de la calidad de agua. Además, están relativamente inmóviles; tienden a formar comunidades características que se asocian con condiciones físicas y químicas muy particulares y fáciles de colectar[155].

Un indicador biológico acuático es considerado como aquel cuya presencia y abundancia señala el estado del sistema en el cual habita, en especial si lo que se busca es el manejo del recurso hídrico. Un contaminante o cualquier otro evento particular que perturbe las condiciones iniciales de un sistema acuático provocarán una serie de cambios en los organismos, cuya magnitud dependerá del tiempo que dure la perturbación, su intensidad y su naturaleza. La acción puede ser indirecta (cambios en el medio) o directa (ingestión o impregnación) [156]. Los índices más importantes con los cuales se puede realizar análisis de bioindicación son:

Índice de diversidad de Shannon Weaver: mientras más uniforme es la distribución de especies que componen la comunidad, mayor es el valor[157].

$$H' = - \sum_{i=1}^R \frac{n_i}{N} \ln \left(\frac{n_i}{N} \right)$$

n_i = Número de individuos por especie

N = Número total de individuos

\ln = logaritmo natural

El índice de diversidad oscila entre 0.0 y 5.0 y se interpreta de la siguiente manera:

0.0 – 1.5	Baja diversidad	alta contaminación
1.6 – 3.0	Mediana diversidad	mediana contaminación
3.1 – 5.0	Alta diversidad	baja contaminación

Índice de calidad BMWP: En 1970 el Biological Monitoring Working Party (BMWP) estableció en Inglaterra un método simple de puntaje para todos los grupos de organismos macro invertebrados identificados hasta nivel de familia y que requiere solo datos cualitativos (presencia / ausencia). El puntaje va de 1 a 10 de acuerdo con su tolerancia a la contaminación orgánica. Las familias más sensibles reciben una puntuación de 10; en cambio las más tolerantes a la contaminación reciben una puntuación de 1. Familias intolerantes a la contaminación tienen puntajes altos y las tolerantes puntajes bajos [158]:

$$BMWP = \sum_{i=1}^n A_i$$

Donde, A_i son las variables elegidas para el estudio.

Los organismos de la muestra recolectada son identificados taxonómicamente a nivel de familia, se asigna el puntaje correspondiente a cada familia y mediante la sumatoria se obtiene el valor del índice BMWP. El índice solo permite obtener puntuaciones para comparar situaciones de calidad pero no para emitir juicios respecto de la misma. En Colombia diferentes estudios adecuaron el sistema BMWP, con la finalidad de apropiarse las metodologías de trabajo de campo en el análisis de calidad de agua [156].

3.1.2. Físicoquímica del Agua

Efecto que se deriva de la penetración lumínica y por consiguiente de la cantidad de energía calórica absorbida por el cuerpo de aguas naturales. El aspecto térmico de un ecosistema acuático es importante considerarlo puesto que incide en características físicas como: densidad del agua, solubilidad de gases, reacciones químicas y en procesos biológicos como por ejemplo: distribución de poblaciones, migraciones y procesos de degradación de materia orgánica [159].

Normalmente la temperatura disminuye en función de la profundidad. Los niveles de estratificación térmica influyen tanto en los ciclos físicos y químicos de las aguas como en los procesos que se llevan a cabo en los ecosistemas acuáticos, algunas de las variables que se tienen en cuenta en los análisis físicoquímicos del agua son:

- **Oxígeno disuelto:** Es uno de los gases más importantes en la dinámica y caracterización de sistemas acuáticos. La difusión del oxígeno en un ecosistema acuático se lleva a cabo por la circulación y movimiento del agua provocados por diferencia de densidad en las capas de agua y los vientos. Otra fuente de oxígeno es la fotosíntesis, generalmente en ecosistemas acuáticos de tipo léntico [158]. El volumen de oxígeno disuelto depende de tres factores fundamentales: la temperatura, la presión parcial de gases atmosféricos y la concentración de las sales disueltas (salinidad).
- **Gas carbónico disuelto:** Es el segundo gas en importancia presente en el agua. Se genera por la descomposición de la materia orgánica, la respiración de animales y plantas y las lluvias. La lluvia arrastra consigo el gas carbónico presente en la atmósfera aumentando la concentración del mismo en los cuerpos de aguas naturales. Su concentración en el agua está influenciada por

procesos fotosintéticos, procesos de respiración y descargas de materia orgánica que elevan los valores hasta concentraciones que afectan el ecosistema acuático. [160]

- **Ph:** Para el ecosistema acuático el pH fisiológico óptimo está entre 7.0 – 7.4. Las reacciones biológicas como la fotosíntesis, tienden a disminuir la concentración de gas carbónico, aumentando los valores de pH. Actividades como la respiración, la degradación de materia orgánica y en general las reacciones químicas que se presentan en el agua elevan la concentración del gas carbónico, disminuyendo los valores de pH [160].
- **Alcalinidad:** Es la capacidad del agua para neutralizar ácidos. El concepto de alcalinidad denota las formas en las cuales se encuentra el dióxido de carbono en el agua. Determina la capacidad del ecosistema para mantener procesos biológicos. Las aguas que poseen valores altos de alcalinidad resisten mejor los cambios drásticos de pH, por consiguiente son aptas para el desarrollo normal del ecosistema acuático [160].
- **Conductividad:** Mide la resistencia de una solución al flujo eléctrico, la conductividad incrementa su valor con la disminución del contenido de sales. [160] Proporciona una idea aproximada del estado de un ecosistema acuático en factores como: actividad iónica, diversidad biótica, balance hídrico, procesos de descomposición de materia orgánica y contaminación. Valores habituales son: 50 microsiemens ($\mu\text{s/cm}$) o menos en aguas muy puras y de 500 a 1000 $\mu\text{s/cm}$ para aguas fuertemente mineralizadas.
- **Salinidad:** Concentración total de componentes iónicos como carbonatos, cloruros, sulfatos, fosfatos y nitratos; se encuentran combinados con calcio, sodio, potasio, hierro y magnesio, formando sales ionizables [160].
- **Dureza:** Está definida por la cantidad de iones de calcio y magnesio presentes en el agua. Las aguas con valores bajos de dureza se llaman “blandas” y son biológicamente poco productivas. Aguas con valores altos de dureza se denominan “duras” y usualmente favorecen al desarrollo del ecosistema [160].

3.1.3. Indicadores de caracterización del agua

La calidad del agua es un factor determinante para estimar la disponibilidad del recurso y el rango de sus posibles usos. Sólo recientemente se ha avanzado en el diseño de políticas, programas y proyectos orientados a corregir la situación de los acueductos del país. Los principales indicadores que caracterizan el recurso hídrico son:

- **Índice de escasez:** Relación porcentual entre la oferta y la demanda de agua que ejercen en su conjunto las actividades sociales y económicas para su uso y aprovechamiento [161].

$$I_e = \frac{D}{O} \times 100\%$$

D - Demanda de agua

O – Oferta hídrica superficial neta

- **Índice de Calidad del Agua (ICA):** Agrupa de manera simplificada algunos parámetros físico-químicos entre ellos: oxígeno disuelto, coniformes totales, nitratos, cloruros y fosfatos; que

indican el deterioro de la calidad del agua, comunicando y evaluando la calidad de los cuerpos de agua [162].

$$ICA = \sum_{i=1}^n S_i W_i$$

S – variable

W – Peso de la variable

- **Índice Diatómico General (IDG):** El principio de este índice es el mismo que para los índices bióticos y viene referido para todos los ecosistemas de agua dulce. Se calcula mediante una fórmula que relaciona la sensibilidad a la polución, el valor indicativo de la especie y la Abundancia [158].

$$IDG = \frac{\sum_{j=1}^n A_j S_j V_j}{\sum_{j=1}^n A_j V_j}$$

S - sensibilidad a la polución

V - Valor indicativo de la especie

A - abundancia

- **Vulnerabilidad por disponibilidad de agua:** Se define como indicador cualitativo del grado de fragilidad del sistema hídrico para el abastecimiento y la amenaza de sequía cuando se presenta condiciones hidroclimáticas medias y extremas.

3.2. El metamodelo del proceso de gestión ambiental

El metamodelo para la gestión de los recursos hidrobiológicos, pretende realizar una abstracción del proceso de análisis ambiental hidrológico, de forma tal que permita generar diferentes modelos dependiendo del tipo de análisis que se requiera. Por ejemplo el metamodelo debe permitir instanciar un modelo de análisis para calcular el BMWP (biological monitoring working party) y también el índice diatómico General manteniendo su estructura fundamental. Para la descripción del metamodelo, a continuación se presenta una descripción general del funcionamiento

El metamodelo debe permitir modelar todo el proceso de gestión ambiental; dicho proceso se compone de tres fases:

- a) La fase de monitoreo: en esta fase se visita un lugar que formalmente se conoce como *punto de monitoreo*, y se realiza la medición de alguna *variable* cualquiera (p.e. ph, oxígeno disuelto), dicho valor se registra acompañado de una *unidad*, que puede ser por ejemplo en el caso del peso gramos o kilogramos.
- b) Con estos valores obtenidos, posteriormente se realiza el cálculo de cualquiera de los índices aplicando un *algoritmo* o *fórmula matemática* que arroja el valor de un indicador.
- c) Finalmente, el valor del indicador, es analizado de acuerdo con unos *rangos* establecidos para cada estudio, por ejemplo el índice de diversidad oscila entre 0.0 y 5.0 y se interpreta de la siguiente forma: si el valor del indicador está entre 0.0 – 1.5 el agua tiene baja diversidad y alta contaminación; por otro lado, si el valor está entre 1.6 – 3.0, se dice que existe una mediana contaminación en el agua.

Como el objetivo de nuestro metamodelo de los estudios ambientales es abstraer los detalles generales del proceso de análisis hidrológico, a continuación se presenta una vista general del

proceso de gestión ambiental de los recursos hídricos, que permitirá identificar los componentes, módulos y características del metamodelo propuesto.

Para la definición del metamodelo se presentará una extensión de la especificación UML 2, donde se definen las relaciones permitidas entre los elementos del modelo a especificar y el uso de las metACLases de un metamodelo dentro de un Perfil UML, como mostraremos a continuación. De los elementos definidos en la especificación de UML 2 (Clases, Extensiones, estereotipos) solamente vamos a utilizar las clases con los elementos de estereotipos. Como aparece en muchos de los perfiles de la OMG [163], el metamodelo se captura como un modelo orientado a objetos expresado utilizando un subconjunto restringido de la notación UML. Los elementos que utilizaremos para nuestro modelo son: clases con atributos y operaciones, asociaciones binarias y paquetes

Las características más importantes del metamodelo son:

- La estructura general del metamodelo se muestra en paquetes, dependiendo del nivel de detalle
- Al mas bajo nivel, se utilizan diagramas de clases para describir las interacciones del metamodelo
- Debido a que no es necesario especificar restricciones no se utilizará OCL
- La semántica de cada elemento se describe junto a cada diagrama.

3.2.1. Descripción por paquetes

A un alto nivel de abstracción, el metamodelo se puede visualizar como dos módulos generales que agrupan toda la funcionalidad (representados como paquetes la figura 3.1). El paquete de monitoreo abarca toda la información asociada con el registro de la toma de muestras. Y en segundo lugar, el paquete de estudio describe el análisis que se realiza de los valores monitoreados.



Figura 3.1. Descripción por paquetes del metamodelo

A continuación se describen estos dos módulos:

3.2.2. Módulo de estudios

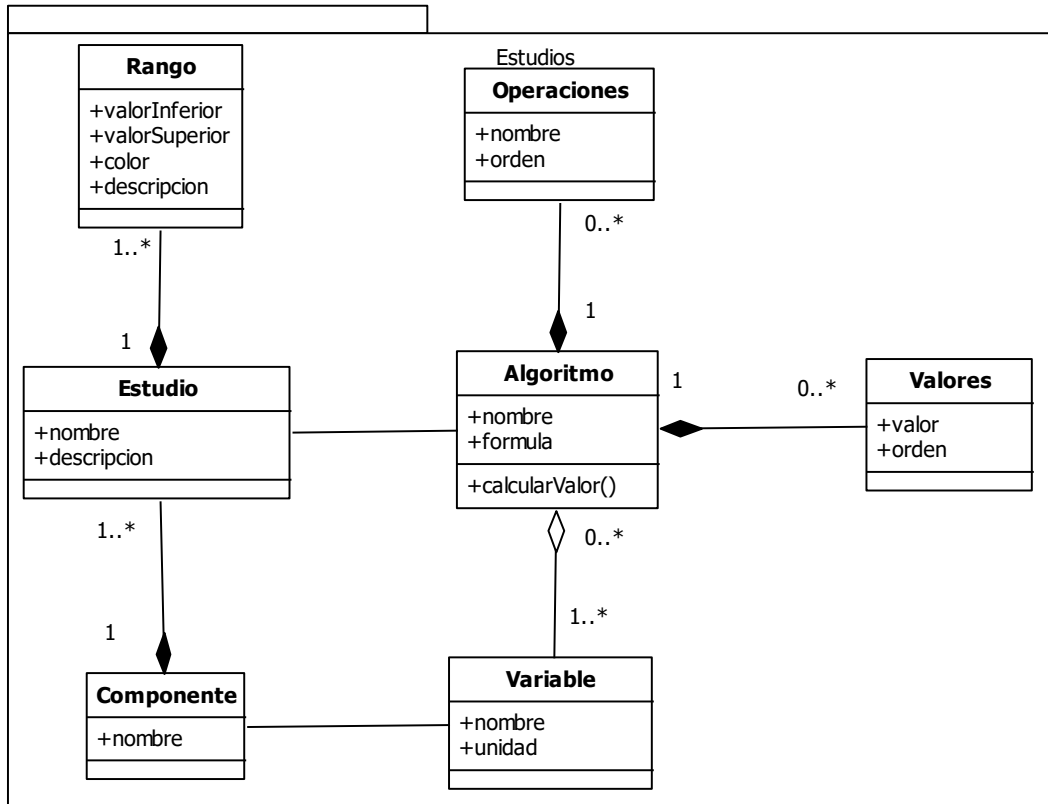


Figura 3.2. Descripción por clases del módulo de estudios

A. *Componente*

Un componente describe la clasificación general de las variables y los estudios biológicos. Un ejemplo de componente puede ser: fisicoquímico o biológico. Los componentes permiten crear clasificaciones generales de las variables y los estudios.

B. *Variable*

Corresponde a un objeto que representa las variables a medir en un estudio ambiental, ejemplos de estas variables puede ser Temperatura o Conductividad, cuyas **unidad** de medida son grados centígrados y microsiemens por segundo respectivamente.

C. *Algoritmo*

Representa la formula matemática que se aplica para conseguir los resultados de un estudio. Por ejemplo en el caso de un estudio se puede requerir que la formula sea

$$\text{Algoritmo}_1 = (\text{Familia}_1 * 0.3) + (\text{Familia}_2 * 0.4)$$

Donde Familia₁ y Familia₂ corresponden a nombres de dos *variables* que se requieren para cálculo del estudio. Para completar la definición de una formula además se cuenta con operaciones y valores, en el ejemplo una *operación* puede ser multiplicación o sumatoria; mientras que un *valor* puede ser 3 ó 0.4

D. Estudio

Representa el *estudio* que se va a realizar, como por ejemplo ICA o BMWP. Los *estudios* poseen un algoritmo cuya aplicación (calculo de la formula matemática con los valores de una muestra en particular) arroja el resultado de un indicador. De igual forma, en un estudio se definen los *rangos* que puede tomar los valores resultantes, los cuales están asociados a unos colores, por ejemplo en el caso de un estudio puede clasificarse su resultado:

Rango	Color	Descripción
0 - 5	Rojo	Malo
5 - 10	Amarillo	Regular
11 - 15	Verde	Bueno

Tabla 3.1. Ejemplo de clasificación de un estudio

E. Rango

Corresponde a la clasificación que posee un estudio según los valores que puede tomar un indicador, como se describió anteriormente, un rango puede estar asociado a un color.

3.2.3. Módulo de monitoreo

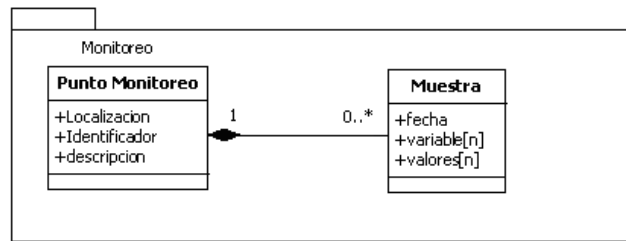


Figura 3.3. Descripción por clases del módulo de monitoreo

F. Punto de monitoreo

Corresponde a una ubicación espacial en donde se toma la *muestra*, sus atributos son:

- La localización que puede ser un identificador geográfico o un punto en un sistema de coordenadas.
- Un identificador del punto que puede ser un nombre o un código si existe
- Y una descripción para información adicional.

G. Muestra

Corresponde a la toma de los valores de una serie de *variables* en un momento dado. para realizar el registro de una muestra se requiere de una fecha de toma y un conjunto de *variables* con sus *valores*.

Como se puede ver en la figura 3.4, la relación entre los dos módulos identificados se realiza entre las clases *Variable* y *Muestra*. Dado que la *muestra* contiene las variables medidas a las cuales se les asocia un *valor*.

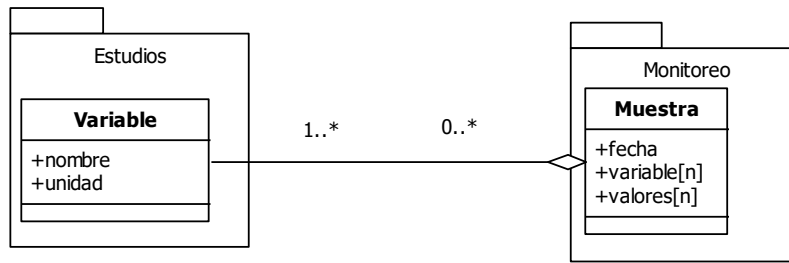


Figura 3.4. Relación entre los módulos de monitoreo y de estudios

3.3. Resumen

En vista de que nuestra propuesta define que uno de los pasos más importantes en la consecución de un consenso alrededor de la construcción de una herramienta para la gestión ambiental, es el desarrollo de un metamodelo de los estudios ambientales que abstraiga los detalles del proceso, que sirva de base para la generación de métodos estándares para la gestión ambiental y la generación de herramientas software igualmente estandarizadas. En el presente capítulo se describe un metamodelo para la gestión de los estudios ambientales hidrológicos, lo cual representa un avance significativo, al ofrece un modelo para la implementación de herramientas software y metodologías de análisis ambiental. El metamodelo explicado en este capítulo define un contexto general dentro del cual se definirá la arquitectura de descubrimiento de servicios geográficos propuesta en esta tesis de maestría.

En primer lugar se presentó una revisión del proceso de gestión de los recursos hídricos, la cual, permitió deducir los componentes, módulos y características del metamodelo a proponer. Primero se explicaron los tres grandes grupos de análisis ambientales: Servicios de gestión hidrológica ambiental; servicios de datos y servicios de procesamiento; posteriormente, se describieron los métodos más importantes en el área del análisis de los recursos hídricos: Bioindicación o Análisis de Macroinvertebrados y Fisicoquímica Del Agua. Y finalmente se presentaron los indicadores de caracterización del agua.

Finalmente, presentó el metamodelo de gestión de estudios ambientales utilizando UML. Para la definición del metamodelo se utilizaron componentes de la especificación UML 2: Clases con atributos y operaciones, Asociaciones binarias y paquetes. En el proceso de gestión ambiental del recurso hídrico se identificaron tres fases, que corresponden con los paquetes de más alto nivel del metamodelo: la fase de monitoreo, el cálculo de cualquiera de los índices aplicando un *algoritmo* o *fórmula matemática* que arroja el valor de un indicador. Y el análisis de los resultados de acuerdo a unos *rangos* establecidos para cada uno de los estudios.

Capítulo 4. Emparejamiento de comportamiento de Servicios

El problema a tratar en este trabajo, es como aplicar el proceso descubrimiento de servicios Web genéricos a los servicios de geoprocesamiento del dominio ambiental, y particularmente de la gestión de los recursos hidrológicos, para ello se propone el enriquecimiento de los mecanismos de consulta existentes en la actualidad con información sobre el comportamiento de los servicios (estructura de los servicios Web compuestos) y con los criterios de calidad del servicio. En este capítulo se presenta la primera parte de nuestra propuesta, la inclusión de la información sobre el comportamiento de los servicios. Debido a que el modelado del comportamiento de los servicios Web geográficos no difiere de los procesos Web genéricos, en este capítulo se presenta la propuesta aplicable a todo tipo de servicios. En capítulos posteriores se abordarán todas las particularidades del descubrimiento de servicios Web geográficos ambientales hidrológicas.

El descubrimiento de servicios es una tarea esencial en el proceso de desarrollar aplicaciones orientadas a servicios. En un escenario típico de descubrimiento de servicios, el solicitante del servicio tiene expectativas específicas sobre los servicios candidatos. En general, existen cuatro tipos de características deseables para un servicio: este debe ser (a) capaz de ejecutar una tarea específica (i.e. realizar una conversión de coordenadas), (b) exponer una interfase particular (i.e. coordenadas de entrada, salida y proyecciones) y (c) comportarse de cierta manera (i.e. ignorar solicitudes con tipos de proyecciones inválidas) (d) Contar con ciertas características de calidad del servicio específicas del dominio de aplicación (utilizar proyecciones estándares, y cartografía actualizada). Tales expectativas motivan a los desarrolladores a buscar dentro de repositorios de servicios Web, para encontrar y seleccionar el servicio que más se acoja a sus necesidades.

Cuando estas dimensiones no son consideradas, la precisión de los procesos de descubrimiento tiende a ser limitado. Por ejemplo, utilizando solo la dimensión (a) en un método de emparejamiento entre un servicio y una solicitud, no hay garantía de que estos dos sean interoperables, dado que la interoperabilidad entre dos servicios depende de la información estructural y otras características de calidad del servicio. Adicionalmente, cuando la interfase de un servicio se compara con la interfase solicitante y los tipos de parámetros no tienen una descripción detallada, no es posible establecer exactamente como los datos de los parámetros y las operaciones corresponden con la especificación del solicitante, ni tampoco se garantiza que el servicio descubierto cumpla con las especificaciones de calidad del servicio esperadas por el solicitante. Finalmente, sin una descripción detallada de las interfaces, el emparejamiento de interfaces puede tornarse confuso porque las definiciones de las operaciones no están significativamente claras.

En este trabajo se argumenta que, en muchas situaciones como es el caso del dominio ambiental, el proceso de descubrimiento de servicios requiere una fase de emparejamiento basada en la especificación del comportamiento de componentes y de una fase de selección de servicios basada en criterios de calidad de servicio. Luego de obtener los servicios que tienen funcionalidades equivalentes (utilizando por ejemplo, emparejamiento de interfaces, ontología, etc.), muchas aplicaciones requieren de encontrar entre los servicios candidatos el servicio que tenga el comportamiento más similar con respecto a las necesidades de los usuarios y que satisfaga los requerimientos de calidad de servicio. Mientras el primer paso del descubrimiento ha sido sujeto de

excelentes trabajos de investigación ([72], [77], etc.), para el segundo paso, no existen soluciones que permitan clasificar la lista de candidatos con respecto al modelo de comportamiento y los criterios de calidad de servicio.

En este capítulo explicamos nuestra aproximación basada en grafos al emparejamiento de comportamiento, así como una aplicación de esta. Primero, introducimos el problema de emparejamiento de grafos explicando su definición y notación, luego, en vista de que nuestra propuesta se concentra en el emparejamiento aproximado de servicios, se presenta un método para medir la similaridad de dos grafos. Posteriormente, se presenta en detalle el emparejamiento de subgrafos por corrección de error. Finalmente, mostramos un caso de aplicación de este algoritmo al protocolo WSCL (Web Services Conversation Language).

4.1. El problema de emparejamiento de grafos

Los grafos son una herramienta poderosa y universal ampliamente usada en el procesamiento de información. Numerosos métodos para el análisis de grafos han sido desarrollados, los cuales son muy importantes en las ciencias de la computación y la ingeniería. Muchos campos como la visión por computador, química y biología molecular tienen diversas aplicaciones donde es necesario descubrir e identificar similaridad entre objetos. Cuando este proceso se realiza de forma automática por un computador sin intervención de un experto humano, una forma útil de representar el conocimiento es utilizando grafos. En conclusión, los grafos han sido probados como una forma efectiva de representar objetos del mundo real y de esta forma el problema de recuperar objetos se convierte en la determinación de la similaridad de grafos, lo cual es conocido como emparejamiento de grafos. En esta sección se presenta una revisión del fondo sobre el emparejamiento de servicios, posteriormente se presenta un método para medir la similaridad de dos grafos, seguidamente, explicamos el emparejamiento de subgrafos por corrección de error. Y finalmente, presentamos las extensiones para expandir el algoritmo EC (de corrección de error) al emparejamiento de comportamiento de servicios.

4.1.1. Antecedentes sobre el emparejamiento de grafos

A continuación describimos las definiciones y las notaciones utilizadas dentro de la teoría de grafos.

Definiciones y notación

Un grafo $G = (V, E)$ en su forma básica se compone de vértices y aristas. V es el conjunto de vértices (llamados también nodos) y $E \subseteq V \times V$ es el conjunto de aristas (también conocidas como arcos o líneas) del grafo G . El orden (o medida) de un grafo G está definido como el número de vértices de G y se representa como $|V|$ y el número de aristas $|E|$. Si dos vértices en G , u y $v \in V$, están conectados por una arista $e \in E$, esto se denota por $e = (u, v)$ y los dos vértices se dice que son adyacentes o vecinos. Se dice que las aristas son no dirigidas cuando ellas no tienen dirección, y un grafo que contiene solamente este tipo de aristas se llama no dirigido.

Cuando todas las aristas tienen dirección y entonces (u, v) y (v, u) pueden distinguirse, se dice que el grafo es dirigido. Aunque nuestra propuesta trabaja principalmente con grafos dirigidos, el emparejamiento de grafos utilizado también puede aplicarse a los no dirigidos. Por otro lado, un grafo dirigido $G = (V, E)$ se llama completo cuando hay siempre una arista $(u, u') \in E = V \times V$ entre cualquier dos vértices u, u' en el grafo. Los vértices del grafo y las aristas también pueden contener información, cuando esta información es una etiqueta simple (i.e. un nombre o un número) el grafo se llama *grafo etiquetado*, donde $\alpha: E \rightarrow L_V$ es la función de etiquetado de vértices y $\beta: E \rightarrow L_E$ es la

función de etiquetado de aristas. De esta forma un grafo etiquetado dirigido se define por el cuádruplo $G = (V, E, \alpha, \beta)$. Otras veces, los vértices y las aristas contienen alguna información adicional. Estos se conocen como atributos de vértices y aristas, y entonces el grafo se conoce como un *grafo atribuido*. Usualmente, este concepto está especificado más profundamente por medio de la distinción entre grafos compuestos de vértices con atributos (o grafos con peso) y grafos compuestos de aristas con atributos.

Emparejamiento de Grafos

Dados dos grafos – *el grafo consulta* G y *el grafo objetivo* G' El procedimiento para compararlos consiste en revisar si son similares o no. Generalmente hablando, los conceptos estándares en el emparejamiento de grafos incluyen isomorfismo de grafos e isomorfismo de subgrafos.

Isomorfismo de Grafos: Dos grafos se dice que son isomórficos si ellos tienen una estructura idéntica. Mas formalmente, un isomorfismo entre dos grafos G y G' es un mapeo biyectivo entre los nodos de G y G' que preserve la estructura de las aristas.

Así: sean G y G' un isomorfismo de grafos entre G y G' es un mapeo biyectivo $f: V \rightarrow V'$ Tal que:

- $\alpha(v) = \alpha'(f(v))$ para todo $v \in V$
- Para todo vértice $e = (u, v) \in E$ existe una arista $e' = (f(u), f(v)) \in E'$ tal que $\beta(e) = \beta'(e')$ y para cada arista $e' = (u', v') \in E'$ existe una arista $e = (f^{-1}(u'), f^{-1}(v')) \in E$ tal que $\beta(e) = \beta'(e')$.

Este tipo de problema se conoce como emparejamiento exacto de grafos.

Isomorfismo de subgrafos es otro concepto popular en la comparación de grafos. Dados dos grafos existe un isomorfismo de subgrafos si un grafo contiene un subgrafo que es isomórfico al otro grafo. El isomorfismo de subgrafos es muy útil para encontrar si un objeto dado es parte de otro objeto o aun de una colección de muchos objetos. De esta manera:

Si $f: V \rightarrow V'$ es un isomorfismo de grafos entre los grafos G y G' , y G'' es un subgrafo de otro grafo G'' , i.e. $G' \subset G''$, entonces f es un isomorfismo de subgrafos desde G a G'' .

Cuando un isomorfismo no puede ser encontrado entre dos grafos a comparar porque el número de vértices es diferente entre los grafos de consulta y objetivo, entonces el problema no consiste en buscar la forma exacta de emparejar los vértices de un grafo con los vértices del otro, sino en encontrar el mejor emparejamiento entre ellos. Esto conduce a una clase de problema conocido como emparejamiento inexacto de grafos. Existen dos tipos de isomorfismo de grafos: el subgrafo común máximo y el supergrafo común mínimo. El subgrafo común máximo de dos grafos G y G' es el grafo más grande que es isomórfico de un subgrafo de G' y G . el subgrafo común máximo es útil para medir la similaridad de dos objetos. Claramente, entre más grande sea el subgrafo común máximo entre dos grafos, más similares serán estos dos grafos. De otro lado, el subgrafo común mínimo de un par de grafos G' y G es el grafo más pequeño que contiene subgrafos isomórficos a G' y G (ver [164]).

4.1.2. Distancia de edición: Un método para medir la similaridad de dos grafos

Un método para medir la similaridad de dos grafos es la distancia de edición. Este método es una generalización de la distancia de edición de cadenas, también conocida como distancia Levenshtein [61]. En la distancia de edición de grafos, uno introduce un conjunto de operaciones de edición de

grafos. Estas operaciones de edición son utilizadas para modelar distorsiones que transforman un modelo ruidoso en una representación de objetos ideal. Los conjuntos comunes de operaciones de edición de grafos incluyen borrar, insertar y sustituir de nodos y aristas. Dado un grupo de operaciones de edición, la distancia de edición de grafos se define como el mínimo número de operaciones necesarias para transformar un grafo en otro. Usualmente se asigna un costo a cada operación de edición; el costo es dependiente de la aplicación y es usado generalmente para modelar las características de las distorsiones correspondientes. Típicamente, entre más probable que sea la aparición de una distorsión, el costo es menor. Si se asigna un costo para cada operación de edición entonces la distancia de edición de dos grafos, G y G' , se define como el menor costo calculado sobre todas las secuencias de operaciones de edición que transforman G en G' . La distancia de edición de grafos y las medidas de similaridad relacionadas han sido discutidas en ([36],[35],[19]).

Dado un grafo G , una operación de edición de grafos δ en G es cualquiera de las siguientes:

- Sustituir la etiqueta $\alpha(v)$ del vértice v por l .
- Sustituir la etiqueta $\beta(e)$ de la arista e por l .
- Eliminar el vértice v de G (para la corrección de vértices no existentes). Se eliminan también las aristas que unan este vértice con otros.
- Eliminar la arista e de G (para la corrección de aristas no existentes)
- Insertar una arista entre dos vértices existentes (para la corrección de aristas)

Grafico editado Dado un grafo y una operación de edición δ , el grafo editado $\delta(G)$ es un grafo en el cual fue aplicada la operación δ . Dado un grafo G y un secuencia de operaciones $\Delta = (\delta_1, \delta_2, \dots, \delta_k)$, el grafo editado $\Delta(G)$ es un grafo $\Delta(G) = \delta_k(\dots\delta_2(\delta_1(G)))\dots$.

En vista de que esta tesis se concentra en el emparejamiento aproximado de servicios, para efectos de complementariedad, en el resto de esta sección se expondrá la distancia de edición de subgrafos.

4.1.3. Emparejamiento de subgrafos por corrección de error

Dados dos grafos G y G' , un isomorfismo de subgrafos por corrección de error (ec) f de G a G' es una 2-upla $f = (\Delta, f_\Delta)$ donde Δ es una secuencia de operaciones de edición y f es un isomorfismo de subgrafos de $\Delta(G)$ a G' .

Para cada operación de edición δ , se asigna un costo $C(\delta)$. El costo de un isomorfismo de subgrafos por corrección de error $f = (\Delta, f_\Delta)$ es el costo de las operaciones de edición Δ , i.e.: $C(f) = \sum_{\delta \in \Delta} C(\delta)$. Usualmente, hay más de una secuencia de operaciones de edición para conseguir un isomorfismo de subgrafos desde $\Delta(G)$ a G' . y consecuentemente, hay más de un isomorfismo de subgrafos por corrección de error de G a G' . Es evidente que nos interesa encontrar en el isomorfismo de subgrafos por corrección de error con el menor costo. Entonces:

Distancia de edición de subgrafos. Sean G y G' dos grafos. La distancia de subgrafos de G a G' , $ed(G, G')$ esta dada por el mínimo costo tomado sobre todos los isomorfismos de subgrafos por corrección de error f de G a G' .

Algoritmo de detección de isomorfismo de subgrafos por corrección de error.

En esta sección se presenta un algoritmo conocido para el problema de detección de isomorfismo de subgrafos por corrección de error [165].

La detección de isomorfismo de subgrafos se basa en una búsqueda en el espacio de estados utilizando un algoritmo similar a A* [36]. La idea básica de una búsqueda en el espacio de estados son precisamente los estados, los cuales representan soluciones parciales del problema dado, en este tipo de búsquedas se definen transiciones de un estado a otro, así, un estado representa una solución más completa que la del estado previo. Para cada estado s hay una función de evaluación $f(s)$ que describe la calidad de la solución representada. Los estados se expanden a sí mismos de acuerdo al valor de f . En el caso de cada detección de isomorfismo de subgrafos, dado un grafo consulta G y un grafo objetivo G_I , un estado s en el espacio de búsqueda representa un emparejamiento parcial desde G a G_I . Cada emparejamiento parcial implica un número de operaciones de edición y su costo puede ser usado para definir la función de evaluación $f(s)$.

En otras palabras, el algoritmo comienza mapeando el primer nodo de G con todos los nodos de G_I y escoge el mejor mapeo (con costo mínimo). Esto representa un mapeo parcial que será extendido por la adición un nodo al tiempo. El proceso termina cuando se alcance un estado representando un isomorfismo de subgrafos ec (emparejamiento por isomorfismo de subgrafos por corrección de error) de G a G_I ; o cuando todos los estados en el espacio de búsqueda tengan costos de edición que excedan el umbral de aceptación. El costo del mapeo $C(p')$ representa el costo de extender el mapeo actual p con el siguiente nodo en el grafo consulta. Extender el mapeo, por medio del mapeo de un vértice v (en el grafo objetivo que aún no ha sido mapeado) a un vértice w en el grafo de consulta (que no pertenece al mapeo actual) implica operaciones de edición de nodos y operaciones de edición de aristas. Primero, los atributos de v deben ser sustituidas por atributos de w , y en segundo lugar, para cada par de nodos ya mapeados (v', w') debe asegurarse que cada arista (v', v) en el grafo consulta pueda ser mapeado a una arista (w', w) en el grafo objetivo por medio de operaciones de edición de aristas [165].

Algoritmo 1 Detección de isomorfismo de subgrafos por corrección de error ($G(V)$, $G_I(V_I)$)

1: Inicializar OPEN: mapear el nodo de actividad en V en cada nodo de actividad en V_I (mapeo de nodos), i.e. crear un mapeo p . Calcular el costo de este mapeo $O(p)$ y agregar p a OPEN .

2: IF OPEN esta vacío THEN EXIT.

3: Seleccionar p de OPEN tal que $O(p)$ es mínimo y remover p de OPEN

4: IF $O(p) > \text{Umbral mínimo}$ THEN EXIT.

5: SI p representa un mapeo completo de G a G_I THEN el resultado es p . Fijar $\text{umbral mínimo} = C(p)$.

IR A 2.

6: LET $p = \{(v_1, w_1), \dots, (v_k, w_k)\}$ el mapeo actual de k nodos de G .

7: FOR EACH *nodo de interacción* w en V_I que no ha sido mapeado todavía a un nodo correspondiente en V

7.1: Extender el mapeo actual p a p' mapeando el nodo v_{k+1} de V a w ,
 $p' = \{(v_1, w_1), \dots, (v_k, w_k), (v_{k+1}, w)\}$ y calcular el costo de este mapeo $C(p')$

7.2: Agregar p' a OPEN

8: GO TO 2

Ejemplo. Considere los grafos g_1 y g_2 en la figura 4.1 y el problema de encontrar el isomorfismo de subgrafos ec desde g_1 a g_2 . Note que los vértices de g_1 y g_2 están etiquetados con letras del alfabeto latino mientras las aristas no tienen etiquetas. Adicionalmente, los vértices de g_2 están numerados de forma única para propósito de identificación. El costo de las operaciones de edición se define como sigue. La sustitución de la etiqueta l_1 por una etiqueta l_2 se define como la distancia de las letras l_1 y l_2 en el orden del alfabeto. Por ejemplo, si $l_1 = a$ y $l_2 = g$ el costo de sustituir a por g , o g por a , es 6. El costo de borrar un vértice este definido constante a 3 mientras el costo de borrar e insertar una

arista se fija en 1. El espacio de búsqueda que es expandido por el algoritmo tradicional se muestra en la figura 4.1. En la izquierda del espacio de búsqueda, el grafo g_1 es redibujado con sus vértices dados en el orden en el que ellos son emparejados. Cada estado s se indica por un emparejamiento parcial $(v)[c]$ donde (v) denota el número de vértices de g_2 que es emparejado con el vértice correspondiente de g_1 y $[c]$ es el costo total que está implicado por el emparejamiento parcial representado en el estado s y sus estados predecesores. Los estados son expandidos de acuerdo al costo de c . Para representar la eliminación de un vértice, se utiliza el símbolo \$.

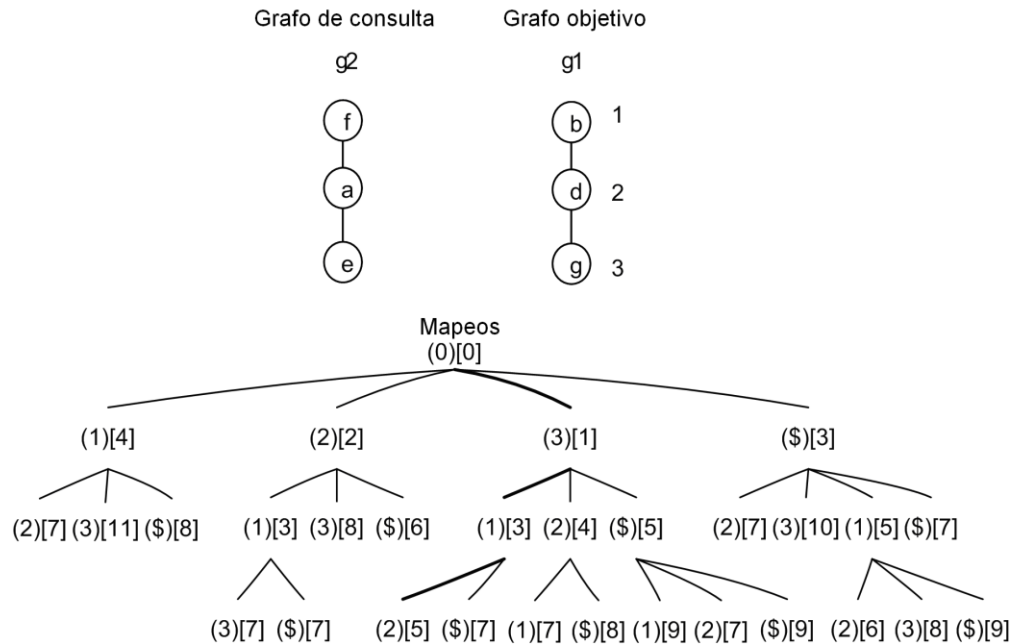


Figura 4.1: Ejemplo de la detección de isomorfismo de subgrafos por corrección de error.

Al inicio, solo hay un estado raíz, $(0)[0]$, para el cual se generan los estados sucesores. A continuación, el estado con el menor costo, que es el tercer estado desde la izquierda, $(3)[1]$ se expande, y así sucesivamente. El camino al estado representando el isomorfismo de subgrafos está representado en negrilla en la figura 4.1. Note que el isomorfismo de subgrafos desde g_1 a g_2 empareja el vértice f con el vértice g , el vértice a sobre el vértice b y el vértice e sobre el vértice d . La arista entre a y f en g_1 es eliminada y se inserta la arista entre f y e en g_1 . El costo total de las operaciones de edición equivale a 5.

4.1.4. Extensiones del Algoritmo EC

Con el fin de expandir el algoritmo EC al emparejamiento de comportamiento de servicios hemos considerado dos extensiones. La primera extensión se centra en el nivel de granularidad de los servicios y la segunda se refiere a las medidas de similaridad. Tales medidas deben tener en cuenta el número de interacciones o el número de secuencias de interacciones en el servicio objetivo que son cubiertas por el servicio de consulta. A continuación se describen estas extensiones.

Extensiones de la distancia de edición de subgrafos

Dado que los modelos emparejados pueden tener diferentes niveles de granularidad para conseguir la misma funcionalidad, se requiere de nuevas operaciones de edición. Por ejemplo, un servicio

tiene una operación simple para realizar cierta funcionalidad, mientras en otro servicio esta misma funcionalidad se consigue por medio de la composición de muchas operaciones.

Dado un grafo G , extendemos la definición de operación de edición δ en G agregando dos operaciones:

- Descomponer un vértice v en dos vértices v_1, v_2 .
- Unir dos vértices v_1, v_2 en un vértice v .

Nos limitamos a un caso simple de descomposición, cuando un vértice se descompone en una secuencia de dos vértices. Este tipo simple de descomposición es suficiente para el tipo de aplicaciones que analizamos, una operación de descomposición mas general sería descomponer un vértice en un subgrafo conectado, esto es sujeto de trabajo futuro.

La operación de descomponer un vértice v en dos vértices v_1, v_2 se ejecuta en la siguiente forma:

- Todas las aristas que tengan como destino el vértice v tendrán destinación de v_1 ;
- Todas las aristas que tengan como fuente el vértice v , tendrán como fuente el vértice v_2 ;
- Se agrega una arista entre el vértice v_1 y v_2 .

La operación de unión se ejecuta en una forma similar.

Estas dos nuevas operaciones de edición permiten modelar dependencias uno a muchos entre los vértices de dos grafos (i.e., un vértice en un grafo corresponde a dos vértices en el segundo grafo). Las operaciones de edición clásicas tienen en cuenta solo mapeos uno a uno entre los vértices de los dos grafos. Por ejemplo, si un vértice v en el primer grafo corresponde a la composición de dos vértices en el segundo grafo (v_1 seguido por v_2), un algoritmo de emparejamiento basado en la distancia de edición clásica mapearía v a v_1 y suprime v_2 . Estos no permitirían describir que v esta mapeado a una composición de v_1 y v_2 .

4.1.5. Medida de la similaridad para el emparejamiento de comportamientos

La distancia de edición de subgrafos definida previamente expresa el costo de transformación necesaria para adaptar el grafo consulta (Q) con el fin descubrir un subgrafo en el modelo objetivo (T). Esta distancia es asimétrica y representa la distancia desde el grafo consulta al grafo objetivo. De otro lado, la similaridad de los grafos y la distancia entre ellos están muy relacionados tanto que a menudo se confunden. Mientras el término distancia se usa formalmente en un sentido matemático, el significado particular del término similaridad usualmente depende de las circunstancias y su campo de aplicación. Teniendo en cuenta lo anterior, las medidas de similaridad para el emparejamiento de comportamiento de servicios se calculan de la siguiente forma:

- Considerando que una medida de similaridad no debería ser cualitativa (dando información sobre similitudes y diferencias entre los dos grafos) sino cuantitativa (indicando que tanta similaridad existe entre los dos grafos) consideramos que la similaridad entre dos grafos puede basarse en la medida de la distancia. De esta manera dos grafos se consideran similares cuando existe una pequeña distancia de edición entre ellos. La función de similaridad puede definirse como:

$$S_{ed} = \frac{1}{d_{ed}}$$

Donde $editDist(Q, T)$ es el costo mínimo de las operaciones de edición necesarias para transformar el grafo de consulta en el grafo objetivo.

- con el fin de clasificar los grafos objetivos, la medida de similaridad debe tener en cuenta el número de vértices en el grafo objetivo que fueron cubiertos por el grafo de consulta. Si dos grafos objetivos tienen la misma distancia de subgrafos al grafo consulta ($Sim_{edit}(Q, T)$) pero son emparejados a subgrafos con diferentes número de nodos, el que empareje un subgrafo con más nodos será seleccionado. Entonces la función de similaridad es:

$$Sim_{node}(Q, T) = Sim_{edit}(Q, T) * \frac{|N_q \cap N_T|}{|N_q|}$$

Donde $|N_q \cap N_T|$ es el número de nodos que aparecen tanto en el grafo de consulta como en el grafo objetivo. □

- La tercera medida de similaridad se basa en el número de diferentes secuencias entre dos grafos.



En la misma forma que Sim_{node} , si dos grafos objetivos tienen la misma medida de distancia con respecto al grafo de consulta ($Sim_{edit}(Q, T)$) pero están emparejados a subgrafos con diferente número de secuencias de nodos, se preferirá el que corresponda con un subgrafo con más secuencias de nodos.

En esta función de similaridad el parámetro $Nseq$ representa el número de nodos consecutivos ($Nseq(Q)$ para el grafo consulta y $Nseq(T)$ para el grafo objetivo) y $Nseq(Q) \cap Nseq(T)$ representan las secuencias mapeadas dentro del grafo consulta con respecto al grafo objetivo. Por ejemplo, si seleccionamos $Nseq = 3$ (tri secuencia), entonces la función considerará todas las secuencias de los tres nodos consecutivos del grafo objetivo y de consulta y su intersección. Así, se calcularán $tri(Q)$, $tri(T)$ y $|tri(Q) \cap tri(T)|$.

4.2. Emparejamiento de protocolos de conversación

En esta sección se ilustra el uso del algoritmo de emparejamiento de grafos por corrección de error para el emparejamiento de protocolos de conversación. Para ejemplificar el emparejamiento de protocolos de negocios hemos seleccionado el modelo WSCL [166]. La misma aproximación puede aplicarse a otros modelos tales como BPEL, siempre y cuando pueda transformarse de forma unívoca el protocolo de conversación en un grafo (representaciones equivalente de un protocolo de conversación se reducen al mismo grafo proceso). El emparejamiento de protocolos de conversación se compone de los siguientes pasos (ver figura 4.2). Primero, los protocolos de conversación a comparar se transforman en grafos, luego, ambos grafos son expandidos con el fin de tener el mismo nivel de granularidad. A continuación, se aplica el algoritmo de emparejamiento de grafos y se aplica la función de similaridad entre los grafos. Finalmente, los niveles de granularidad son comparados y los costos correspondientes a las diferencias identificadas se agregan a la distancia total.

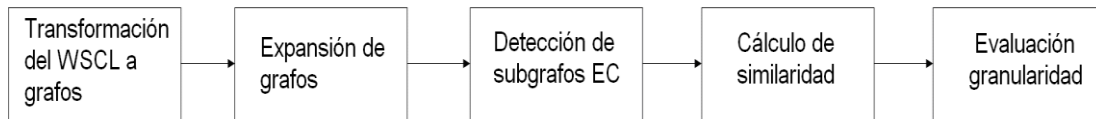


Figura 4.2: proceso de emparejamiento WSCL

A continuación, damos un vistazo general del lenguaje de conversación de servicios Web, y posteriormente discutimos cada paso del emparejamiento en detalle

4.2.1. Lenguaje de conversación de servicios Web: WSCL

WSCL es un lenguaje de definición de conversaciones simple, que ofrece las construcciones básicas para modelar la secuencia de interacciones u operaciones de una interfase. Este lenguaje complementa la definición de interfaces especificando la orden de invocación de las operaciones. Una conversación en WSCL se especifica utilizando las siguientes construcciones básicas.

- Descripciones de tipo de documento: especifica los tipos (esquemas) de los documentos XML que el servicio puede aceptar y transmitir en el curso de una conversación. Los esquemas de los documentos intercambiados no son especificados como parte del documento de especificación WSCL; estos esquemas son documentos XML separados referenciados por su URL en los elementos de tipo de documentos XML de la especificación de la conversación.
- Las interacciones modelan las acciones de la conversación como intercambios de documentos entre dos participantes. WSCL soporta cinco tipos de interacciones: Send (el servicio envía un documento saliente); Recieve (el servicio recibe un documento entrante); SendReceive (el servicio envía un documento saliente y espera recibir un documento entrante como respuesta); ReceiveSend (el servicio recibe un documento entrante y envía un documento como respuesta); Empty (no contiene documentos intercambiados, solo se usa para modelar el inicio y el fin de una conversación). Cada interacción especifica el tipo (esquema) de documento XML que se espera como entrada o es producida como salida.
- Las transiciones especifican las relaciones de orden entre las interacciones.

Un documento WSCL contiene información adicional sobre la conversación, incluyendo el nombre de la conversación y el nombre de los documentos intercambiados entre los participantes. Las conversaciones pueden ser vistas como interfaces o procesos públicos soportados por un servicio. Se diferencian de las interfaces como se definen en CORBA (Common Object Request Broker Architecture) o en el lenguaje de programación Java porque también especifican el orden posible de las operaciones (i.e. la posible secuencia en la cual los documentos pueden intercambiarse). Una definición de la conversación describe la conversación desde la perspectiva de uno de los participantes. generalmente, una conversación publicada en un directorio de servicios está definida desde la perspectiva del oyente; la primera interacción que ocurre es una interacción Receive o ReceiveSend. El iniciador puede derivar su definición de conversación del oyente simplemente convirtiendo las interacciones Receive y ReceiveSend, en Send y SendReceive, y viceversa. Los dos participantes pueden interactuar si las definiciones de la conversación que utilizan son duales del otro.

La figura 4.3 muestra una conversación WSCL que describe la funcionalidad de un servicio que calcula un indicador correspondiente a un Indicador ambiental para la gestión del recurso hídrico. En el ejemplo, primero el servicio espera la recepción del mensaje *solicitudLogin* (interacción *Login*). El servicio envía una respuesta un documento *rtaLoginValido* o *rtaLoginInvalido*

dependiendo del tipo y el contenido del mensaje recibido. En caso de un proceso de validación exitoso, el servicio espera un mensaje *solicitudTipoEstudio* (interacción *VerificarTipoEstudio*). Dependiendo del contenido del documento recibido, la interacción puede responder con *rtaEstudioExistente* o *rtaEstudioNoExistente*. Si la respuesta es un *rtaEstudioExistente* la conversación espera un mensaje *parametrosIndicador* y puede enviar como respuesta un mensaje con el valor del indicador ambiental calculado *rtaIndicador* o *rtaParamInvalido* al usuario.

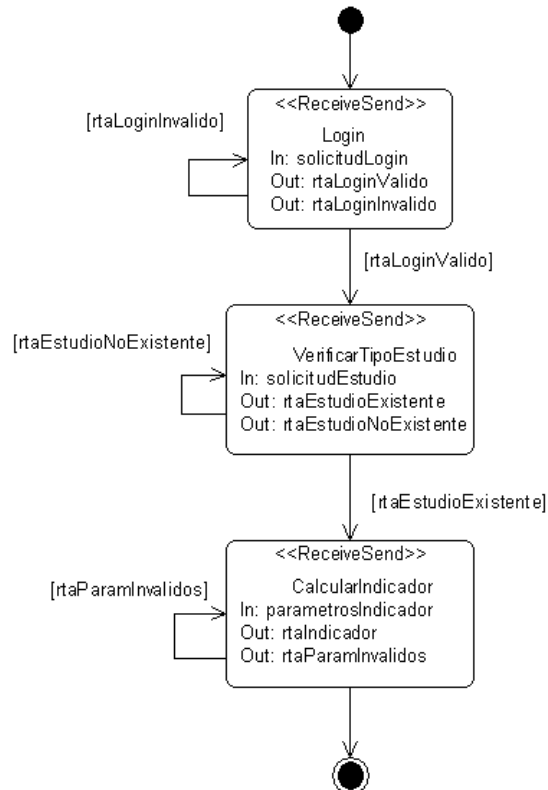


Figura 4.3: Ejemplo de una conversación WSCL

La descripción XML de la conversación contiene dos sub elementos: la lista de interacciones, que contienen las interacciones de conversación y la lista de transiciones. Otra parte de la definición del posible orden de las interacciones es la especificación de las primeras y últimas interacciones de una Conversación. En la descripción XML esto se hace por los atributos *InitialInteraction* y *FinalInteraction* del elemento *conversación*. Para el ejemplo presentado en la figura 4.3 la interacción inicial es *Login* y la interacción final es *CalcularIndicador*.

Una conversación puede pasar de una interacción a otra de acuerdo a la secuencia permitida definida en los elementos de transición. La figura 4.4 muestra la transición XML del ejemplo descrito. *SourceInteraction* referencia una interacción que puede preceder la *DestinationInteraction* cuando se ejecuta la conversación. Similarmente, *DestinationInteraction* referencia una de las interacciones que pueden seguir la *SourceInteraction* cuando se ejecuta la conversación. Juntas, todas las transiciones especifican todas las posibles secuencias de interacción. *SourceInteractionCondition* es una restricción adicional sobre la transición. Esta se necesita cuando la *SourceInteraction* especifica más de un posible documento a intercambiar y el tipo de documento intercambiado tiene una influencia sobre las interacciones siguientes.

```

<ConversationTransitions>
<Transition>
  <SourceInteraction href="Login"/>
  <DestinationInteraction href="VerificarTipoEstudio"/>
</Transition>
<Transition>
  <SourceInteraction href="Login"/>
  <DestinationInteraction href="Login"/>
  <SourceInteractionCondition href="rtaLoginInvalido"/>
</Transition>
<Transition>
  <SourceInteraction href="VerificarTipoEstudio"/>
  <DestinationInteraction href="CalcularIndicador"/>
</Transition>
<Transition>
  <SourceInteraction href="VerificarTipoEstudio"/>
  <DestinationInteraction href="VerificarTipoEstudio"/>
  <SourceInteractionCondition href="rtaEstudioNoExiste"/>
</Transition>
<Transition>
  <SourceInteraction href="VerificarTipoEstudio"/>
  <DestinationInteraction href="CalcularIndicador"/>
</Transition>
<Transition>
  <SourceInteraction href="CalcularIndicador"/>
  <DestinationInteraction href="CalcularIndicador"/>
  <SourceInteractionCondition href="rtaParamInvalidos"/>
</Transition>
</ConversationTransitions>

```

Figura 4.4: Representación XML de la transición WSCL

La figura 4.5 ejemplifica la representación XML de la interacción *Login*. Cada interacción *ReceiveSend* es la unidad lógica de recibir una petición y retornar una respuesta. La interacción no está completa hasta que la respuesta ha sido enviada. Una interacción *ReceiveSend*. Puede especificarse más de un documento XML de salida. Esta capacidad permite modelar el tipo de casos en los cuales hay muchos tipos de mensajes de respuesta que un servicio puede retornar en respuesta a una solicitud específica. La posibilidad de tener respuestas adicionales posibles se usa principalmente para mensajes de error. Si una interacción especifica más de un tipo de documento de salida, solo uno de ellos es intercambiado en tiempo de ejecución.

```

<ConversationInteractions>
...
<Interaction interactionType="ReceiveSend" id="Login">
  <InboundXMLDocument id="solicitudLogin"
    hrefSchema="http://sira.unicauca.edu.co/solicitudLogin.xsd" />
  <OutboundXMLDocument id="rtaLoginValido" hrefSchema="
    http://sira.unicauca.edu.co/rtaLoginValido.xsd " />
  <OutboundXMLDocument id="rtaLoginInvalido" hrefSchema="
    http://sira.unicauca.edu.co/rtaLoginInValido.xsd " />
</Interaction>
...
</ConversationInteractions>

```

Figura 4.5: Representación XML de una interacción ReceiveSend

4.2.2. Transformación WSCL a Grafos

La función parser transforma un modelo de conversación WSCL en un grafo cuyos vértices representan interacciones y cuyas aristas representan transiciones. Cada nodo tiene los siguientes atributos: nombre, tipo de interacción y documentos. La tabla 4.1 muestra la correspondencia entre constructores WSCL y elementos del grafo.

Construcción WSCL	Elementos del grafo
Interaction	Vértice
Tipo	Atributo del vértice
Documento de entrada	Atributo del vértice
Documento de salida	Atributo del vértice
ID	Etiqueta
Transición	Arista
Condición de transición	Atributo de la arista

Tabla 4.1. Correspondencia entre elementos WSCL y elementos del grafo.

4.2.3. Descomposición de las interacciones WSCL

Después de transformar los protocolos de conversación en grafos, el siguiente paso en el emparejamiento de comportamientos es la expansión del grafo. Las operaciones de descomposición se aplican con el fin de tener el mismo nivel de granularidad en ambos modelos. Las operaciones de descomposición dependen del metamodelo de protocolos a emparejar. Por ejemplo, para el metamodelo WSCL, es posible que en un protocolo una interacción esté modelada como una interacción *SendReceive*, mientras en el segundo protocolo se consigue la misma funcionalidad se consigue con una interacción *Send* seguida de una interacción *Receive*. Así, la descomposición transformará las interacciones de tipo *SendReceive* y *ReceiveSend* en interacciones atómicas: *Send* y *Receive*. Una interacción *SendReceive* se descompone en una interacción *Send* seguida por una interacción *Receive* de la siguiente forma:

- Todas las aristas que tengan como destino la interacción *SendReceive* tendrán como nuevo destino la interacción *Send*.
- Todas las aristas que tengan como inicio la interacción *SendReceive* tendrán como nuevo destino la interacción *Receive*.
- Se agregará una arista desde la interacción *Send* a la interacción *Receive*.
- Si la interacción *SendReceive* tiene un documento de salida *a* y un documento de entrada *b*, entonces la interacción *Send* tendrá como documento de salida a *a*, y la interacción *Receive* tendrá como documento de entrada a *b*.

En forma similar una interacción *ReceiveSend* se descompone en una interacción *Receive* seguida de una interacción *Send*.

Una función de descomposición podría también ser aplicada para tratar la diferencia de granularidad de los mensajes intercambiados; esto significa que si un mensaje *D* puede descomponerse en *n* mensajes atómicos, entonces la interacción que envía el documento *D* (*Send D*) se puede descomponer en *n* interacciones de acuerdo con *n* partes de los documentos. Sin embargo, en nuestro trabajo no analizamos el contenido de los mensajes y solo se propone la operación de descomposición descrita anteriormente.

Esta función de descomposición es específica para el modelo WSCL. Para otras aplicaciones, el usuario puede especificar una función de descomposición diferente. La función de descomposición tiene siempre la misma forma: toma como argumento un vértice y devuelve dos vértices resultantes de la descomposición (que se supone secuencial). El comportamiento de la función es específico a la aplicación (metamodelo de los protocolos a emparejar) y consiste en especificar como las etiquetas y los atributos de los nuevos vértices se obtienen a partir del vértice descompuesto.

4.2.4. Reglas de comparación de los Grafos WSCL

Las reglas de comparación describen todas las funciones dependientes de la aplicación que permiten calcular el costo de las operaciones de edición de grafos. Estas funciones son usadas por el algoritmo de emparejamiento de isomorfismo de subgrafos ec para calcular la distancia entre los grafos (ver sección 4.1.3). Con el fin de soportar aplicaciones con funciones de costos especializados, se puede registrar la función de costo definida por el usuario como una regla. A continuación se explica la función de costo utilizada para la el emparejamiento de protocolos de conversación.

El costo de insertar, eliminar aristas y vértices puede definirse como constante. El costo de editar una arista se calcula por la función VertexMatch (ver algoritmo 2). Como los vértices representan interacciones WSCL este costo expresa la distancia entre dos interacciones WSCL. Cada interacción tiene una etiqueta (Id) y dos atributos: el tipo de interacción (Type) y el conjunto de documentos (D) (documentos de entrada o salida). El emparejamiento da prioridad a la comparación de tipos, y si dos interacciones tienen el mismo tipo, compara la similaridad del conjunto de documentos *TotalSD*; Si hay una similaridad entre ellos ($TotalSD > 0$), calcula la similaridad de los nombres de las interacciones (*SimId*).

Algoritmo 2 Función VertexMatch

INPUTS: (Node_i, Node_j)

Node_i: Struct (Id_i, Type_i, D_i), Node_j: Struct (Id_j, Type_j, D_j)

OUTPUT: DistanceNode

if Type_i ≠ Type_j (tipos diferentes) **Then**

Devolver DistanceNode = 1

else

 Calcula la similaridad de los documentos *TotalSD*

if *TotalSD* > 0 **Then**

 Calcula la similaridad de Ids SimId = LS(Id_i, id_j)



 Devolver DistanceNode

else

 Devolver DistanceNode = 1

end if

end if

La función $SD(D_i, D_j)$ donde D_i, D_j es el conjunto de documentos del $Nodo_i$ y el $Nodo_j$ respectivamente, calcula el mejor emparejamiento que se puede obtener entre los documentos de los dos conjuntos.



El número de mapeos establecidos es $Min(|D_i|, |D_j|)$. La función LS calcula la similaridad lingüística entre los nombres de los documentos y se explica en la siguiente sección. Finalmente la similaridad total de los conjuntos de documentos es:

$$TotalSD = \frac{SD(D_i, D_j)}{k}$$

Donde, k = Numero de documentos del conjunto D_i . El resultado de aplicar la función SD se normaliza con el número de documentos del nodo pertenecientes al grafo objetivo.

Los pesos w_d y w_i indican la contribución de TotalSD (similaridad de los documentos intercambiados) y $SimId$ (similaridad de los nombres de las interacciones) respectivamente en la distancia total del nodo ($0 \leq w_d \leq 1$ y $0 \leq w_i \leq 1$).

4.2.5. Comparación Lingüística de los atributos WSCL

La comparación lingüística calcula la similaridad lingüística entre dos etiquetas basándose en sus nombres [81]. Las etiquetas se forman usualmente por una palabra o por la combinación de palabra y puede contener abreviaciones.

Para obtener la distancia lingüística entre dos cadenas, utilizamos algoritmos existentes: *NGram*, *Check synonym*, y *Check abbreviation*. El algoritmo *NGram* estima la similaridad de acuerdo con el número de qgram comunes entre los nombres de las etiquetas [167]. El algoritmo *Check synonym* utiliza un diccionario lingüístico (e.g. Wordnet [168] en nuestra implementación) para encontrar los sinónimos entre los nombres de las etiquetas mientras que *Check abbreviation* utiliza un diccionario de abreviaturas acorde con el dominio de aplicación.

Primero, las cadenas son divididas en tokens (unidades básicas) con base en la puntuación y las letras mayúsculas. Las palabras innecesarias se remueven de la lista de tokens, utilizando una lista de palabras predefinidas. Si estos tokens individuales no pueden ser emparejados, estos son divididos utilizando el algoritmo Porter Stemmer y se trata de emparejarlos utilizando la técnica *NGram*. Si alguno de estos algoritmos retorna un emparejamiento completo 1 en la escala de 0 ó 1, entonces se retorna un puntaje de 1 para la similaridad lingüística. Por otro lado, si el algoritmo retorna 0, significa que no existe emparejamiento entre las etiquetas. Si el valor *NGram* y el valor de *Check abbreviation* son iguales a 0, y el valor de *Check Synonym* está entre 0 y 1, la similaridad lingüística total será igual al valor de *Check Synonym*. Finalmente, si los valores de los 3 algoritmos están entre 0 y 1, la similaridad lingüística LS es el promedio de ellos.

$$\begin{array}{l}
 \frac{1}{3} (s_1 a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_2 a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_3 a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_4 a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_5 a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_6 a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_7 a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_8 a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_9 a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{10} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{11} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{12} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{13} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{14} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{15} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{16} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{17} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{18} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{19} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{20} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{21} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{22} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{23} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{24} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{25} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{26} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{27} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{28} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{29} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{30} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{31} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{32} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{33} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{34} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{35} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{36} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{37} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{38} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{39} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{40} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{41} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{42} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{43} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{44} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{45} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{46} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{47} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{48} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{49} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{50} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{51} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{52} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{53} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{54} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{55} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{56} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{57} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{58} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{59} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{60} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{61} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{62} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{63} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{64} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{65} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{66} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{67} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{68} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{69} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{70} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{71} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{72} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{73} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{74} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{75} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{76} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{77} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{78} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{79} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{80} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{81} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{82} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{83} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{84} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{85} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{86} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{87} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{88} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{89} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{90} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{91} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{92} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{93} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{94} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{95} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{96} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{97} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{98} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{99} a_1 + m_1 a_2 + m_2 a_3) \\
 \frac{1}{3} (s_{100} a_1 + m_1 a_2 + m_2 a_3)
 \end{array}$$

Donde, $m_1 = \text{Sim}(\text{NGram})$, $m_2 = \text{Sim}(\text{Sinónimos})$ y $m_3 = \text{Sim}(\text{Abreviaciones})$.

Existen otras formas posibles de medir la similaridad de nombres: el algoritmo de distancia de edición de Levenshtein, técnicas tomadas del área de recuperación de información como TF-IDF o una combinación de estas técnicas. Sin embargo, definir una función clara para la similaridad sintáctica entre cadenas está más allá del alcance de este trabajo, puesto que nos centramos en la similaridad de comportamiento.

4.2.6. Comparación de los niveles de Granularidad de las interacciones WSCL Mapeadas

El isomorfismo de subgrafos ec (emparejamiento por isomorfismo de subgrafos por corrección de error) se aplica a los grafos que fueron expandidos, i.e., contiene solo interacciones atómicas Receive o Send. La comparación de granularidad verifica si las interacciones que fueron mapeadas por el algoritmo de isomorfismo de subgrafos ec tienen el mismo nivel de granularidad en ambos modelos. Por ejemplo, suponga que en un grafo objetivo tenemos una interacción *SendReceive*. Esta fue dividida por la función de descomposición en una interacción Send seguida de una interacción Receive que fueron mapeadas con dos interacciones correspondientes el grafo consulta (por el algoritmo de isomorfismo de subgrafos ec). Si estas interacciones fueran atómicas en el grafo consulta, el costo de la operación de unión debe ser agregada a la distancia total de grafos. (Línea 5 en la Tabla 4.2).

Los costos para las diferencias de granularidad que tienen que ser tomados en cuenta para la distancia total para todos los casos (interacciones atómicas contra no atómicas en el grafo consulta y objetivo) se resumen en la Tabla 4.2.

Tipo de interacción del grafo consulta	Tipo de interacción del grafo objetivo	Costo de diferencia de granularidad
S atómico	S atómico	0
R atómico	R atómico	0
SR	SR	0
RS	RS	0
SR (or RS)	S atómico + R atómico	c_j
SR (or RS)	S no atómico. + R no atómico.	$c_d + c_j$
SR (or RS)	S no atómico. + R atómico ó S atómico. + R no atómico.	$c_d/2 + c_j$
S atómico	S no atómico	$c_d/2$
R atómico	R no atómico	$c_d/2$

S=Send, R=Receive, SR= SendReceive, RS=ReceiveSend

Tabla 4.2 Costos para las diferencias de granularidad

Por motivos de claridad, la tabla no presenta los casos para interacciones que no tienen correspondencia en el otro grafo. Si las interacciones mapeadas tienen el mismo nivel de granularidad (ambos tienen atómicas o no atómicas) no se agrega costo a la distancia de edición. Un caso más complicado (línea 7 en la tabla 4.2) es cuando una interacción *SendReceive* SR_1 en el grafo objetivo se mapea con una interacción atómica *Send* SM seguida de una interacción *Receive* RM que no es

atómica (pertenece a un *SendReceive* SR_M) en el grafo consulta. En este caso, el costo es $c_j + c_d/2$ (c_j = costo de unir S_M y R_M ; $c_d/2$ = costo de obtener R_M por descomposición de SR_M en el grafo consulta).

4.3. Resumen

En este capítulo explicamos nuestra aproximación basada en grafos para el emparejamiento de comportamientos y se mostró una aplicación de este. Primero, introducimos el problema de emparejamiento de grafos explicando su definición y notación utilizada dentro de la teoría de grafos. Como nuestra propuesta para el descubrimiento de servicios se centra en el emparejamiento aproximado de servicios, se describe además un método para medir la similitud de dos grafos. A continuación se presenta en detalle el algoritmo de emparejamiento de subgrafos por corrección de error (Algoritmo EC). Con el fin de expandir el algoritmo-ec al emparejamiento de comportamiento de servicios, presentamos dos extensiones. La primera se centra en el nivel de granularidad de los servicios, mientras que la segunda extensión se refiere a las medidas de similitud. Tales medidas tienen que tener en cuenta el número de interacciones o el número de secuencias de interacciones en el servicio objetivo que sean cubiertos por el servicio de consulta.

Finalmente, mostramos un caso de aplicación de este algoritmo a los protocolos WSCL (Web Services Conversation Language), aunque la misma aproximación puede aplicarse a otros modelos. El proceso de emparejamiento de protocolos de conversación se compone de los siguientes pasos: primero, los protocolos de conversación a comparar se transforman en grafos; posteriormente, ambos grafos se expanden con el fin de conseguir el mismo nivel de granularidad; luego se aplica el algoritmo de emparejamiento de grafos por corrección de error y se evalúa la función de similitud entre los grafos. Finalmente, se comparan los niveles de granularidad y se agregan los costos correspondientes a las diferencias identificadas al costo total.

Capítulo 5. Descubrimiento de Servicios Web geográficos ambientales basado en calidad del servicio

Si los servicios no pueden ser encontrados y entendidos apropiadamente, prácticamente no existen. Aun si un servicio es encontrado por un usuario, no podrá interactuar efectivamente con él, si aparte de conocer su funcionalidad, no es capaz de determinar la calidad del servicio. Preguntas como ¿quién lo creo?, ¿para qué fue creado?, ¿que pretende hacer?, ¿quién lo soporta?, ¿cuánto cuesta?, ¿cuál es su reputación?, son importantes para determinar su utilidad en una aplicación específica. Distinguir y diferenciar la calidad de los servicios es tan importante como descubrirlos; particularmente en el campo de la gestión ambiental del recurso hídrico como se discutió en el capítulo uno.

La calidad del servicio (QoS) está ligada al proceso de descubrir y componer servicios. Todos los trabajos de investigación en el área del descubrimiento de servicios Web geográficos (Ver sección 2.3), destacan la necesidad de considerar los atributos de calidad para el descubrimiento de servicios, con el fin de realizar búsquedas más precisas. Este factor es particularmente importante para el dominio geográfico, donde la información sobre calidad es un factor decisivo para conocer y caracterizar adecuadamente un conjunto de datos y servicios geográficos de una organización determinada [169].

Por otro lado, muchas aproximaciones en este campo de investigación ([97],[126], [127]), proponen la adición de parámetros asociados con la calidad del servicio (e.g. precio, tiempo de respuesta) a las descripciones básicas de los servicios (e.g las especificadas utilizando WSDL) o de los procesos (e.g las especificadas utilizando BPEL). Este hecho hace que el proceso de realizar emparejamiento basado en calidad del servicio pueda implementarse como una “capa extra” sobre los demás niveles de descubrimiento. Por este motivo, es muy común que se conjugue el emparejamiento basado en calidad del servicio con otros tipos de emparejamiento, para formar acercamientos de emparejamiento de tipo híbridos (Ver sección 2.3.2).

En vista de la relación intrínseca existente entre los criterios de calidad del servicio y el dominio de aplicación específico (criterios como tiempo de respuesta, o confiabilidad pueden ser irrelevantes para sistemas de soporte a redes sociales, pero son indispensable para áreas como las aplicaciones como domótica o el comercio electrónico), para poder desarrollar mecanismos de descubrimiento de servicios basados en calidad del servicio, es indispensable la definición de un perfil de calidad propio para cada dominio. En este capítulo se presenta precisamente un perfil de calidad para servicios geográficos ambientales hidrológicos, y se extienden las especificaciones de Metadatos geográficos para incluirlo.

El resto de este capítulo se encuentra organizado de la siguiente forma: primero se presenta el perfil de calidad para los servicios geográficos ambientales hidrológicos, luego se presenta una descripción del proceso seguido para la definición de nuestra arquitectura de descubrimiento de

servicios hidrológicos basado en criterios de calidad del servicio; posteriormente, se presenta y describe la arquitectura de descubrimiento de servicios geográficos ambientales basado en QoS; y finalmente, se muestra como dicha arquitectura se integra a nuestra propuesta de descubrimiento basada en comportamiento descrita en el capítulo anterior, ambas enmarcadas dentro del metamodelo descrito en el capítulo dos.

5.1. Perfil de QoS para los servicios geográficos ambientales.

En la Tabla 5.1 se describen el conjunto de atributos escogidos para conformar un perfil de QoS para ser utilizado dentro de la arquitectura propuesta en este trabajo, teniendo en cuenta las perspectivas de usuario, aplicación y recurso, pero segregándolas en dos subniveles de gestión distintos denominados nivel cuantificable y nivel cualitativo.

De manera general, la selección de estos atributos atiende a su relevancia para el dominio de la gestión ambiental del recurso hídrico, en donde atributos como: la confiabilidad, disponibilidad, reputación y conformidad son claves para el proceso de toma de decisiones. Mientras que, aspectos como costo, desempeño, localización e interactividad, son atributos que garantizan que la interacción entre los servicios sea llevada a cabo de manera óptima y fácil.

Niveles/ Subniveles	Cuantificable	Cualitativo
Usuario	Costo	Reputación, interactividad, localización
Aplicación	Desempeño, Confiabilidad, Disponibilidad	Conformidad

Tabla 5.1. Conjunto de atributos del perfil de QoS.

Dentro del nivel cuantificable se agrupan los atributos de calidad por naturaleza medibles. Estos son importantes para establecer acuerdos de nivel de servicio (SLA) y estimaciones combinadas en procesos de composición como se describe en [119]. Se incluyen las propiedades típicas a nivel de recurso y aplicación relacionadas con las operaciones de transporte, comunicación y procesamiento que se llevan a cabo para prestar el servicio. Estos son: desempeño, confiabilidad, disponibilidad y costo. A continuación se describen cada uno de estos atributos:

- El *Desempeño* es la consecución del servicio de forma adecuada y en el tiempo previsto. A nivel de aplicación, es la duración de tiempo en segundos entre el instante en que una solicitud es realizada y el instante en que la respuesta está disponible [129]. Se incluye en el perfil porque es un parámetro importante en el momento de estimar la duración en la planeación de composiciones dinámicas [129],[170].
- La *Confiabilidad* es una característica típicamente de nivel de aplicación y representa la probabilidad de que el servicio sea ejecutado correctamente o adhiera acuerdos de nivel de servicio [129],[170].
- La *Disponibilidad*, es el porcentaje de tiempo en que el servicio esta accesible [170]. También se utiliza al definir acuerdos de nivel de servicio.
- El *Costo* se define como el precio cobrado por ejecutar la operación [129],[170]. Existen diferentes esquemas de cobro como por ejemplo: valor fijo, según el tiempo empleado en proveer el servicio, cantidad de datos solicitados, etc. Es utilizado a nivel de aplicación para estimar costos en servicios de composición [170].

En el Nivel Cualitativo se agrupan los atributos de calidad percibidos por los usuarios, que se pueden cualificar pero cuyas valoraciones dependen del dominio de aplicación y el contexto en que se usen. Las valoraciones en este nivel no son absolutas y son: reputación, interactividad, localización y conformidad. A continuación se describen cada uno de estos atributos:

- La *Reputación* es un juicio emitido con base en la experiencia en el uso, pre concepciones o recomendaciones del usuario [171]. Se define como la medida de fiabilidad percibida del servicio. Se incluye en el esquema porque permite utilizar la preferencia de los usuarios en el descubrimiento de contenidos [172].
- El *Soporte de interactividad*, [170] (también conocido en ambientes multimedia como usabilidad [173]), representa el grado de acción que puede ejercer una persona que interactúe con el servicio. Se incluye porque es un parámetro clave en el descubrimiento de servicios a nivel de abstracción de usuario [173]. Este parámetro depende estrictamente de los objetivos del servicio, por ejemplo: una evaluación de interactividad de un servicio móvil, no es comparable con un servicio tradicional Web.
- La *localización* es la región geográfica a la que se asocian los datos entregados por el servicio. Es definido por [170] y aunque no es estrictamente un criterio de calidad, es un aspecto importante a nivel de aplicación para definir la procedencia de datos en servicios Web de procesamiento o consulta. Por ejemplo un servicio de muestras hidrológicas que entregue datos específicos de acuerdo a la posición.
- La *Conformidad* se incluye para especificar el estándar y versión que caracterizan al servicio. Aunque no se ha tomado en cuenta como atributo de calidad geográfica por algunos autores [170], este atributo es utilizado típicamente a nivel de aplicación en contextos multimedia [174] y servicios digitales [173] para conocer las capacidades un servicio de forma temprana. Se incluye en el perfil porque permite realizar una valoración previa de funcionalidad, antes de acceder a las interfaces programáticas.

5.2. Proceso de definición de la arquitectura propuesta

En este trabajo partimos de la premisa que la calidad de servicio a nivel de aplicación puede definirse utilizando descripciones esquematizadas en forma de metadatos asociados a los servicios. En este sentido, el perfil de calidad para servicios geográficos ambientales presentado anteriormente se implementará como una extensión de las especificaciones de Metadatos geográficos utilizando repositorios de servicios Web estándares como UDDI. Esta sección describe cada uno de los elementos tenidos en cuenta para el diseño de la arquitectura propuesta. A continuación se presenta una revisión de los metadatos geográficos, así como una comparación de los repositorios de servicios más utilizados en el campo de los servicios Web.

5.2.1. Metadatos para datos y servicios geográficos

Según [175], los datos describen el mundo real y son un modelo de la realidad; los metadatos en cambio describen los datos y se utilizan para tomar decisiones acerca de los mismos. Los metadatos constituyen la información, en forma documentada y estructurada que permite que los datos sean entendidos, compartidos y explotados de manera eficaz por todo tipo de usuarios a lo largo del tiempo. Se utilizan para poder identificar, acceder y usar los datos. Describir un recurso en determinado contexto, como en el caso de la calidad de servicio, requiere la adición de

descripciones en forma de metadatos. Una definición apropiada de los metadatos espaciales puede ser: “una leyenda mucho más detallada que la del mapa en formato papel, que describe los autores que han producido los datos, las fuentes documentales utilizadas en la producción, los atributos que poseen los datos, la fecha de publicación, el sistema de referencia, la frecuencia de mantenimiento, etc.” [176]

Los principales usos de los metadatos son: Organizar un conjunto de datos, proporcionar información para catálogos de datos y centros de distribución de metadatos (clearinghouse) y proveer información necesaria para interpretar y procesar datos transferidos por otra organización. Los metadatos pueden ser clasificados en tres grandes categorías [176]:

- Metadatos de Descubrimiento: Identifican los datos disponibles.
- Metadatos de Exploración: determinan si el conjunto de datos poseen las características necesarias para determinado propósito.
- Metadatos de Explotación: permiten conocer el procedimiento de adquisición y utilización de los datos.

Nuestra propuesta de metadatos, intenta describir para cada servicio: sus potencialidades, datos disponibles, características (de calidad del servicio) y comportamiento. Además las descripciones básicas de los servicios (WSDL, BPEL) cuentan generalmente con información sobre el acceso a los servicios. Por lo anterior, nuestra propuesta de metadatos ofrece información de las tres categorías mencionadas anteriormente.

Cuando se realizan búsquedas en la Web, los resultados obtenidos muchas veces difieren de lo que se espera obtener. Pueden existir incongruencias en los datos, la autenticidad de la información y del sitio Web y la precisión de éste. Para mejorar las búsquedas, existen varios lenguajes estándar, basados en XML, orientados a asociar información descriptiva que relacione el contenido que se desea obtener de las búsquedas. Los estándares más populares, principalmente son: Dublin Core, PRISM (Publishing Requirements for Industry Standard Metadata), XMP (Extensible Metadata Platform) y RDF (Resource Description Framework).

Para la descripción de información geográfica también existen una serie de lenguajes estándar basados en XML, generados por la ISO (Organización de Estandarización Internacional), que permiten agregar información a cualquier tipo de recursos. La familia de normas ISO 19100 es una familia de normas internacionales para la información geográfica desarrollada por el Comité Técnico 211, llamado Geomática/Información Geográfica dentro de de la Organización de Estandarización Internacional. Las normas relacionadas con los metadatos son principalmente tres: ISO 19115, 19119 y 19139.

El estándar 19115, permite documentar los datos geográficos digitales, proporcionando un esquema y un conjunto común de terminología, que permita realizar una correcta gestión y explotación de los datos. No sólo para uso interno de una organización, sino para poderlos compartir e intercambiar entre los diferentes productores y usuarios. Los metadatos son almacenados en catálogos o bases de datos distribuidas en forma de documentos XML definidos por la norma ISO 19115, denominada Metadatos de Información Geográfica. de acuerdo con el estándar 19115 los metadatos se encuentran organizados en siete categorías: Información del Metadato (fechas de creación y actualización del metadato, estándar y la versión en que está escrito, las restricciones de uso y acceso de los datos, los sistemas de seguridad y la información de contacto del responsable de los datos); Información de identificación (conjunto de datos, periodo temporal, la actualidad, el

dominio espacial y palabras clave para indexación de registros); Información de calidad de los datos (estado de consistencia y completitud de los datos); Información de la representación espacial (raster o vectoriales, dimensiones, resolución, escala); Información del sistema de referencia; Información de contenido (entidades y atributos de los datos descritos); Información de distribución (contacto, horarios costo). Por su parte, el pre-estándar ISO 19139 desarrolla una implementación en XML del modelo abstracto de metadatos descrito por ISO 19115 y, como tal, define un conjunto de esquemas que declaran elementos para describir los metadatos de una capa. [177] [178]

De otro lado, el borrador de especificación ISO 19119 identifica y define los patrones arquitecturales de las interfaces de servicio usadas en información geográfica, describiendo la taxonomía de este tipo de servicios. ISO 19119 provee un marco de trabajo abstracto que permite realizar el desarrollo coordinado de servicios específicos, habilita la interoperabilidad de servicios mediante la estandarización de las interfaces, soporta el desarrollo de un catálogo de servicios mediante la definición de metadatos a nivel de servicios, separa las instancias de datos y servicios y permite el uso de proveedores de servicios y de datos independientes. El marco de trabajo descrito por ISO 19119 abre la puerta a la construcción de soluciones software que permitan a los usuarios acceder y procesar datos geográficos procedentes de una variedad de fuentes, utilizando una interfaz computacional genérica, en ambientes tecnológicos abiertos. A nivel nacional, el Instituto Colombiano de Normas Técnicas y Certificación ICONTEC ha generado una norma para metadatos Geográficos (NTC 4611) con el objetivo de definir el esquema requerido para describir la información geográfica. Esta norma es una adaptación de la norma internacional ISO 19115, por tal motivo categoriza el metadato geográfico en una jerarquía de relaciones que comprende nueve secciones, similares a la norma de ISO.

Nuestra arquitectura implementará los estándares: ISO 19115 y NTC 4611, el borrador de especificación ISO 19119 no será tenido en cuenta por dos razones: la primera es: en el momento de escribir este documento se encuentra en fase preliminar de desarrollo; la segunda es que gran parte de los lineamientos de la especificación ISO 19115 puede adaptarse al descubrimiento de servicios y sirven de base para el borrador ISO 19119.

Atendiendo a que en este trabajo argumentamos que la especificación de los criterios de calidad del servicio puede integrarse a las descripciones existentes en los repositorios de servicios, el siguiente paso en el proceso de definición de la arquitectura es la revisión de los dos estándares mas importantes en el área de los repositorios de servicios Web geográficos: UDDI y el registro ebXML.

5.2.2. Repositorios para servicios Web geográficos

Las búsquedas en Internet se basan en la interpretación y almacenamiento de etiquetas de páginas Web y los links que apuntan a estas páginas. Sitios Web de búsquedas como Google o Yahoo almacenan en sus bases de datos registros con la información necesaria de forma que facilite la consecución de los datos. En el caso de los Servicios Web, se delega esta tarea al registro de servicios que intenta optimizar el mecanismo de búsqueda usando descripciones estándar de los servicios basadas en XML. Mediante un registro especializado, un usuario puede encontrar servicios de acuerdo al nombre de la compañía que lo presta, el nombre del servicio o la URL. Al buscar un servicio específico, los usuarios pueden consultar las características de los Servicios Web y determinar cuales de ellos se ajustan completamente a sus necesidades antes de usarlos. Estos registros tienen mecanismos para asociar metadatos descriptivos o categorías a la información, por esta razón, En el presente trabajo se utilizan este tipo de repositorios de servicios Web, para almacenar descripciones sobre las características de calidad del servicio, en forma de metadatos.

Hoy en día, las implementaciones más importantes de los repositorios de Servicios Web geográficos se basan en los dos registros más difundidos en el campo de los Servicios Web convencionales: el registro de ebXML y el registro UDDI. Por ejemplo, la implementación del *servicio de catálogo* (CS-W) propuesto por la OGC, se acoge al metamodelo del registro ebXML [147]. A pesar de que la OGC respalda algunas propuestas de implementación de repositorios de Servicios Web geográficos como la mencionada anteriormente, también es categórico al afirmar que ningún modelo satisface completamente las necesidades del contexto geográfico [179] por lo que deja abierta la posibilidad de usar UDDI, mecanismos basados en CORBA, Z39.50 [107] y búsquedas de texto no estructurado como motores de búsqueda y descubrimiento de servicios geográficos. A continuación se presenta una revisión de los repositorios más importantes en el área de los Servicios Web:

Repositorio de Servicios UDDI

La especificación UDDI fue creada como un conjunto de modelos de datos, codificados en XML, organizados en una jerarquía con el objetivo de facilitar las búsquedas de registros. UDDI no es un repositorio de datos o servicios, su función es únicamente la de almacenar información para encontrar e invocar un servicio. UDDI define unas estructuras de datos principales que son pasadas como parámetros de entrada y salida en los mensajes de la API. Estas estructuras de datos están organizadas jerárquicamente, A continuación se describe la información que representa cada una de estas estructuras [180]:

businessEntity (Organización): representa la información básica de una organización.

publisherAssertion(Declaración): usada para establecer relaciones públicas entre dos entidades de negocio.

businessService (Servicio): representa una clasificación de servicio a nivel lógico.

<bindingTemplate> (Plantilla del contrato): contiene apuntadores a descripciones técnicas y a la URL que sirve como punto de acceso al servicio, no contiene detalles de las especificaciones de los servicios.

<tModel> (Modelo Técnico): Descripción abstracta de una especificación o comportamiento en particular al que un Servicio Web se adhiere. Determina específicamente como se interactúa con un Servicio Web en particular. Este, sin embargo, no provee directamente la especificación del Servicio Web, en lugar de esto, contiene apuntadores a las direcciones de las especificaciones. Este elemento es utilizado por las compañías para determinar que tan compatible es un Servicio Web con sus requerimientos.

Para facilitar la búsqueda de recursos dentro del registro, UDDI permite asignar categorías a la información; sin esta categorización, localizar datos dentro del registro UDDI sería muy difícil. La categorización es la característica más importante en UDDI, puesto que es la forma en que la especificación atribuye metadatos a los servicios registrados [181]. Para el descubrimiento de negocios, servicios, bindings (contratos) o tipos de servicios desconocidos con anterioridad, es indispensable que los datos registrados correspondientes sean marcados en una serie de categorías que puedan ser buscadas universalmente. Un ejemplo de las clasificaciones utilizadas se encuentra en la UNSPSC (Universal Standard Products and Services Classification), la cual define un conjunto de códigos que representan productos y categorías, que son usados para especificar un producto, o servicio ofrecido por un negocio de manera formal o normalizada.

Repositorios ebXML

A diferencia de los Servicios Web convencionales, ebXML implementa estándares a medida creados por los usuarios para el manejo eficiente del contenido informativo y los metadatos asociados, a su vez, provee mecanismos flexibles para el descubrimiento de contenido y la entrega eficaz del mismo a los usuarios y organizaciones. EbXML facilita una cooperación automática entre aplicaciones empresariales distribuidas. Soporta los principios de la infraestructura B2B: intercambio de mensajes electrónicos basado en XML, acuerdos entre socios comerciales y soporte para registros. Esta iniciativa es un esfuerzo del UN/CEFACT (Centro de las Naciones Unidas para Facilitar el Comercio y los Negocios Electrónicos) y OASIS (Organización para el Avance de la Estructura de Estándares de Información). EbXML también permite a las empresas encontrar socios mediante mensajes electrónicos basados en XML y posibilita los procesos de negocio. Una de las diferencias principales con UDDI es que ebXML también es un repositorio de datos. Desde una perspectiva más amplia, ebXML puede considerarse como un sistema de gestión que cumple las funciones básicas de un repositorio para sistemas distribuidos y abiertos. Dos elementos muy importantes de la especificación de ebXML Registry son El Modelo de Registro de Información (RIM) y el registro de Servicios y Protocolos; el primero especifica los metadatos y el contenido que puede ser almacenado en el registro ebXML; Establece perfiles de registro para dominios específicos, definiendo restricciones y extensiones en el uso del mismo, entre ellos se tienen: servicios Web, HL7 y el perfil para SIG. Por su parte, el *registro de Servicios y Protocolos ebXML* define las interfaces y los servicios provistos por ebXML, como también una API para el acceso.

Comparación de los repositorios de servicios Web

UDDI se ha impuesto de facto como modelo de registro de Servicios Web en la mayoría de las compañías porque es más fácil de utilizar, esto se debe a que define una estructura de datos fija que simplifica su manejo. Al ser comparada con el registro de ebXML, no se puede hablar de superioridad de uno sobre otro, en realidad lo que existe es una diferencia de enfoque, y en muchos casos se complementan [182]. El modelo del registro de ebXML fue diseñado como registro y repositorio de recursos, lo que incluye el almacenamiento de interfaces, objetos, documentos y demás productos generados a nivel corporativo, que dan mayor robustez en etapas posteriores a la realización del contrato con el proveedor. UDDI por otro lado es un registro estrictamente hablando, que procura simplicidad en su estructura para garantizar integraciones rápidas entre compañías, focalizándose en facilitar los procesos de búsqueda y descubrimiento en la etapa previa a la consecución del contrato con un proveedor.

Como este trabajo se centra en el emparejamiento de servicios geográficos, y considerando primero: que el OGC es categórico al afirmar que ningún modelo satisface completamente las necesidades del contexto geográfico [179] deja abierta la posibilidad de usar UDDI, y segundo; que en el dominio de aplicaciones SIG, las especificaciones del OGC promueve el uso de estándares abiertos como SOAP, WSDL y UDDI. Consideramos que el estándar UDDI provee una arquitectura flexible, abierta y sencilla, que permitirá a futuro extender nuevas capacidades. Por lo tanto, se utilizará UDDI en la arquitectura de descubrimiento basada en QoS.

5.3. Arquitectura para el descubrimiento de servicios Web geográficos basada en QoS

En esta sección se presenta nuestra propuesta para el descubrimiento de servicios Web geográficos basada en QoS. La arquitectura que se detalla a continuación tiene como núcleo central un repositorio de servicios, el cual permite enriquecer la descripción de los Servicios al incluir

información sobre Metadatos ISO 19115 y NTC 4611 en un repositorio de Servicios de manera que se puedan manejar descripciones de calidad del servicio para el entorno ambiental.

El resto de esta sección se organiza de la siguiente forma: primero se presenta una vista general de la arquitectura propuesta, posteriormente se explica detalladamente el modelo de información utilizado en el registro UDDI, y finalmente, se muestra como la arquitectura para el descubrimiento de servicios Web geográficos basado en QoS, se integra con la propuesta de descubrimiento basada en comportamiento presentada en el capítulo 4.

5.3.1. Arquitectura para el descubrimiento de servicios Web geográficos basada en QoS

La demanda de Servicios Web geográficos más complejos, que involucran encadenamiento y operación en conjunto de aplicaciones y servicios, requiere de catálogos que definen esquemas específicos de metadatos ajustados al ámbito del servicio[183]. En este sentido, la especificación del catálogo OGC (uno de los referentes más importantes en el área de los SIG), describe el servicio orientado a gestionar metadatos como un elemento fundamental. Dentro de este catálogo, se pueden almacenar metadatos sobre uno o más tipos de elementos, como pueden ser: servicios geográficos, coberturas, contextos de Mapas, perfiles GML, documentos WSDL, documentos BPEL, entre otros. En general, los elementos que se pueden describir no están limitados por la especificación.

Por otro lado, la arquitectura multicapa distribuida, propuesta por la OGC para la composición de servicios geográficos, deja abierta la posibilidad de realizar el descubrimiento de servicios mediante un componente desplegado en la capa aplicación que haga las veces de agente o facilitador (broker) del servicio de descubrimiento; nuestra propuesta, busca definir justamente un broker encargado de realizar el descubrimiento de servicios geográficos, este descubrimiento se basa en los metadatos geográficos y de calidad, compatibles con las especificaciones ISO19115 y NTC4611.

Las características más importantes que debe tener la arquitectura propuesta son las siguientes:

- Centrada en la red: utiliza protocolos W3C que habilitan la comunicación entre diferentes Servicios Web geográficos desplegados, esto permite compartir componentes y componer Servicios de forma abierta y estándar.
- Manejo de especificaciones de metadatos estándar para calidad e información geográfica, que se almacenan de forma centralizada.
- Estructura en múltiples capas, que agrupan diferentes niveles de funcionalidad para procesamiento geográfico.
- Habilidad como agente o Bróker para descubrimiento de servicios web geográficos para la gestión ambiental del recurso hídrico, mediante atributos de Calidad del servicio.

Nuestro sistema de descubrimiento basado en calidad de servicios, permite el registro de información básica sobre los servicios, así como información adicional sobre calidad del servicio y metadatos geográficos. De esta manera, los usuarios de los servicios cuentan con un punto central de consulta acerca de las características funcionales y no funcionales de los servicios. La Figura 5.1 muestra la arquitectura de descubrimiento de servicios geográficos ambientales propuesta basada en [129].

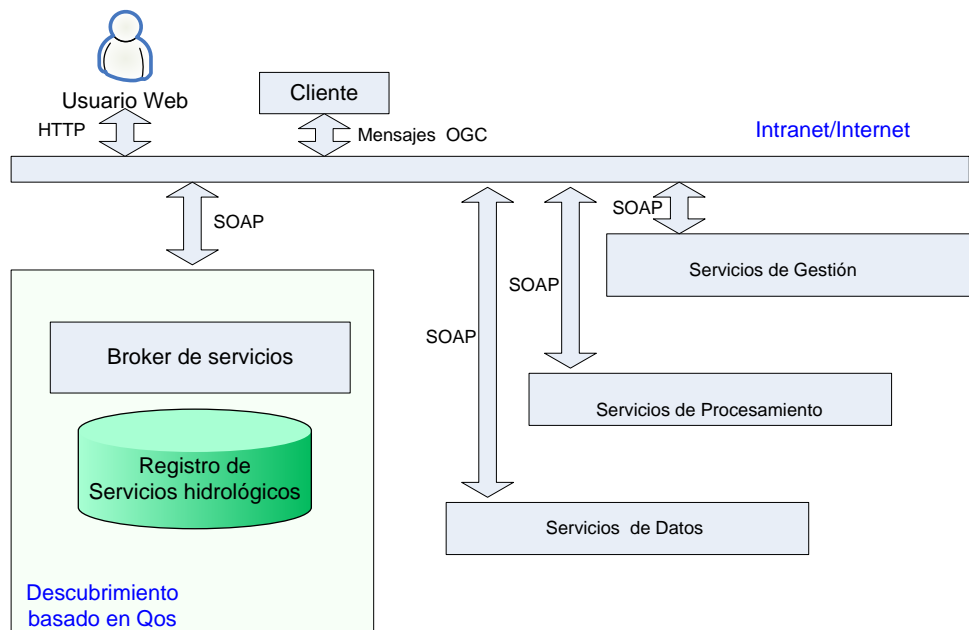


Figura 5.1. Vista General de la arquitectura.

A continuación se describen los elementos restantes de la arquitectura:

Usuario Web: persona que utiliza las funcionalidades del broker de descubrimiento a través de una interfaz Web.

Cliente: Aplicación software externa que utiliza las prestaciones del catalogo mediante las interfaces remotas definidas por la especificación OGC-CSW. Un cliente común puede ser un Gestor de composición de Servicios [129]. Este gestor es el encargado de planear y ejecutar los esquemas de composición de servicios, su funcionamiento se basa en dos mecanismos: Un gestor, encargado de generar el plan de composición y un operador, que ejecuta el plan generado por el componente anterior. En el momento de realizar la composición, el operador contacta al “*broker de servicios*” para identificar los servicios candidatos para realizar las tareas del proceso

Broker de servicios: Permite a los proveedores publicar la descripción de Servicios Web en el registro, además se encarga de la búsqueda de servicios.

Registro de servicios Hidrológicos: Almacena físicamente las descripciones de los servicios.

Como puede verse en la figura 5.1, existe un conjunto de servicios Web geográficos para la gestión ambiental del recurso hídrico, estos han sido clasificados en

- *Servicios de acceso a datos*, como por ejemplo los registros de *Muestras* hidrológicas.
- *Servicios de Procesamiento*: los cuales contienen servicios orientados a realizar cálculos y procesos. Como por ejemplo el cálculo de *Índices de Calidad del Agua (ICA)* o el *conversor de coordenadas*.
- *Servicios de Gestión*: los cuales soportan requerimientos directos de los clientes y usuarios facilitando el reuso, por ejemplo: *Workflow* para definir procesos de composición, *gestor de composición* para gestionar su ejecución, el *servicio de Balances Hídricos* para la toma de decisiones en abastecimiento de Agua.

5.3.2. El broker de servicios

El *broker de servicios* permite a los proveedores de servicios, publicar la información sobre sus servicios Web en un repositorio, además, se encarga de realizar las búsquedas de servicios dentro de este mismo registro. Este componente se basa en el broker presentado en [129]. Para soportar descripciones de servicios geográficos y sus atributos de calidad. El broker toma como base el modelo de información básico de UDDI, y extiende la descripción de los servicios utilizando los siguientes elementos:

- El *Perfil de calidad para el dominio ambiental* que contiene la información sobre los criterios de calidad del servicio para el dominio ambiental hidrológico.
- El *Servicio de Metadatos* que gestiona las descripciones de servicios geográficos con base en la especificación ISO 19115 y NTC4611.

Con el fin de conseguir la inclusión de información sobre los metadatos definidos por los estándares ISO 19115 y NTC 4611 y sobre calidad del servicios, se modificó el modelo de información del registro UDDI, como se muestra en la figura 5.2, las tablas a la izquierda de la tabla servicio representan la extensión para los metadatos geográficos, mientras que las tablas a la izquierda extienden el modelo para soportar la consulta basada en criterios de calidad del servicio.

El *broker de servicios* cumple dos funciones fundamentales: registrar la información y soportar las búsquedas de servicios. Para el registro de información el broker ofrece una interfaz gráfica de usuario para el registro de la información básica, así como los parámetros de calidad del servicio y los metadatos geográficos (una implementación de este sistema puede verse en la sección 6.3).

Extensión para QoS

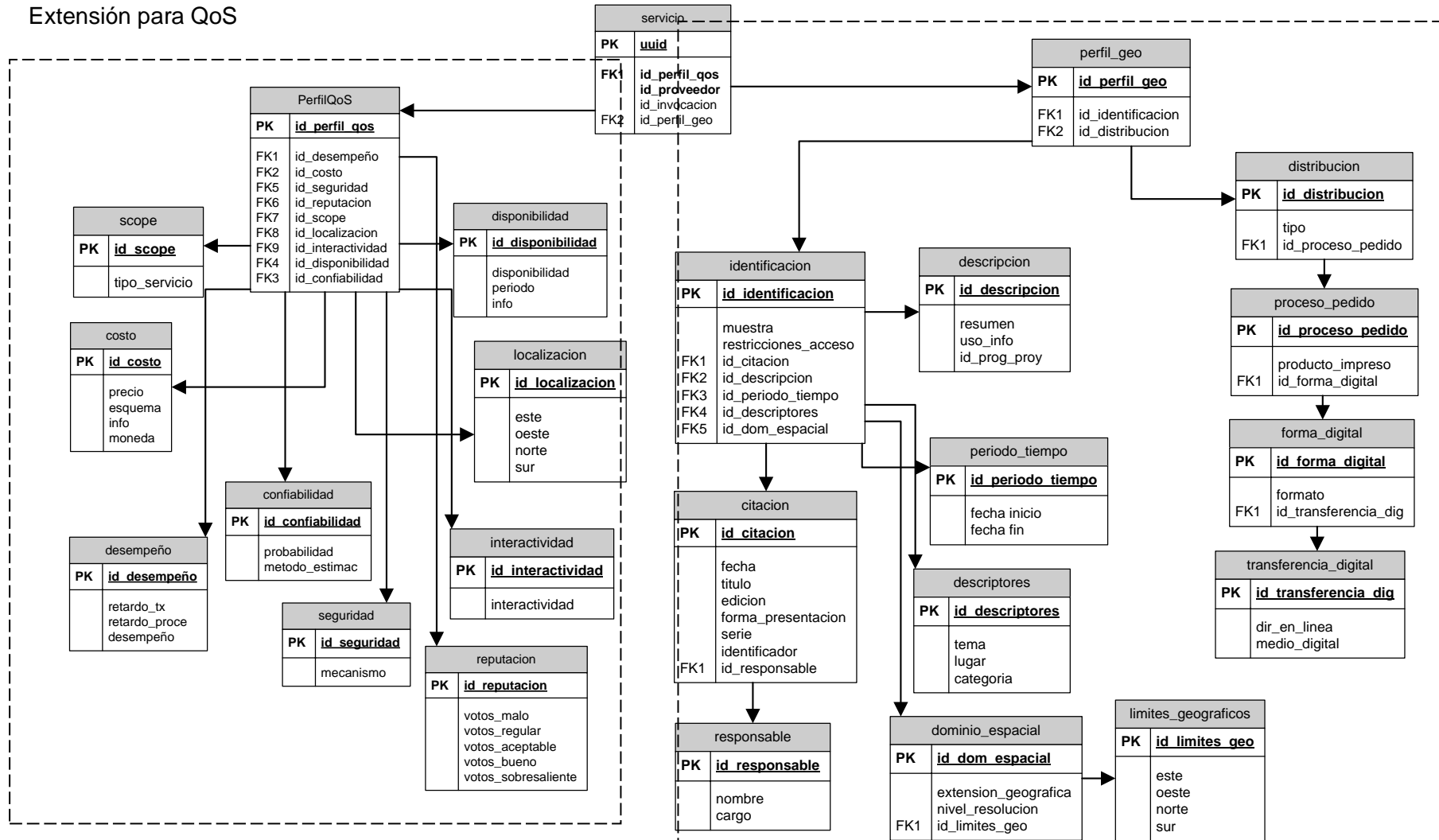


Figura 5.2. Modelo de información del sistema de descubrimiento.

A la hora de registrar un servicio, es necesario incluir la información básica sobre el servicio, para ello se puede utilizar un documento WSDL o WSCL, y adicionalmente, se debe agregar información sobre la calidad del servicio y los metadatos geográficos, lo cual se realiza en nuestra arquitectura (ver figura 5.1) utilizando un documento en formato XML como el que se muestra a continuación (ver figura 5.3):

```
<?xml version="1.0" encoding="utf-8" ?>
<QS_METADATA>
  <QS_Desempeño>
    <RetardoTx>1.2</RetardoTx>
    <RetardoPx>.25</RetardoPx>
    <Desempeño>1.45</Desempeño>
  </QS_Desempeño>
  <QS_Costo>
    <Precio>20</Precio>
    <Moneda>USD</Moneda>
    <Esquema>/Kbyte</Esquema>
    <Info> información adicional</Info>
  </QS_Costo>
  <QS_Confiabilidad>
    <Probabilidad>.65</Probabilidad>
    <Estimacion>Polansky</Estimacion>
  </QS_Confiabilidad>
  <QS_Disponibilidad>
    <Disponibilidad>50 min</Disponibilidad>
    <Periodo>60 min</Periodo>
    <Info>Información adicional</Info>
  </QS_Disponibilidad>
  <QS_Seguridad>
    <Mecanismo>Rest. Loggin, encripcion</Mecanismo>
  </QS_Seguridad>
  <QS_Reputacion>
    <Valoracion>No es mecanismo efectivo, cambiar</Valoracion>
  </QS_Reputacion>
  <QS_Interactividad>Mapping, zoom, popups</QS_Interactividad>
  <QS_Scope>Gestion, procesamiento, análisis, default</QS_Scope>
</QS_METADATA>
```

Figura 5.3. Información sobre QoS y metadatos de los servicios.

Por otro lado, el usuario del *broker de servicios* puede enviar sus consultas especificando cualquiera de los parámetros definidos en el modelo de información (una implementación de las capacidades de consulta del broker se presenta en el capítulo seis).

5.3.3. Arquitectura para el descubrimiento de servicios Web geográficos basada en QoS y comportamiento.

Nuestro trabajo buscar mejorar el proceso de descubrimiento de servicios Web geográficos ambientales hidrológicos, para ello se propone el enriquecimiento de los mecanismos de consulta existentes en la actualidad con información sobre el comportamiento de los servicios y con los

critérios de calidad del servicio del dominio de la gestión ambiental del recurso hídrico. Para la utilización de las consultas enriquecidas en el entorno ambiental se propone además la definición de un meta-modelo del proceso de gestión ambiental que sirva de base para el proceso de búsqueda.

Como se discutió anteriormente, todas las aproximaciones revisadas en el estado del arte, proponen la adición de ciertos parámetros asociados con la calidad del servicio a las descripciones básicas de los servicios (e.g las especificadas utilizando WSDL o WSCL). Este hecho hace que el proceso de realizar emparejamiento basado en calidad del servicio pueda implementarse como una capa sobre los demás niveles de descubrimiento, razón por la cual, es muy común que se conjugue el emparejamiento basado en calidad del servicio con otros tipos de emparejamiento, para formar acercamientos de emparejamiento de tipo híbridos. En nuestra arquitectura, primero se define un perfil de calidad de servicios para el entorno ambiental que se integra al módulo de descubrimiento basado en calidad de servicio. Este módulo se ubica como una capa o filtro (*broker de servicios* en la figura 5.1) sobre el módulo de descubrimiento basado en comportamientos WSCL descrito en el capítulo 4 (ver figura 5.4) Todo lo anterior, se enmarca dentro del metamodelo del proceso de gestión ambiental del recurso hídrico.

Como se puede apreciar en la arquitectura de la figura 5.4, el cliente de la arquitectura presentada anteriormente en la figura 5.1, se reemplazó por el módulo de descubrimiento de servicios de SIRA, atendiendo a que uno de los objetivos del presente trabajo es ofrecer la funcionalidad de descubrimiento al sistema SIRA. Este cambio es posible dado que dicho cliente es un componente genérico que hace uso de los servicios del sistema de descubrimiento. De la misma forma, el sistema de descubrimiento de servicios geográficos hidrológicos puede prestar sus servicios a cualquier otro sistema que requiera de este tipo de búsquedas especializadas.

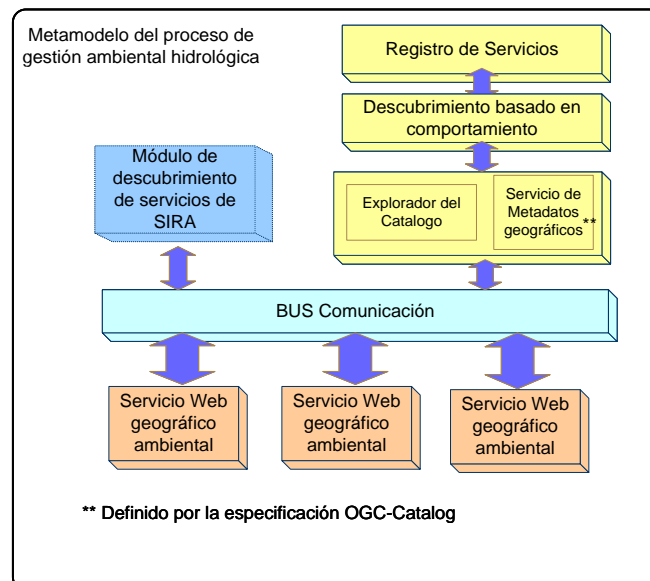


Figura 5.4. Arquitectura de descubrimiento para servicios Web geográficos hidrológicos basada en comportamiento y en QoS.

5.4. Resumen

En este capítulo se presentó nuestra propuesta para el descubrimiento de servicios Web geográficos hidrológicos, En primer lugar y atendiendo a la relación intrínseca existente entre los criterios de calidad del servicio y el dominio de aplicación específico se presenta el perfil de calidad para los servicios geográficos ambientales, cuyos atributos se seleccionaron teniendo en cuenta su relevancia para el dominio de la gestión ambiental del recurso hídrico, los atributos seleccionados son: la confiabilidad, disponibilidad, reputación, conformidad, costo, desempeño, localización e interactividad.

Posteriormente, se presenta una descripción del proceso seguido para la definición de nuestra arquitectura de descubrimiento de servicios geográficos ambientales basado en criterios de calidad del servicio, en primer lugar se presenta una revisión de los metadatos geográficos y posteriormente, se ofrece una comparación de los repositorios de servicios mas utilizados en el campo de los servicios Web: UDDI y ebXML. Para la primera parte, se estudiaron los estándares más importantes para documentar los datos geográficos digitales: a nivel internacional el ISO 19115, y a nivel nacional, el estándar NTC 4611. Por otro lado, luego de comparar los repositorios de servicios Web, se estableció que el estándar UDDI provee las capacidades necesarias para incluirse en la arquitectura de descubrimiento basada en QoS por proveer una arquitectura flexible, abierta, sencilla y estándar en comparación con el registro ebXML.

Posteriormente se presentó nuestra arquitectura para el descubrimiento de servicios Web geográficos basada en QoS. Esta arquitectura puede verse como un Middleware (el middleware es un software de conectividad que ofrece un conjunto de servicios que hacen posible el funcionamiento de aplicaciones distribuidas sobre plataformas heterogéneas) que extiende el modelo de los Servicios Web al incluir las especificaciones de Metadatos ISO 19115 y NTC 4611 en un repositorio de Servicios UDDI de manera que se puedan manejarse descripciones de calidad del servicio para el entorno ambiental.

Finalmente, se mostró la integración de los mecanismos de búsqueda de servicios propuestos en esta tesis basados en comportamiento y calidad del servicio; para ello el módulo de calidad del servicio se ubica como un filtro sobre el módulo de descubrimiento basado en comportamientos WSCL. Todo lo anterior, enmarcado dentro del metamodelo del proceso de gestión ambiental del recurso hídrico. La arquitectura propuesta pretende ofrecer capacidades de búsqueda al módulo de descubrimiento de servicios de SIRA.

Capítulo 6. Prototipos y Experimentación

Los capítulos 3, 4 y 5 describen nuestra propuesta para el descubrimiento de servicios Web geográficos para la gestión del recurso hídrico basado en comportamiento y calidad del servicio. Todo lo anterior, enmarcado dentro del metamodelo del proceso de gestión ambiental del recurso hídrico. En este capítulo se ilustran los prototipos, desarrollados para validar nuestra propuesta:

- a) La primera herramienta permite la ejecución del algoritmo para el emparejamiento de servicios basado en comportamiento, el prototipo utiliza un conjunto de WSCL de servicios geográficos para la gestión ambiental del recurso hídrico.
- b) La segunda, es una aplicación Web que implementa el descubrimiento de servicios geográficos hidrológicos soportado en los criterios de Calidad del servicio.
- c) Finalmente, para validar la efectividad del metamodelo propuesto, se desarrolló una aplicación Web (que constituye el núcleo del sistema SIRA) y que permite definir, calcular y mostrar informes sobre los indicadores ambientales hidrológicos definidos en el metamodelo de gestión ambiental.

Además, hemos construido una herramienta para evaluar la efectividad del emparejamiento basado en comportamientos. Esta herramienta permite crear un ranking de servicios basada en la comparación manual entre un servicio consulta y un conjunto de servicios en un repositorio. Utilizando esta herramienta es posible comparar el resultado obtenido por la plataforma y el ranking definido por los usuarios.

En este trabajo proponemos que para optimizar el proceso de descubrimiento de servicios ambientales, se debe realizar una integración de los mecanismos de búsqueda basados en comportamiento y calidad del servicio; para ello el módulo de calidad del servicio se ubica como un filtro sobre el módulo de descubrimiento basado en comportamientos WSCL. Por lo anterior, para comprobar la efectividad de los mecanismos, se realizará una experimentación apropiada para cada mecanismo por separado. Las siguientes secciones describen nuestros prototipos y las experimentaciones: las tres primeras secciones presentan las tres aplicaciones desarrolladas para validar el proceso de descubrimiento, describiendo sus funcionalidades, arquitectura e interfaz de usuario. La cuarta sección expone los detalles sobre la herramienta de evaluación de la calidad del descubrimiento basado en comportamiento, mostrando igualmente, sus funcionalidades, arquitectura e interfaz de usuario. Finalmente, en la quinta sección se presentan las pruebas de evaluación de la calidad de la búsqueda para cada uno de los dos métodos (comportamiento y calidad), describiendo los escenarios de aplicación de prueba considerados, las estrategias de prueba y los resultados obtenidos.

6.1 Plataforma para el emparejamiento de servicios basada en comportamiento.

Desarrollamos un sistema prototipo que implementa nuestro emparejamiento de comportamientos para el descubrimiento de servicios basado en comportamiento. La herramienta fue construida utilizando el lenguaje de programación Java (JDK 1.6.0) y el entorno de desarrollo integrado (IDE) Netbeans 5.0. Se desarrolló una aplicación de escritorio, aunque también esta disponible esta funcionalidad como servicio Web. El sistema toma como entradas dos archivos WSCL que describen los servicios geográficos para la gestión ambiental del recursos hídrico y calcula la similaridad entre ellos (<http://ariadna.unicauca.edu.co/matching/>), luego genera el script de operaciones de edición requeridas con el fin de transformar el primer protocolo con el segundo. A continuación describimos las funcionalidades provistas por el sistema. Luego, la arquitectura del sistema y finalmente, se muestran las interfaces de usuario provistas por el prototipo.

6.1.1. Funcionalidades del Sistema.

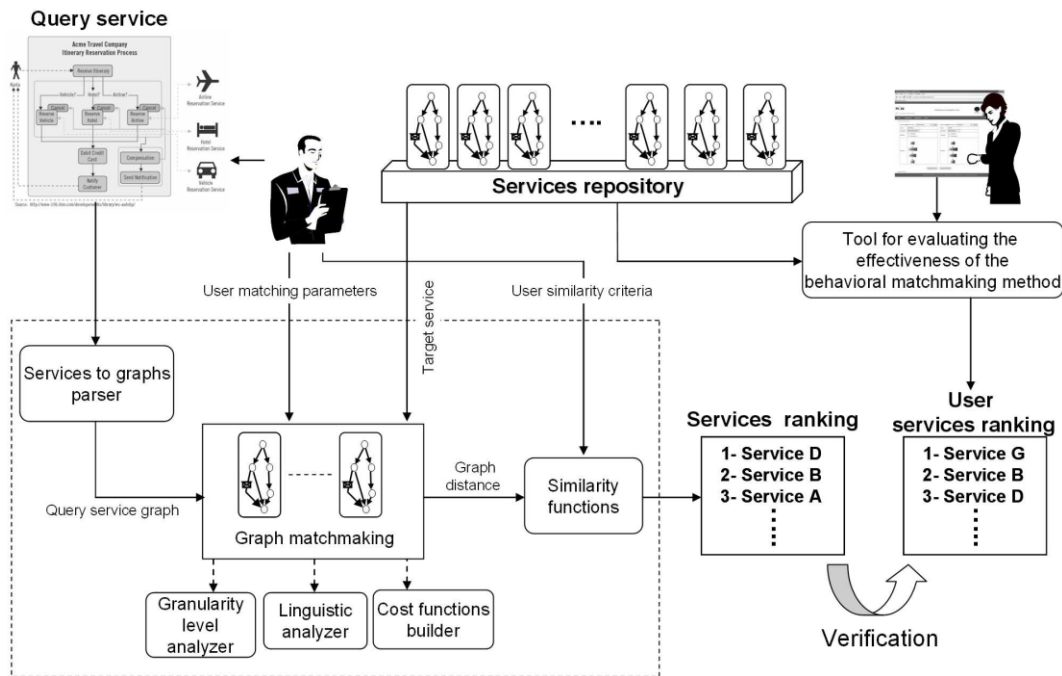


Figura 6.1. Plataforma para el ranking de servicios basada en emparejamiento de comportamientos

Dado un conjunto de servicios publicados con funcionalidades equivalentes (correspondientes a un dominio dado, por ejemplo, la gestión ambiental), el objetivo de esta plataforma es crear una clasificación (*ranking*) de los servicios con respecto a su utilidad para resolver los requerimientos del usuario. Suponemos que el usuario expresa sus requerimientos utilizando un modelo de comportamientos (e.g. utilizando WSCL) y la plataforma lo ayuda a identificar los servicios más similares. En nuestra plataforma el ranking de servicios se basa en el emparejamiento de

comportamiento de servicios presentado en el capítulo 4, el cual es reducido a un problema de emparejamiento de grafos. El *ranking* de servicios se construye teniendo en cuenta diferentes medidas (distancia de edición de grafos, similaridad simple y similaridad basada en: número de nodos mapeados y número de secuencias de nodos mapeados).

El sistema de descubrimiento de servicio basado en comportamiento se presenta en la Figura 6.1 y se compone de los siguientes módulos:

- *Procesador (Parser) de Servicios a grafos*: este modulo transforma una descripción de comportamiento de servicio (e.g., WSCL) a un grafo.
- *Emparejamiento de grafos*: este modulo toma como entrada los dos grafos producidos por el *Parser* presentados antes y encuentra la distancia semántica entre ellos basada en el isomorfismo de subgrafos por corrección de error con costo mínimo.
- *Constructor de funciones de costo*: este modulo agrupa las funciones de costos para las operaciones de edición de grafos que permiten calcular la distancia entre grafos. Los costos asignados a diferentes operaciones de edición de grafos reflejan la importancia relativa de disimilitudes entre los diferentes atributos de los grafos. Así ellos dependen del metamodelo de comportamiento de servicios y sobre el dominio de la aplicación.
- *Analizador lingüístico*: este componente calcula la similaridad lingüística entre dos cadenas utilizando los siguientes algoritmos: NGram, Check synonym, Check abbreviation y tokenization.
- *Analizador de nivel de granularidad*: éste chequea si las operaciones de composición y descomposición son necesarias y agrega su costo a la medida de distancia. Estas operaciones de edición son necesarias cuando la misma funcionalidad se modela a diferentes modelos de granularidad en los dos grafos (por ejemplo, utilizando dos nodos en un grafo y solo un nodo en el otro grafo).
- *funciones de similaridad*: este módulo define las funciones de similaridad que permiten construir el *ranking* de servicios. Aquí se utiliza el resultado del modulo de emparejamiento de grafos (la distancia de edición, el mapeo de nodos, etc.).
- *Herramienta para evaluar la efectividad del método de emparejamiento de comportamientos*: esta herramienta permite crear un *ranking* de servicios de usuario basado en las comparaciones manuales (humanas) entre un servicio consulta y los servicios en el repositorio. La herramienta permite comparar el resultado obtenido por la plataforma y el *ranking* definido por los usuarios.

Dado que la parametrización de las funciones de costos es dependiente del dominio, y además, muy importantes para la efectividad del método de emparejamiento, es necesario que la herramienta permita calibrar los parámetros óptimos de emparejamiento a usar para un dominio dado. En esta sección, explicaremos en detalle esta herramienta.

6.1.2. Arquitectura del Sistema

La arquitectura lógica presentada en la figura 6.2 organiza las clases software en paquetes, subsistemas y capas. Esta está implementada en tres capas diferentes (Aplicación, Mediación y base [184]). La figura 6.2 muestra las capas de la arquitectura y la interacción entre ellas, así como los paquetes más relevantes que componen cada capa.

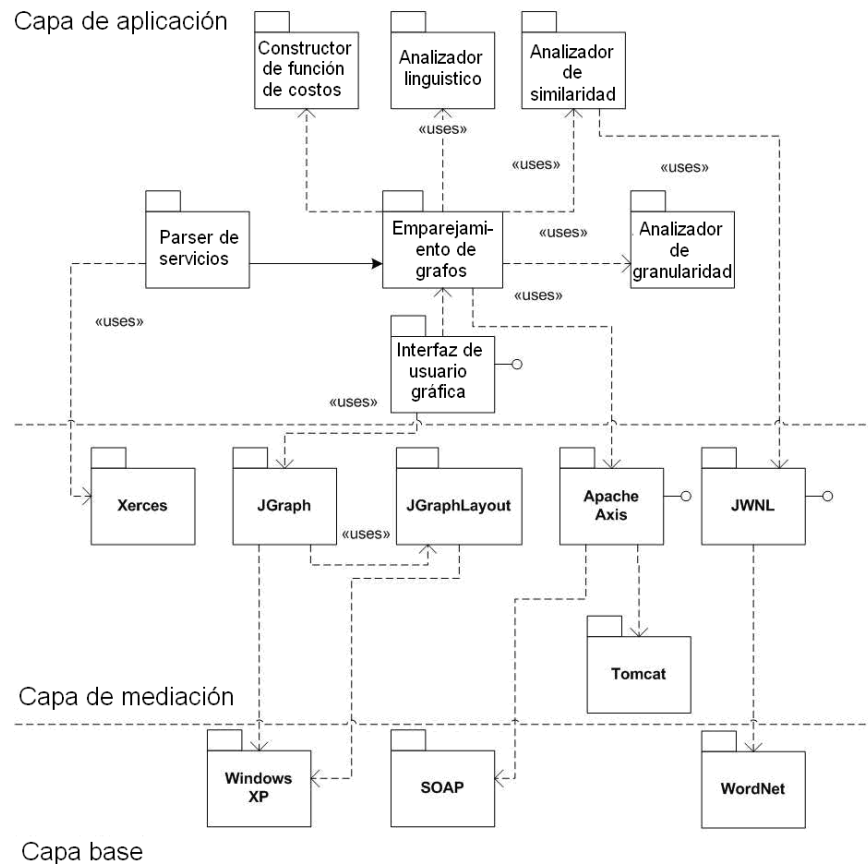


Figura 6.2. Arquitectura lógica del prototipo

- **La capa de aplicación:** Esta capa incluye las funcionalidades del prototipo, y se compone de los siguientes paquetes:

* *Parser de Servicios:* este paquete permite registrar las funciones que transforman el metamodelo de servicios consumidos por el usuario a un grafo. Para la herramienta desarrollada hemos implementado las funciones WSCL, pero es posible registra otras como las de WS-CDL.

* *Emparejamiento de grafos:* este paquete contiene las clases que implementan la detección de isomorfismo de subgrafos. Este paquete utiliza los paquetes: *analizador de similitud*, *el analizador de Granularidad*, *constructor de funciones de costo*, *analizador lingüístico* para calcular el emparejamiento entre los dos grafos de servicios.

* *Analizador de granularidad:* contiene las clases de descomposición y composición para el metamodelo.

* *Constructor de funciones de costo:* este paquete permite registrar las reglas de comparación para emparejar los metamodelos del servicio.

* *Analizador lingüístico:* contiene las clases que implementan los algoritmos para calcular la similitud lingüística entre cadenas de caracteres. Por ejemplo para la herramienta desarrollada

hemos usado los algoritmos Ngram, Token, Sinonym y Abbreviation, aunque puede agregarse otros algoritmos.

* *Analizador de Similaridad*: contiene las funciones que permiten calcular la similaridad basada en la distancia total entre los dos grafos. De la misma forma de los paquetes anteriores, este permite registrar otras funciones de similaridad.

* *Interfaz gráfica de usuario*: con el fin de tener una representación visual, este paquete contiene todas las clases que implementan las interfaces gráficas del prototipo.

- **Capa de mediación** contiene todas las interfaces de programas de aplicación (APIs) utilizadas en el prototipo. La capa se compone de los siguientes paquetes:

* *Xerces*: es una familia de paquetes software para procesar XML. La librería implementa un número de APIs estándares para el procesamiento XML, incluyendo DOM, SAX y SAX2.

* *Apache Axis*: es un marco de trabajo (*framework*) de software libre de Servicios Web. Este consiste en una implementación Java y C++ del servidor SOAP, y muchas utilidades y APIs, para generar aplicaciones de Servicios Web. Utilizando Apache Axis el emparejamiento de grafos puede ser publicados como un Servicio Web.

* *JWNL*: (Librería WordNet Java) es una API para acceder a diccionarios relacionales WordNet.

* *JGraph*: Conjunto de clases java para manejo de grafos, que cumple con los principios de diseño de Swing. Contiene toda la funcionalidad de visualización e interacción de grafos. Este paquete soporta la visualización de la interfaz gráfica de usuario.

* *JGraph Layout*: es una librería de diseño de grafos de alto rendimiento que posiciona automáticamente el grafo, diagrama o red en una forma adecuada para su análisis visual.

- **Capa base**: incluye el software básico que permite el rendimiento del prototipo. Esta capa está compuesta de los siguientes paquetes:

* *Protocolo de Acceso de Objetos Simple (SOAP)*: es un protocolo para intercambiar mensajes basados en XML sobre redes de computadores, normalmente utiliza HTTP/HTTPS. Este paquete soporta la publicación del emparejamiento de grafos como un servicio Web.

* *WordNet*: es un diccionario semántico para el lenguaje inglés. Agrupa palabras en inglés en conjuntos de sinónimos llamados *synsets*; provee definiciones generales cortas, y registra las relaciones semánticas entre estos conjuntos de sinónimos. El propósito es doble: producir una combinación de diccionario que es más usable intuitivamente; y soportar análisis de textos automático y aplicaciones de inteligencia artificial. Este paquete es utilizado por el analizador lingüístico (a través del paquete JWNL) para verificar la relación semántica entre dos palabras.

* *Windows XP professional*: es el sistema operativo que soporta el prototipo.

6.1.3. Interfaces de Usuario

En esta sección, presentamos la interfaz de usuario gráfica (GUI) del prototipo. El proceso de emparejamiento comienza cuando el usuario carga el archivo WSCL, posteriormente se ejecuta el procesador WSCL. A continuación, el usuario selecciona la función de descomposición. El cuarto

paso permite al usuario asignar el costo para las operaciones de edición de grafos. Finalmente, se presenta el resultado del emparejamiento al usuario. La figura 6.3 muestra la GUI en la cual el usuario escoge el documento WSCL a comparar. La GUI muestra también los grafos resultantes del procesamiento de documentos. En la figura 6.3 los nodos atómicos y los no atómicos tienen diferentes colores.

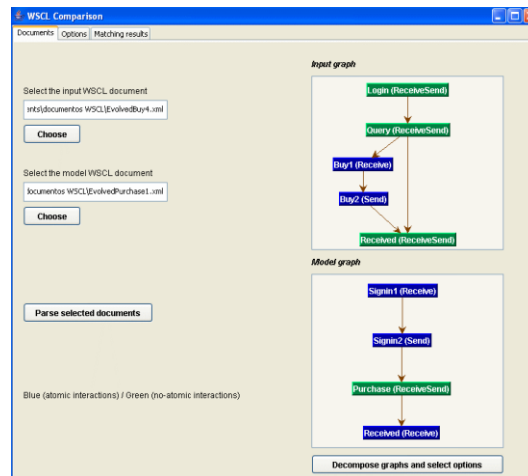


Figura 6.3. Interface de documentos WSCL

En la GUI de la figura 6.4, el usuario puede configurar los costos de las operaciones de edición de grafos WSCL (Interacciones y transiciones). El costo de: la granularidad, el peso de similitud para nombres e interacciones de documentos (W_i y W_d), y el valor de umbral para la distancia de edición de grafos (valor aceptable) se configuran en esta interfaz. De otro lado, la interfaces muestra la descomposición del grafo de entrada y el grafo modelo (primer y segundo grafo).

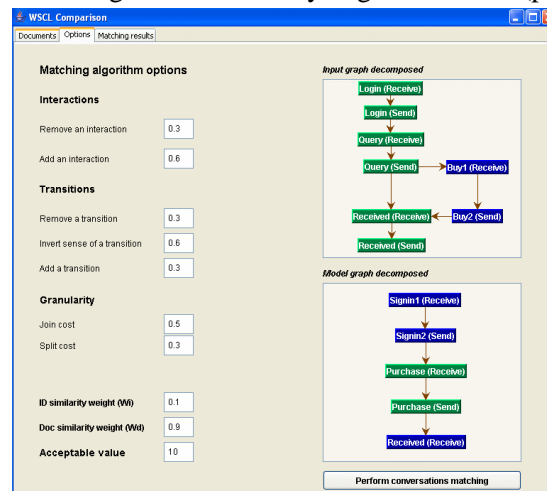


Figura 6.4. Interfase de opciones WSCL

Finalmente, la figura 6.5 muestra el resultado del algoritmo de emparejamiento. Estos resultados son desplegados de forma gráfica y textual.

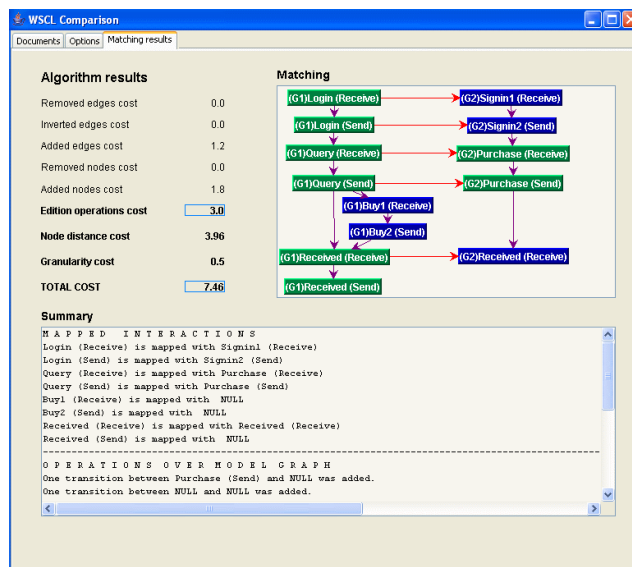


Figura 6.5. Interfaces de resultados del emparejamiento WSCL

6.2 Un sistema de descubrimiento de servicios Web geográficos basado en calidad del servicio

El sistema presentado en esta sección se basa en la arquitectura definida en el capítulo cinco, y permite el descubrimiento de servicios en un repositorio, enriqueciendo la definición y las consultas de los servicios utilizando información sobre la calidad del servicio (ingresada previamente en el catálogo). Nuestro sistema incluye una implementación de la norma NTC4611, soportada en GeoAPI (un framework Java que ofrece un conjunto estándar de interfaces derivadas de los estándares OGC/ISO, para más información ver discusión en el Anexo 1: *Herramientas para el desarrollo de aplicaciones Web SIG de software libre*).

La herramienta fue construida utilizando el lenguaje de programación Java (JDK 1.6.0) y el entorno de desarrollo integrado (IDE) de MyEclipse 6.0. El sistema es una aplicación Web que esta disponible en <http://gea.unicauca.edu.co/qos>. El resto de esta sección, describe las funcionalidades, la arquitectura, y las interfaces de usuario del sistema.

6.2.1. Funcionalidades del sistema

Nuestra implementación incluye dos roles básicos, uno de registro de servicios, encargado de registrar la información de servicios; y otro de consulta de los servicios del repositorio; estas dos funciones determinan los roles básicos del sistema:

- Usuario de registro de servicios
 - Gestión de usuarios: La herramienta permite crear nuevos usuarios o editar los ya existentes
 - Gestionar de Servicios: El sistema permite agregar información sobre los servicios

geográficos existentes, pudiéndose agregar los criterios de calidad del servicio (definidos en el perfil descrito en el capítulo 5).

- Usuario de consulta

- Consulta de Servicios: La herramienta permite consultar los servicios de acuerdo con los diferentes criterios de calidad del servicio.

6.2.2. Arquitectura del servicio

La figura 6.6 muestra las capas de la arquitectura del sistema y la interacción entre ellas, así como los paquetes más relevantes que componen cada capa:

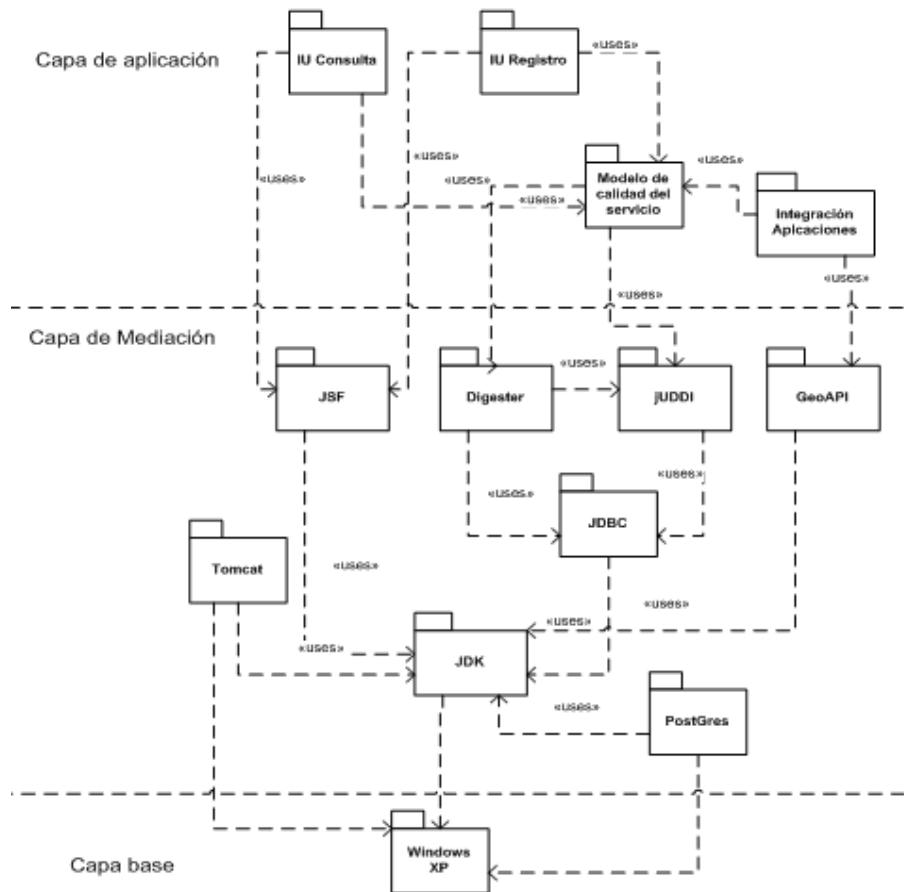


Figura 6.6. Arquitectura lógica de la herramienta de gestión de indicadores ambientales

- **La capa de Aplicación:** Esta capa se compone de los siguientes paquetes:
 - *Interfaz de usuario de Consulta:* contiene todas las clases que implementan las interfaces de usuario (IU) de consulta de Servicios.
 - *Interfaz de usuario de Registro:* contiene las clases de IU del registro de Servicios.

- *Módulo de calidad del servicio*: este paquete contiene todas las clases que conforman el modelo de metadatos, soporta la consulta y el registro de servicios.
 - *Integración de aplicaciones*: este paquete permite al servicio de catálogo integrarse con otras aplicaciones como por ejemplo la plataforma SIRA. Para conseguir lo anterior, agrupa un conjunto de interfaces basadas en los estándares de la OGC.
- **Capa de mediación**: se compone de los siguientes paquetes:
- *JSF*: Es un framework para aplicaciones Java basadas en Web que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE.
 - *Digester*: Jakarta Commons Digester, es un API Java de software libre que permite mapear documentos XML en objetos Java.
 - *JUDDI*: es una implementación Java de software libre de la especificación UDDI
 - *GeoAPI*: Framework Java con interfaces derivadas de los estándares OGC/ISO.
 - *JDBC*: Es una API Java que permite el acceso a una fuente de datos de forma estándar. Provee métodos para consulta y actualización de datos en una base de datos. Esta interfaz es utilizada por paquetes de la capa de aplicación requieren de la interacción con la base de datos Firebird.
 - *JDK* (Java Development Kit): entorno de desarrollo integrado (IDE) para codificar aplicaciones Java. Consiste de un entorno de ejecución que se ubica sobre la capa del sistema operativo.
 - *Tomcat*: provee un entorno para ejecutar la herramienta como una aplicación Web.
 - *PostgreSQL*: Sistema gestor de bases de datos.
- **Capa base**: Esta capa se compone de *Windows XP professional*.

6.2.3. Interfaces de Usuario.

Como se puede observar en la figura 6.7, la interfaz permite el ingreso de información sobre el perfil de calidad y el geográfico; esta información debe estar en formato XML (Ver capítulo 5 para más detalles sobre el contenido del documento).

Figura 6.7 Registro de información de Servicios



Figura 6.8 Interfaz del catalogo para búsqueda de Servicios

Finalmente, el usuario de consulta puede ingresar los criterios de búsqueda para realizar las consultas (Ver figura 6.8).

6.3 Una Herramienta para la gestión de los indicadores ambientales basada en el metamodelo de gestión ambiental

La herramienta implementada permite la gestión y análisis de indicadores ambientales para cualquier tipo de estudio ambiental del recurso hídrico, para ello el sistema permite la definición de elementos conceptuales como: tipos de variable, estudios o algoritmos relacionado con la gestión ambiental. Con base en lo anterior, el sistema construido soluciona una gran limitante existente hasta ahora en los sistemas de información ambiental, los cuales únicamente podían calcular el valor de un solo indicador (como el índice de calidad del agua), y generar informes estadísticos asociados con un único subdominio de aplicación (por ejemplo, análisis físico-químico). El sistema propuesto soluciona este inconveniente al proveer una calculadora de indicadores en la cual pueden registrarse fórmulas matemáticas y asociarlas con el cálculo de cualquier indicador, pudiéndose de esta manera realizar el cálculo y análisis ambiental de cualquier recurso hídrico (aunque este sistema se diseñó para el dominio Hidrológico, la utilidad del metamodelo y del sistema en general es extensible a cualquier campo del dominio ambiental). La herramienta fue construida utilizando el lenguaje de programación Java (JDK 1.6.0) y el entorno de desarrollo integrado (IDE) de MyEclipse 6.0. El sistema es una aplicación Web que esta disponible en <http://gea.unicauca.edu.co/sira>. El resto de esta sección, describe las funcionalidades, la arquitectura, y las interfaces de usuario del sistema.

6.3.1 Funcionalidades del sistema

Nuestra implementación incluye cuatro roles básicos: el usuario de consulta, el usuario de ingreso de datos y el usuario de administración y el usuario de administración de mapas. A continuación se presentan las funcionalidades para cada usuario.

- Usuario de administración

- Gestión de usuarios: La herramienta permite crear nuevos usuarios o editar los ya existentes.
 - Gestionar elementos del metamodelo: Una vez registrado en el sistema, el administrador puede gestionar elementos como: componentes, rangos, variables, estudios y algoritmos (para una ampliación de estos conceptos véase capítulo 3). De esta manera el administrador puede configurar el sistema hacia un dominio específico (conteniendo sus propias variables, estudios, algoritmos)
- Usuario de administración de mapas
- Gestión de capas cartográficas: La herramienta permite cargar capas cartográficas y mostrarlas sobre Internet.
 - Gestionar mapas temáticos: El administrador de mapas, puede crear sus propios mapas temáticos de acuerdo con las capas cargadas en el sistema, estos mapas pueden incluir información relacionada con los análisis ambientales.
- Usuario de ingreso
- Gestión de puntos de monitoreo: El usuario de ingreso, puede registrar la información espacial (geográfica) y temática sobre los lugares de monitoreo, o donde se hayan tomado muestras.
 - Gestión de muestras: el sistema permite ingresar los valores de las variables (definidas por el usuario de administración según un dominio particular) medidas en un punto específico.
- Usuario de consulta
- Consulta de informes temáticos: La herramienta permite visualizar reportes gráficos (en diagramas de tortas, barras y tendencias) sobre la distribución de una variable o de un estudio (configurados por el administrador) en un punto de monitoreo o en una muestra específica.
 - Consulta de mapas: el sistema permite visualizar los mapas configurados por el administrador de mapas.

6.3.2 Arquitectura del servicio

La figura 6.9 muestra las capas de la arquitectura del sistema y la interacción entre ellas, así como los paquetes más relevantes que componen cada capa.

- **La capa de Aplicación:** se compone de los siguientes paquetes:
- *Interfaz de usuario de Administración:* clases que implementan las interfaces de usuario (IU) de administración.
 - *Interfaz de usuario de Administración SIG:* clases asociadas con IU de administración de mapas.
 - *Interfaz de usuario de ingreso:* clases que implementan las IU de ingreso.
 - *Interfaz de usuario de Consulta:* clases que implementan las IU de consulta.
 - *Estudios:* contiene las clases que implementan el metamodelo adaptado a un dominio específico (variables, estudios, rangos, componentes).
 - *Monitoreo:* contiene la información sobre puntos de monitoreo y muestras de datos.

Capa de mediación: se compone de los siguientes paquetes:

- *JDBC*: API Java que permite el acceso a una fuente de datos de forma estándar.
- *Postgres*: sistema de gestión de base de datos utilizado para almacenar los datos del sistema.

Capa base: incluye el software básico que permite ejecutar el prototipo.

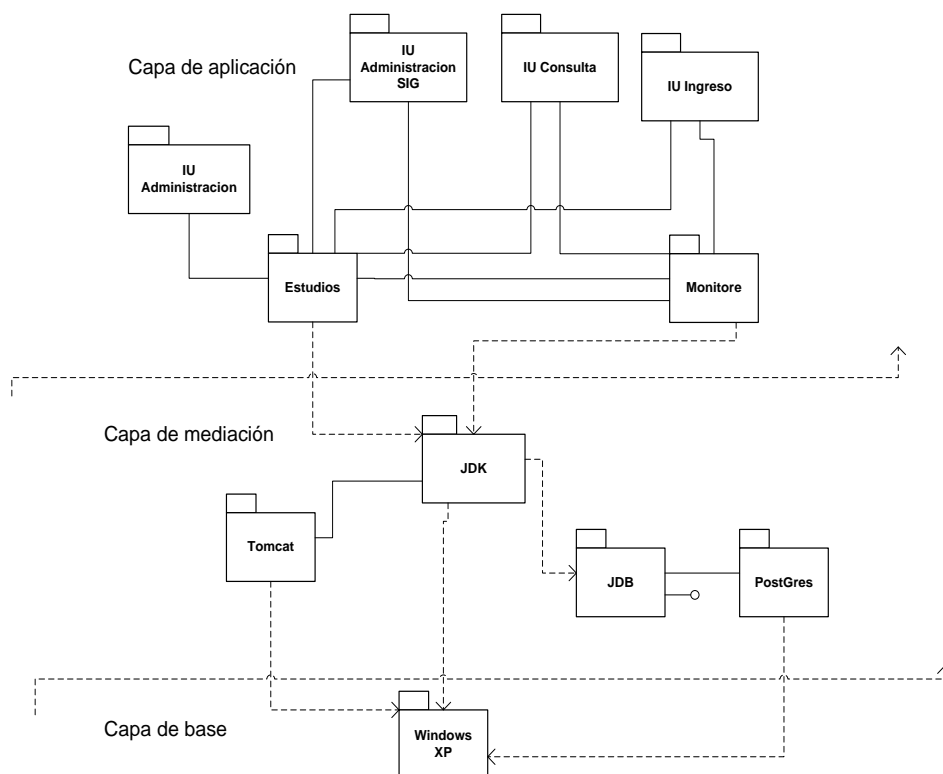


Figura 6.9: Arquitectura lógica de la herramienta de gestión de indicadores ambientales

6.3.3 Interfaces de usuario

El usuario puede asumir cualquiera de cuatro roles descritos anteriormente dependiendo de los permisos asignados durante el proceso de autenticación. Si el usuario es un administrador del sistema, puede definir las variables, estudios y formulas matemáticas para el cálculo de cada uno de los indicadores ambientales hidrológicos, para ello la interfaz muestra todas las variables definidas previamente para el dominio específico (agrupadas por componentes ambientales) y el administrador puede crear formulas matemáticas utilizando las variables ambientales y un conjunto de operaciones matemáticas disponibles (ver figura 6.10).

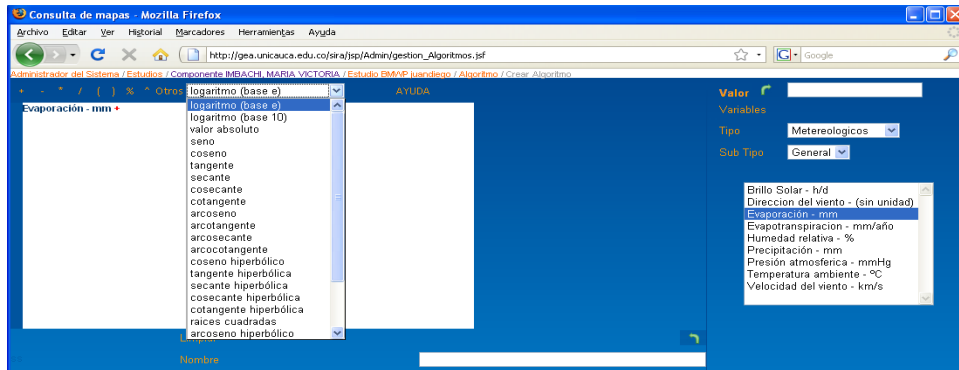


Figura 6.10. Interfase de comparación de servicios

Por su parte, el *usuario de ingreso* puede registrar las muestras en cualquier punto de monitoreo, seleccionando las variables y asociando una fecha de toma y un valor. De otro lado, el *usuario de consulta* puede consultar el valor de los estudios en un punto dato, así como la distribución de las variables y las tendencias de una variable o un estudio en un punto o zona geográfica. Por ejemplo, en la figura 6.11 se muestra la distribución promedio de una variable entre diferentes puntos de monitoreo distribuidos geográficamente. En figura 6.12 se muestra la distribución temporal de una variable o un indicador ambiental a través del tiempo en un punto de monitoreo, en la parte inferior se muestra el valor del indicador (calculado automáticamente utilizando la formula registrada por el administrador y con los valores registrados por el usuario de ingreso).

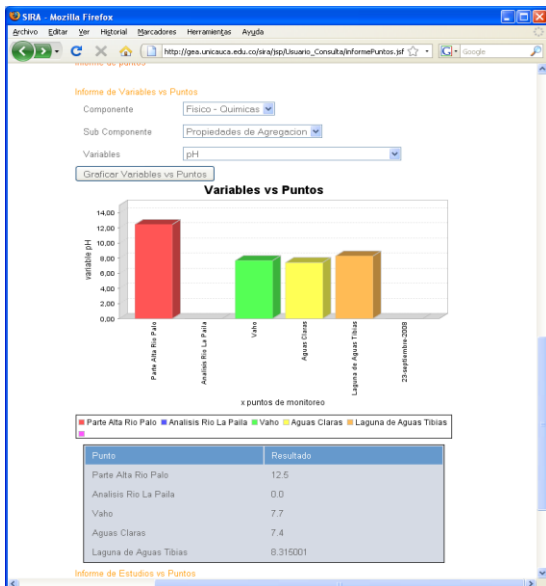


Figura 6.11: Ejemplo de análisis multiespacial de una variable

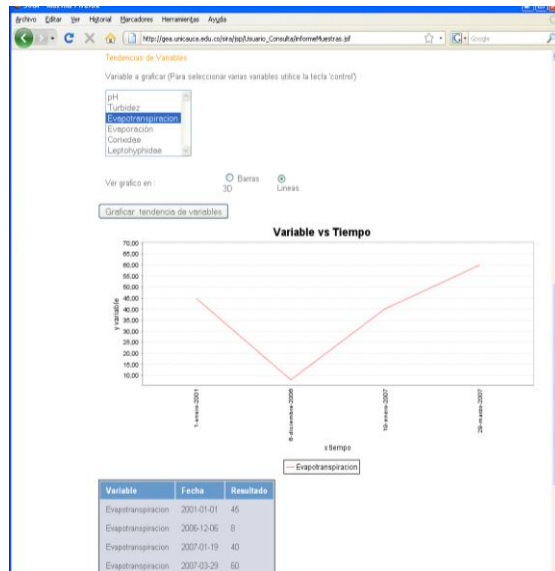


Figura 6.12: Ejemplo de análisis multitemporal de una variable

6.4 Una herramienta para evaluar la efectividad del método de emparejamiento de comportamiento.

La herramienta implementa un método que permite a los usuarios comparar manualmente los modelos de un modelo de consulta y un servicio objetivo, y posteriormente crear un ranking de servicios de acuerdo con los resultados de la comparación. Los modelos representan las características más importantes de los procesos Web. De esta forma, la herramienta permite comparar el resultado obtenido por la plataforma de emparejamiento de servicios y el ranking definido por los usuarios. Dado el hecho que la parametrización de la función de costo depende del dominio y muy importante para la efectividad del método de emparejamiento, es importante tener una herramienta que permita determinar los parámetros óptimos a usar para un dominio dado y criterios de similaridad. La herramienta fue construida utilizando el lenguaje de programación Java (JDK 1.6.0) y el entorno de desarrollo integrado (IDE) de MyEclipse 6.0. El sistema es una aplicación Web que esta disponible en <http://ariadna.unicauca.edu.co/pertinence>. En el resto de esta sección describimos las funcionalidades; la arquitectura y las interfaces de usuario de la herramienta.

6.4.1 Funcionalidades del sistema

En nuestra implementación se han definido dos roles básicos: el usuario de comparación y el usuario de administración. A continuación presentamos las funcionalidades más representativas.

- Usuario de comparación
 - Realizar comparación: El sistema presenta al usuario dos servicios a comparar, posteriormente, el usuario puede asignar una medida de similaridad (entre 0 y 5) entre los dos servicios de acuerdo con los criterios de comparación que considere relevantes. Los criterios de similaridad propuestos son: nombre del servicio, descripción del servicio, conjunto de actividades y estructura del servicio. En el caso del conjunto de actividades, la herramienta permite además realizar una comparación interacción por interacción.
- Usuario de administración
 - Gestión de usuarios: La herramienta permite crear nuevos usuarios o editar los ya existentes (tanto usuarios de comparación como de administración).
 - Gestión de documentos: el sistema permite registrar documentos (pueden ser en formato BPEL) de consulta y objetivos en el repositorio de servicios.
 - Análisis de los resultados de comparación: con este componente funcional la herramienta puede crear una clasificación de los servicios más similares a un servicio consulta de acuerdo con los criterios de comparación. La lista es una clasificación (Top) n ($1 < n < 10$), donde el primer servicio es el servicio más similar.

6.4.2 Arquitectura del servicio

La figura 6.13 muestra las capas de la arquitectura del sistema y la interacción entre ellas, así como los paquetes más relevantes que componen cada capa.

- **La capa de Aplicación** gestiona los paquetes que implementan las funcionalidades del sistema. Esta capa se compone de los siguientes paquetes:
 - *Interfaz de usuario de comparación*: con el fin de ofrecer una representación visual, estos paquetes contienen todas las clases que implementan las interfaces gráficas del usuario de comparación.
 - *Interfaz de usuario de Administración*: este paquete contiene todas las clases que implementan las interfaces de usuario de administración.
 - *Proceso de comparación*: contiene la implementación necesaria para comparar manualmente dos servicios.
 - *Proceso de ranking*: este paquete calcula el ranking de servicios basado en los resultados del paquete de comparación.
 - *Gestión de usuarios*: este paquete implementa la lógica para registrar y editar la información de los usuarios que interactúan con la herramienta.
 - *Gestión de documentos*: contiene las clases que implementan la lógica para gestionar los documentos que definen los procesos web de consulta y objetivos.

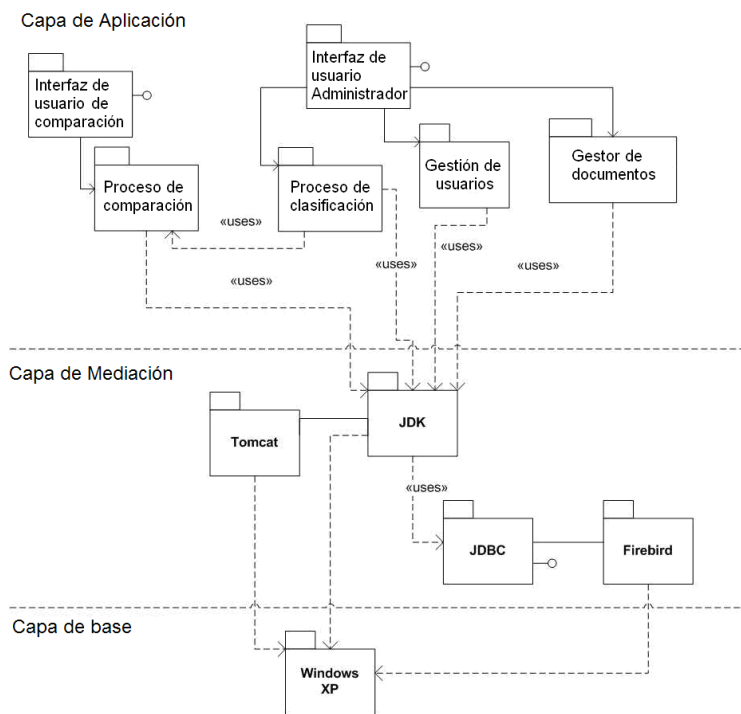


Figura 6.13. Arquitectura lógica de la herramienta

- **Capa de mediación:** se compone de los siguientes paquetes:
 - *JDBC*: Es una API Java que permite el acceso a una fuente de datos de forma estándar. Provee métodos para consulta y actualización de datos en una base de datos. Esta interfaz es utilizada por paquetes de la capa de aplicación requieren de la interacción con la base de

datos Firebird.

- *Firebird*: es el sistema de gestión de base de datos utilizado para almacenar los datos de usuario y los análisis del ranking y emparejamiento de servicios.
- *JDK* (Java Development Kit) es un entorno de desarrollo integrado (IDE) para codificar aplicaciones Java. Consiste de un entorno de ejecución que se ubica sobre la capa del sistema operativo y permite compilar y ejecutar la herramienta de efectividad, la cual está escrita en Java.
- *Tomcat*: provee un entorno para ejecutar la herramienta como una aplicación Web.

- **Capa base:** Esta capa se compone de *Windows XP professional*.

6.4.3 Interfaces de usuario

En esta sección mostramos las principales interfaces de la herramienta. Como se explicó anteriormente, luego de realizar el proceso de validación en la herramienta de evaluación, el usuario selecciona el servicio de consulta a analizar. El proceso de comparación comienza con la comparación del servicio de consulta contra el primer servicio objetivo (Ver figura 6.14) y termina cuando se compara el servicio objetivo.



Figura 6.14: interfase de comparación de servicios

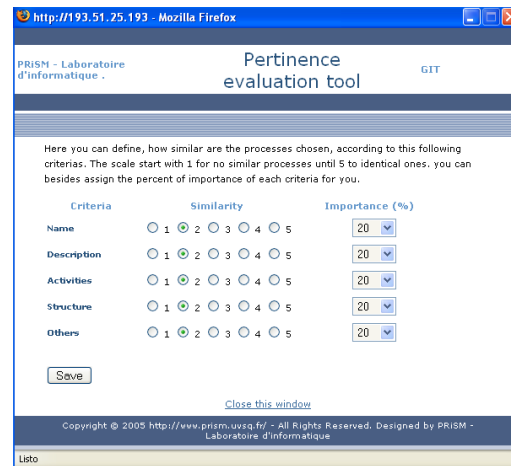


Figura 6.15: Interfase de selección de criterios

Cada par de servicios se muestra en una página Web, la cual ofrece una interfaz de usuario gráfica sencilla, y permite una definición apropiada de los parámetros de comparación. Como se puede ver en la figura 6.15, el usuario puede definir la similitud entre los dos procesos en una escala de 1 (sin similitud) a 5 (idénticos). Con botones de elección simples. Además, los usuarios pueden especificar como se realiza la comparación: 1. por nombre del servicio, 2. por descripción del servicio, 3. por conjunto de actividades, 4. por estructura del servicio y 5. Utilizando otros métodos de evaluación. Finalmente, con el fin de capturar el aspecto subjetivo de la medida de la similitud, la herramienta permite especificar la importancia de cada criterio a juicio del usuario.

Si el usuario selecciona el conjunto de actividades como un criterio, la herramienta permite analizar cada rama del grafo de servicios, comparando interacción por interacción (ver figura 6.16). Finalmente, la herramienta crea una clasificación de los servicios más similares para cada servicio de consulta analizado de acuerdo con los criterios de comparación o utilizando una combinación de criterios (ver figura 6.17). Finalmente se muestra el TOP n, este TOP se organiza en orden descendente y si ha sido comparado utilizando muchos criterios, se utiliza un promedio para calcular la similaridad.

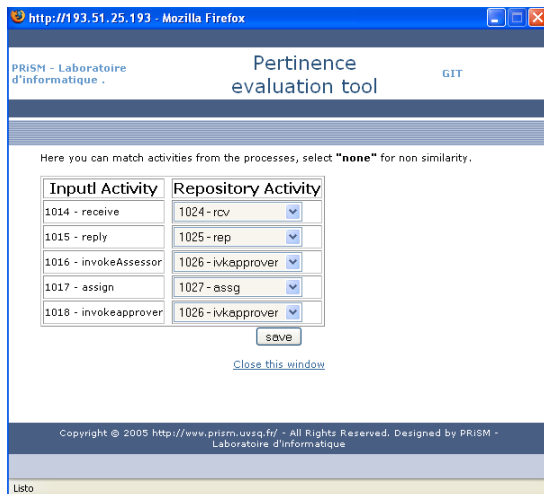


Figura 6.16: Interfase de comparación de ramas del servicio



Figura 6.17: Interfaz de ranking de servicios

6.5 Evaluación experimental

Uno de los problemas en la evaluación del descubrimiento de servicio es la inexistencia de un estándar de comparación (benchmark) de la similaridad de servicios. Básicamente, el problema de desarrollar un benchmark de similaridad de servicios es que este requiere muchos datos de prueba y experimentación, lo cual consume tiempo. Por consiguiente, la mayoría de los métodos de evaluación tienen lugar en un nivel de sistemas estadísticos. Por otro lado, un benchmark debería involucrar la evaluación humana, lo cual hace que benchmark de este tipo no sean propuestos frecuentemente. De esta manera es necesario tener en cuenta la interacción humana si se quiere capturar la realidad. Por lo tanto, en este trabajo hacemos uso de la técnica de validez de construcción para validar los resultados de la herramienta de comparación de servicios basada en comportamiento y en calidad del servicio. La validez tiene que ver con la relación entre la teoría y la observación. Se refiere al grado en el cual una medida dada caracteriza acertadamente la construcción bajo estudio. Para llevar a cabo esta validez analizamos la efectividad de los métodos de descubrimiento propuestos (basados en calidad y comportamiento del servicio), realizando una comparación entre los resultados obtenidos con los prototipos descritos en las secciones 6.1 y 6.2 y la evaluación humana (en el caso del mecanismo de comportamiento se implementó un prototipo adicional descrito en la sección 6.4).

En vista de que proponemos que el proceso de descubrimiento de servicios puede optimizarse

utilizando dos mecanismos: uno basado en calidad y otro basado en el comportamiento de los servicios; y atendiendo además, al hecho de que argumentamos que el mecanismo de descubrimiento basado en calidad del servicio puede implementarse como un filtro sobre el mecanismo basado en comportamiento; Nuestra evaluación experimental se divide en dos partes, que corresponden con los dos mecanismos propuestos:

a) Para el mecanismo basado en comportamiento, evaluamos la calidad del descubrimiento comparando el resultado obtenido por el prototipo (ver sección 6.1) con el resultado guardado en la herramienta de evaluación (ver sección 6.5). En vista de que este trabajo se centra en las técnicas de descubrimiento y no en el ranking de servicios, para esta parte de la evaluación experimental nos centramos en la medida de la calidad del descubrimiento. Aunque el caso de la herramienta de evaluación (sección 6.2) permite crear un ranking de servicios, para la evaluación experimental se utilizará únicamente la funcionalidad que permite analizar cada rama del grafo de servicios, comparando interacción por interacción (ver figura 6.16).

b) Para el mecanismo basado en calidad del servicio, evaluamos la calidad del descubrimiento de servicios ambientales basado en calidad del servicio realizado con el prototipo (ver sección 6.2) por medio de una simulación de usuarios de prueba.

6.5.1 Evaluación experimental del mecanismo basado en comportamiento

A continuación se describen los objetivos de la evaluación, posteriormente, se muestran: el entorno de pruebas, el conjunto de datos de los experimentos, así como los resultados y sus análisis. Estos resultados muestran la mejor función de costo para obtener un resultado de emparejamiento óptimo. Además, se exploran los límites del algoritmo con respecto al tiempo de ejecución.

6.5.1.1 Objetivo de la evaluación experimental

El objetivo de la evaluación experimental era caracterizar el desempeño y la calidad del proceso de emparejamiento. En particular, deseamos encontrar lo siguiente:

- Parametrización de la función de costo
- Estimación de la calidad del proceso de emparejamiento
- Desempeño del algoritmo de emparejamiento
- Conformidad de los resultados del emparejamiento calculado por el prototipo con los resultados intuitivos registrados por el usuario.

Para realizar la parametrización y criterios medibles de la relevancia, ejemplificamos la expresión de parametrización de la función de costos, y en particular la definición de la calidad de emparejamiento en términos de precisión y retentiva (*recall*)

Parametrización de la función de costo:

Un elemento muy importante a tener en cuenta para el descubrimiento basado en comportamiento, es la definición de las operaciones de edición de grafos y la correspondiente función de costo. La elección de una operación de edición específica a aplicar en un grafo de consulta depende de su

costo. Por ejemplo, considérese que la eliminación de costos es de 0.5 y que las etiquetas de tres vértices en el grafo de consulta son diferentes del grafo objetivo (siendo la distancia entre las etiquetas de 0.1). Entonces la distancia de edición de grafos es de 0.3. De otro lado, si el error de etiquetas crece, entonces el algoritmo considerara la posibilidad de borrar un vértice. La relación entre el costo de eliminación de un nodo y el costo de sustitución de un nodo influencia enormemente el resultado y el comportamiento del algoritmo. El objetivo de nuestros experimentos es parametrizar la función de costo con el fin de obtener una buena calidad del algoritmo en emparejamiento. A continuación, describimos como puede evaluarse la calidad del algoritmo de emparejamiento.

Estimación de la calidad del emparejamiento:

Para proveer una base para la evaluación de la calidad del algoritmo de emparejamiento, primero hemos desarrollado manualmente el emparejamiento identificando las semejanzas entre las interacciones de los dos modelos de comportamiento (nodos del grafo). Para evaluar la calidad del algoritmo de emparejamiento comparamos los emparejamientos (P) obtenidos por nuestro algoritmo, con los emparejamientos obtenidos manualmente (R). Posteriormente, determinamos el conjunto de verdaderos positivos (i.e. emparejamientos identificados correctamente), I, así como un conjunto de falsos positivos (i.e. falsos emparejamientos), F = P/I, y falsos negativos (i.e. emparejamientos faltantes) M= R/I. Basándose en la cardinalidad de estos conjuntos, se calculan las siguientes medidas de calidad:

- *Precisión* = $|I|/|P| = |I|/(|I| + |F|)$ estima la confiabilidad del emparejamiento.
- *Recall* = $|I|/|R|$ especifica el porcentaje de emparejamientos efectivos.
- *Overall* = $1 - (|F| + |M|)/R = (|I| - |F|)/|R| = Recall * (2 - 1/Precision)$ Representa una medida combinada para la calidad del emparejamiento (ver [185]) tomando en cuenta el esfuerzo post-emparejamiento necesario tanto para remover emparejamientos falsos y para agregar emparejamientos faltantes.

6.5.1.2 Metodología de experimentación y resultados

Como hemos mencionado, evaluar el prototipo de descubrimiento basado en calidad del servicio requiere comparar los resultados del emparejamiento contra los resultados base. De esta manera el sistema es comparado contra los resultados del emparejamiento de una evaluación humana que fue realizada utilizando la herramienta para la evaluación de la efectividad del método de emparejamiento de comportamientos. Estudiamos la influencia de la función de costo, mas precisamente de la relación entre el costo de borrar y sustituir, sobre la calidad del proceso de emparejamiento. Para nuestra evaluación utilizamos 5 archivos WSCL, que contenían entre 10 y 15 nodos (los servicios Web simples son relativamente simples, con un número pequeño de operaciones). Para cada archivo, generamos copias distorsionadas (con sintaxis diferentes, pero con la misma función) de la siguiente forma:

- Cambiando el nombre de las etiquetas con sinónimos y abreviaciones (caso 1).
- Cambiando los nombres de las etiquetas y cambiando el orden de las interacciones (borrando,

insertando aristas) (caso 2).

- Cambiando los nombres de las etiquetas, borrando vértices e insertando nuevos vértices (caso 3).

En resumen generamos 5 servicios de consulta y 15 servicios objetivos como datos de prueba. Utilizando la herramienta para evaluar la efectividad del método de emparejamiento de comportamiento generamos un resultado de emparejamiento base que permite evaluar la calidad del emparejamiento del prototipo de descubrimiento basado en comportamiento.

Precisamente, tomamos los datos de prueba presentados arriba y posteriormente comparamos el servicio de consulta contra los servicios objetivos utilizando la funcionalidad de la herramienta que permite comparar interacción por interacción entre dos grafos de servicios. Consecuentemente, 5 usuarios evaluaban 5 servicios de consulta (Q_j) contra 15 servicios objetivos (T_j). Cada usuario cubría un conjunto de servicios. La tabla 6.1. Muestra que 5 usuarios (U1 a U5) llevaban a cabo 15 comparaciones para cada servicio de consulta respectivamente (e.g. el usuario U1 comparaba el servicio de consulta Q1 contra 15 servicios objetivos) de la misma forma se trata el servicio de consulta Q2. En resumen, se analizaron 150 comparaciones utilizando la herramienta y dos comparaciones se hicieron para cada servicio de consulta, dado que un servicio de consulta fue evaluado al menos una vez por dos usuarios diferentes. De otro lado, evaluamos los 5 servicios de consulta contra los 15 servicios objetivos (el mismo conjunto de datos de prueba) utilizando el prototipo de descubrimiento basado en comportamiento. Con el objetivo de estudiar la influencia de la función de costo en el resultado de emparejamiento, para cada servicio de consulta analizado modificamos las operaciones de edición de costos. A continuación contrastamos los resultados con los resultados base de emparejamiento obtenidos con la herramienta de evaluación para determinar la mejor función de costo.

Usuario	Servicio de consulta	Comparaciones
U1	Q1 y Q2	30
U2	Q2 y Q3	30
U3	Q3 y Q4	30
U4	Q4 y Q5	30
U5	Q5 y Q1	30

Tabla 6.1. Conjunto de comparación

Los experimentos fueron llevados a cabo en un maquina Dell, con procesador Pentium 4 de 2.30GHz y 1000 MB RAM. El espacio total en disco fue de 80 GB. La máquina estaba corriendo bajo el sistema operativo Windows XP. En el resto de esta sección, mostramos los resultados obtenidos de la evaluación para calibrar el sistema, y el tiempo de ejecución utilizado por el proceso de emparejamiento de dos metamodelos WSCL.

6.5.1.3 Resultados obtenidos

En nuestros experimentos, hemos cambiado los costos de eliminar un vértice, Od de 0.1 a 0.9, mientras los pesos de eliminar e insertar nodos fueron constantes a 0.2. El costo de sustituir una etiqueta de nodo y sus atributos asociados es definido como se explica en la sección 3.2. Su valor está entre 0 (similaridad completa) y 1 (no emparejamiento).

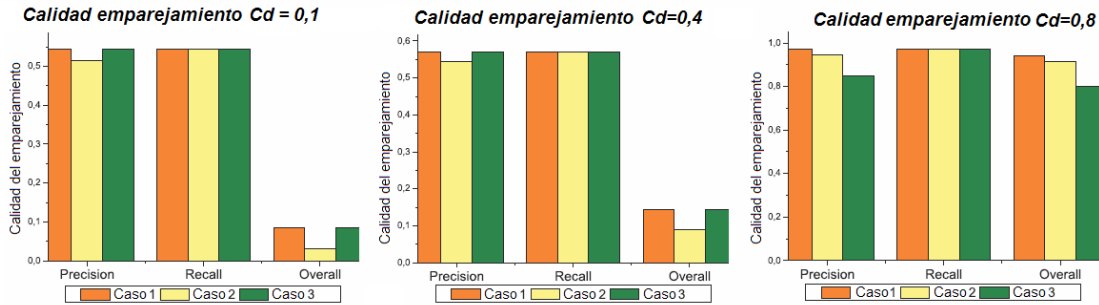


Figura 6.18. Calidad del emparejamiento para diferentes funciones de costo

La figura 6.18 muestra la *precisión*, *recall* y *overall* promedio obtenidos cuando se comparan los archivos WSCL con sus copias distorsionadas para diferentes valores de costos de eliminación de vértices, Od ($Od = 0,1$, $Od = 0,4$, $Od = 0,8$). Mostramos las medidas de calidad de forma separada para los tres casos presentados arriba. Los experimentos muestran que unos buenos resultados de calidad pueden obtenerse con la siguiente parametrización de función de costo: costo de eliminación de vértices = 0,8, costo de eliminación de nodos = 0,2, costo de inserción de aristas = 0,2.

La Figura 6.19 muestra el tiempo de ejecución promedio para los tres casos y los diferentes valores para los costos de eliminación de vértices. Los experimentos muestran que para la parametrización que producía los mejores resultados de calidad ($Od = 0,8$), el algoritmo toma más tiempo comparado con los otros casos. La explicación es la siguiente: la relación entre el costo de eliminación de nodos y la sustitución de nodos influye fuertemente el espacio de búsqueda del algoritmo. Cuando el costo de eliminar un vértice es alto en comparación con el costo de inserción, antes de eliminar cualquier nodo el algoritmo intenta sustituir etiquetas en el grafo de consulta por etiquetas en el grafo objetivo. Como el algoritmo de emparejamiento de nombres consume bastante tiempo, el tiempo de ejecución es mayor. Cuando se decreta el costo de eliminar, muchos de los mapeos se vuelven muy costosos y no son considerados.

En el siguiente experimento (ver figura 6.20) tratamos de explorar los límites del algoritmo con respecto al tamaño del grafo objetivo. La complejidad teórica del algoritmo de emparejamiento del grafo [165] es $O(m^2 n^2)$ en el mejor caso (cuando la distancia entre el grafo objetivo y el grafo de consulta es mínima) y $O(m^n n)$ en el peor de los casos (m = el número total de vértices en el grafo objetivo; n = el número total de vértices en el grafo consulta).

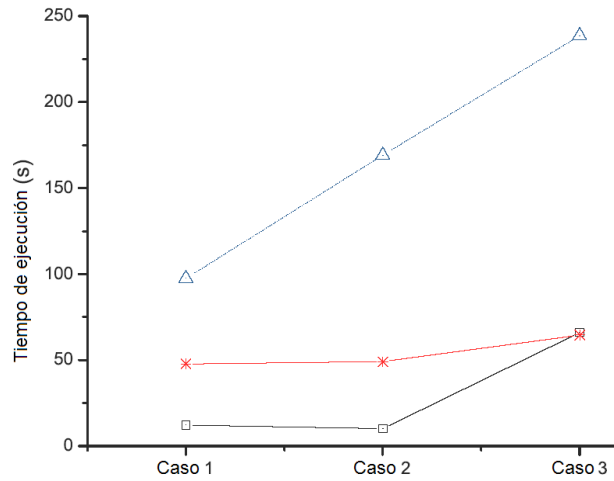


Figura 6.19 Tiempo de ejecución para diferentes funciones de costos

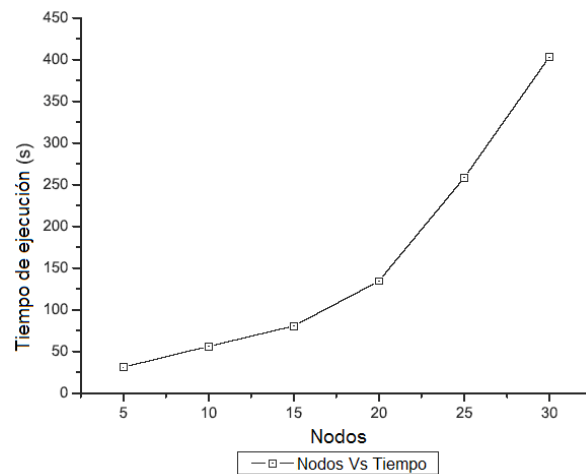


Figura 6.20: Tiempo de ejecución para un número creciente de nodos

En el experimento anterior notamos la búsqueda de sinónimos en Wordnet lleva a la búsqueda de grafos conceptuales muy grandes, lo cual consume mucho tiempo. Con el fin de reducir este tiempo hemos introducido un diccionario *ad hoc* en el cual se puede realizar la búsqueda a un costo muy reducido. (Este diccionario específico del dominio puede ser enriquecido con correspondencias confirmadas a nivel de nombre de interacción y documentos, permitiendo reutilizar los resultados del emparejamiento.). En este contexto, dado un conjunto de grafos objetivos cuya medida varía entre 5 vértices a 30 vértices, los tiempos de ejecución correspondientes se presentan en la figura 6.20. A pesar de que el costo teórico exponencial, la gráfica muestra que el algoritmo de emparejamiento puede utilizarse, con un bajo costo, para las especificaciones WSCL teniendo menos de 30 interacciones, las cuales corresponden a una medida razonable.

6.5.2 Evaluación experimental del mecanismo basado en Calidad

Objetivo de la evaluación experimental

En esta sección se describen los resultados de las pruebas realizadas al proceso de descubrimiento de servicios basado en calidad del servicio. El objetivo es encontrar la incidencia de considerar los criterios de calidad del servicio en el descubrimiento de servicios geográficos para la gestión ambiental del recurso hídrico.

Metodología de experimentación y resultados

Para la experimentación se emplearon solamente tres atributos de calidad con el objetivo de simplificar la implementación del proceso de selección. En el escenario de prueba, tres usuarios del catálogo buscan servicios que cumplan sus requerimientos de calidad mediante el explorador del catálogo. Los servicios a consultar son los siguientes: Cálculo del estudio del índice de calidad del agua (ICA), un servicio para análisis del índice de escasez y un servicio de datos de muestras hidrológicas.

El explorador del catálogo consulta en el registro UDDI los servicios que cumplen la funcionalidad solicitada, posteriormente consulta el modelo de metadatos formulado y los clasifica de acuerdo a las características de calidad solicitadas. Los experimentos fueron llevados a cabo en una máquina Dell, con procesador Pentium 4 de 3GHz y 1000 MB RAM. El espacio total en disco fue de 80 GB. La máquina estaba corriendo bajo el sistema operativo Windows XP.

El proceso de clasificación (ranking) de servicios se realiza teniendo en cuenta: la funcionalidad del servicio, el conjunto de atributos de calidad requeridos y el peso de cada atributo asignados por el usuario (a criterio propio). Antes de efectuar la búsqueda, el usuario configura los atributos de calidad relevantes en la búsqueda (que en adelante serán llamados “dominantes”) y les asigna peso a cada uno según su criterio. La evaluación de los servicios se realiza mediante puntajes asignados con base en el/los atributos dominantes de acuerdo a la siguiente ecuación.

$$ValoracionTotal = \sum_{i=0}^n W_i (PuntajeAtributoQoS_i)$$

n – Número total de atributos dominantes.

W_i , es el peso del atributo definido por el usuario.

$PuntajeAtributoQoS_i$, es la calificación para el i-esimo atributo de calidad.

$$PuntajeAtributoQoS_i = \begin{cases} \frac{Valor_i}{MejorValoracion_i} & \text{Si el incremento del atributo refleja mejoramiento en el desempeño.} \\ \frac{MejorValoracion_i}{Valor_i} & \text{Si el decremento del atributo refleja mejoramiento en el desempeño.} \end{cases}$$

$Valor_i$, es el valor del i-esimo atributo.

$Mejor_{i}$, es el valor más alto/bajo encontrado en el registro para el i-esimo atributo. Los servicios son calificados con valores entre cero y uno.

Resultados de la experimentación

Nuestra primera evaluación busca conocer si la probabilidad de seleccionar un servicio adecuado se incrementa si se especifican los criterios de calidad en el momento de realizar la búsqueda. Recordando la clasificación de los servicios hídricos: procesamiento, datos y gestión; se presupone que los criterios de calidad específicos para cada tipo son más relevantes. Así por ejemplo para un servicio de procesamiento como el cálculo de un estudio como el BMWP podrá ser más importante el desempeño mientras que para un servicio de gestión como el Índice de Escasez será más importante la confiabilidad o la interactividad.

Para las pruebas se establecen requerimientos de calidad de tres usuarios hipotéticos: U1 requiere un estudio para Gestión del Índice de Escasez, U2 un servicio de procesamiento para un análisis de Índice de Calidad de Agua y U3 datos de Muestras Hidrológicas. En el registro se ingresan 27 servicios, para las 3 funcionalidades, con diferentes atributos de calidad específicos como se ve en la Tabla 6.2.

Interactividad	Confiabilidad	Desempeño		
		Alto	Medio	Bajo
Alto	Alto	S1	S10	S19
	Medio	S2	S11	S20
	Bajo	S3	S12	S21
Medio	Alto	S4	S13	S22
	Medio	S5	S14	S23
	Bajo	S6	S15	S24
Bajo	Alto	S7	S16	S25
	Medio	S8	S17	S26
	Bajo	S9	S18	S27

Tabla 6.2 Características de Calidad de los Servicios Prueba

En la Tabla 6.3 se observa los valores de los criterios de Calidad:

	Desempeño (seg)	Confiabilidad (%)	Interactividad
Bajo	0.08	99	1
Medio	0.05	99.9	5
Alto	0.02	99.99	8

Tabla 6.3 Valores para los Criterios de Calidad

Cada usuario realiza cuatro tipos de búsquedas (12 en total), utilizando distintos atributos de calidad dominantes como se detalla en la Tabla 6.4 En esta tabla, un SI quiere decir que el usuario está

interesado en servicios con las mejores prestaciones de calidad posibles en dicho atributo, un NO quiere decir que el atributo no es relevante en el contexto de aplicación (a criterio del usuario).

Usuario	Interactividad	Confiabilidad	Desempeño
U1, U2, U3	No	No	No
U1	Si	No	No
U2	No	Si	No
U3	No	No	Si
U1	Si	Si	No
U2	No	Si	Si
U3	Si	No	Si
U1, U2, U3	Si	Si	Si

Tabla 6.4 Atributos de Calidad Dominantes utilizados en la prueba

Los casos de búsquedas se clasificaron de la siguiente forma:

Caso 1. U1, U2 y U3 no están interesados en la calidad: los usuarios buscan funcionalidad, y no están interesados en el desempeño del servicio.

Caso 2. U1, U2 y U3 solicitan un atributo de calidad: cada usuario busca el servicio con las mejores prestaciones de calidad en un atributo específico. (Un atributo dominante)

Caso 3. U1, U2, U3 solicitan dos atributos de calidad: cada usuario busca el servicio con las mejores prestaciones de calidad en dos atributos. (Dos atributos dominantes)

Caso 4. U1, U2, U3 solicitan tres atributos de calidad: cada usuario busca el servicio con las mejores prestaciones de calidad en tres atributos. (Tres atributos dominantes)

En esta simulación se asignan pesos iguales a los atributos, de manera que la clasificación depende exclusivamente de la elección de los atributos dominantes. A continuación se consignan los resultados para 25 búsquedas por cada caso de pruebas, donde se toma como referencia el primer servicio de la lista ordenada entregada como respuesta.

Caso1. Solo funcionalidad:

Se obtiene la lista de servicios al azar, sin ningún orden específico, como se ve en la Figura 6.21.

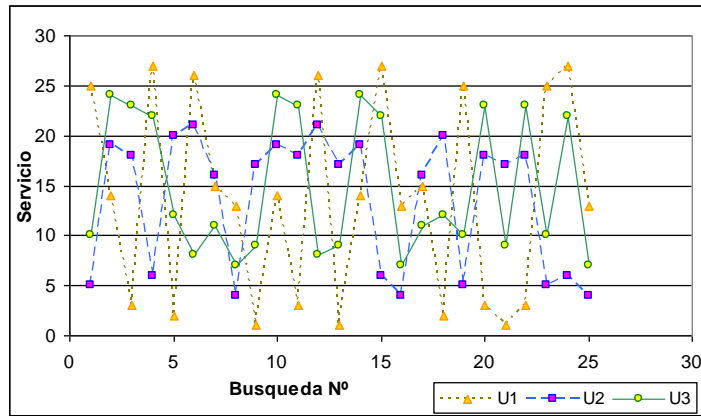


Figura 6.21 Búsqueda de solo funcionalidad

Cada punto representa el servicio ubicado en el tope de la lista retornada como resultado en cada una de las 25 búsquedas. El sistema encuentra los servicios que cumplen la funcionalidad solicitada y entrega una lista aleatoria.

Caso 2. Búsqueda con un atributo de calidad dominante:

Al ingresar un atributo dominante, se tiene como resultado que el número de servicios ubicados en la primera posición de la lista ordenada se reduce, como se en Figura 6.22.

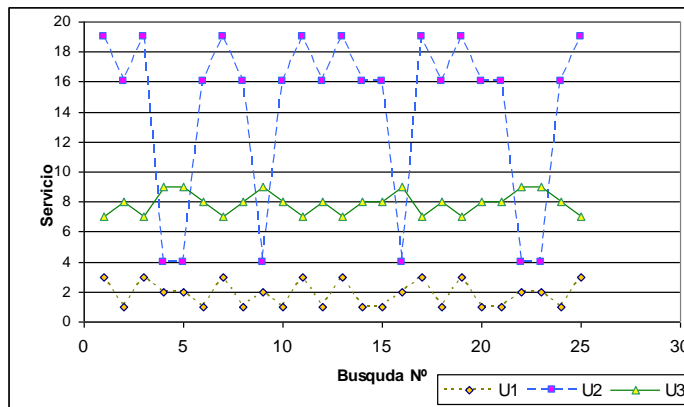


Figura 6.22 Resultado de la búsqueda con un atributo de calidad dominante

Esto indica que utilizar un atributo de calidad dominante permite clasificar los servicios entregados como respuesta.

Caso 3. Búsquedas con dos atributos de calidad dominantes:

Con dos atributos dominantes, los servicios elegibles por el sistema se reducen aun mas como se ve en la Figura 6.23.

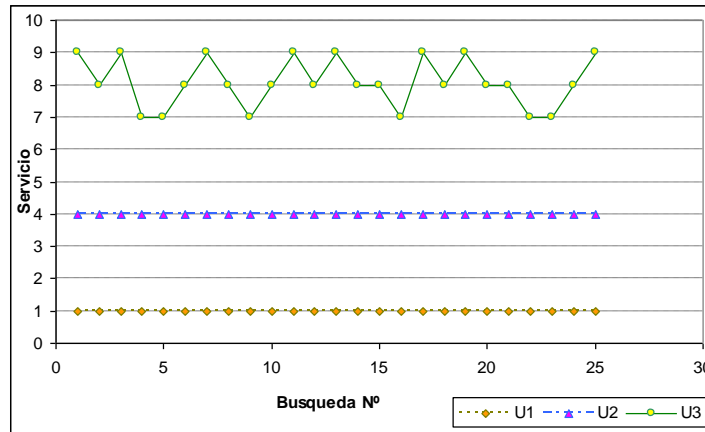


Figura 6.23. Búsqueda con dos atributos de calidad dominantes

Como se puede observar, U1 y U2 obtienen siempre en el primer lugar de la lista un único servicio que cumple con las mejores prestaciones de calidad en los dos atributos solicitados. La traza de U3 se mantiene variando entre tres servicios con similares prestaciones, indicando que el atributo de calidad adicional no influye en la calificación general de calidad.

Caso 4. Búsqueda con tres atributos de calidad dominantes:

Con tres atributos, se discriminan mejor los servicios, al punto que en la primera posición aparece siempre un único servicio como se puede ver en la Figura 6.24.

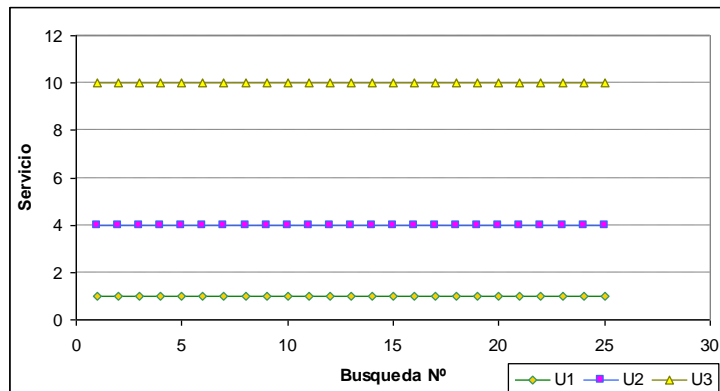


Figura 6.24 Búsqueda con tres criterios de calidad dominantes

Como se puede apreciar, las trazas se estabilizan en el servicio con mejores prestaciones en los tres atributos. La respuesta para U3 deja de variar.

6.6 Resumen

En este capítulo se presentaron tres prototipos creados: el primero es una herramienta que permite la ejecución del algoritmo para el emparejamiento de servicios basado en comportamiento, la segunda, es una aplicación Web que posibilita el registro y consulta de servicios Web geográficos utilizando criterios de Calidad del servicio y la tercera, es una aplicación Web (que constituye el núcleo del

sistema SIRA) y que permite definir, calcular y mostrar informes sobre los indicadores ambientales definidos en el metamodelo de gestión ambiental descrito en el capítulo 2. Además, se presentó una herramienta para evaluar la efectividad del emparejamiento basado en comportamientos. Esta herramienta permite crear un ranking de servicios basada en la comparación manual entre un servicio consulta y un conjunto de servicios en un repositorio. Utilizando esta herramienta es posible comparar el resultado obtenido por la plataforma y el ranking definido por los usuarios.

Para la validación de nuestra propuesta se realizó un análisis de la efectividad del descubrimiento utilizando cada una de las dos propuestas: basada en comportamiento y basada en calidad del servicio.

La evaluación experimental de nuestra propuesta de descubrimiento de servicios basada en comportamiento, se realiza de dos forma, primero queremos analizar la calidad del proceso de emparejamiento. En segundo lugar, queremos probar el tiempo de ejecución del método de emparejamiento utilizando estos protocolos. A pesar del costo teórico exponencial, el algoritmo de emparejamiento puede utilizarse con un costo bajo, para las especificaciones WSCL teniendo menos de 30 interacciones, lo cual constituye un tamaño razonable de servicios para el problema.

En cuanto a las pruebas correspondientes con el descubrimiento basado en calidad del servicio, se comprobó por medio de simulaciones de búsqueda que la definición de los criterios de calidad del servicio, permiten mejorar enormemente el resultado de la consulta de servicios.

Capítulo 7. Conclusiones

7.1 Logros alcanzados con el trabajo

El problema a tratar en este trabajo es la optimización del proceso de descubrimiento de servicios Web geográficos en el dominio de la gestión ambiental del recurso hídrico. Para ello nuestra propuesta se enmarca dentro de dos elementos para mejorar el descubrimiento de servicios de geoprocetamiento en el entorno ambiental: el primer elemento es la aplicación de los dos mecanismos para mejorar el descubrimiento de servicios (búsqueda de servicios basada en el comportamiento de los servicios y en parámetros de calidad del servicio); el segundo elemento es el diseño de un meta-modelo del proceso de gestión de los recursos hidrológicos que permita la aplicación de estos mecanismos de búsqueda en el dominio ambiental hidrológico. En cuanto al primer elemento, relacionado con la aplicación de los dos mecanismos para mejorar el descubrimiento de servicios tenemos: Para la definición de un mecanismo de búsqueda basado en comportamiento se definieron técnicas de emparejamiento de servicios que operan sobre modelos de grafos, que representan el comportamiento de los servicios. Estas técnicas permitan identificar semejanzas parciales, así como la evaluación de la distancia semántica entre estas semejanzas con los requerimientos del usuario. Consecuentemente si no existe un servicio que satisfaga exactamente los requerimientos del usuario, se obtienen los más similares y propuestos para su reuso o modificación. Para conseguir esto, se redujo el problema del emparejamiento de comportamiento de los servicios a un problema de emparejamiento de grafos, adaptando algoritmos existentes para este propósito. Por otro lado, la evaluación experimental de nuestra propuesta de descubrimiento de servicios basada en comportamiento, se realizó de dos formas: primero se analizó la calidad del proceso de emparejamiento; y seguidamente, se probó el tiempo de ejecución del método de emparejamiento utilizando estos protocolos. A pesar del costo teórico exponencial, el algoritmo de emparejamiento puede utilizarse con un costo bajo, para las especificaciones WSCL teniendo menos de 30 interacciones, lo cual constituye un tamaño razonable de servicios para el problema.

En cuanto al mecanismo de búsqueda de servicios Web geográficos para la gestión ambiental del recurso hídrico basada en parámetros de calidad del servicio, se incluyó información sobre la calidad de servicio a nivel de aplicación de los servicios, utilizando descripciones esquematizadas en forma de metadatos asociados a los servicios. De esta manera se definió un perfil de calidad para servicios geográficos hidrológicos, y se extendieron las especificaciones de Metadatos ISO 19115 y NTC 4611 para incluir descripciones de calidad que pudieran ser confrontadas con los requerimientos de calidad de los usuarios. Igualmente, dada la relación intrínseca existente entre los criterios de calidad del servicio y el dominio de aplicación específico, para poder desarrollar mecanismos de descubrimiento de servicios basados en calidad del servicio, es indispensable la definición de un perfil de calidad propio para cada dominio. El perfil de calidad propuesto en este trabajo incluye los siguientes criterios: desempeño, confiabilidad, disponibilidad, costo, reputación, soporte a la interactividad, localización y conformidad. En cuanto a las pruebas correspondientes

con el descubrimiento basado en calidad del servicio, se comprobó por medio de simulaciones de búsqueda que la definición de los criterios de calidad del servicio, permiten mejorar enormemente el resultado de la consulta de servicios.

El segundo elemento presentado como parte de la solución fue el diseño de un meta-modelo del proceso de gestión de los recursos hidrológicos, que permita la aplicación de los mecanismos de búsqueda descritos anteriormente, en el dominio de la gestión ambiental del recurso hídrico. El metamodelo para la gestión de los recursos hidrobiológicos, pretende realizar una abstracción del proceso de análisis ambiental hidrológico, de forma tal que permita generar diferentes modelos dependiendo del tipo de análisis que se requiera. El metamodelo permite modelar todo el proceso de gestión ambiental, que básicamente se compone de tres fases: la fase de monitoreo, el cálculo de los índices y el análisis de los mismos.

7.2 Principales contribuciones

Las principales contribuciones de esta tesis son las siguientes:

- **Un metamodelo para la gestión de estudios ambientales.** Se definió un metamodelo de la gestión de los estudios ambientales que ofrece una descripción global de dichos procesos, lo anterior representa un avance significativo, al brindar un modelo general para la implementación de herramientas software y metodologías de análisis ambiental. Este metamodelo define elementos propios del dominio de la gestión de recursos hidrológicos (como estudios, operaciones, variables); ofreciendo, de esta manera, un marco general y estándar para el descubrimiento y reutilización de servicios geográficos ambientales.
- **Un prototipo para validar el metamodelo para la gestión de estudios ambientales.** Se desarrolló un prototipo para la validación del metamodelo dentro del módulo de análisis Hidrológico de la plataforma SIRA (Sistema de Información Regional del Agua). La herramienta consiste de una Aplicación Web que permite la definición de estudios y el cálculo de algoritmos de gestión ambiental y la aplicación de los mismos sobre un conjunto de datos preestablecido. El sistema implementado sirve de base para el cálculo de todos los indicadores ambientales del sistema. Además, debido a su naturaleza abierta y flexible, el prototipo que implementa el metamodelo, permitió resolver el problema de la adaptación del sistema SIRA [6]. A los diferentes estudios ambientales hidrológicos, lo cual había sido una enorme limitante de las versiones anteriores de SIRA. Esta limitante, persiste aún en la mayoría de las herramientas de gestión ambiental.
- **Un análisis detallado del descubrimiento de servicios Web geográficos Hidrológicos:** los principales resultados de este análisis son: (i) un análisis de las descripciones formales que ofrecen características de representación con el fin de describir, intercambiar y ejecutar servicios. (ii) una descripción de las técnicas existentes para el emparejamiento de servicios genéricos basadas en los formalismos para el modelado de procesos (iii) Una revisión de las propuestas específicas para el descubrimiento de Servicios Web geográficos basadas en calidad del servicio (vi) una revisión analítica del modelamiento de procesos de gestión ambiental del recurso hídrico.

- **Aplicación del emparejamiento de servicios al protocolo WSCL** Con el objetivo de validar el método de descubrimiento basado en comportamiento, se ejemplifica nuestra propuesta para los protocolos de conversación expresados utilizando el modelo WSCL. El proceso de emparejamiento de protocolos de conversación se compone de los siguientes pasos: Primero, los protocolos WSCL se convierten en grafos, a continuación, en ambos grafos se aplica el algoritmo de emparejamiento de grafos con corrección de error y finalmente la función de similaridad evalúa la semejanza entre los grafos.

- **Un prototipo del descubrimiento de servicios basado en comportamiento.** Se desarrolló un prototipo que implementa la propuesta de descubrimiento basado en emparejamiento de comportamientos. La herramienta (disponible como aplicación Web y como Servicio Web) permite la ejecución de algoritmos para emparejamiento de servicios descritos con WSCL.

- **Aplicación de criterios de calidad del servicio al descubrimiento de servicios geográficos ambientales.** Partiendo de la premisa de que la calidad de servicio a nivel de aplicación puede definirse utilizando descripciones esquematizadas en forma de metadatos asociados a servicios y datos; se presenta un perfil de calidad para servicios geográficos, y se extienden las especificaciones de Metadatos ISO 19115 y NTC 4611 para incluir descripciones de calidad que puedan ser confrontadas con los requerimientos de calidad de los usuarios, igualmente se define una arquitectura para el descubrimiento de servicios hidrológicos basado en calidad del servicio.

- **Un prototipo para el descubrimiento de servicios basado en Calidad del servicio.** Se desarrolló un prototipo que implementa la propuesta de descubrimiento basado en calidad del servicio. La herramienta permite la definición de criterios de calidad y la realización de búsquedas de servicios basadas en esos requerimientos. Se realizó la validación de la propuesta utilizando extensiones a los estándares de Metadatos ISO 19115 y NTC 4611, en un repositorio UDDI.

- **Un prototipo para evaluar la efectividad del método de emparejamiento de comportamientos.** Se implementó una herramienta para evaluar la efectividad del método de emparejamiento de comportamiento. Esta herramienta permite crear un ranking de servicios basada en comparaciones manuales entre un servicio de consulta y los servicios en el repositorio. La herramienta permite comparar el resultado obtenido por el prototipo de búsqueda y el ranking definido por los usuarios.

7.3 Trabajo futuro

El trabajo futuro puede enfocarse en las siguientes líneas:

7.3.1. Técnicas de indexación para repositorios de servicios

El soporte para la indexación es necesario para repositorios grandes ya que el número de servicios a comparar es muy alto. De esta manera, la búsqueda en repositorios grandes de servicios para analizar la similaridad entre dos servicios es un proceso computacional costoso [93]. Dado a que nuestra propuesta de descubrimiento basada en comportamiento, se basa en emparejamiento de grafos, y considerando que muchas aproximaciones han sido propuestas para la indexación de bases

de datos (Ver [186],[187]), un trabajo futuro puede explorar cual es el mecanismo de indexación mas apropiado para el proceso de descubrimiento de servicios.

7.3.2. Información semántica para el descubrimiento basado en QoS

Otros de los ejes en los cuales se puede mejorar el proceso de descubrimiento es la inclusión sobre semántica asociada con la calidad del servicio. Nuestra propuesta no incluye información semántica sobre los servicios. La inclusión de este tipo de información puede enriquecer la tanto la descripción de los servicios como la definición de los requerimientos de consulta del usuario [188].

7.3.3. Evaluación de la calidad del descubrimiento

La evaluación de la calidad del proceso de descubrimiento de servicios es un proceso dispendioso debido a la cantidad de datos y la necesidad de la intervención humana. En este trabajo hacemos uso de la técnica de validez de construcción para validar los resultados de la herramienta de comparación de servicios basada en comportamiento y en calidad del servicio. Sin embargo, un futuro campo de investigación puede ser la exploración de nuevas técnicas que mejoren la evaluación de la calidad del proceso de descubrimiento.

7.3.4. Composición de servicios geográficos ambientales

El descubrimiento es solo un parte del proceso de composición automático de servicios geográficos para el dominio de la gestión ambiental del recurso hídrico. En este trabajo solamente se presentan aportes en el área del descubrimiento de servicios. Por lo anterior, una línea muy importante de investigación futura se asocia con la composición automática de servicios geográficos, que involucra muchos otros aspectos, entre los que se encuentra la generación automática de adaptadores para resolver las diferencias entre los servicios emparejados de forma aproximada.

Bibliografía

- [1] L. Burbano, "Indicadores Biológicos en Ecosistemas Acuáticos Continentales". Universidad del Cauca. Tech, Rep. Ag. 2006. <http://ariadna.unicauca.edu.co/indicadores.doc>. Last accessed on June 5, 2007.
- [2] Z. Peng, M. Tsou. "Internet GIS: Distributed Geographic Information Services for the Internet and Wireless Networks" John Wiley and Sons, 2003
- [3] J. Zhang, et al. "Design and development of Distributed Virtual Geographic Environment system based on web services". *Information Sciences: an International Journal* 177(19): 3968-3980, 2007.
- [4] "OpenGIS Abstract Specification, Topic 12: OpenGIS Service Architecture,". Jan. 2002; www.opengis.org/techno/abstract02-112.pdf. Last accessed: Jan 2007.
- [5] J.C. Corrales, D. Grigori, M. Bouzeghoub, and A. Ordonez. Conversation protocol matchmaking. In Proc. of Euro American Conference on Telematics and Information Systems., pages 67–74, 2006.
- [6] A. Ordoñez, et al. "Sistema Integrado de Información sobre el Recurso Agua SIRA". Proc. GIS Day Universidad del Cauca. Popayán, Noviembre.2006.
- [7] D. Grigori, J.C. Corrales, and M. Bouzeghoub. Behavioral matchmaking for service retrieval. In Proc. of The IEEE International Conference on Web Services (ICWS), pages 145–152, 2006.
- [8] J.C. Corrales, D. Grigori, and M. Bouzeghoub. Behavioral matchmaking for service retrieval: application to conversation protocols. In Proc of 22^emes Journées Bases de Données Avancées, BDA, 2006.
- [9] J.C. Corrales, D. Grigori, and M. Bouzeghoub. Decouverte de services basee sur leurs protocolesde conversation. In *Ingenierie des systemes d'information (ISI)*, volume 12, pages 9–32, 2007.
- [10] A. Barros, M. Dumas, and P. Oaks. Standards for web service choreography and orchestration: Status and perspectives. In Proc. of Business Process Management Workshops, 2006.
- [11] R. Milner. *A calculus of communicating systems*. Springer-Verlag NewYork, Inc, 1982.
- [12] C. A. R. Hoare. *Communicating sequential processes*. Prentice Hall International, 1985.
- [13] J.A. Bergstra, A. Ponse, and S.A (eds) Smolka. *Handbook of process algebra*. North Holland, Elsevier, 2001.
- [14] M. Mazzara and R. Lucchi. A pi-calculus based semantics for WS-BPEL. *Journal of Logic and Algebraic Programming*, 2006.
- [15] J. L. Peterson. *Petri net theory and the modeling of systems*. In Prentice-Hall, 1981.
- [16] J. Paananen. Tik-110.501 seminar on network security. <http://www.tml.tkk.fi/Opinnot/Tik-110.501/1995/intfo.html>, 1995. Last accessed on January 2007.
- [17] W. Van der Aalst and K. V. Hee. *Workflow management- models, methods, and systems*. In MIT Press, 2002.
- [18] W. van der Aalst. *Interorganizational workflows: An approach based on message sequence charts and petri nets*. 1999.
- [19] R. Hopcroft, J. E. and Motwani and J. D. Ullman. *Introduction to automata theory, languages, and computation*. 2001.
- [20] D. Berardi, F. ROSA, L. SANTIS, and M. MECELLA. Finite state automata as conceptual model for e-services. 2003.
- [21] G. Weikum. *Towards guaranteed quality and dependability of information services*. 1999.
- [22] H. Bunke. *Graph matching: Theoretical foundations, algorithms, and applications*. pages 82–88, 2000.
- [23] O. Gerbe, R. Keller, , and G. Mineau. *Conceptual graphs for representing business processes in corporate memories*. 1998.
- [24] J. Mendling and J. Ziemann. Transformation of bpeL processes to epcS. In Proc. of the 4th GI Workshop on Event-Driven Process Chains (EPK2005), 2005.

- [25] W.J. Christmas, J. Kittler, and M. Petrou. Structural matching in computer vision using probabilistic relaxation. *IEEE Trans. PAMI* 8, pages 749–764, 1995.
- [26] R. Wilson and E. Hancock. Graph matching by discrete relaxation. *Pattern Recognition in Practice IV: Multiple Paradigms, Comparative Studies and Hybrid Systems*, pages 165–176, 1994.
- [27] A. Branca, E. Stella, and A. Distanto. Qualitative scene interpretation using planar surfaces. *Autonomous Robots*, 8(2):129139, 2000.
- [28] C. Pautasso and G. Alonso. Visual composition of web services. In *Proceedings of the 2003 IEEE Symposium on Human Centric Computing Languages and Environments*, pages 92–99, 2003.
- [29] Special section on graph algorithms and computer vision. *IEEE Trans. PAMI*, 23(10):1049–1151, 2001.
- [30] J.R. Ullman. An algorithm for subgraph isomorphism. *Journal of the Association for Computing*, 23(1):31–42, 1976.
- [31] J. McGregor. Backtrack search algorithms and the maximal common subgraph problem. *Software-Practice and Experience*, 12:23–34, 1982.
- [32] G. Levi. A note on the derivation of maximal common subgraphs of two directed or undirected graphs. *Calcolo*, 9:341–354, 1972.
- [33] M. Pelillo. A unifying framework for relational structure matching. In *Proceedings of the 14th ICPR*, 1998.
- [34] M.A. Eshera and K.S. Fu. A graph distance measure for image analysis. *IEEE Trans. SMC* 14, pages 398–408, 1984.
- [35] A. Sanfeliu and K.S. Fu. A distance measure between attributed relational graphs for pattern recognition. *IEEE Transactions on Systems, Man, and Cybernetics*, 13(5):353–363, 1983.
- [36] L. G. Shapiro and R. M. Haralick. Structural descriptions and inexact matching. *IEEE Trans. Pattern Anal. Mach. Intell.*, 3, 1981.
- [37] W.H. Tsai and K.S. Fu. Error-correcting isomorphisms of attributed relational graphs for pattern recognition. *IEEE Trans. SMC*, 9:757–768, 1979.
- [38] E.K. Wong. Three-dimensional object recognition by attributed graphs. *Syntactic and Structural Pattern Recognition-Theory and Applications*, pages 381–414, 1990.
- [39] H. Bunke and K. Shearer. A graph distance metric based on the maximal common subgraph. *Pattern Recognition Letters*, 19(3-4):255259, 1998.
- [40] J. Lladós, E. Martí, and J. Villanueva. Symbol recognition by error-tolerant subgraph matching between region adjacency graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10):11371143, 2001.
- [41] H. Bunke. Error correcting graph matching: on the influence of the underlying cost function. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(9):917922, 1999.
- [42] B. T. Messmer and H. Bunke. A new algorithm for error-tolerant subgraph isomorphism detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(5):493504, 1998.
- [43] H. Bunke. On a relation between graph edit distance and maximum common subgraph. *Pattern Recognition Letters*, 18(8):689694, 1997.
- [44] M. Fernandez and G. Valiente. A graph distance metric combining maximum common subgraph and minimum common supergraph. *Pattern Recognition Letters*, 22(6-7):753758, 2001.
- [45] J. F. Sowa. *Conceptual structures: Information processing in mind and machine*. Addison-Wesley, 1984.
- [46] John F. Sowa, *Knowledge Representation: Logical, Philosophical, and Computational Foundations*, Brooks Cole Publishing Co., Pacific Grove, CA, ©2000. Actual publication date, 16 August 1999.
- [47] J. Baget and M. Mugnier. Extension of simple conceptual graphs: the complexity of rules and constraints. *Artificial Intelligence Research*, 16:425465, 2002.
- [48] S. Gao and J. Shah. Automatic recognition of interacting machining features based on minimal condition subgraph. *Computer Aided Design*, 30(9):727739, 1998.
- [49] R. Bacik. *Structure of Graph Homomorphism*. PhD thesis, Simon Fraser University, Canada, 2001.
- [50] H. Bunke and S. Gunter. Weighted mean of a pair of graphs. *Computing*, 67(3):209224, 2001.
- [51] X. Jiang, A. Munger, and H. Bunke. On median graphs: Properties, algorithms, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10):11441151, 2001.

- [52] B. T. Messmer and H. Bunke. Efficient subgraph isomorphism detection: a decomposition approach. *IEEE Transactions on Knowledge and Data Engineering*, 12(2):307323, 2000.
- [53] H. Bunke. Recent advances in structural pattern recognition with applications to visual form analysis. In *Proceedings of the Fourth International Workshop on Visual Form IWVF4*, 2001.
- [54] J. Feng, M. Laumy, and M. Dhome. Inexact matching using neural networks. *Pattern Recognition in Practice IV: Multiple Paradigms, Comparative Studies and Hybrid Systems*, pages 177–184, 1994.
- [55] L. Xu and E. Oja. Improved simulated annealing, boltzmann machine, and attributed graph matching. In L. Almeida (ed): *LNCS 41 Springer Verlag*, pages 151–161, 1990.
- [56] D. Riviere, J. Mangin, D. Papadopoulos, J. Martinez, V. Frouin, and J. Regis. Automatic recognition of cortical sulci of the human brain using a congregation of neural networks. *Medical Image Analysis*, 6(2):7792, 2002.
- [57] A. Cross, R. Wilson, and E. Hancock. Genetic search for structural matching. In *Proceedings of the Computer Vision - ECCV*, pages 514–525, 1996.
- [58] Y.-K. Wang, K.-C. Fan, and J.-T. Horng. Genetic-based search for error-correcting graph isomorphism. *IEEE Trans. SMC* 27, 4:588–597, 1997.
- [59] R. Myers and E. R. Hancock. Least-commitment graph matching with genetic algorithms. *Pattern Recognition*, 34(2):375394, 2001.
- [60] S. Umeyama. An eigendecomposition approach to weighted graph matching problems. *IEEE Trans. PAMI* 10, pages 695–703, 1988.
- [61] H. Almohamed. A linear programming approach for the weighted graph matching problem. *IEEE Trans. PAMI* 15, pages 522–525, 1993.
- [62] V. et al. Cantoni. 2-d object recognition by multiscale tree matching. *Pattern Recognition* 31, pages 1443–1455, 1998.
- [63] K. Oflazer. Error-tolerant retrieval of trees. *IEEE Trans. PAMI* 19, pages 1376–1380, 1997.
- [64] M. Pelillo, K. Siddiqi, and S. Zucker. Matching hierarchical structures using associated graphs. *IEEE Trans. PAMI* 21, pages 1105–1120, 1999.
- [65] J. et al. Wang. An algorithm for finding the largest approximately common substructure of two trees. *IEEE Trans. PAMI* 20, pages 889–895, 1998.
- [66] K. Shearer, H. Bunke, and S. Venkatesh. Video indexing and similarity retrieval by largest common subgraph detection using decision trees. *Pattern Recognition*, 34(5):10751091, 2001.
- [67] B. T. Messmer and H. Bunke. Error-correcting graph isomorphism using decision trees. *International Journal of Pattern Recognition and Artificial Intelligence*, 12(6):721742, 1998.
- [68] B. T. Messmer and H. Bunke. A decision tree approach to graph and subgraph isomorphism detection. *Pattern Recognition*, 32:1979–1998, 1999.
- [69] N. Kokash, W. van den Heuvel, and V. D’Andrea. Leveraging web services discovery with customizable hybrid matching. In *Proc. of ICSOC*, pages 522–528, 2006.
- [70] E. Stroulia and Y. Wang. Structural and semantic matching for assessing web-service similarity. *Int. J. Cooperative Inf. Syst.*, 14(4):407–438, 2005.
- [71] M. Saboua and J. Panb. Towards semantically enhanced web service repositories. *Web Semantics: Science, Services and Agents on the World Wide Web*, 5:142–150, 2007.
- [72] L. Dong, A. Halevy, J. Madhavan, E. Nemes, , and J. Zhang. Similarity search for web services. In *Proc. of VLDB*, 2004.
- [73] M. David, M. Paolucci, S. McIlraith, M. Burstein, D. McDermott, D. McGuinness, B. Parsia, T. Payne, M. Sabou, M. Solanki, N. Srinivasan, and K. Sycara. Bringing semantics to web services: The owl-s approach. In *Proc of the First International Workshop on Semantic Web Services and Web Process Composition*, 2004.
- [74] M. Dean and G. Schreiber. Owl web ontology language reference. In *w3c recommendation*, pages <http://www.w3.org/TR/owl-ref>. Last accessed on September 2007, 2004.
- [75] L. C. Chiat, L. Huang, and J. Xie. Matchmaking for semantic web services. In *Proc of Services Computing, 2004 IEEE International Conference on (SCC’04)*, 2004.

- [76] L. Li and I. Horrocks. A software framework for matchmaking based on semantic web technology. In Proc of WWW 2003, pages 331–339, 2003.
- [77] M. Paolucci, T. Kawamura, T. R. Payne, and K. Sycara. Semantic matching of web services capabilities. In Proc. of ISWC, 2002.
- [78] A. Bernstein and M. Klein. Towards high-precision service retrieval. In Proc. of ISWC, 2002.
- [79] B. Benatallah, M.S. Hacid, C. Rey, and F. Toumani. Semantic reasoning for web services discovery. In Proc. of ESSW, 2003.
- [80] D. Trastour, C. Bartolini, and J. Gonzalez-Castillo. A semantic web approach to service description for matchmaking of services. In Proc. of SWWS, 2001.
- [81] Abhijit Patil, Swapna Oundhakar, A. Sheth, and Kunal Verna. Meteor-s web service annotation framework. In Proc. of WWW Conference, 2004.
- [82] K. Sivashanmugam, K. Verma, A. Sheth, , and J. Miller. Adding semantics to web services standards. In Proc of the the 1st International Conference on Web Services (ICWS'03), 2003.
- [83] T. Kawamura, J.A. De Blasio, T. Hasegawa, M. Paolucci, and K. Sycara. A preliminary report of a public experiment of a semantic service matchmaker combined with a uddi business registry. In Proc. of ICSOC, 2003.
- [84] J. Cardoso and A. Sheth. Semantic e-workflow composition. Journal of Intelligent Information Systems, 21:191–225, 2003.
- [85] J. Wu and Z. Wu. Similarity-based web service matching. In Proc. of IEEE SCC, 2005.
- [86] S. S. Bansal and J. M. Vidal. Matchmaking of web services based on the DAML-S service model. In Proc. of AAMAS, pages 926–927, 2003.
- [87] G. Piccinelli, G. Di Vitantonio, and L. Mokrushin. Dynamic service aggregation in electronic marketplaces. Computer Networks, 2(37), 2001.
- [88] A. Wombacher, B. Mahleko, P. Fankhauser, and E. Neuhold. Matchmaking for business processes based on choreographies. In Proc. of EEE, 2004.
- [89] B. Benatallah, F. Casati, and F. Toumani. Analysis and management of web services protocols. In Proc. of ER, 2004.
- [90] L. Bordeaux, G. Salan, D Berardi, and M. Mecella. When are two web services compatible? In Proc. of TES, 2004.
- [91] N. Ould, A. M'Bareck, and S. Tata. Bpel behavioral abstraction and matching. In Business Process Management Workshops, pages 495–506, 2006.
- [92] Li Kuang, Ying Li, Shuiguang Deng, Jian Wu, Wei Shi, and Zhaohui Wu. Expressing service and query behavior using pi-calculus for matchmaking. In Proc of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence, pages 629–632, 2006.
- [93] B. Mahleko and A. Wombacher. Indexing business processes based on annotated finite state automata. In Proc. of The IEEE International Conference on Web Services (ICWS'06), pages 303–311, 2006.
- [94] Z. Shen and J. Su. Web services discovery based on behavior signatures. In Proc. of IEEE SCC, 2005.
- [95] L. Fuentes y A. Vallecillo. “Una Introducción a los Perfiles UML”. Novática: Revista de la Asociación de Técnicos de Informática, ISSN 0211-2124, N°. 168, 2004 (Ejemplar dedicado a: UML e Ingeniería de Modelos), pages. 6-11
- [96] J. Davidson. 2001. OGC General Services Model GSM En Línea, disponible: <http://www.intl-interfaces.net/service/gsm/gsm-2001-08-15.html>. OGC.
- [97] D. Gouscos, M. Kalikakis, and P. Georgiadis. 2003. “An Approach to Modeling Web Service QoS and Provision Price”. Proceedings of the 1st International Web Services Quality Workshop, Rome, Italy, pp.1-10, 2003.
- [98] OpenGIS® Simple Features Implementation Specification for OLE/COM 1.0.
- [99] OpenGIS® Simple Features Implementation Specification for CORBA 1.0.
- [100] S. Tu y M. Abdelguerfi. 2006. Web Services for Geographic Information Systems. IEEE Internet Computing, Vol.10, No5.
- [101] OGC 2005. Open GIS Web Services Architecture Description. Best Practices Paper Reference number OGC 05-042r2. Arliss Whiteside Ed.

- [102]A. Rajabifard, F. Williamson, I.P. (2001). Spatial Data Infrastructures: Concept, SDI Hierarchy and Future directions, Proceedings of GEOMATICS'80 Conference, 29 April-2 May 2001, Tehran, Iran
- [103] L. McKee. (1996) Building the GSDI—discussion paper for the September 1996 Emerging Global Spatial Data Infrastructure Conference. Proceedings of the 1996 Conference on Emerging Global Spatial Data Infrastructure, Königswinter, Bundesrepublik, Deutschland (Germany). September, 1996. European Umbrella Organization for Geographical Information (EUROGI): 19 Pp
- [104]D.Nebert. Whiteside, A. 2004. Open Gis Consortium: Open Gis Catalogue Services Specification. OGC-04-021r2 .v.2.0.
- [105]GeoNetwork. Geonetwork project. <http://sourceforge.net/projects/geonetwork>
- [106] Deegree. Bloques de construcción para Infraestructuras de Datos Espaciales basados en las especificaciones del OGC. (<http://www.deegree.org/>)
- [107]American National Standards Institute. (1985). American national standard Z39.2-1985: Bibliographic information interchange. New York:American National Standards Institute.
- [108] A. Busch, F. Willrich, 2000. “Quality management of atkis data”. Paper presented at OEEPE/ISPRS joint workshop on Spatial Data Quality Management, Istanbul.
- [109] ITU-T, 1994. “Terms and Definitions related to Quality of Service and Network Performance including Dependability”. ITU-T Recommendation E.800. ITU-T.
- [110] L. Franken, 1996. Quality of Service Management: A Model-Based Approach. PhD thesis, Centre for Telematics and Information Technology.
- [111]ITU/ISO, 1995. “Open Distributed Processing - Reference Model Part 2: Foundations” International standard 10746-2, ITU-T Recommendation X.902.
- [112] M.Woodside, D. Menascé. Application-Level QoS. IEEE Internet Computing. Vol. 10, Num. 3. 2006
- [113] D. Menascé. Composing Web Services: A QoS View. IEEE Internet Computing. 2004.
- [114]R. Onchaga. 2004 Modelling For Quality Of Services In Distributed Geoprocessing. International Institute for Geo-Information Science and Earth Observation ITC. Netherlands
- [115] ISO/TC211 2002. ISO/DIS 19119 Geographic Information Services. ISO.
- [116]R. Al-Ali, O. Rana, D. Walker S. Jha & S. Sohail. 2002. “G-QoS: Grid Service Discovery Using QoS Properties”. Department of Computer Science, Cardiff University. Department of Computer Science and Engineering, UNSW, Sydney, Australia.
- [117] L. O'Brien, L. Bass, P. Merson., 2005 “Quality attributes and service oriented architectures” Software Architecture Technology Initiative. CMU-SEI.
- [118] ISO/IEC, 1998. ISO/IEC 13236 “Information Technology – Quality of Service: Framework”, Geneva, Switzerland.
- [119] R. Onchaga, 2005a. “On Quality of Service and Geo-service Compositions”. International Institute for Geo-Information Science and Earth Observation ITC.
- [120]R. Devillers, M. Gervais, Y. Bedard, R. Jeansoulin, 2002. “Spatial data quality: From metadata to quality indicators and contextual end-user manual”. Paper presented at OEEPE/ISPRS joint workshop on Spatial Data Quality Management, Istanbul.
- [121] A. Smith, 2002. “Spatial data quality management at ordinance survey”. Paper presented at OEEPE/ISPRS joint workshop on Spatial Data Quality Management, Istanbul.
- [122] R. Onchaga. 2006 Modelling For Quality Of Services In Distributed Geoprocessing. International Institute for Geo-Information Science and Earth Observation ITC. Netherlands.
- [123]D. Xu, 1994. “An integrated and QoS-Aware framework for multimedia service management”. Doctoral Thesis in philosophy in computer science. B.S Zhongshan University.
- [124] M. Radwan. and Morales, J., 2002. “Extending geoinformation services: a virtual architecture for spatial data infrastructures”. The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 34.
- [125]A. Simonis. 2005. “Quality of Service in a Global SDI”. FIG Working Week 2005 and GSDI-8. Session TS 14 - SIM and Quality; Paper TS14.6 Cairo, Egypt April 16-21.
- [126] S. Ran. 2004. “A Model for Web Services Discovery with QoS”. SIGecom Exchanges, vol. 4, no. 1, 2004, pp. 1–10.

- [127] E. Maximilien. y M. Singh. 2002. "Conceptual Model of Web Services Reputation". ACM SIGMOD Record, 36-41, 2002.
- [128] C. Zhou, L. Chia y B. Lee. 2004. "DAML-QoS Ontology for Web Services". Proceedings. IEEE International Conference on Web Services, 472-479, 2004.
- [129] A. Zeng, B. Benatallah, A. Ngu, M. Dumas, Kalagnanam, y J. Chang. "QoS-Aware Middleware for Web Services Composition". IEEE Transactions on software Engineering. Vol. 30.N.5.
- [130] F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, M. Stal. 1996. Pattern-Oriented Software Architecture. 2da. Ed, Volume 1: A System of Patterns. Wiley.
- [131] M. Rodríguez. "Arquitectura cognitiva para el diseño de entornos telemáticos de enseñanza y aprendizaje". Tesis doctoral licenciado en informática. universidad nacional de educación a distancia escuela técnica superior de ingenieros industriales departamento de ingeniería eléctrica, electrónica y de control 2002:
- [132] M. Niño. "Metamodelo de evaluación la educación en línea". Tesis de maestría, escuela de ingeniería de sistemas universidad industrial de santander. 2003.
- [133] J. Fuentes et al. "Errors in the uml metamodel?". ACMsigsoft software engineering notes, 28(6), nov 2003
- [134] L. Plazaola, E. Silva, N. Vargas, J. Flores, M. Ekstedt, "A Metamodel for Strategic Business and IT Alignment Assessment" In proceeding of conference on systems engineering research (CSER 2006), Los Angeles, USA, Abril 2006.
- [135] A. Kleppe, S. Warmer, W. Bast, "MDA Explained. The Model Driven Architecture: Practice and Promise", Addison- Wesley, April 2003
- [136] J.M. Favre, "Foundations of Model (driven) (Reverse) Engineering - Episode I: Story of the Fidis Papyrus and the Solarus", post-proceedings of Dagstuhl Seminar on Model Driven Approaches for Language Engineering, May 2004 in Series "From Ancient Egypt to Model Driven Engineering". <http://www-adele.imag.fr/mda>. Last accessed on January 2007.
- [137] J.M. Favre, "Foundations of Meta-Pyramids: Languages and Metamodels - Episode II: Story of Thotis the Baboon", postproceedings of Dagstuhl Seminar on Model Driven Approaches for Language Engineering, May 2004, in Series "From Ancient Egypt to Model Driven Engineering". <http://www-adele.imag.fr/mda>. Last accessed on January 2007.
- [138] Meta Object Facility (MOF) Specification" Version 1.4, April 2002.
- [139] E. Seidewitz, "What Models Mean", IEEE Software, September 2003.
- [140] J. Martínez et. al. "Cauca región líder en el manejo integral del agua, recurso articulador entre el entorno natural y la actividad humana para la competitividad regional," Grupo de Estudios Ambientales, Universidad del Cauca, Ago. 2006. http://ariadna.unicauca.edu.co/CAUCA_1.pdf. Last accessed on June 2007.
- [141] GSDI. "Spatial Data Infrastructure Cookbook v2.0" GSDI, Jan. 2004. <http://www.gsd.org/gsdicookbookindex.asp>. Last accessed on June 2007.
- [142] UNESCO. 2006. Water a shared responsibility. The United Nations World Water Development Report 2. Berghahn Books.
- [143] Minambiente. 2007. Ministerio de Ambiente, Vivienda y Desarrollo Territorial. Gestión Integral del Agua. Dirección de Agua Potable y Saneamiento Básico y Ambiental. En línea, disponible en: <http://www.minambiente.gov.co/admin./contenido/documentos/gestionintegraldelagua.pdf>. Bogotá.
- [144] IDEAM. 2007. Instituto de Hidrología, Meteorología y Estudios Ambientales. Ministerio del Medio Ambiente. <http://www.ideam.gov.co>.
- [145] IDEAM. 2006. Estudio Nacional del Agua. Instituto de Hidrología, Meteorología y Estudios Ambientales. Ministerio del Medio Ambiente. <http://www.ideam.gov.co/biblio/paginaabierta/guia.pdf>
- [146] M. Vargas. Nelson O. 2004. Monitoreo de Aguas Subterráneas. Documento sin publicar.
- [147] OGC 2006. OpenGIS® Catalogue Services - eBRIM ISO/TS 15000-3 profile of CSW. V 1.0.
- [148] OMS. 2007. Guidelines for Drinking Water Quality: Vol.1 Recommendations. World Health Organization. Geneva. En línea. Disponible en: http://www.who.int/water_sanitation_health/dwq/gdwq3.
- [149] Cinara. 2002. Seminario Taller Aprovechamiento Integral del Recurso Hídrico con Énfasis en Riego, Drenaje y Sostenibilidad. Universidad del Valle. Santiago de Cali; CINARA; 540 p.

- [150] N. Navarro. Congreso Ibérico sobre Gestión y Planificación de Aguas. 2001. 750 p. Universidad de Zaragoza. Zaragoza, España.
- [151] M. Hernández, M. Estudio de la Calidad del Agua, Estado Trófico y Caracterización de Factores Físicos de la Laguna de Sonso. Universidad del Valle Escuela de Recursos Naturales y el Ambiente. Maestría en Ingeniería Énfasis en Ingeniería Sanitaria y Ambiental. Santiago de Cali.
- [152] E. Geldreich. Microbial Quality of Water Supply in Distribution Systems. 504 p.
- [153] L. Barba. Química Ambiental. Universidad del Valle: Postgrado de Ingeniería Sanitaria y Ambiental. Santiago de Cali. 55 p.
- [154] E. De Lange, Ocampo, P., Muñoz, N., Visscher, J. 2001. Simplified Water Quality Assessment for Multi-Stage Filtration. Instituto de Investigación y Desarrollo en Agua Potable, Saneamiento Básico y Conservación del Recurso Hídrico. The Hague; International Water and Sanitation 38 p.
- [155] J. Gutiérrez, R. Ospina. 2004. Bioindicación de la Calidad del Agua con Macroinvertebrados Acuáticos en la Sabana de Bogotá, utilizando redes neuronales artificiales. Universidad Nacional de Colombia, Bogotá. Disponible en: <http://www.unal.edu.co/icn/publicaciones/caldasias/261/11F.pdf>.
- [156] A. Pinilla. 1998. Indicadores biológicos en ecosistemas acuáticos continentales de Colombia. Compilación bibliográfica. Santafé de Bogotá. Centro de Investigaciones Científicas. Fundación Universitaria de Bogotá Jorge Tadeo Lozano. P. 11-13, 40.
- [157] R. Muñoz. 2005. Macroinvertebrados bioindicadores de la calidad del agua en Cuba. [En Línea] Disponible en: <http://www.dama.gov.co>. Last accessed January 2008.
- [158] L. Burbano. 2006a. Indicadores Biológicos en Ecosistemas Acuáticos Continentales. Grupo de Estudios Ambientales – Universidad del Cauca.
- [159] G. Vasquez, G. Zamora, G. Naundorf. 2001. Estudio Limnológico del río Cauca, sector embalse de la Salvajina – puente El Hormiguero. Popayán. Universidad del Cauca.
- [160] G. Roldan. 1992. Limnología y eutrofización de embalses en Colombia, segunda circular. Río Negro, Antioquia.
- [161] H. Rivera, E. Domínguez, R. Ramirez, R. Venegas. 2004. Metodología para el cálculo del índice de escasez de agua superficial. IDEAM.
- [162] M. Burbano. 2006. Índice De Calidad Del Agua. Grupo de Estudios Ambientales –Universidad del Cauca.
- [163] J. Bézivin, O. Gerbé, "Towards a Precise Definition of the OMG/MDA Framework", ASE'01, November 2001.
- [164] H. Bunke, X. Jiang, and A. Kandel. On the minimum common supergraph of two graphs. Computing, 65(1):13–25, 2000.
- [165] B. Messmer. Graph Matching Algorithms and Applications. PhD thesis, University of Bern, 1995.
- [166] A. Banerji, C. Bartolini, D. Beringer, V. Chopella, K. Govindarajan, A. Karp, H. Kuno, M. Lemon, G. Pogossians, S. Sharma, and S. Williams. Web services conversation language (wscl) 1.0. In W3C, 2002.
- [167] R. C. Angell, G. E. Freund, and P. Willett. Automatic spelling correction using a trigram similarity measure. Information Processing and management, 19(4):255–261, 1983.
- [168] G. Miller. Wordnet: A lexical database for english. Communications of the ACM, 38(11):39–41, 1995.
- [169] ICONTEC 2000. Información geográfica. Metadatos. NTC 4611. Instituto Colombiano de Normas Técnicas y Certificación. Santafé de Bogotá.
- [170] R. Onchaga, 2005b. "Modelling For Quality of Services In Distributed Geoprocessing". International Institute for Geo-Information Science and Earth Observation ITC Enschede, The Netherlands.
- [171] D. Garvin, 1988. Managing Quality: The strategic and Competitive Edge. New York.
- [172] Z. Xu, 2006. "Reputation-Enhanced Web Services Discovery with QoS". Thesis submitted to the School of Computing for the degree of Master of Science. Queen's University. Canada.
- [173] G. Griffiths, 2007. User satisfaction as a measure of system performance. Journal of Librarianship and Information Science. 39: 142-152.
- [174] A. Kluwer, 2000. "Managing Qos in Multimedia Networks and Services". Third International Conference on Management of Multimedia Networks and Services Mmns'2000 - ISBN: 9780792379621.
- [175] IDEE 2005. Núcleo Español de Metadatos NEM v1.0. Infraestructura de Datos Espaciales Española.

- [176] M. Manso. 2004 .Metadatos en los sistemas de información geográfica ISO- 19115. Escuela Técnica Superior de Ingenieros en Telecomunicaciones.
- [177] A. Zabala, J. Masó. 2005 . Aplicación del estándar ISO 19139 a un modelo relacional d-e capa, tablas y campos. Univ. Autónoma de Barcelona.
- [178] ISO 2004 . 19139 Geographic information – Metadata – XML schema implementation. ISO Project Information.
- [179] S. Bacharach. 2007. OGC Adopts ebRIM Metamodel for Catalogues. Cover Pages. Wayland, USA. En línea, disponible: <http://xml.coverpages.org/OGC-ebRIM-200701.html>
- [180] Oasis 2002. UDDI Version 2.03 Data Structure Reference. UDDI Committee Specification.
- [181] K. Januszewski. 2002 . The Importance of Metadata: Reification, Categorization, and UDDI. Microsoft Corporation.
- [182] T. Aditya, T. y Lemmens, R. 2006. “Chaining Distributed GIS Services”. http://www.itc.nl/library/Papers_2003/non_peer_conf/aditya.pdf. Last accessed on January 2008.
- [183] J. Clark. 2005. A Framework for Autonomic Web Service Selection. Department of Computer Science University of Saskatchewan. Saskatoon.
- [184] I. Jacobson, G. Booch, and J. Rumbaugh. The unified software development process. In Addison-Wesley, 1998.
- [185] H. Melnik, S. and Garcia and E. Rahm. Similarity flooding : A versatile graph matching algorithm and its application to schema matching. In 18th ICDE, 2002.
- [186] D.W. Williams, J. Huan, and W. Wang. Graph database indexing using structured graph decomposition. In 23rd International Conference on Data Engineering (ICDE), 2007.
- [187] W. Wang, C. Wang, Y. Zhu, B. Shi, J. Pei, X. Yan, and J. Han. Graphminer: a structural patternmining system for large disk-based graph databases and its applications. In Proc of the ACM SIGMOD International conference on Management of data(SIGMOD), 2005.
- [188]I. Roussaki, I. Papaioannou. “QoS awareness support in Web-Service semanticsTsesmetzis”, M.E.Telecommunications, 2006. AICT-ICIW apos;06. Proceedings International Conference on Internet and Web Applications and Services/Advanced International Conference on Volume , Issue , 19-25 Feb. 2006 Pages: 128 – 128

Anexo 1. Herramientas de desarrollo de aplicaciones Web SIG de software libre

A continuación se presenta una descripción de las principales herramientas de desarrollo de aplicaciones Web SIG de software libre.

1. MapServer.

Originalmente fue un proyecto desarrollado por la Universidad de Minnesota (UMN) y actualmente continúa con el patrocinio de la NASA. Es un ambiente de desarrollo para construir aplicaciones Web que soporten manejo de información espacial; no es un sistema de información geográfica (SIG), ni pretende serlo, en lugar de ello, sobresale por la interpretación de información espacial como mapas e imágenes en ambientes Web. MapServer además de facilitar búsquedas de información geográfica, permite al usuario crear sus propios mapas e imágenes con datos asociados y finalmente publicarlos en un portal para compartirlos con otras personas. El proyecto es mantenido por un gran número de desarrolladores en todo el mundo y recibe fondos de un grupo de organizaciones [1]. Entre sus principales características tenemos:

- La escala de la cartografía mostrada depende de las características del mapa y su aplicación.
- Mejor control para las características de etiquetado en los mapas.
- Totalmente configurable.
- Soporta fuentes TrueType.
- Provee elementos de información en mapas de manera automática, como: barra de escala, convenciones y leyendas.
- Los mapas temáticos tienen soporte en clases lógicas.
- Soporta múltiples ambientes integrados de desarrollo, como también lenguajes de programación como: PHP, Python, Perl, Ruby, Java y C#.
- Funciona en varias plataformas: Linux, Windows, Mac OS X, Solaris y más.
- Trabaja sin problema con multitud de formatos de datos: TIFF/GeoTIFF, EPPL7, ESRI shapefiles, PostGIS, ESRI ArcSDE, Oracle Spatial, MySQL y otras.
- Tiene buena capacidad de hacer proyecciones cartográficas.
- Basado en las especificaciones Web del OGC como: WMS, WFS, WMC, WCS, Filter Encoding, SLD, GML y SOS.

2. GeoTools.

Proyecto de software libre que reúne varias librerías Java ajustadas a los métodos y estándares propuestos para el manejo de datos espaciales, ideal para desarrollar sistemas de información geográfica (SIG). El conjunto de librerías implementadas siguen las especificaciones del OGC con la colaboración de los proyectos GeoAPI y GeoWidgets [2]. Entre sus principales características tenemos:

- Bibliotecas programadas en Java para el desarrollo de soluciones SIG.
- Tiene una arquitectura modular que permite añadir nuevas funcionalidades de manera sencilla en mapas geográficos interactivos.
- Soporta OpenGIS y otros estándares de desarrollo para sistemas geográficos.

3. Deegree.

Deegree es un framework desarrollado en Java que ofrece los bloques constitutivos para construir infraestructuras de datos espaciales. Su arquitectura ha sido desarrollada con base en los estándares del OGC y el comité ISO/TC 211. Deegree interpreta datos en formatos vector y ráster de diferentes fuentes y entrega esta información a cualquier cliente que realice solicitudes http, Get y Post. Los formatos de almacenamiento soportados son:

- PostgreSQL/ PostGIS
- Oracle(Spatial/Locator)
- Shapefiles
- GML2 y GML3 para WFS
- Imágenes JPEG, GIF, PNG, BMP, TIFF y GeoTIFF para WCS.

4. GeoNetwork Open Source.

Es una herramienta y ambiente para el manejo de información espacial, es descentralizada y basada en estándares, diseñada para acceder a bases de datos georreferenciadas, cartografía y metadatos relacionados ubicadas en diversas fuentes, potencializa el intercambio y compartimiento de información espacial al soportarse en Internet. El alcance del manejo de información espacial ha llegado al punto de servir a una gran comunidad de usuarios cuando éstos utilizan mapas temáticos en la toma de decisiones. El principal objetivo de GeoNetwork es lograr mayor accesibilidad a una amplia variedad de datos, información espacial asociada, en diferentes escalas y ubicadas en múltiples fuentes las cuales son organizadas y documentadas en un estándar de manera consistente[3]. Provee:

- Búsquedas rápidas en catálogos geoespaciales locales o distribuidos.
- Carga y descarga de datos, documentos, PDF's y cualquier tipo de archivo.
- Posee un visor WebMap interactivo para combinar servicios web geográficos desde cualquier servidor en el mundo.
- Editor de metadatos online con un avanzado sistema de plantillas.
- Sincronización de metadatos entre catálogos distribuidos.
- Acceso finamente controlado.
- Gestión de grupos y usuarios.

Bibliografía.

- [1] Sitio oficial de Mapserver. <http://mapserver.gis.umn.edu>. Last accessed January 2008.
- [2] GeoAPI 2007 . GeoAPI Working Group. Open GIS Consortium Technical Committee. <http://geoapi.sourceforge.net/stable/site/charter.html>. Last accessed January 2008.
- [3] GeoNetwork 2007. GeoNetwork opensource. En línea, disponible: <http://geonetwork-opensource.org/>

Anexo 2. Integración de la plataforma de descubrimiento basada en Calidad del Servicio

SIRA[1] es una herramienta software cuyo objetivo es soportar la gestión eficiente de los recursos hídricos; es una aplicación Web de fácil manejo, expandible a nuevos análisis y funcionalidades y que permite agregar cartografía nueva, desarrollada con base en una arquitectura abierta y orientada al servicio, con tecnología robusta y software libre. En SIRA existen cuatro roles o usuarios: administrador de cartografía, administrador del componente ambiental, usuario de ingreso de datos y usuario de consulta. La integración con SIRA, como con cualquier otra aplicación se realiza exponiendo el servicio del catalogo como servicio Web. Existen dos posibilidades, utilizar las especificaciones de los servicios Web convencionales o utilizar las especificaciones de servicios OGC. La primera opción tiene el inconveniente de utilizar estructuras de mensajes no estandarizadas. La mejor opción es implementar las operaciones y los mensajes definidos por la especificación catalogo de OGC, a continuación se listan dichas operaciones[2]:

- OGC_Service.GetCapabilities: Como respuesta se obtiene el documento que describe las características del servicio de catalogo. Esta operación es obligatoria.
- CSW-Discovery.GetRecords: Ejecuta una consulta y genera el documento de respuesta con los resultados. En el modelo general existen dos operaciones search y present, estas dos operaciones son combinadas en la operación GetRecords. Esta operación es obligatoria.
- CSW-Discovery.GetRecordById: Consulta registros del repositorio utilizando su identificador y genera un documento de respuesta. Es la implementación de la operación Present definida por el modelo general del catalogo. Se asume que previamente se conocen los identificadores que serán usados en la operación. Esta operación es un subconjunto de la operación GetRecord utilizada para apuntar registros específicos en el catalogo. Esta operación es obligatoria.
- CSW-Discovery.DescribeRecord: Permite a un cliente del catalogo descubrir elementos del modelo de información soportados por el catalogo. Todas las estructuras de información pueden ser descritas.
- CSW-Discovery.GetDomain: Determina el dominio y genera un documento de respuesta. Es usado para obtener el rango de valores permitido en los metadatos o en los elementos de respuesta en tiempo de ejecución.
- CSW-Publication.Transaction: Ejecuta una transacción y genera un documento de respuesta. Define una interfaz para crear, modificar o borrar registros del catalogo.
- CSW-Publication.Harvest: Genera un documento de respuesta que confirma la recepción de la petición y la ejecuta de forma asíncrona. El modelo general define dos operaciones que pueden ser utilizadas para crear o actualizar registros en el catalogo. Estas son, la operación Transactions y la operación HarvestRecords. La operación Harvest referencia los datos a ser insertados o actualizados en el catalogo, quien posteriormente se responsabiliza por traer la información y procesarla. Tiene dos formas de operación: Síncrona o asíncrona.

Como se puede apreciar, implementando las tres operaciones obligatorias se cumple con la especificación OGC Catalog y se obtiene un perfil del catálogo con funcionalidades básicas. Cada operación tiene asociado duplas de mensajes petición / respuesta, como se aprecia en la figura 1.

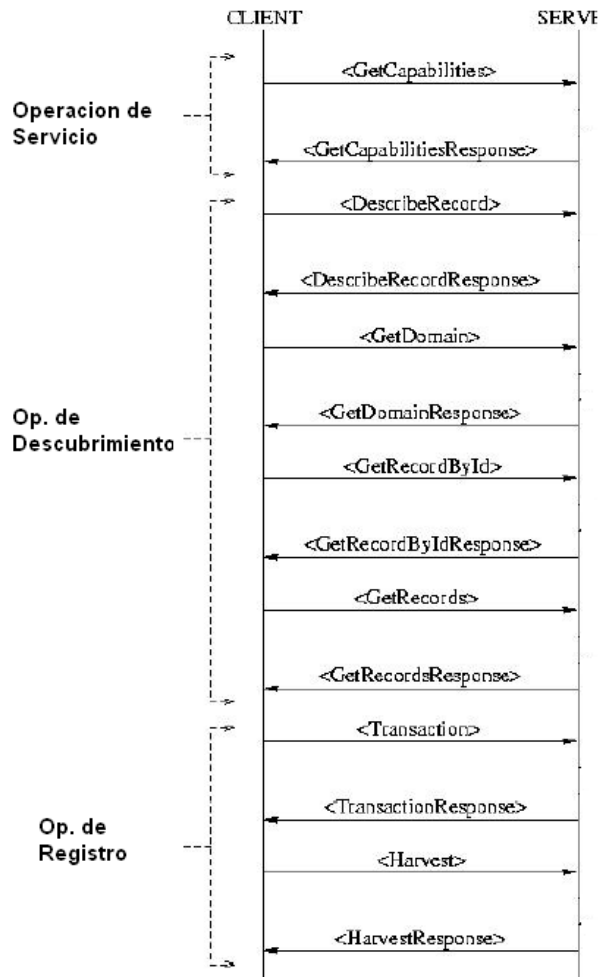


Figura 1. Mensajes propuestos por OGC Catalog

Como contenedor se especifica el protocolo SOAP, de esta forma cuando el cliente o el servidor envían una petición o respuesta, se genera un mensaje SOAP cuyo contenido es la operación definida por el mensaje OGC. El formato de los mensajes puede consultarse en la especificación. Las operaciones, por el contrario, no están especificadas y se pretende que estas sean construidas de acuerdo al contexto de aplicación[2]. El catalogo implementa las siguientes operaciones:

Nombre de la Operación Propuesta	Equivalencia OGC
Metadatos()	OGC_Service.GetCapabilities
BuscarQoSInfo()	CSW-Discovery.GetRecords,
DominioQoS()	CSW-Discovery.GetDomain
BuscarServicio()	CSW-Discovery.GetRecordById
IngresarServicio()	CSW-Publication.Transaction
ActualizarServicio()	CSW-Publication.Transaction
BorrarServicio()	CSW-Publication.Transaction

Tabla 1. Operaciones propuestas para la implementación.

La operación BuscarQoSInfo permite extraer registros ordenados del catalogo de acuerdo a un parámetro de calidad solicitado. DominioQoS entrega los criterios de calidad definidos por el esquema y el formato

definido que debe utilizarse en el momento de realizar la consulta. En general el proceso de gestión de servicios y la arquitectura no cambia, a nivel programático se agregan estas operaciones para lograr que diversas aplicaciones accedan al catálogo utilizando un mecanismo estándar. En la Figura A2.2. Puede apreciarse la relación del módulo de integración propuesto con el resto del sistema.

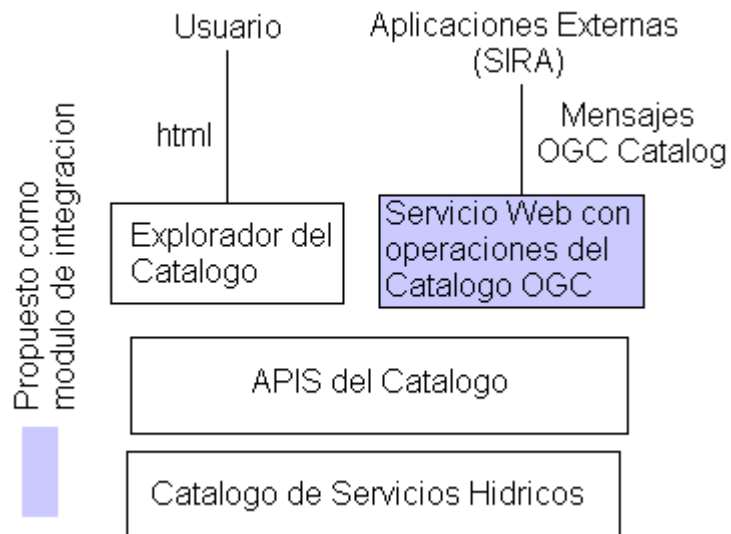


Figura A2.2 Módulo de integración

Bibliografía del Anexo

- [1] A. Ordoñez, et al. “Sistema Integrado de Información sobre el Recurso Agua SIRA”. Proc. GIS Day Universidad del Cauca. Popayán, Noviembre.2006.
- [2] OGC 2005. Open GIS Web Services Architecture Description. Best Practices Paper Reference number OGC 05-042r2. Arliss Whiteside Ed.