

Arquitectura de Referencia para la Composición de Servicios en Ambientes de Computación Ubicua



Anexos

Víctor Alberto Hermida Quintero

Director: Mag. Oscar Mauricio Caicedo Rendón

**Universidad del Cauca
Instituto de Postgrados en Electrónica y Telecomunicaciones
Maestría en Ingeniería, Área Ingeniería Telemática
Departamento de Telemática
Línea de Investigación en Servicios Avanzados de Telecomunicaciones
Popayán, enero de 2011**

ANEXO A

ANEXO A

ARTEFACTOS OBTENIDOS EN LA PLANEACIÓN, CONSTRUCCIÓN Y ELABORACIÓN DE LA PLATAFORMA DE COMPOSICIÓN

A.1 Fase de Planeación y Elaboración

Esta sección presenta los resultados del proceso de ingeniería llevado a cabo para el desarrollo de la arquitectura que soporta la plataforma propuesta.

Para el desarrollo de los componentes software que conforma la plataforma propuesta, se utilizó la metodología sugerida en el Modelo de Construcción de Soluciones (Serrano, 05). La cual plantea el desarrollo de un prototipo de valoración considerando las siguientes fases: Establecimiento de responsabilidades, Descripción de la solución, Definición de la Arquitectura, Implementación y Puesta a Punto, y Evaluación. Para este proyecto la etapa de Implementación y Puesta a Punto se desarrolló en iteraciones incrementales incluyendo actividades de Verificación y Validación al final de cada iteración.

Se plantearon las siguientes iteraciones:

Iteración 1: Implementación del prototipo de Emparejamiento de Actividades BPEL, Integración del Repositorio de Servicios con el módulo de Emparejamiento (ServiceMatch), Verificación y Validación.

Iteración 2: Definición del metadato que describe los requerimientos de contexto del servicio, Implementación del mecanismo de consulta de los metadatos de contexto. Adaptación del prototipo ServiceMatch al contexto, Integración del prototipo ServiceMatch con el Repositorio de Capacidades de Dispositivos (SeMatch-Context), Implementación del Repositorio de Usuarios, Integración del prototipo SeMatch-Context con el Repositorio de Usuarios (U-ServiceMatch), Verificación y Validación.

Iteración 3: Implementación del prototipo de composición de servicios (UBeComp) e integración con el prototipo de Emparejamiento de Actividades BPEL considerando el contexto (SeMatch-Context).

A.1.1 Casos de Uso

En esta sección se presenta el diagrama de casos de uso de la plataforma desarrollada en el presente trabajo de maestría. Los casos de uso se identificaron a partir de las funcionalidades que ofrece el sistema.

A.1.1.1 Diagrama de Casos de Uso de la Plataforma U-ServiceMatch

La Figura A1 presenta el diagrama de casos de uso general para la plataforma, donde se pueden apreciar los tipos de usuarios que interactúan con la aplicación y las funcionalidades del sistema.

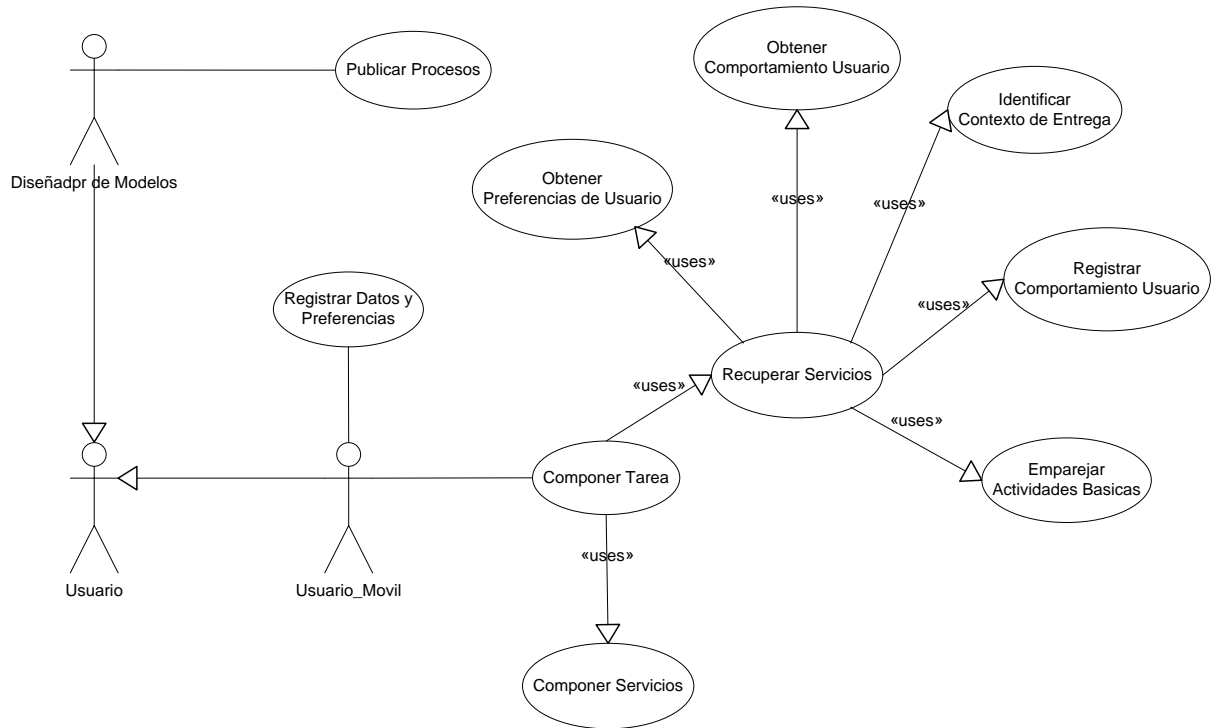


Figura A 1. Diagrama de Casos de Uso de la Plataforma de Composición.

A.1.1.2 Casos de uso en formato expandido

En esta sección se presenta los casos de uso en formato expandido para la Plataforma de Composición.

Caso de uso:	REGISTRAR DATOS Y PREFERENCIAS	
Actores:	Usuario – Cliente Móvil (Iniciador).	
Propósito:	Ingresar al sistema.	
Resumen:	Permite registrar información personal y preferencias de usuario, para posteriormente crear su perfil dentro del sistema.	
Tipo:	Primario y extendido.	
Curso normal de eventos		
	Acción de los Actores	Respuesta del sistema

<ol style="list-style-type: none"> Este caso de uso inicia cuando un usuario quiere ingresar al sistema por primera vez. El usuario ingresa sus datos y preferencias personales para realizar su ingreso y posteriormente crear el <i>Perfil de Usuario</i>. El usuario ingresa a la plataforma. 	<ol style="list-style-type: none"> El sistema presenta la interfaz de Registro de la Plataforma al nuevo usuario.
---	--

Tabla A 1. Caso de uso en formato expandido Registrar Datos y Preferencias.

Caso de uso:	PUBLICAR PROCESOS	
Actores:	Usuario – Diseñador de Modelos (Iniciador).	
Propósito:	Publicar modelos de comportamiento.	
Resumen:	Permite publicar los documentos BPEL y los requerimientos del contexto asociado a dicho proceso en el Repositorio de Procesos.	
Tipo:	Primario y extendido.	
Curso normal de eventos		
	Acción de los Actores	Respuesta del sistema
	<ol style="list-style-type: none"> Este caso de uso inicia cuando un diseñador de modelos quiere publicar sus procesos en el <i>Repositorio de Procesos</i> del sistema. El usuario selecciona los procesos que desea publicar. 	<ol style="list-style-type: none"> El sistema presenta al proveedor la interfaz de Publicación de Servicios. El sistema valida los tipos de documentos enviados por el usuario El sistema almacena los documentos enviados por el usuario el <i>Repositorio de Modelos</i>

Tabla A 2. Caso de uso en formato expandido Publicar Procesos.

Caso de uso:	COMPONER TAREA	
Actores:	Usuario – Cliente Móvil (Iniciador).	
Propósito:	Componer una tarea de usuario.	
Resumen:	Este caso de uso permite a un cliente móvil enviar un proceso de consulta especificando la tarea de usuario requerida y recibir como respuesta un proceso compuesto por los servicios disponibles que más se acerquen al requerimiento de usuario.	
Tipo:	Primario y extendido.	
Curso normal de eventos		
	Acción de los Actores	Respuesta del sistema

<ol style="list-style-type: none"> Este caso de uso inicia cuando un cliente móvil desea usar la funcionalidad: <i>Componer Tarea</i> ofrecida por la Plataforma. El usuario envía el proceso de consulta y pide que se inicie la composición de la tarea especificada. El usuario recibe el proceso compuesto. 	<ol style="list-style-type: none"> El sistema presenta al usuario la interfaz para ingresar el proceso de consulta. El sistema inicia el proceso de composición de los servicios más pertinentes a la consulta del usuario. El sistema retorna el proceso compuesto generado para el requerimiento enviado por el usuario
--	--

Tabla A 3. Caso de uso en formato expandido Componer Tarea.

Caso de uso:	RECUPERAR SERVICIOS	
Actores:	Usuario – Cliente Móvil (Iniciador).	
Propósito:	Recuperar Servicios.	
Resumen:	Este caso de uso permite a un cliente móvil consultar y recuperar los servicios más relevantes que se ajusten a su petición, mediante una fase emparejamiento a nivel atómico de actividades BPEL, considerando tanto las restricciones del contexto, como las preferencias del usuario.	
Tipo:	Primario y extendido.	
Curso normal de eventos		
Acción de los Actores	Respuesta del sistema	
<ol style="list-style-type: none"> Este caso de uso inicia cuando un cliente móvil desea usar la funcionalidad: <i>Buscar Servicios</i> ofrecida por la Plataforma. El usuario solicita a la plataforma encontrar los servicios más adecuados a los requerimientos de su petición. El usuario recibe el ranking de servicios recuperados 	<ol style="list-style-type: none"> El sistema presenta al usuario la interfaz de consulta de servicios. El sistema inicia el proceso de recuperación de los servicios más pertinentes a la consulta del usuario. El sistema retorna un ranking de los servicios más relevantes encontrados que pueden ser consumidos por el cliente móvil. 	

Tabla A 4. Caso de uso en formato expandido Recuperar Servicios.

Caso de uso:	OBTENER COMPORTAMIENTO DEL USUARIO
Actores:	Usuario – Cliente Móvil (Iniciador).

Propósito:	Obtener el comportamiento del Usuario
Resumen:	Este es caso de uso permite a la plataforma obtener información del comportamiento del usuario capturada dinámicamente, tal como servicios consumidos y patrones de uso del sistema.
Tipo:	Primario y extendido.
Curso normal de eventos	
Acción de los Actores	Respuesta del sistema
<ol style="list-style-type: none"> Este caso de uso inicia cuando un cliente móvil realiza las siguientes acciones: Consulta servicios, seleccionar servicios consultados y consumir servicios. El usuario solicita a la plataforma encontrar los servicios más adecuados a los requerimientos de su petición. El usuario recibe el ranking de servicios sugeridos de acuerdo a su perfil. 	<ol style="list-style-type: none"> El sistema presenta al usuario la interfaz de consulta de servicios. El sistema inicia el proceso de recuperación de los servicios más pertinentes a la consulta del usuario. El sistema retorna un ranking de los servicios sugeridos previamente al usuario, o a usuarios con un perfil similar que hayan consultado los mismos servicios.

Tabla A 5. Caso de uso en formato expandido Obtener Comportamiento del Usuario.

Caso de uso:	OBTENER PREFERENCIAS DEL USUARIO
Actores:	Usuario – Cliente Móvil (Iniciador).
Propósito:	Obtener las preferencias del Usuario
Resumen:	Este caso de uso permite obtener el perfil de usuario para la fase de descubrimiento.
Tipo:	Primario y extendido.
Curso normal de eventos	
Acción de los Actores	Respuesta del sistema
<ol style="list-style-type: none"> El usuario solicita a la plataforma encontrar los servicios más adecuados a los requerimientos de su petición. 	<ol style="list-style-type: none"> El sistema obtiene el perfil del usuario y lo compara con perfiles similares para ofrecer una mayor cantidad de posibles servicios relevantes a la solicitud (historial de servicios consultados y consumidos), con el fin de agilizar el proceso de recuperación.

Tabla A 6. Caso de uso en formato expandido Obtener Preferencias del Usuario.

Caso de uso:	REGISTRAR COMPORTAMIENTO DE USUARIO
---------------------	--

Actores:	Usuario – Cliente Móvil (Iniciador).
Propósito:	Obtener y Registrar el comportamiento del Usuario
Resumen:	Una vez el proceso de recuperación de servicios ha sido ejecutado, el sistema registra tanto los servicios consumidos, como los que simplemente han sido consultados. Este caso de uso permite registrar dicho comportamiento en el <i>Repositorio de Usuarios</i> y actualizar el perfil de los usuarios.
Tipo:	Primario y extendido.
Curso normal de eventos	
Acción de los Actores	Respuesta del sistema
1. El usuario selecciona los servicios recuperados más pertinentes que satisfagan sus requerimientos.	2. El sistema registra el comportamiento del usuario, es decir: los servicios consultados y consumidos, y los dispositivos usados para ello.

Tabla A 7. Caso de uso en formato expandido Registrar Comportamiento de Usuario.

Caso de uso:	EMPAREJAR ACTIVIDADES BÁSICAS	
Actores:	Usuario – Cliente Móvil (Iniciador).	
Propósito:	Comparar dos actividades para determinar su similitud	
Resumen:	Este caso de uso permite comparar dos nodos, que representan actividades básicas de BPEL (<i>receive</i> , <i>invoke</i> y <i>reply</i>), y calcular la distancia semántica entre ellos.	
Tipo:	Primario y extendido.	
Curso normal de eventos		
Acción de los Actores	Respuesta del sistema	
1. Este caso de uso inicia cuando el usuario solicita a la plataforma encontrar servicios que se adecuen los requerimientos de su petición.	<p>2. Verifica que los nodos a comparar tengan el mismo tipo de actividad (<i>receive</i>, <i>invoke</i> y <i>reply</i>).</p> <p>3. El sistema toma dos nodos del mismo tipo de actividad y los compara mediante el uso de algoritmo que permiten determinar la distancia lingüística entre ellos.</p> <p>4. Una vez comparados los nodos <i>Query</i> (consulta del usuario) contra los nodos <i>Target</i> (almacenados en el repositorio). Organiza los nodos más pertinentes según su valor de similitud. Entregando un ranking de los servicios más adecuados.</p>	

Tabla A 8. Caso de uso en formato expandido Emparejar Actividades Básicas.

Caso de uso:	IDENTIFICAR EL CONTEXTO DE ENTREGA	
Actores:	Usuario – Cliente Móvil (Iniciador).	
Propósito:	Comparar dos actividades para determinar su similitud	
Resumen:	Este caso de uso permite identificar las restricciones del solicitante y obtener los requerimientos del servicio. Información relevante para el proceso de descubrimiento de servicios en entornos tan dinámicos como los ambientes ubicuos	
Tipo:	Primario y extendido.	
Curso normal de eventos		
Acción de los Actores	Respuesta del sistema	
1. Este caso de uso inicia cuando el usuario ingresa y solicita a la plataforma encontrar los servicios relevantes para ejecutar un comportamiento deseado.	2. El sistema identifica las cabeceras HTTP y recolecta la información relacionada con el dispositivo dentro de WURFL y UAProf. 3. Obtiene las capacidades del cliente móvil y crea el contexto de entrega. 4. EL sistema obtiene las restricciones del contexto en el cual el servicio recuperado puede ser consumido.	

Tabla A 9. Caso de uso en formato expandido Identificar el Contexto de Entrega.

Caso de uso:	COMPONER SERVICIOS	
Actores:	Usuario – Cliente Móvil (Iniciador).	
Propósito:	Componer un conjunto de servicios recuperados de acuerdo al comportamiento especificado por la tarea de usuario requerida	
Resumen:	Este caso de uso permite coordinar los servicios recuperados durante el proceso de descubrimiento de acuerdo al comportamiento requerido en la tarea de usuario recibida como petición en la plataforma.	
Tipo:	Primario y extendido.	
Curso normal de eventos		
Acción de los Actores	Respuesta del sistema	
1. Este caso de uso inicia cuando el usuario móvil ingresa a la plataforma y solicita componer una tarea. 3. El usuario ingresa el comportamiento de la tarea que desea realizar y lo envía a la plataforma.	2. El sistema despliega la interfaz de para ingresar el comportamiento requerido. 4. El sistema inicia el descubrimiento de servicios y obtiene el ranking de servicios recuperados. 5. El sistema toma el ranking de servicios recuperados y crea las síntesis de	

	<p>composición, de acuerdo a la conformidad de las interfaces de los diferentes servicios y el flujo de ejecución del comportamiento solicitado por el usuario.</p> <p>6. Se selecciona la síntesis de composición más adecuada al requerimiento del usuario</p> <p>7. El sistema retorna al usuario el comportamiento compuesto.</p>
--	---

Tabla A 10. Caso de uso en formato expandido Componer Servicios.

A.2 Fase de Construcción

A.2.1 Diagramas de clases del Sistema

En esta sección se expone los diagramas de clases de las aplicaciones software definidas que componen la plataforma de composición: *ServiceMatch*, *SeMatch-Contex* y *UBeComp*. Las descripciones de las clases de cada diagrama son también presentadas en esta sección.

A.2.1.1 Diagrama de Clases del Analizador BPEL - Grafos

La Figura A2 presenta las clases empleadas por el traductor de BPEL a Grafos. A continuación se explica de manera general el funcionamiento y relaciones entre las clases.

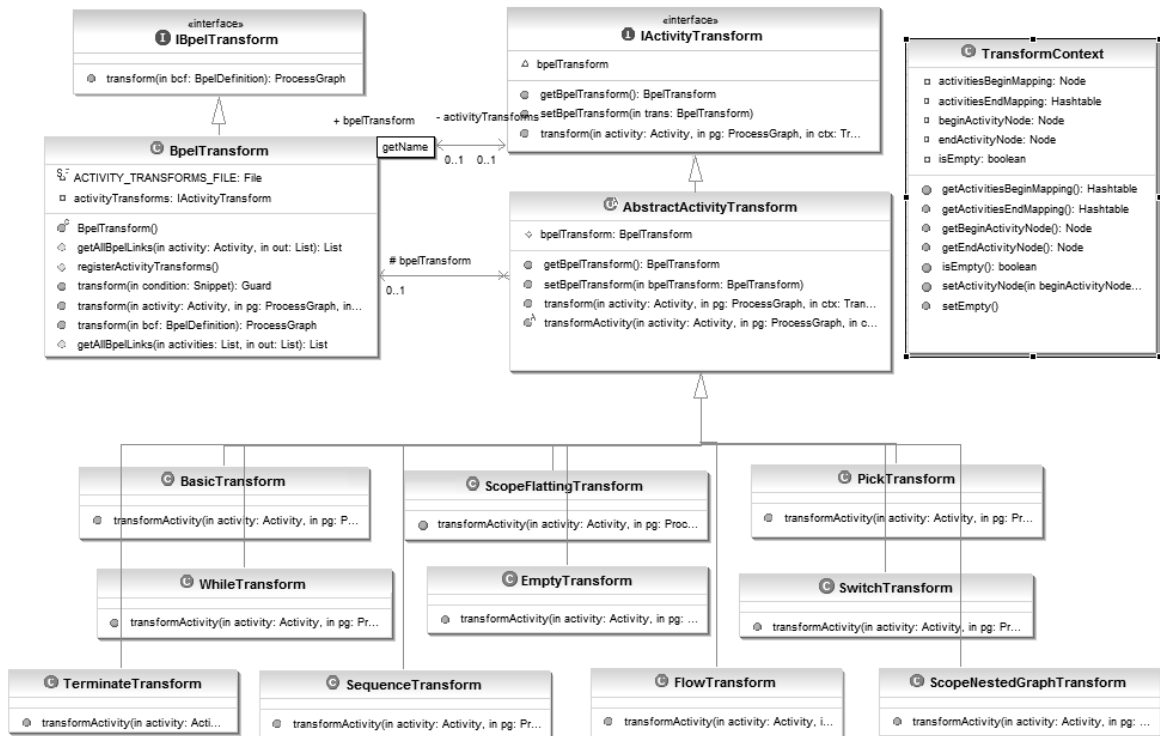


Figura A 2. Diagrama de Clases para la Transformación de un documento BPEL a Grafos

Inicialmente la clase *BpelReader* carga el archivo (.bpel) y escoge un lector especializado para cada actividad. Estos lectores no son más que clases, las cuales implementan la interfaz *IActivityReader*. Esta interfaz se configura manipulando un archivo XML el cual especifica la lectura de cada actividad gracias a las etiquetas *<activityReader>* las cuales tienen dos atributos, el primero corresponde al nombre de la actividad que va a ser procesada y el segundo a la clase Java que realiza la lectura. Seguido, se almacena todo lo concerniente a una representación en objetos del documento BPEL leído, para cada parte del archivo de entrada se extraen sus componentes y se crean objetos que representan fielmente al proceso de negocio.

La parte más importante del analizador es la clase *BpelTransform*, responsable de transformar el metamodelo de BPEL en un objeto de tipo *ProcessGraph*. Éste es el punto de entrada o el punto de inicio para la transformación de un proceso BPEL en un Grafo. La interfaz *IActivityTransform*, debe ser implementada por cualquier clase que quiera transformar una actividad en particular. En conclusión, la clase *BpelTransform*, tiene las estrategias de transformación de cada actividad BPEL. Otra clase importante del paquete en cuestión y en cuanto a estrategias se refiere, es la *AbstractActivityTransform*, la cual contiene todas las estrategias de transformación comunes a todas las actividades.

A.2.1.2 Diagrama de Clases de la aplicación ServiceMatch

La Figura A3 presenta el diagrama de clases de la aplicación *ServiceMatch*, las cuales han sido agrupadas según la funcionalidad que implementen: Gestión del Repositorio de Servicios, Emparejamiento de Servicios, Manejo de Grafos y Presentación.

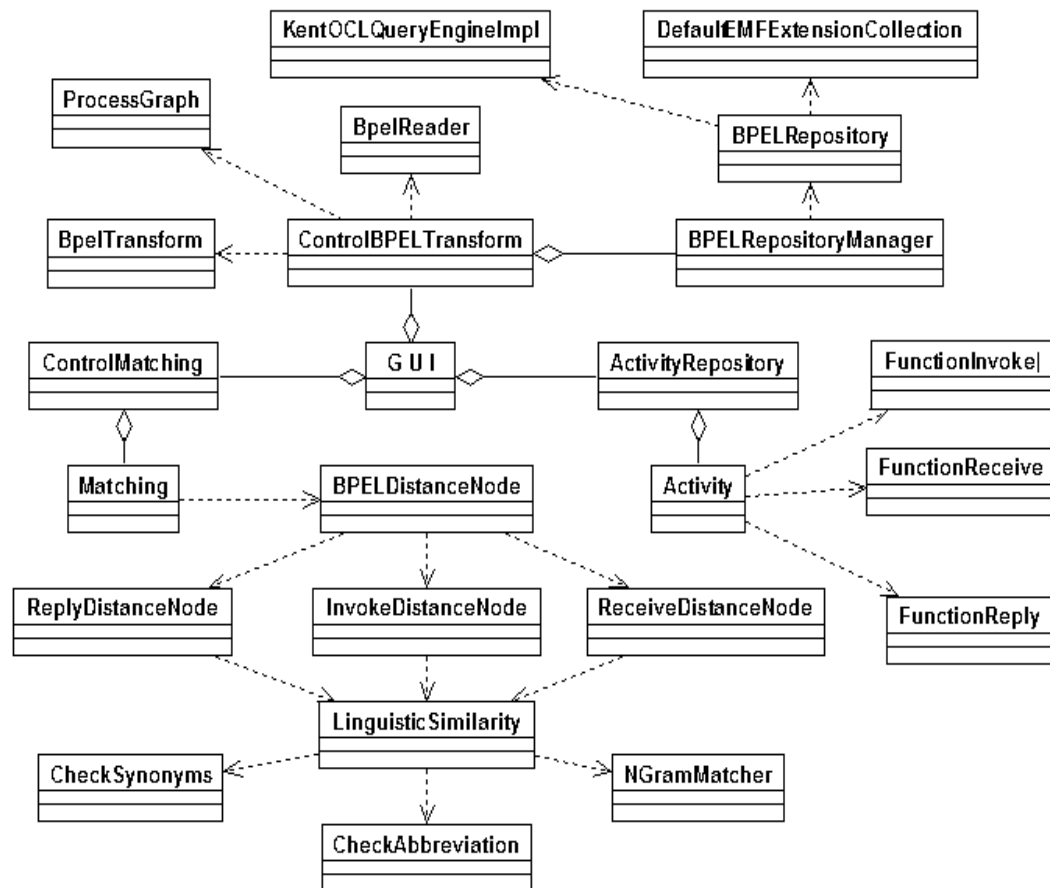


Figura A 3. Diagrama de Clases aplicación ServiceMatch

A continuación se presenta una descripción detallada de las clases que conforman la aplicación ServiceMatch.

❖ **Gestión del Repositorio de Servicios:** a continuación se describen las clases que implementan funcionalidades que permiten manipular el *Repositorio de Servicios*. Estas clases son:

- *BPELRepositoryManager*: esta clase contiene la lógica que permite gestionar, tanto las consultas como la colección de extensiones EMF (*Eclipse Modeling Framework*) del Repositorio de Procesos BPEL.
- *DefaultEMFExtensionCollection*: una de las más importantes características del Repositorio de Procesos de Negocio es la posibilidad de registrar múltiples modelos EMF, para soportar el almacenamiento de diferentes tipos de archivos, extendiendo así su capacidad. Esta clase permite registrar los modelos para BPEL, WSDL, XSD (por defecto soportados dentro del repositorio) y otros.
- *KentOCLQueryEngineImpl*: esta clase implementa el motor de consulta principal de la versión pública del *Repositorio de Servicios* desarrollado por la Universidad de Kent. Es un motor de consulta OCL de código abierto que implementa el estándar OCL 2.0. Para este prototipo, el motor OCL de Kent obtiene las URI de los archivos BPEL almacenados en el repositorio de procesos de negocio.

- *BPELRepository*: esta clase permite inicializar el Repositorio de Procesos, registrando tanto objetos EMF como los motores OCL de consulta.
- *BpelReader*: esta clase carga los archivos *.bpel* encontrados en el repositorio, los lee, mediante lectores especializados para cada actividad, y almacena toda la información necesaria para el proceso de transformación. Para cada parte del archivo de entrada se extraen sus componentes y se crean objetos que representan finalmente el procesos de negocio.
- *ProcessGraph*: esta clase permite almacenar en un metamodelo de objetos la estructura resultante de la serialización del documento BPEL, representando finalmente un grafo mediante arcos, funciones, conectores y nodos *start/end*. Las funciones son los nodos del grafo que representan acciones específicas. Los nodos *start/end* representan el nodo inicial y el nodo final del grafo. Los conectores son nodos que hacen posible las conexiones entre elementos al igual que pueden llevar a cabo diversas conexiones en el proceso.
- *BpelTransform*: esta clase implementa las estrategias de transformación de cada actividad BPEL, siendo la responsable de transformar el metadato de BPEL en un Objeto Java de tipo *ProcessGraph*. Esta clase contiene la lógica para implementar el *Analizador BPEL a Grafos*, el cual es un proyecto desarrollado en Java, que define el diagrama de clases presentado en la Figura A2 para realizar la transformación de un documento BPEL a grafos.
- *ControlBPELTransform*: esta clase implementa la lógica de control que permite almacenar, encontrar, usar y definir archivos basados en el esquema estándar XML. Además, permite transformar archivos BPEL en su equivalente en Grafos.
- ❖ **Manejo de Grafos:** a continuación se describen las clases que implementan funcionalidades que permiten operar sobre los Grafos abstraídos de los procesos encontrados en el *Repositorio de Servicios*. Estas clases son:
 - *ActivityRepository*: esta clase contiene la lógica para obtener las funciones/nodos de los Grafos de procesos, para organizarlos y almacenarlos por tipo de actividad (*invoke, reply, receive*).
 - *Activity*: esta clase representa la actividad/nodo/función de un Grafo. La cual puede ser: *FunctionInvoke, FunctionReceive, FunctionReply*. Cada función posee cuatro atributos: *Tipos de Actividad, Nombre de la operación PartnerLink* y *PortType*.
- ❖ **Emparejamiento de Servicios:** a continuación se describen las clases que implementan funcionalidades que soportan el proceso de emparejamiento de servicios. Estas clases son:
 - *Matching*: esta clase toma como entradas dos nodos, que representan actividades básicas de BPEL (*receive, invoke, reply*), y calcula la distancia lingüística entre ellos. Para esto utiliza las clases: *InvokeDistanceNode, ReceiveDistanceNode* y *ReplyDistanceNode*; quienes a su vez usan la clase *LinguisticSimilarity*.

- *BPELDistanceNode*: esta clase implementa la lógica que permite verificar que los nodos a ser comparados tengan el mismo tipo de actividad.
 - *InvokeDistanceNode*: esta clase permite obtener la distancia lingüística entre dos nodos tipo *Invoke*, para lo cual se fijan los pesos *Wop*, *Wpl* y *Wpt* que indican la contribución de la similitud de *Operación*, *PartnerLink* y *PortType*, respectivamente, a la similitud de las actividades.
 - *ReceiveDistanceNode*: esta clase permite obtener la distancia lingüística entre dos nodos tipo *Receive*, para lo cual se fijan los pesos *Wop*, *Wpl* y *Wpt* que indican la contribución de la similitud de *Operación*, *PartnerLink* y *PortType*, respectivamente, a la similitud de las actividades.
 - *ReplyDistanceNode*: esta clase permite obtener la distancia lingüística entre dos nodos tipo *Reply*, para lo cual se fijan los pesos *Wop*, *Wpl* y *Wpt* que indican la contribución de la similitud de *Operación*, *PartnerLink* y *PortType*, respectivamente, a la similitud de las actividades.
 - *LinguisticSimilarity*: calcula la similitud lingüística entre dos etiquetas basándose en sus nombres. Para obtener esta medida se utilizan los algoritmos *Ngram*, *Check synonym* y *Check abbreviation*, implementados por las clases *CheckAbbreviation*, *CheckSynonyms* y *NGramMatcher* respectivamente.
 - *CheckAbbreviation*: esta clase implementa la lógica que le permite estimar la similitud de dos cadenas usando un diccionario de abreviaciones adecuado al dominio de aplicación.
 - *CheckSynonyms*: implementa la lógica que le permite estimar la similitud de dos cadenas usando el diccionario lingüístico Wordnet, para identificar sinónimos.
 - *NGramMatcher*: implementa la lógica que le permite estimar la similitud de dos cadenas de acuerdo al número común de *qgramas* entre las etiquetas.
- ❖ **Presentación:** a continuación se describen las clases que implementan funcionalidades de representación visual del prototipo *ServiceMatch*. Estas clases son:
- *GUI*: esta clase implementa la lógica de presentación para las interfaces gráficas del prototipo *ServiceMatch*.
 - *ControlMatching*: esta clase implementa el control de los datos entre los procesos de tratamiento y visualización de la información.

A.2.1.3 Diagrama de Clases de la aplicación SeMatch-Context

La Figura A4 presenta el diagrama de clases de la aplicación SeMatch-Context, las cuales son descritas a continuación.

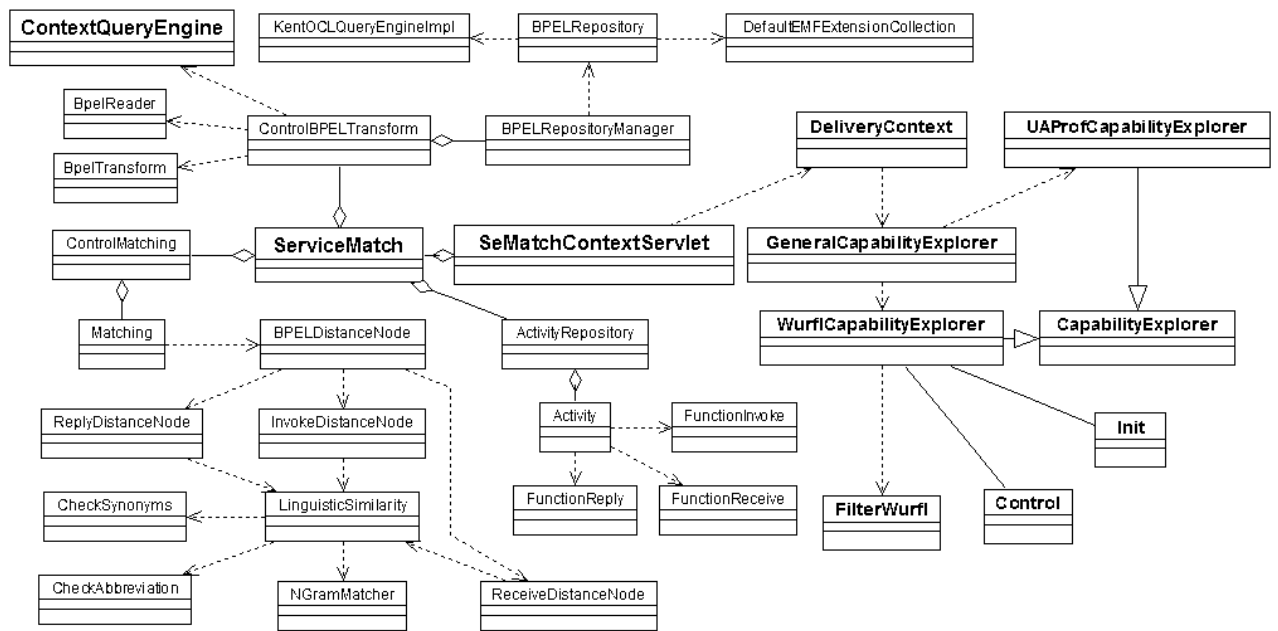


Figura A 4. Diagrama de Clases aplicación SeMatch-Context

- *ContextQueryEngine*: esta clase es utilizada por *ControlBPELTransform* de la aplicación *ServiceMatch* expuesta en la sección anterior, la cual implementa un mecanismo de consulta del valor correspondiente al atributo de contexto (*Tipo de Acceso*) de los servicios publicados en el repositorio.
- *ServiceMatch*: representa el *.JAR (Java ARchive)* del prototipo *ServiceMatch* definido. Los *.jar* son un tipo de archivo que permite ejecutar aplicaciones escritas en lenguaje Java. Dada la lógica implementada en el *ServiceMatch*, permite determinar la similitud entre un nodo *Query* y una colección de nodos *Target* (1:N), y entregar una lista ordenada de las actividades más similares a un nodo de consulta.
- *SeMatchContextServlet*: su función es atender las peticiones provenientes de un cliente móvil a través de su navegador Web. Esta clase implementa la lógica que le permite al prototipo *SeMatch-Context* obtener el contexto de entrega de los clientes que acceden a la aplicación. Además, controla el proceso de recuperación de servicios, considerando tanto el requerimiento del contexto de los servicios, como las restricciones de contexto del usuario.
- *DeliveryContext*: esta clase se encarga de obtener el contexto de entrega, utilizando tres fuentes de información: Cabeceras HTTP, UAProf y WURFL.
- *GeneralCapabilityExplorer*: esta clase permite explorar las capacidades de un dispositivo móvil, obteniendo la información de la biblioteca WURFL o de los repositorios UAProf.
- *UAProfCapabilityExplorer*: esta clase se encarga de explorar las capacidades de un móvil obteniendo la información en los repositorios UAProf.

- *WurflCapabilityExplorer*: esta clase se encarga de explorar las capacidades de un móvil obteniendo la información en la biblioteca WURFL.
- *CapabilityExplorer*: es la Interfaz que define la estructura de los exploradores de capacidades para dispositivos móviles.
- *Init*: clase encargada de cargar en RAM el archivo “*wurfl.xml*” que contiene la información de la biblioteca WURFL.
- *Control*: clase encargada de controlar la obtención de información en las bibliotecas WURFL.
- *FilterWurfl*: esta clase permite buscar las capacidades de un dispositivo en la biblioteca WURFL mediante la cabecera HTTP, “*User-Agent*”.

A.2.1.4. Diagrama de Clases de la aplicación UBeComp

La Figura A5 presenta el diagrama de clases de la aplicación UBeComp, las cuales son descritas a continuación.

- *ControlComposition*: esta clase realiza la coordinación del proceso de composición, recibe las parejas obtenidas después de realizar el descubrimiento de servicios e inicia el proceso de síntesis y selección de la composición.
- *HyperGraph*: es la implementación de los hiper-grafos, es empleado para realizar la síntesis de composición. Posteriormente es consultado para llevar a cabo la selección de la síntesis de composición más adecuada.
- *HyperNode*: cada uno de los nodos que hacen parte del hiper-grafo.
- *HyperArch*: cada una de las hiper-aristas que hacen parte del hiper-grafo. Referencia a un nodo de origen y a uno o más nodos de destino.
- *HyperGraphEngine*: esta clase implementa los algoritmos concebidos para realizar la síntesis de composición. A partir del Proceso de consulta y las parejas encontradas en el descubrimiento, crea el hiper-grafo de composición.
- *EngineComposition*: esta clase implementa los algoritmos concebidos para realizar la selección de una síntesis de composición. Al terminar la selección se obtiene el comportamiento que será entregado al usuario para su posterior ejecución.
- *CompositionException*: representa las excepciones que pueden ser lanzadas durante el proceso de composición, principalmente cuando no es posible encontrar una combinación de servicios para componer la tarea de usuario requerida.

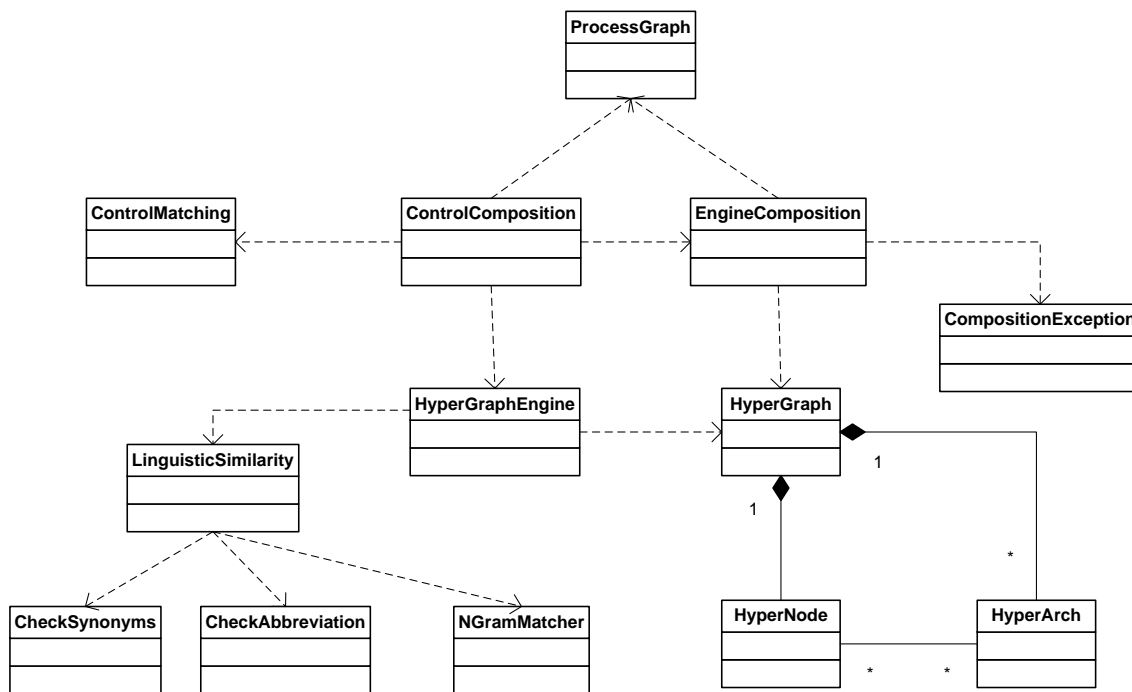


Figura A 5. Diagrama de Clases la Aplicación UBeComp

A.2.2 Diagramas de paquetes del Sistema

La Figura A6 presenta la vista lógica de la plataforma de composición de servicios. A continuación se describen las capas y su interacción, así como los paquetes más relevantes que las componen.

❖ Capa de aplicación:

- *Aplicación Móvil*: representa las aplicaciones móviles que emplea el usuario para interactuar con la plataforma. A través de estas aplicaciones el usuario puede enviar las peticiones al sistema y recibir los procesos compuestos para ejecutar sus tareas.

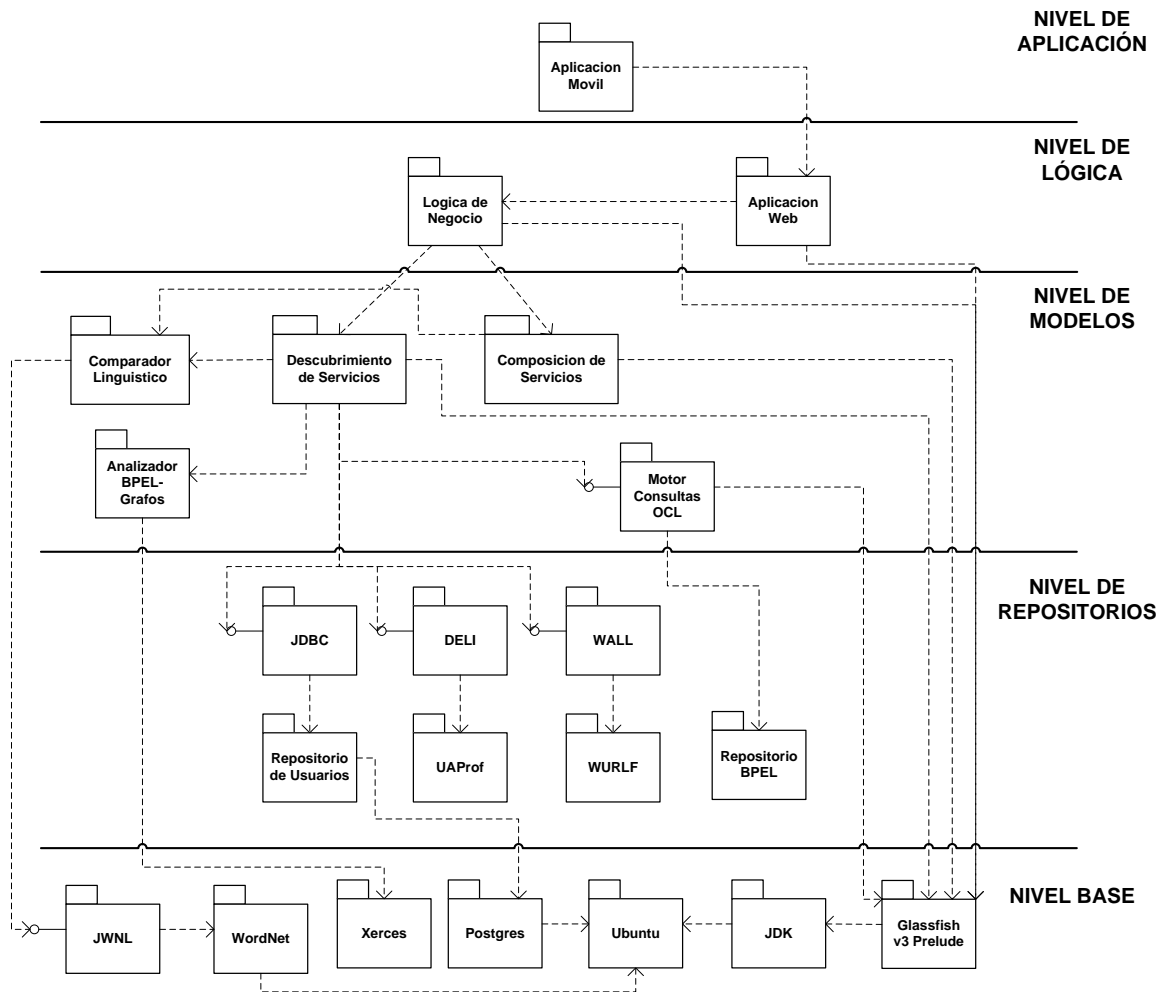


Figura A 6. Vista lógica de las Aplicaciones

❖ Capa de lógica:

- *Aplicación Web:* contiene la aplicación web que se proporciona al usuario para que interactúe con la plataforma de composición. Por medio de esta aplicación web se envía y responden las peticiones de los usuarios.
- *Lógica de Negocio:* este paquete encapsula toda la lógica necesaria para que las aplicaciones de Descubrimiento y Composición de servicios puedan ser invocadas desde la aplicación web que se pone a disposición de los usuarios.
- *Descubrimiento de Servicios:* este paquete contiene todas las funcionalidades que implementan los algoritmos propuestos para el descubrimiento de servicios en los ambientes de computación ubicua.
- *Composición de Servicios:* este paquete contiene todas las funcionalidades que implementan los algoritmos propuestos para la composición de servicios en los ambientes de computación ubicua.

- *Analizador BPEL-Grafos*: este paquete encapsula las funciones que transforman procesos de negocio a su equivalente en grafos. Para el prototipo se ha implementado las funciones de BPEL. Sin embargo, es posible registrar otro tipo de funciones y extender su aplicabilidad a otros modelos como WSDL o WS-CDL.
- *Comparador Lingüístico*: contiene las clases implementan la lógica para calcular la similitud entre dos cadenas. Para el prototipo se ha utilizado los algoritmos *Ngram*, *Synonym* y *Abbreviation*. Sin embargo, otros algoritmos pueden ser incluidos.
- *Motor Consultas OCL*: es una API de Java que permite manipular documentos BPEL (y otros archivos XML relacionados), encontrados en el *Repositorio de Proceso BPEL*, como objetos. Ocultando la serialización y deserialización al sistema. Las consultas son realizadas utilizando un lenguaje de consulta orientado a objetos, OCL.

❖ **Capa de Repositorios:**

- *Repositorio BPEL*: Es un repositorio de procesos de negocio, que soporta la recuperación de archivos BPEL (y otros documentos XML) y provee un poderoso mecanismo de consulta que permite encontrar un proceso con búsquedas en sus propiedades o metadatos relacionados. Este repositorio provee una API (paquete *BPEL Repository*) con el fin de hacer flexible su uso.
- *DELI*: librería que permite a un Servlet Java conocer el contexto de entrega de un dispositivo a través de CCPP o UAProf. El contexto de entrega ayuda a los servidores a identificar en qué clase de entorno se está realizando la petición y con base en esto puede dar una respuesta más adecuada según el dispositivo.
- *WALL: Wireless Abstraction Library* (Librería de Abstracción Móvil) es una librería de tags JSP que permite diseñar páginas Web que, dependiendo de las capacidades del dispositivo, entrega el contenido en formato WML, C-HTML, y XHTML Mobile Profile.
- *WURFL: Wireless Universal Resource FiLe* (Archivo de Recursos Universal Inalámbrico). Es parte del esfuerzo de una comunidad FOSS (Free and Open Source Software, Código Fuente Libre y Abierto) enfocada en el problema de presentar contenido en la amplia variedad de dispositivos móviles. El WURFL es un fichero de configuración XML que contiene información acerca de características y capacidades para una variedad de dispositivos móviles. Dicha información es contribución de desarrolladores de todo el mundo y el WURFL es actualizado de forma frecuente reflejando los nuevos dispositivos móviles que entran en el mercado
- *Repositorio UAProf*: es una especificación basada en el CCPP que se encarga de describir las preferencias de usuario y las capacidades de los dispositivos, con el fin de obtener una experiencia de usuario más personalizada.
- *JDBC*: es una API que permite la ejecución de operaciones sobre bases de datos desde el lenguaje de programación Java, independientemente del sistema operativo

donde se ejecute o de la base de datos a la cual se accede, utilizando el dialecto SQL del modelo de base de datos que se utilice

- *Repositorio de Usuarios*: este paquete representa la implementación del repositorio de usuario realizado sobre el motor de base de datos Postgres.
- ❖ **Capa de Base**: incluye el software básico y API's que permite el funcionamiento del prototipo. Esta capa se compone de los siguientes paquetes:
- *Ubuntu*: es el sistema operativo que soporta el prototipo.
 - *WordNet*: es una enorme base de datos léxica del idioma inglés. Agrupa las palabras en conjuntos de sinónimos llamados 'synsets', proporcionando definiciones cortas y generales, y almacenando las relaciones semánticas entre estos conjuntos de sinónimos. El objetivo de este proyecto es: por un lado producir una combinación de diccionario y tesoro cuyo uso es más intuitivo, y ayudar al análisis automático de textos y a las aplicaciones de inteligencia artificial.
 - *JDK*: el *Java Development Kit* es un entorno de desarrollo integrado (IDE) que permite soportar la implementación de aplicaciones en Java. Consiste en un entorno de ejecución (encontrado en un nivel superior al del sistema operativo) que permite compilar, depurar y ejecutar la herramienta/aplicación escrita en Java.
 - *GlassFish v3Prelude*: es un servidor de aplicaciones open source desarrollado por Sun Microsystems, que implementa las tecnologías definidas en la plataforma Java EE 6 y permite ejecutar aplicaciones que siguen esta especificación.
 - *PostgreSQL*: es un sistema de gestión de bases de datos relacional, utilizado para generar las tablas que contienen la información relacionada con el usuario.
 - *Xerces*: es un Parser XML compuesto por una familia de paquetes software que permiten analizar y manipular archivos XML. La biblioteca implementa una serie de APIs estándar que incluye a DOM, SAX y SAX2.
 - *JWNL*: (Java WordNet Library) es una API que permite acceder al diccionario relacional WordNet. También provee funcionalidades más allá del acceso a datos, tales como el descubrimiento de relaciones y procesamiento morfológico.

A.3 Fase de Implementación

En esta fase se describe la implementación de los prototipos operacionales que integran las Plataformas para el Descubrimiento y Composición de Servicios en Ambientes Ubicuos. Se detallan los modelos de implantación y despliegue del sistema.

A.3.1 Diagrama de Implantación

La Figura A7 muestra el diagrama de implantación del sistema. Se detallan los componentes incluidos en la implementación de referencia y su distribución en los nodos empleados.

En el contenedor Web se despliegan los componentes para el Descubrimiento y Composición de servicios, estos componente son accedido por el usuario a través de la Aplicación Web y la Lógica de negocio. El usuario utiliza un navegador web para poder enviar las peticiones y recibir las respuestas desde la aplicación web.

Los repositorios de modelos y contexto han sido distribuidos en diversos nodos indicando los protocolos que se emplean para realizar las consultas. De igual manera se detallan los componentes (DELI y WALL) que participan como medidores entre el componente de descubrimiento de servicios y los repositorios de dispositivos.

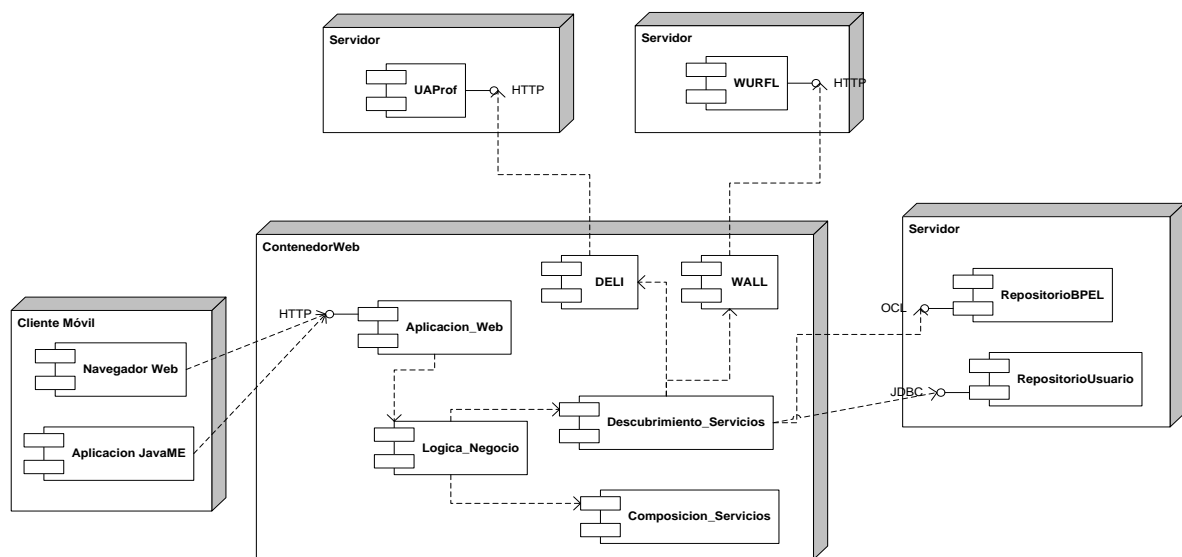


Figura A 7. Modelo de Implantación del Sistema de Descubrimiento y Composición de Servicios

A.3.2 Diagrama de Despliegue

El diagrama de despliegue expuesto en la Figura A8 describe la configuración del sistema de composición para su ejecución en un ambiente del mundo real, presentando la disposición física de los distintos nodos que intervienen en la composición del Sistema, los protocolos de comunicación empleados y la topología hardware de la implementación.

Los clientes móviles acceden a la plataforma mediante el navegador Web, enviando una URL por medio de las cabeceras HTTP, la cual contiene capacidades básicas del dispositivo descritas según el estándar CC/PP. Ésta petición es llevada hasta el *Servidor Web de la Plataforma de Composición*, que identifica las capacidades básicas del dispositivo que está interactuando con la plataforma por medio de las cabeceras HTTP. Sin embargo, dicha información no es suficiente, es aquí donde interviene el *Repositorio de Dispositivos*, el cual determina el contexto de entrega completo del cliente móvil.

El contexto de entrega es construido con base en tres fuentes de información UAProf, WURFL y las cabeceras HTTP, lo que garantiza una descripción amplia y fiel de las capacidades del dispositivo. El orden en el cual el *Repositorio de Dispositivos* obtiene el contexto de entrega es el siguiente: El Servidor inicialmente usa la información que envía el *Servidor Web de Composición* por medio de las cabeceras HTTP, luego los datos ofrecidos por la librería WURFL y por último usa UAProf. Se ha considerado que este es el orden de fidelidad de la información, ya que el navegador debe conocer fielmente sus capacidades, y la librería WURFL ha sido construida por desarrolladores que mediante pruebas obtienen éstos datos; mientras que en el caso de UAProf suele obtenerse información que no detalla el producto, sino una categoría de dispositivos del fabricante.

Una vez el contexto de entrega del cliente móvil ha sido identificado, el usuario es habilitado para usar la plataforma, permitiéndole consultar su historial de servicios (consumidos y consultados) o ejecutar una búsqueda para recuperar nuevos servicios. Esto es posible gracias a los Repositorios de Usuarios y de Procesos del sistema, donde el primero gestiona los *Perfiles de Usuario*, registrando el comportamiento del mismo, servicios consumidos o historial y patrones de uso; y el segundo provee un poderoso mecanismo de consulta que permite obtener información de los procesos BPEL almacenados, mediante búsquedas en sus propiedades o metadatos relacionados.

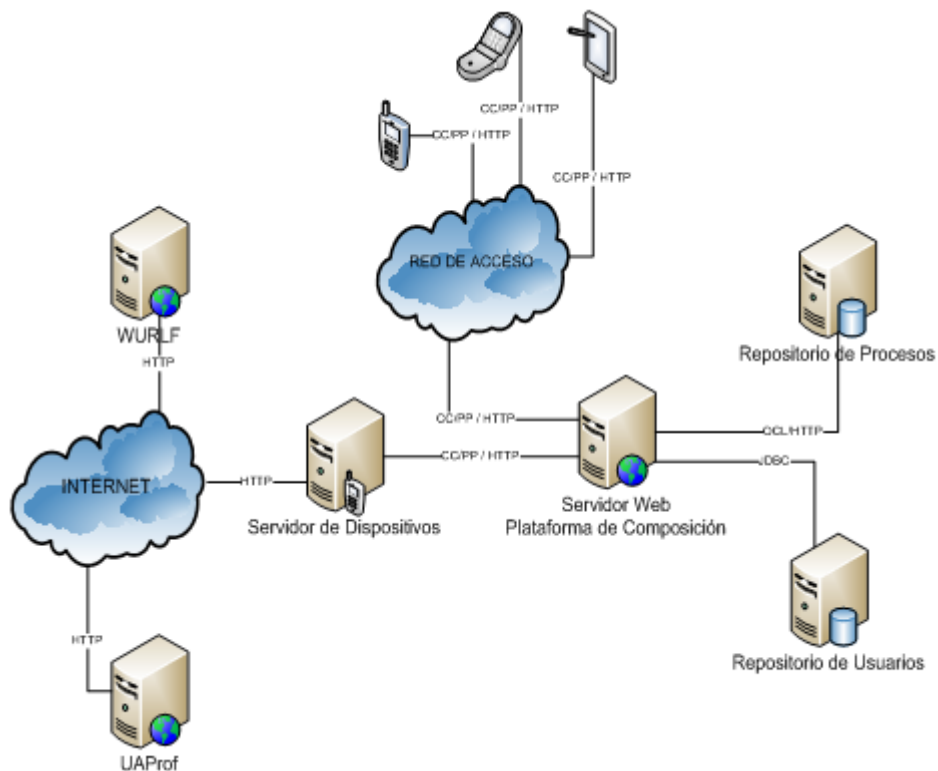


Figura A 8. Modelo de Despliegue del Sistema de Descubrimiento y Composición de Servicios

Protocolos de comunicación y conexión.

Los protocolos son el conjunto de reglas que especifican el intercambio de mensajes durante la comunicación entre las entidades que forman parte de una red. Los utilizados por la plataforma son los siguientes:

- **HTTP** (*HyperText Transfer Protocol*): el Protocolo de transferencia de hipertexto define la sintaxis y la semántica que utilizan los elementos software de la arquitectura web (clientes, servidores) para comunicarse. Es el protocolo de comunicación definido por los repositorios de procesos y de dispositivos (WURLF y UAProf) para realizar las consultas. En cuanto a los dispositivos móviles las cabeceras HTTP permiten identificar las capacidades básicas del dispositivo y realizar posteriores consultas a los repositorios.
- **OCL** (*Object Constraint Language*): este lenguaje permite construir consultas basadas en una representación de objetos e independientes del tipo de almacenamiento empleado en los repositorios (p.ej. bases de datos relacionales o base de datos XML). OCL es definido como el lenguaje de consulta para Repositorio de Procesos.
- **JDBC** (*Java Database Connectivity*): es una API que permite la ejecución de consultas sobre bases de datos relacionales desde el lenguaje de programación Java, independientemente del sistema operativo donde se ejecute o de la base de datos a la cual se accede. Este protocolo es empleado para consultar el repositorio de usuario cuya implementación se soporta sobre una base de datos relacional.

ANEXO B

ANEXO B

TABLAS DE RESULTADOS DE LAS PRUEBAS DE CALIDAD Y RENDIMIENTO DEL SUB-SISTEMA DE DESCUBRIMIENTO

B.1 Recuperación de Actividades Básicas BPEL (ServiceMatch)

B.1.1 Prueba de Calidad de los Resultados PC1

Para el análisis de los resultados se plantean tres escenarios:

Caso 1: evaluación de las medidas de desempeño comparando las actividades de entrada contra las actividades del repositorio que pertenecen al mismo dominio.

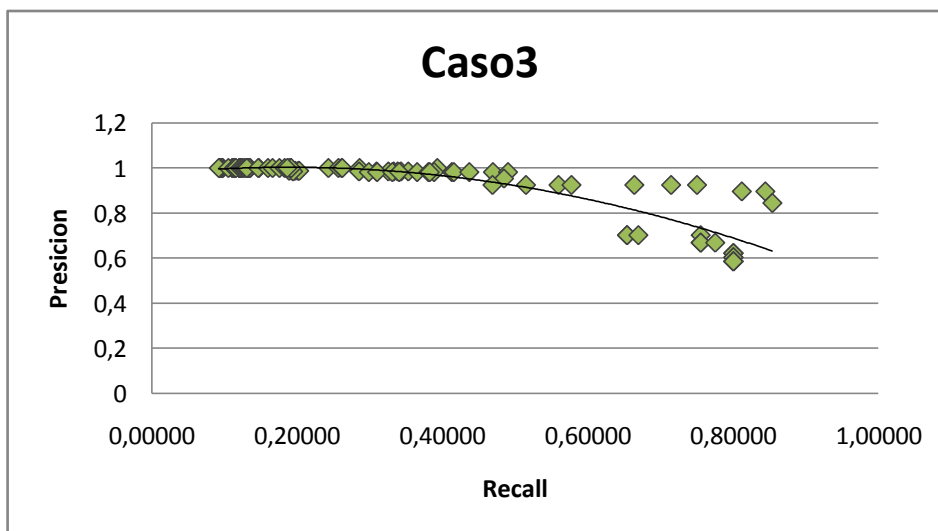
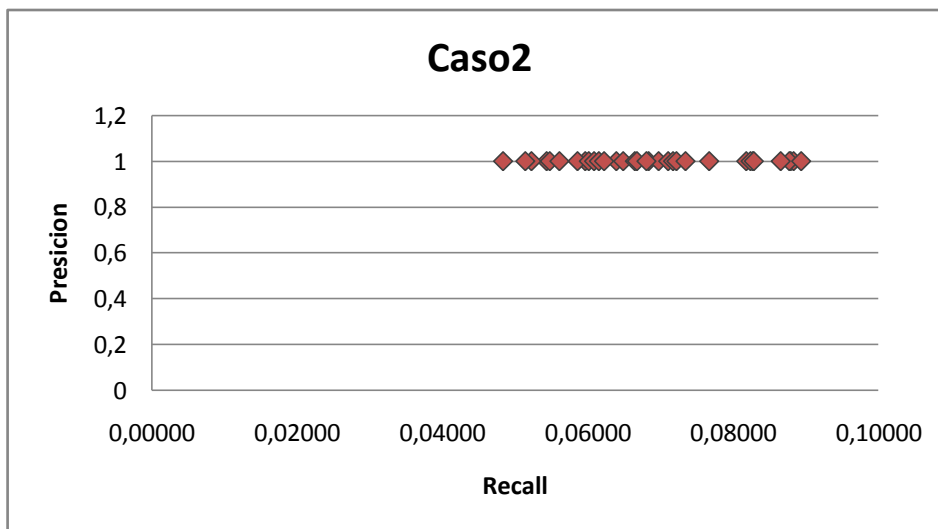
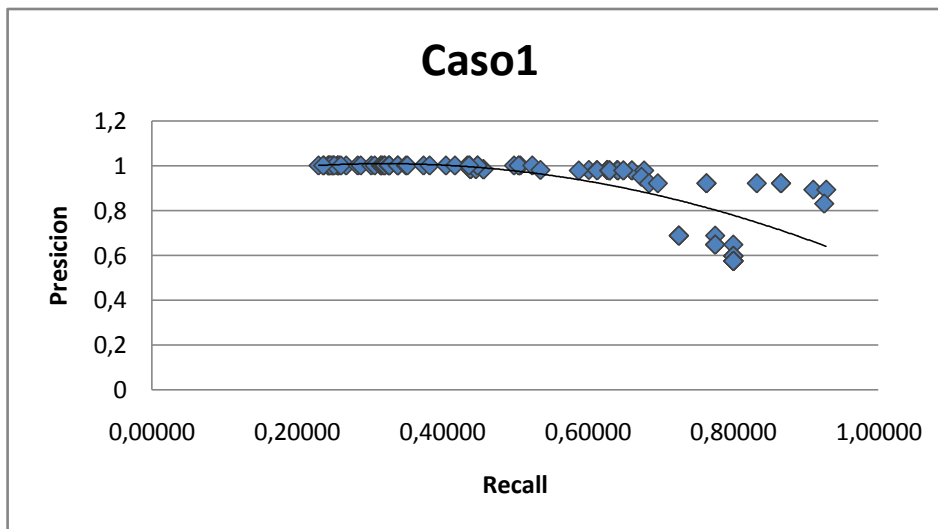
Caso 2: evaluación comparando las actividades de entrada contra las actividades almacenadas que pertenecen a un dominio diferente.

Caso 3: evaluación comparando las actividades de consulta contra todas las actividades contenidas en el repositorio.

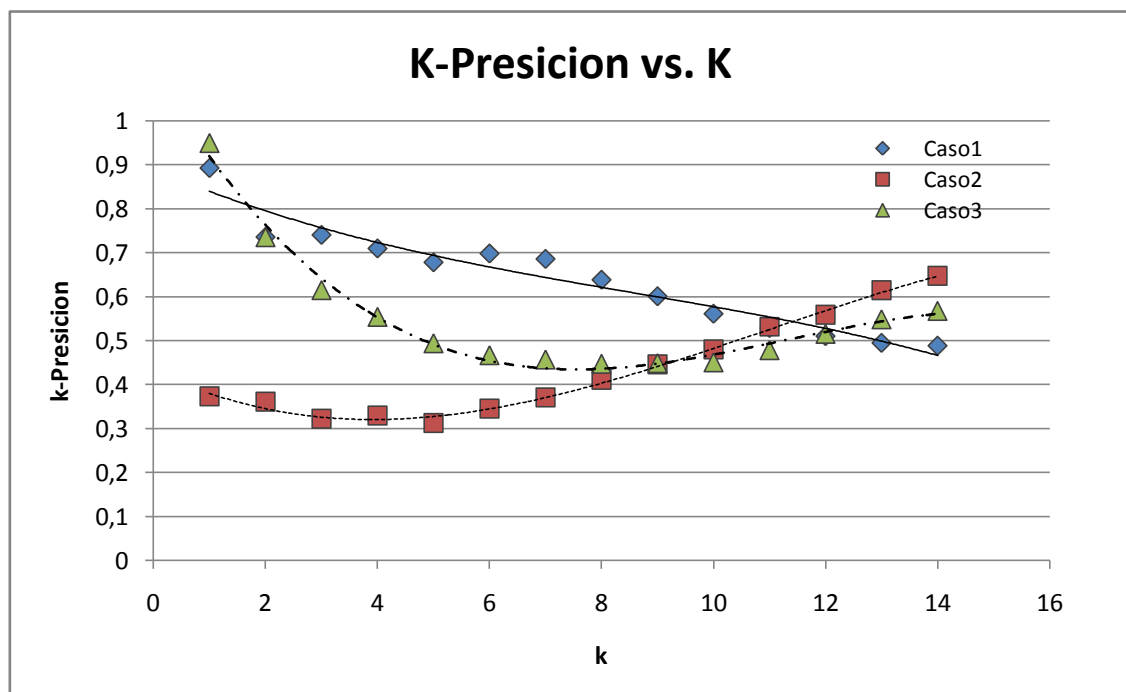
Microaveraging	CASO 1			CASO 2			CASO 3		
Threshhold	Presicion	Recall	Overall	Presicion	Recall	Overall	Presicion	Recall	Overall
0	1	0,43408	0,43408	1	0,04835	0,04835	1	0,39251	0,39251
0,05	1	0,22940	0,22940	1	0,08178	0,08178	1	0,11041	0,11041
0,1	1	0,23658	0,23658	1	0,08268	0,08268	1	0,11332	0,11332
0,15	1	0,24284	0,24284	1	0,08236	0,08236	1	0,11343	0,11343
0,2	1	0,24284	0,24284	1	0,08282	0,08282	1	0,11396	0,11396
0,25	1	0,25568	0,25568	1	0,08788	0,08788	1	0,12024	0,12024
0,3	1	0,25600	0,25600	1	0,08828	0,08828	1	0,12075	0,12075
0,35	1	0,25600	0,25600	1	0,08770	0,08770	1	0,12047	0,12047
0,4	1	0,25827	0,25827	1	0,08651	0,08651	1	0,11975	0,11975
0,45	1	0,26738	0,26738	1	0,08933	0,08933	1	0,12390	0,12390
0,5	1	0,24803	0,24803	1	0,07668	0,07668	1	0,11033	0,11033
0,55	1	0,24980	0,24980	1	0,05964	0,05964	1	0,09730	0,09730
0,6	1	0,24980	0,24980	1	0,05858	0,05858	1	0,09658	0,09658
0,65	1	0,24385	0,24385	1	0,05433	0,05433	1	0,09296	0,09296
0,7	1	0,24452	0,24452	1	0,05476	0,05476	1	0,09394	0,09394
0,75	1	0,24579	0,24579	1	0,05609	0,05609	1	0,09576	0,09576
0,8	1	0,25111	0,25111	1	0,05225	0,05225	1	0,09381	0,09381

0,85	1	0,23578	0,23578	1	0,05140	0,05140	1	0,09175	0,09175
0,9	1	0,26026	0,26026	1	0,06013	0,06013	1	0,10494	0,10494
0,95	1	0,28327	0,28327	1	0,06392	0,06392	1	0,11341	0,11341
1	1	0,28743	0,28743	1	0,06086	0,06086	1	0,11267	0,11267
1,05	1	0,30203	0,30203	1	0,06152	0,06152	1	0,11530	0,11530
1,1	1	0,30663	0,30663	1	0,06486	0,06486	1	0,12090	0,12090
1,15	1	0,31525	0,31525	1	0,06646	0,06646	1	0,12396	0,12396
1,2	1	0,31525	0,31525	1	0,06675	0,06675	1	0,12442	0,12442
1,25	1	0,31735	0,31735	1	0,06835	0,06835	1	0,12698	0,12698
1,3	1	0,31988	0,31988	1	0,06973	0,06973	1	0,12919	0,12919
1,35	1	0,32063	0,32063	1	0,07104	0,07104	1	0,13123	0,13123
1,4	1	0,32597	0,32597	1	0,07172	0,07172	1	0,13246	0,13246
1,45	1	0,32675	0,32675	1	0,07218	0,07218	1	0,13321	0,13321
1,5	1	0,32675	0,32675	1	0,07343	0,07343	1	0,13494	0,13494
1,55	1	0,33848	0,33848	1	0,06806	0,06806	1	0,13375	0,13375
1,6	1	0,33848	0,33848	1	0,06223	0,06223	1	0,13032	0,13032
1,65	1	0,34959	0,34959	-	0,07188	-	1	0,14695	0,14695
1,7	1	0,35157	0,35157	-	0,06635	-	1	0,14603	0,14603
1,75	1	0,37385	0,37385	-	0,07204	-	1	0,15948	0,15948
1,8	1	0,38218	0,38218	-	0,07528	-	1	0,16581	0,16581
1,85	1	0,40461	0,40461	-	0,07780	-	1	0,17519	0,17519
1,9	1	0,41704	0,41704	-	0,08072	-	1	0,18238	0,18238
1,95	1	0,43708	0,43708	-	0,08335	-	1	0,18851	0,18851
2	0,98333334	0,43839	0,42663	-	0,08452	-	0,9875	0,18867	0,18341
2,05	0,98333334	0,43839	0,42663	-	0,08912	-	0,9875	0,19381	0,18826
2,1	0,98333334	0,44585	0,43408	-	0,09097	-	0,9875	0,19708	0,19153
2,15	0,98333334	0,45610	0,44434	-	0,09245	-	0,9875	0,20124	0,19568
2,2	0,98333334	0,45610	0,44434	-	0,09291	-	0,9875	0,20236	0,19680
2,25	0,98181819	0,45645	0,44469	-	0,07589	-	0,98461538	0,19391	0,18820
2,3	1	0,44815	0,44815	-	0,07661	-	1	0,19083	0,19083
2,35	1	0,44815	0,44815	-	0,07706	-	1	0,19140	0,19140
2,4	1	0,43638	0,43638	-	0,07729	-	1	0,18598	0,18598
2,45	1	0,50474	0,50474	-	0,10541	-	1	0,24263	0,24263
2,5	1	0,50704	0,50704	-	0,10887	-	1	0,25641	0,25641
2,55	1	0,49833	0,49833	-	0,11647	-	1	0,26074	0,26074
2,6	1	0,50474	0,50474	-	0,09786	-	1	0,26170	0,26170
2,65	1	0,52346	0,52346	-	0,11043	-	1	0,28544	0,28544
2,7	0,98	0,53466	0,52514	-	0,09089	-	0,98461538	0,28483	0,28120
2,75	0,98	0,60126	0,59073	-	0,09883	-	0,98461538	0,30879	0,30495
2,8	0,98	0,61241	0,60064	-	0,10396	-	0,98461538	0,32514	0,32097
2,85	0,98	0,62652	0,61319	-	0,10507	-	0,98461538	0,33235	0,32801
2,9	0,98	0,62652	0,61319	-	0,10814	-	0,98461538	0,33793	0,33359
2,95	0,98	0,62652	0,61319	-	0,10947	-	0,98461538	0,34252	0,33798
3	0,98	0,64015	0,62681	-	0,10947	-	0,98461538	0,35256	0,34802
3,05	0,97777778	0,58737	0,57309	-	0,07513	-	0,98181819	0,29821	0,29345

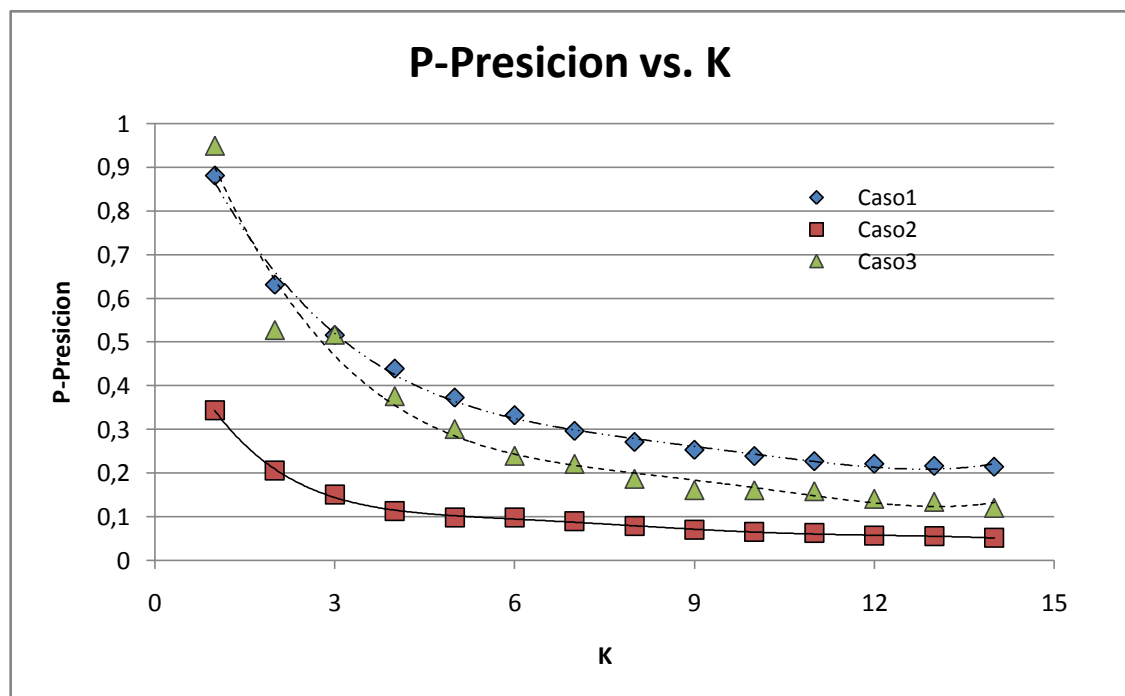
3,1	0,97777778	0,61271	0,59843	-	0,06204	-	0,98181819	0,30966	0,30478
3,15	0,97777778	0,64100	0,62671	-	0,05470	-	0,98181819	0,33060	0,32560
3,2	0,97777778	0,64100	0,62671	-	0,05641	-	0,98181819	0,33996	0,33470
3,25	0,97777778	0,66044	0,64615	-	0,06476	-	0,98181819	0,36472	0,35900
3,3	0,97777778	0,66044	0,64615	-	0,07117	-	0,98181819	0,38801	0,38176
3,35	0,97777778	0,62967	0,61538	-	0,07174	-	0,98181819	0,38028	0,37403
3,4	0,97777778	0,62967	0,61538	-	0,06977	-	0,98181819	0,38209	0,37564
3,45	0,97777778	0,62967	0,61538	-	0,08783	-	0,98181819	0,41247	0,40532
3,5	0,97777778	0,62967	0,61538	-	0,08818	-	0,98181819	0,41526	0,40785
3,55	0,97777778	0,64872	0,63205	-	0,09848	-	0,98181819	0,43655	0,42822
3,6	0,97777778	0,64872	0,63205	-	0,10260	-	0,98181819	0,46906	0,46036
3,65	0,97777778	0,67727	0,66061	-	0,11619	-	0,98181819	0,48980	0,48071
3,7	0,94920635	0,67370	0,62846	-	0,11619	-	0,95324676	0,48424	0,45015
3,75	0,92063493	0,68398	0,62056	-	0,07778	-	0,92467534	0,46842	0,41961
3,8	0,92063493	0,69610	0,63117	-	-	-	0,92467534	0,51451	0,46165
3,85	0,92063493	0,76303	0,69152	-	-	-	0,92467534	0,55925	0,50111
3,9	0,92063493	0,76303	0,69152	-	-	-	0,92467534	0,57712	0,51659
3,95	0,92063493	0,83222	0,75389	-	-	-	0,92467534	0,66364	0,59475
4	0,92063493	0,86556	0,78722	-	-	-	0,92467534	0,71429	0,64444
4,05	0,92063493	0,86556	0,78722	-	-	-	0,92467534	0,74987	0,68003
4,1	0,8920635	0,91000	0,78500	-	-	-	0,89610391	0,81154	0,69115
4,15	0,8920635	0,92778	0,80056	-	-	-	0,89610391	0,84394	0,72227
4,2	0,82984128	0,92500	0,72000	-	-	-	0,84458874	0,85364	0,67227
4,25	0,68698413	0,72500	0,60000	-	-	-	0,70173159	0,65364	0,55227
4,3	0,68698413	0,72500	0,60000	-	-	-	0,70173159	0,65364	0,55227
4,35	0,68698413	0,72500	0,60000	-	-	-	0,70173159	0,66919	0,56783
4,4	0,68698413	0,77500	0,64167	-	-	-	0,70173159	0,75500	0,64167
4,45	0,64698413	0,77500	0,55833	-	-	-	0,66839827	0,75500	0,58167
4,5	0,64698413	0,77500	0,55833	-	-	-	0,66839827	0,75500	0,58167
4,55	0,64698413	0,80000	0,57619	-	-	-	0,66839827	0,77500	0,59722
4,6	0,59619049	0,80000	0,48667	-	-	-	0,62164504	0,80000	0,54500
4,65	0,59619049	0,80000	0,48667	-	-	-	0,62164504	0,80000	0,54500
4,7	0,59619049	0,80000	0,48667	-	-	-	0,62164504	0,80000	0,54500
4,75	0,57396827	0,80000	0,42667	-	-	-	0,60346321	0,80000	0,50571
4,8	0,57396827	0,80000	0,42667	-	-	-	0,60346321	0,80000	0,50571
4,85	0,57396827	0,80000	0,42667	-	-	-	0,5852814	0,80000	0,45333
4,9	0,57396827	0,80000	0,42667	-	-	-	0,5852814	0,80000	0,45333
4,95	0,57396827	0,80000	0,42667	-	-	-	0,5852814	0,80000	0,45333
5	0,57396827	0,80000	0,42667	-	-	-	0,5852814	0,80000	0,45333



K	K-Presicion vs. K		
	Caso1	Caso2	Caso3
1	0,89285715	0,37285715	0,95
2	0,73571429	0,36071429	0,73571429
3	0,7404762	0,32190477	0,61523812
4	0,70964285	0,32857143	0,55428571
5	0,67800001	0,312	0,49428572
6	0,69809522	0,345	0,46714286
7	0,68571429	0,37040817	0,45734695
8	0,6382653	0,41017857	0,4475
9	0,60064627	0,44539684	0,44904763
10	0,56083674	0,47928572	0,45085714
11	0,52826529	0,53103895	0,47857144
12	0,50991033	0,55892856	0,51571428
13	0,49437919	0,61472531	0,54879123
14	0,48809976	0,64693878	0,56785714

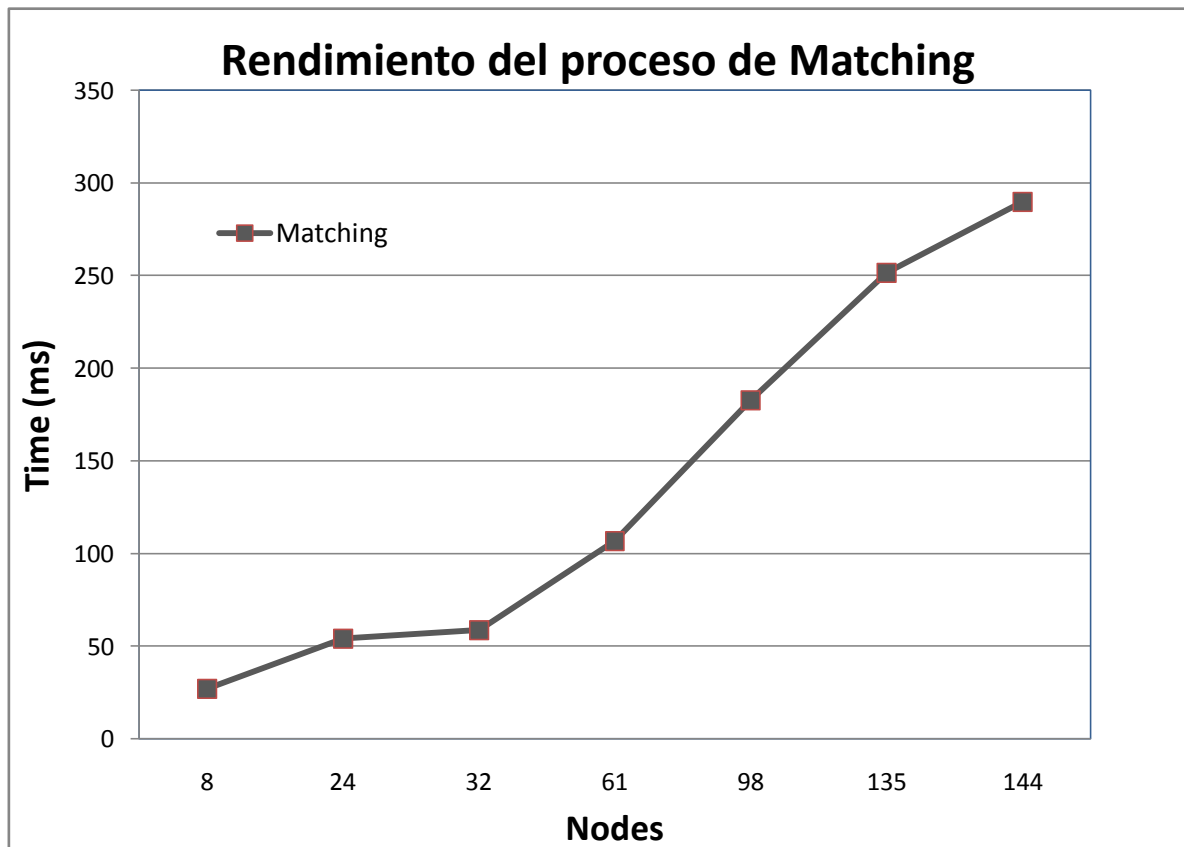


K	P-Presicion vs. K		
	Caso1	Caso2	Caso3
1	0,881428574	0,3442857	0,95
2	0,631428572	0,2064286	0,5278571
3	0,515714296	0,1509524	0,5171429
4	0,438928566	0,1132143	0,3764286
5	0,372571432	0,0985714	0,3011429
6	0,332142864	0,0983333	0,2397619
7	0,296122452	0,09	0,2212245
8	0,270739794	0,07875	0,1869643
9	0,25280329	0,07	0,1611111
10	0,238454082	0,0658571	0,1608571
11	0,226713822	0,0624675	0,1587013
12	0,220869666	0,0572619	0,1411905
13	0,215924612	0,0559341	0,1342857
14	0,213883794	0,0519388	0,1204082



B.1.2 Prueba de Rendimiento PR1

Nodes	Matching time (ms)					Total
	Dominio1	Dominio2	Dominio3	Dominio4	Dominio5	
8	10	10,2	35,6	40,4	39,2	27,08
24	20,8	12,8	71,4	89,6	75,6	54,04
32	22,4	13,6	78	104,6	74,8	58,68
61	49	46,6	160,2	152,8	125	106,72
98	99,2	91,4	236	256,2	231,6	182,88
135	158,8	128,4	353,4	332,6	285	251,64
144	184,4	161	417,4	373,8	312,4	289,8



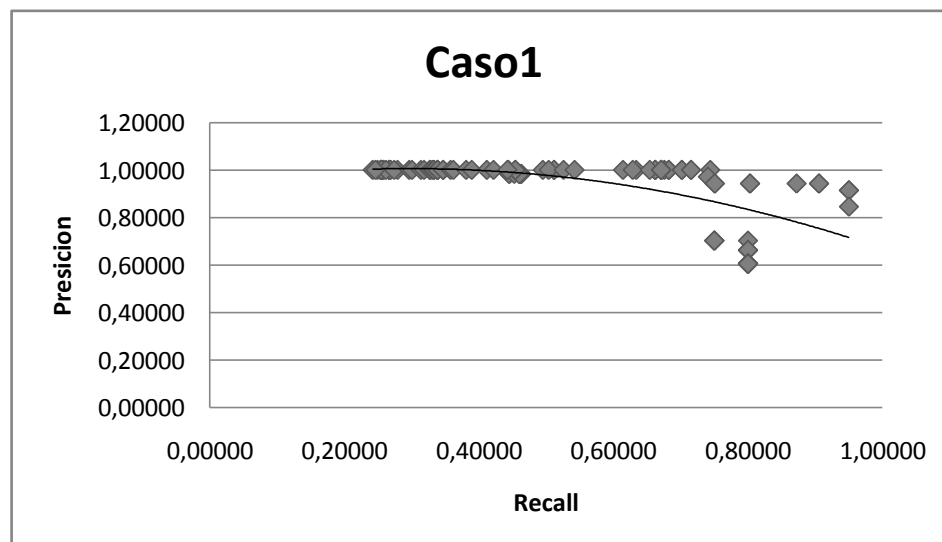
B.2 Recuperación de Actividades Básicas BPEL adaptado al Contexto

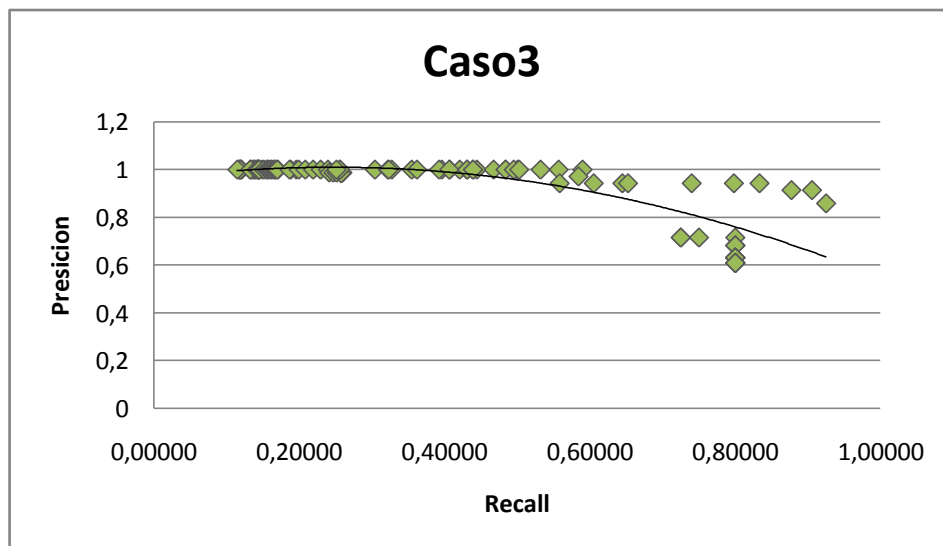
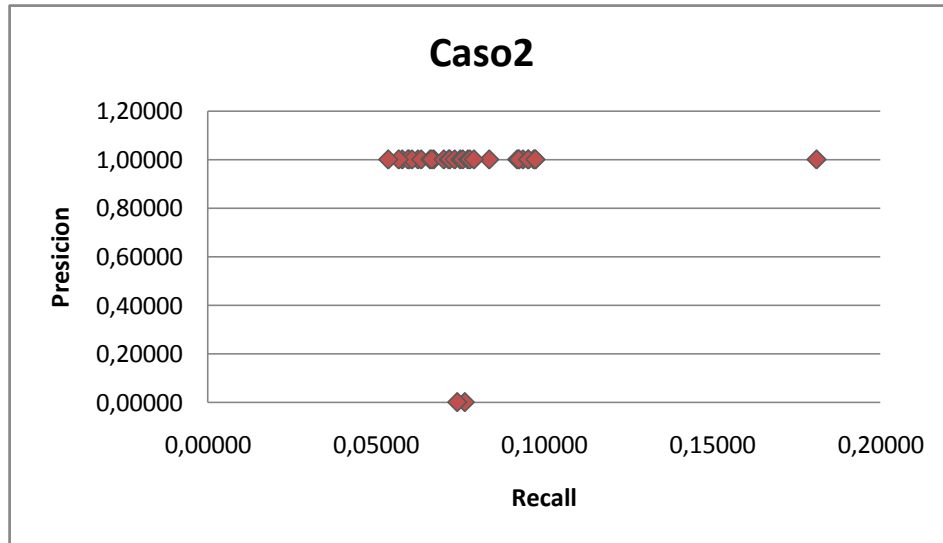
B.2.1 Prueba de Calidad de los Resultados PC2

Microaveraging	CASO 1			CASO 2			CASO 3		
Threreshold	Presicion	Recall	Overall	Presicion	Recall	Overall	Presicion	Recall	Overall
0	1,00000	0,25500	0,25500	1,00000	0,18101	0,18101	1	0,19582	0,19582
0,05	1,00000	0,24217	0,24217	1,00000	0,09248	0,09248	1	0,13487	0,13487
0,1	1,00000	0,24935	0,24935	1,00000	0,09367	0,09367	1	0,13833	0,13833
0,15	1,00000	0,25561	0,25561	1,00000	0,09193	0,09193	1	0,13770	0,13770
0,2	1,00000	0,25561	0,25561	1,00000	0,09239	0,09239	1	0,13821	0,13821
0,25	1,00000	0,26671	0,26671	1,00000	0,09690	0,09690	1	0,14415	0,14415
0,3	1,00000	0,26671	0,26671	1,00000	0,09727	0,09727	1	0,14457	0,14457
0,35	1,00000	0,26671	0,26671	1,00000	0,09522	0,09522	1	0,14350	0,14350
0,4	1,00000	0,26898	0,26898	1,00000	0,09522	0,09522	1	0,14399	0,14399
0,45	1,00000	0,27915	0,27915	1,00000	0,09736	0,09736	1	0,14889	0,14889
0,5	1,00000	0,25561	0,25561	1,00000	0,08363	0,08363	1	0,13283	0,13283
0,55	1,00000	0,25827	0,25827	1,00000	0,06241	0,06241	1	0,11910	0,11910
0,6	1,00000	0,25827	0,25827	1,00000	0,05973	0,05973	1	0,11763	0,11763
0,65	1,00000	0,25410	0,25410	1,00000	0,05773	0,05773	1	0,11634	0,11634
0,7	1,00000	0,25410	0,25410	1,00000	0,05942	0,05942	1	0,11859	0,11859
0,75	1,00000	0,25537	0,25537	1,00000	0,06065	0,06065	1	0,12043	0,12043
0,8	1,00000	0,26192	0,26192	1,00000	0,05668	0,05668	1	0,11906	0,11906
0,85	1,00000	0,24622	0,24622	1,00000	0,05357	0,05357	1	0,11466	0,11466
0,9	1,00000	0,27405	0,27405	1,00000	0,06337	0,06337	1	0,13256	0,13256
0,95	1,00000	0,29646	0,29646	1,00000	0,06708	0,06708	1	0,14120	0,14120
1	1,00000	0,30062	0,30062	1,00000	0,06605	0,06605	1	0,14251	0,14251
1,05	1,00000	0,31394	0,31394	1,00000	0,06651	0,06651	1	0,14474	0,14474
1,1	1,00000	0,31854	0,31854	1,00000	0,07009	0,07009	1	0,15122	0,15122
1,15	1,00000	0,32716	0,32716	1,00000	0,07156	0,07156	1	0,15544	0,15544
1,2	1,00000	0,32716	0,32716	1,00000	0,07185	0,07185	1	0,15590	0,15590
1,25	1,00000	0,33074	0,33074	1,00000	0,07333	0,07333	1	0,15843	0,15843
1,3	1,00000	0,33327	0,33327	1,00000	0,07499	0,07499	1	0,16126	0,16126
1,35	1,00000	0,33327	0,33327	1,00000	0,07573	0,07573	1	0,16235	0,16235
1,4	1,00000	0,33860	0,33860	1,00000	0,07727	0,07727	1	0,16530	0,16530
1,45	1,00000	0,33860	0,33860	1,00000	0,07786	0,07786	1	0,16616	0,16616
1,5	1,00000	0,33860	0,33860	1,00000	0,07908	0,07908	1	0,16771	0,16771
1,55	1,00000	0,34685	0,34685	-	0,07634	-	1	0,17040	0,17040
1,6	1,00000	0,34685	0,34685	-	0,07409	-	1	0,16961	0,16961
1,65	1,00000	0,35796	0,35796	-	0,08489	-	1	0,18804	0,18804
1,7	1,00000	0,36211	0,36211	-	0,07881	-	1	0,18660	0,18660

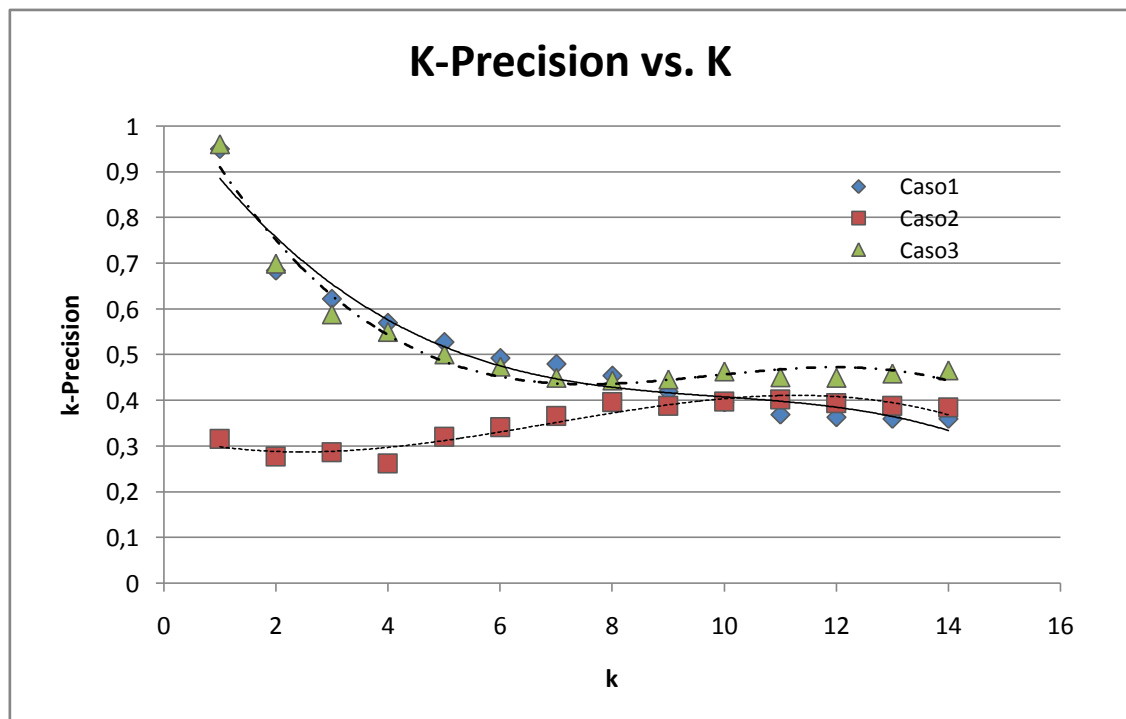
1,75	1,00000	0,38107	0,38107	-	0,08388	-	1	0,19912	0,19912
1,8	1,00000	0,38940	0,38940	-	0,08995	-	1	0,20822	0,20822
1,85	1,00000	0,41183	0,41183	-	0,09286	-	1	0,21901	0,21901
1,9	1,00000	0,42176	0,42176	-	0,09873	-	1	0,22937	0,22937
1,95	1,00000	0,44389	0,44389	-	0,10211	-	1	0,23940	0,23940
2	0,98333	0,44521	0,43344	-	0,10373	-	0,9875	0,24124	0,23597
2,05	0,98333	0,44521	0,43344	-	0,10884	-	0,9875	0,24688	0,24132
2,1	0,98333	0,45266	0,44090	-	0,11168	-	0,9875	0,25144	0,24589
2,15	0,98333	0,46292	0,45115	-	0,11411	-	0,9875	0,25807	0,25252
2,2	0,98333	0,46292	0,45115	-	0,11411	-	0,9875	0,25929	0,25374
2,25	0,98182	0,46018	0,44841	-	0,10079	-	0,98461538	0,25756	0,25184
2,3	1,00000	0,45421	0,45421	-	0,10165	-	1	0,25588	0,25588
2,35	1,00000	0,45421	0,45421	-	0,10165	-	1	0,25588	0,25588
2,4	1,00000	0,44244	0,44244	-	0,10244	-	1	0,25111	0,25111
2,45	1,00000	0,51175	0,51175	-	0,13004	-	1	0,30411	0,30411
2,5	1,00000	0,51175	0,51175	-	0,13989	-	1	0,32192	0,32192
2,55	1,00000	0,49509	0,49509	-	0,14954	-	1	0,32722	0,32722
2,6	1,00000	0,50376	0,50376	-	0,11939	-	1	0,32317	0,32317
2,65	1,00000	0,52588	0,52588	-	0,13553	-	1	0,35466	0,35466
2,7	1,00000	0,54228	0,54228	-	0,12039	-	1	0,36190	0,36190
2,75	1,00000	0,61422	0,61422	-	0,13039	-	1	0,39606	0,39606
2,8	1,00000	0,63380	0,63380	-	0,13039	-	1	0,40690	0,40690
2,85	1,00000	0,66208	0,66208	-	0,13333	-	1	0,42072	0,42072
2,9	1,00000	0,66208	0,66208	-	0,14000	-	1	0,43085	0,43085
2,95	1,00000	0,66208	0,66208	-	0,14000	-	1	0,43085	0,43085
3	1,00000	0,67570	0,67570	-	0,14000	-	1	0,44433	0,44433
3,05	1,00000	0,62864	0,62864	-	0,10589	-	1	0,39244	0,39244
3,1	1,00000	0,65398	0,65398	-	0,08142	-	1	0,40643	0,40643
3,15	1,00000	0,68226	0,68226	-	0,08636	-	1	0,43902	0,43902
3,2	1,00000	0,68226	0,68226	-	0,08636	-	1	0,43902	0,43902
3,25	1,00000	0,70171	0,70171	-	0,10111	-	1	0,46740	0,46740
3,3	1,00000	0,70171	0,70171	-	0,11977	-	1	0,50054	0,50054
3,35	1,00000	0,67094	0,67094	-	0,11977	-	1	0,48387	0,48387
3,4	1,00000	0,67094	0,67094	-	0,11977	-	1	0,48387	0,48387
3,45	1,00000	0,67094	0,67094	-	0,12937	-	1	0,49508	0,49508
3,5	1,00000	0,67094	0,67094	-	0,13889	-	1	0,50258	0,50258
3,55	1,00000	0,71538	0,71538	-	0,14603	-	1	0,53202	0,53202
3,6	1,00000	0,71538	0,71538	-	0,14881	-	1	0,55687	0,55687
3,65	1,00000	0,74394	0,74394	-	0,17167	-	1	0,58977	0,58977
3,7	0,97143	0,74037	0,71180	-	0,17167	-	0,97142857	0,58421	0,55921
3,75	0,94286	0,75065	0,70390	-	0,11333	-	0,94285715	0,55830	0,51791
3,8	0,94286	0,75065	0,70390	-	-	-	0,94285715	0,60518	0,56122
3,85	0,94286	0,80303	0,75152	-	-	-	0,94285715	0,64466	0,59594
3,9	0,94286	0,80303	0,75152	-	-	-	0,94285715	0,65235	0,60235
3,95	0,94286	0,87222	0,81389	-	-	-	0,94285715	0,74000	0,68444

4	0,94286	0,90556	0,84722	-	-	-	0,94285715	0,79810	0,74254
4,05	0,94286	0,90556	0,84722	-	-	-	0,94285715	0,83333	0,77778
4,1	0,91429	0,95000	0,84500	-	-	-	0,91428572	0,87727	0,77227
4,15	0,91429	0,95000	0,84500	-	-	-	0,91428572	0,90556	0,80056
4,2	0,84571	0,95000	0,76167	-	-	-	0,85873016	0,92500	0,75500
4,25	0,70286	0,75000	0,64167	-	-	-	0,71587301	0,72500	0,63500
4,3	0,70286	0,75000	0,64167	-	-	-	0,71587301	0,72500	0,63500
4,35	0,70286	0,75000	0,64167	-	-	-	0,71587301	0,75000	0,66000
4,4	0,70286	0,80000	0,68333	-	-	-	0,71587301	0,80000	0,70167
4,45	0,66286	0,80000	0,60000	-	-	-	0,68253969	0,80000	0,64167
4,5	0,66286	0,80000	0,60000	-	-	-	0,68253969	0,80000	0,64167
4,55	0,66286	0,80000	0,60000	-	-	-	0,68253969	0,80000	0,64167
4,6	0,60571	0,80000	0,50667	-	-	-	0,63174605	0,80000	0,56286
4,65	0,60571	0,80000	0,50667	-	-	-	0,63174605	0,80000	0,56286
4,7	0,60571	0,80000	0,50667	-	-	-	0,63174605	0,80000	0,56286
4,75	0,60571	0,80000	0,50667	-	-	-	0,63174605	0,80000	0,56286
4,8	0,60571	0,80000	0,50667	-	-	-	0,63174605	0,80000	0,56286
4,85	0,60571	0,80000	0,50667	-	-	-	0,60952383	0,80000	0,52000
4,9	0,60571	0,80000	0,50667	-	-	-	0,60952383	0,80000	0,52000
4,95	0,60571	0,80000	0,50667	-	-	-	0,60952383	0,80000	0,52000
5	0,60571	0,80000	0,50667	-	-	-	0,60952383	0,80000	0,52000

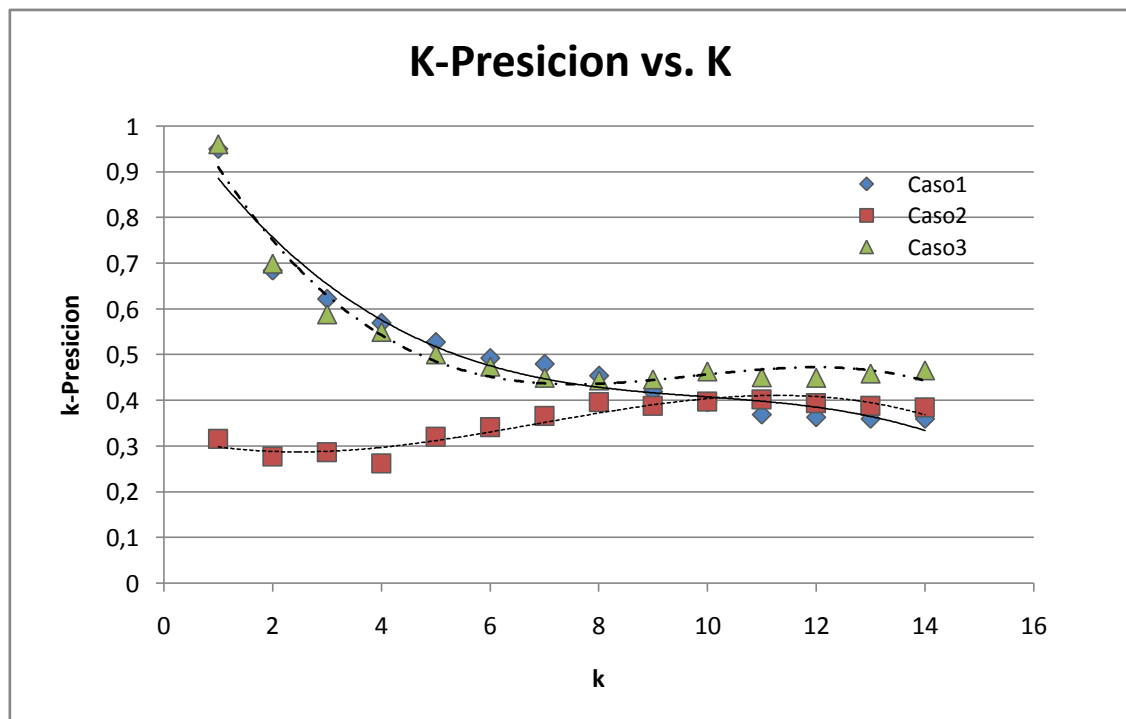




K	K-Precision		
	Caso1	Caso2	Caso3
1	0,95	0,31571429	0,96
2	0,68428572	0,27714286	0,69857143
3	0,62238096	0,28571429	0,58857145
4	0,56928571	0,26142857	0,54964285
5	0,52742857	0,32	0,50085715
6	0,49285714	0,34047619	0,47333333
7	0,47959185	0,36571429	0,44959184
8	0,45423469	0,39535714	0,44392857
9	0,4192347	0,38746032	0,44619048
10	0,39523469	0,39628571	0,46342857
11	0,3683256	0,40194805	0,45025974
12	0,36347713	0,39345237	0,44916666
13	0,35937456	0,38780221	0,45835165
14	0,35937456	0,38367348	0,46561224

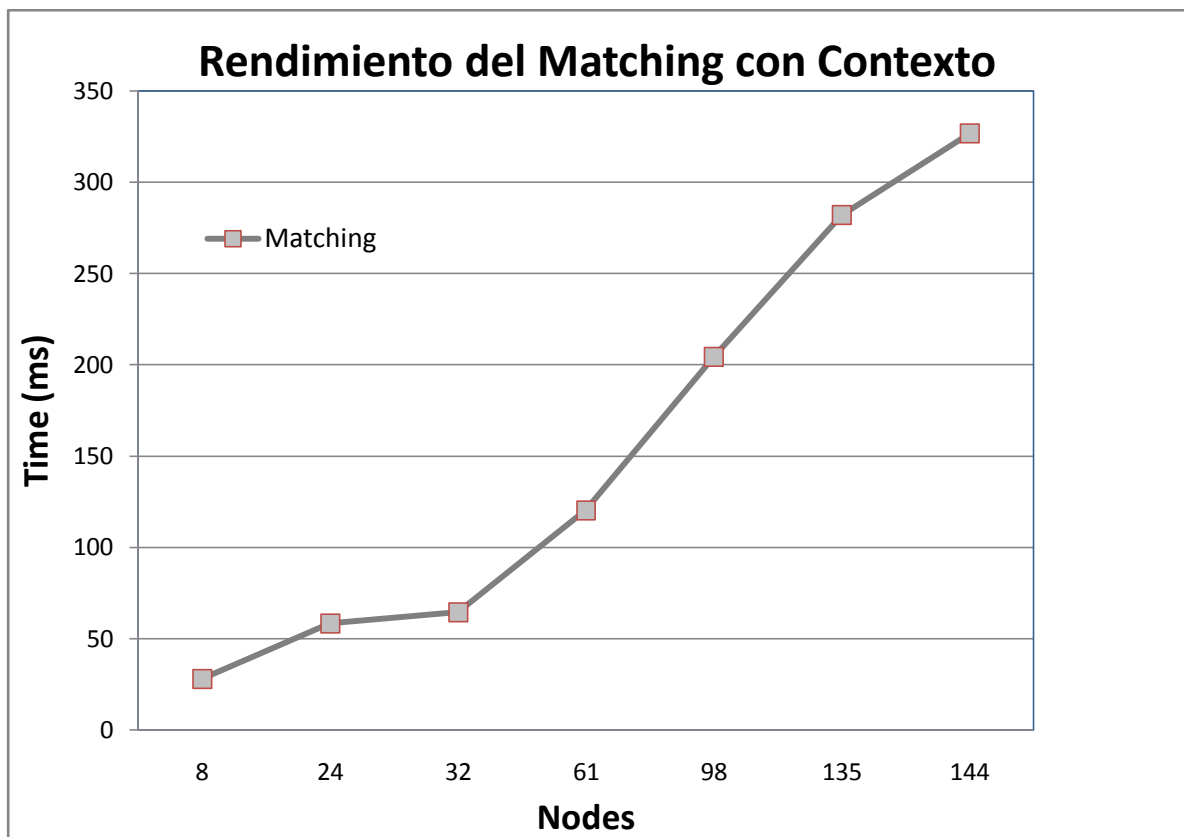


K	P-Precision		
	Caso1	Caso2	Caso3
1	0,91	0,3157143	0,96
2	0,59428572	0,1778571	0,5535714
3	0,466190486	0,1414286	0,462381
4	0,38214286	0,1132143	0,3464286
5	0,313714286	0,0985714	0,2634286
6	0,27476191	0,0971429	0,2711905
7	0,241224494	0,0832653	0,242449
8	0,22015306	0,0728571	0,1871429
9	0,203486396	0,0647619	0,1819048
10	0,19015306	0,0622857	0,1757143
11	0,17924397	0,0566234	0,1668831
12	0,174092454	0,0519048	0,1488095
13	0,16973348	0,050989	0,1327473
14	0,16973348	0,0473469	0,1182653



B.2.2 Prueba de rendimiento PR2:

Nodes	Matching time (ms)					Total
	Dominio1	Dominio1	Dominio1	Dominio1	Dominio1	
8	6,4	11,6	39	42,6	39,8	27,88
24	22	12,2	76,4	97,6	83	58,24
32	22,4	12,6	87	112,6	87,8	64,48
61	50,2	50,2	191,2	167	142,4	120,2
98	108,8	103,2	259	290,8	259,6	204,28
135	156,6	145,2	400,4	382,2	325,8	282,04
144	192,6	180,8	492	417,2	351	326,72



ANEXO C

ANEXO C

TABLAS DE RESULTADOS DE LAS PRUEBAS DE RENDIMIENTO DEL SUB-SISTEMA DE COMPOSICIÓN

C.1 Pruebas de Rendimiento

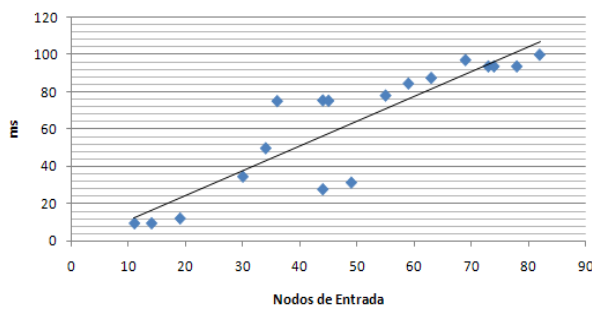
C.1.1 Prueba de Rendimiento PR3

Determina el tiempo que tarda el módulo de composición en entregar un resultado ante un proceso de consulta entrante. Se varía el número de actividades de consulta y se mantiene constante el número de actividades recuperadas.

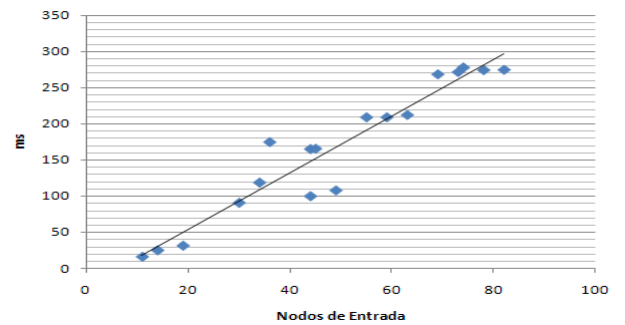
PROCESO DE CONSULTA			Tiempo Promedio de Composición (ms)					
No. Actividades	No. Datos	Total Nodos Híper-grafo	K = 3	K = 4	K = 5	K = 6	K = 7	K = 8
3	8	11	9,6	9	15,6	16	15,6	31,2
4	10	14	9,6	9,4	18,8	25	15,8	40,6
5	14	19	12,2	12,6	31,4	31,4	22	46,6
6	30	36	75	84,4	140,6	175	175	262,6
7	27	34	49,8	74,6	106,2	118,8	146,6	231,4
8	36	44	75,4	90,6	131,4	165,4	187,6	322
9	36	45	75,2	81,2	125	165,8	175	293,8
10	20	30	34,6	37,4	43,6	90,8	93,8	149,8
11	44	55	78	97	140,8	209,4	219	312,4
12	47	59	84,4	102,6	141	209,4	228	325
13	50	63	87,4	109,4	141	212,6	234,2	324,8
14	30	44	27,8	43,8	46,8	100	100	153,2

15	34	49	31,4	31,4	46,8	108	100	156,2
16	53	69	97	125	171,8	269	293,6	359,4
17	56	73	93,6	134,4	181,4	272,2	297	359,2
18	56	74	93,6	128,2	171,6	278,4	296	359,4
19	59	78	93,6	134,6	171,6	274,8	300	368,6
20	62	82	99,8	134,2	178	275,2	309,4	371,8

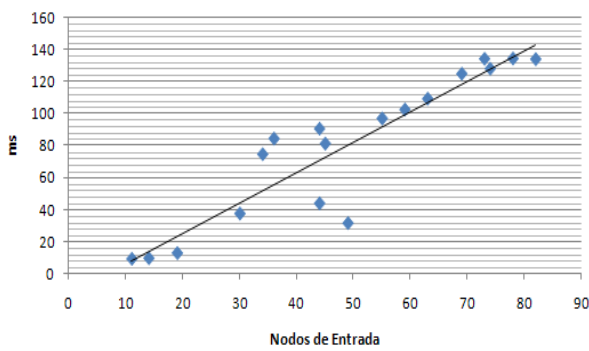
3 Actividades Recuperadas



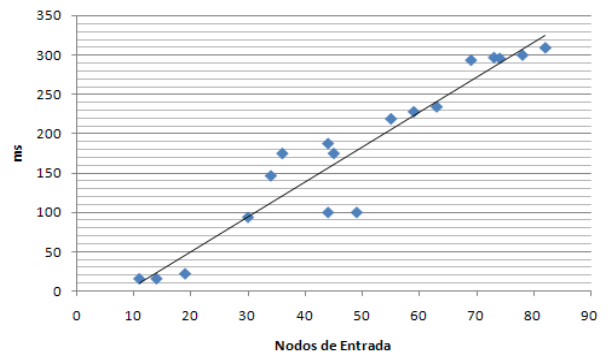
6 Actividades Recuperadas



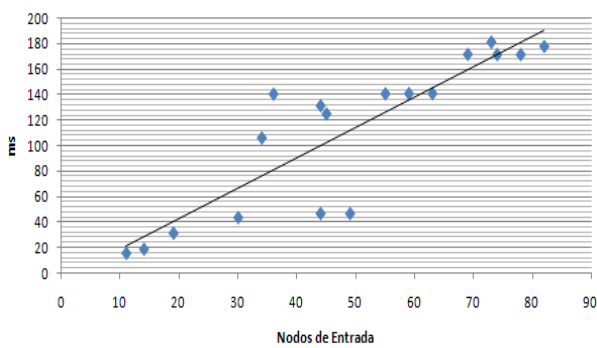
4 Actividades Recuperadas



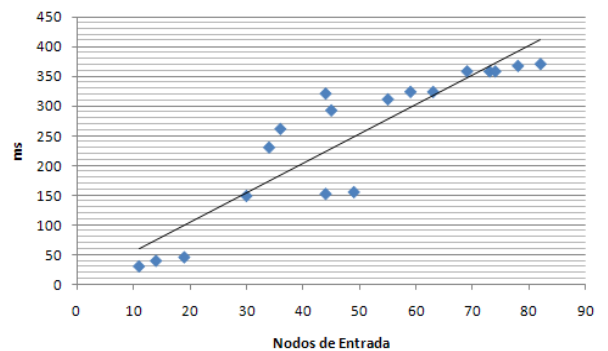
7 Actividades Recuperadas



5 Actividades Recuperadas

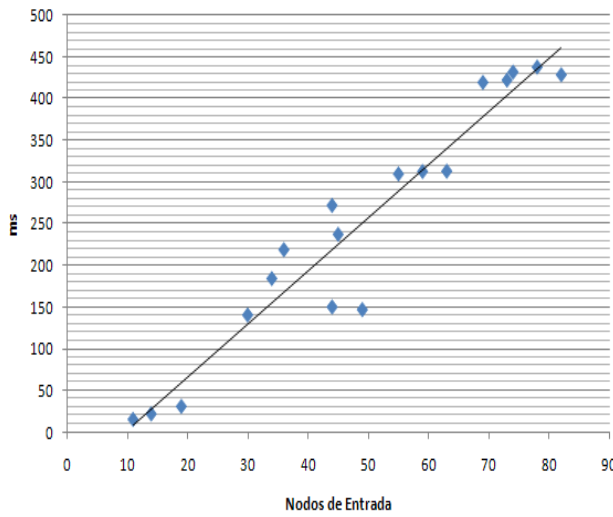


8 Actividades Recuperadas

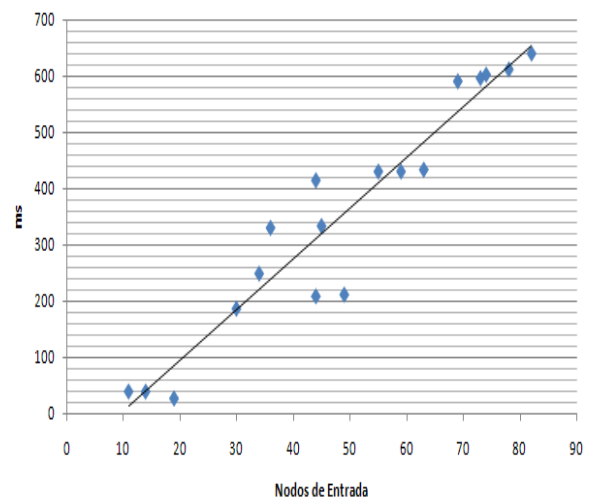


PROCESO DE CONSULTA			Tiempo Promedio de Composición (ms)					
No. Actividades	No. Datos	Total Nodos Híper-grafo	K = 9	K = 10	K = 11	K = 12	K = 13	K = 14
3	8	11	15,6	21,6	25,2	40,6	40,6	40,6
4	10	14	22,2	24,8	41	40,4	40,6	40,6
5	14	19	31,2	31	41	28,2	28,4	31,4
6	30	36	218,8	250	350	331,2	387,6	446,8
7	27	34	184,2	212,2	297	250	274,6	296
8	36	44	271,8	324,8	456,2	415,6	459,6	487,6
9	36	45	237,2	268,6	378,2	334,4	359,2	397
10	20	30	140,6	265	250	187,6	203	222
11	44	55	309,4	352,2	481	431,2	465,8	506,2
12	47	59	312,4	353,2	484,4	431,2	465,6	515,8
13	50	63	312,8	356,4	506,2	434,4	472	515,8
14	30	44	150,2	168,4	265,4	209,6	218,6	237,8
15	34	49	147	175,2	278	212,6	222,2	246,8
16	53	69	419	484,2	543,6	591,4	640,6	700
17	56	73	421,8	490,6	537,6	597	653,2	693,6
18	56	74	431,4	497	550	603,2	665,6	712,4
19	59	78	437,4	496,8	552,8	612,4	665,6	718,8
20	62	82	428,2	506	562,4	640,8	668,8	725

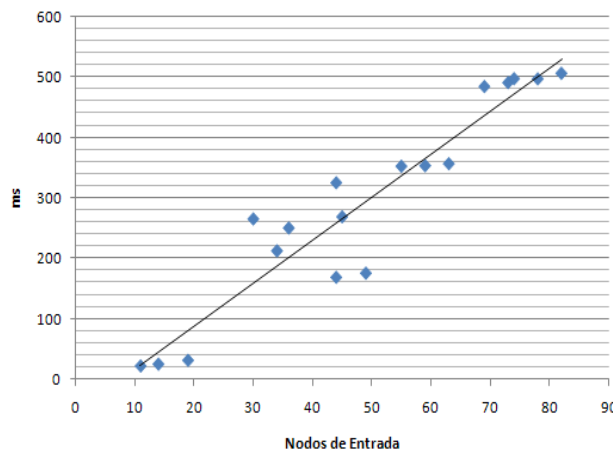
9 Actividades Recuperadas



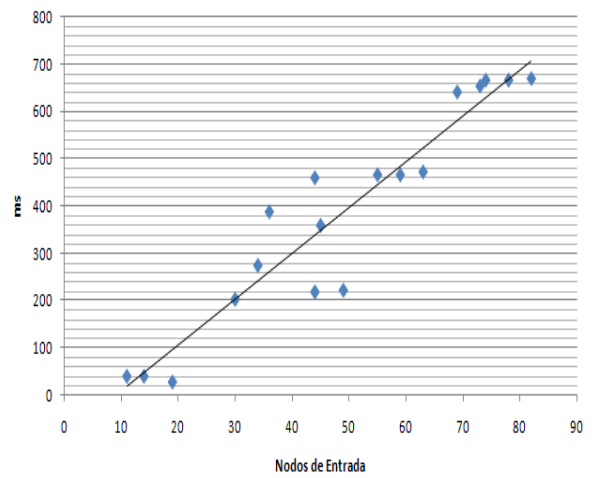
12 Actividades Recuperadas



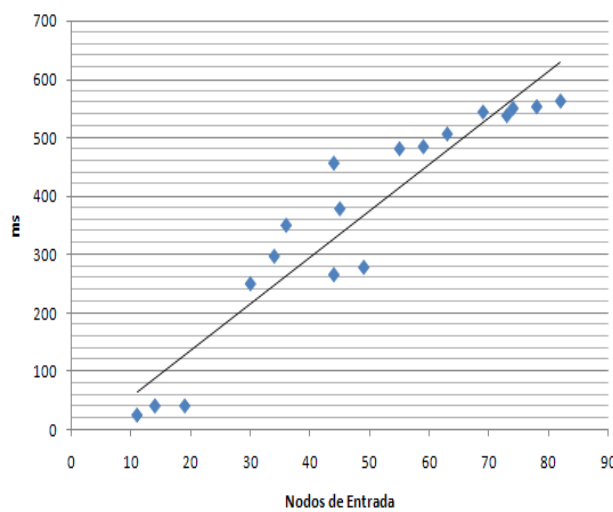
10 Actividades Recuperadas



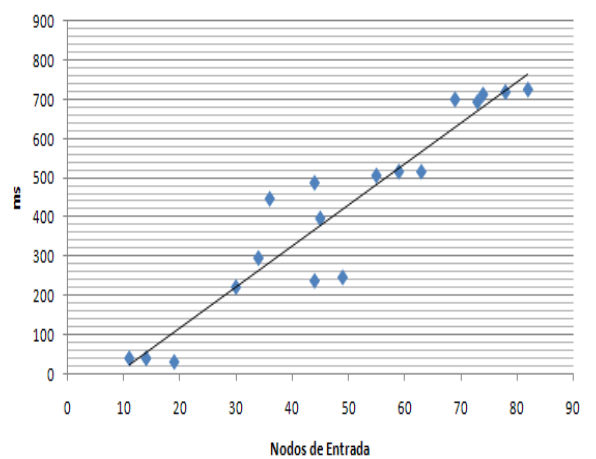
13 Actividades Recuperadas



11 Actividades Recuperadas



14 Actividades Recuperadas

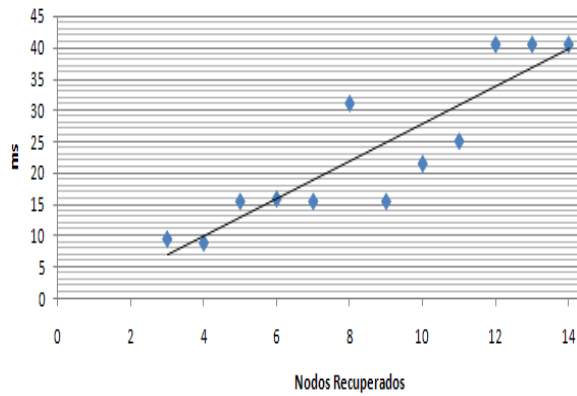


C.1.2 Prueba de Rendimiento PR4

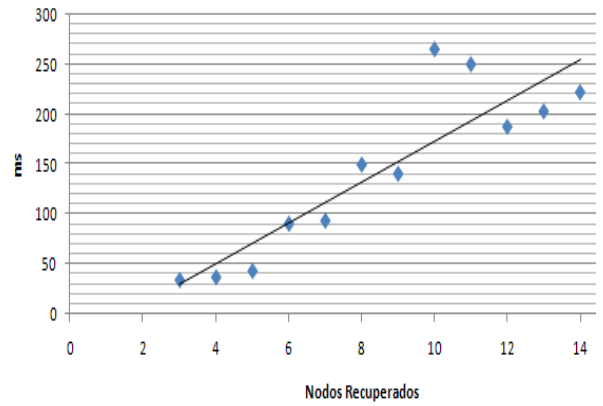
Determina el tiempo que tarda el módulo de composición en entregar un resultado ante un proceso de consulta entrante variando número de actividades recuperadas, se mantiene constante la cantidad de actividades que conforman el proceso de consulta.

Actividades Recuperadas K	Tiempos Promedio de Composición (ms)					
	Híper-Nodos = 11	Híper-Nodos = 14	Híper-Nodos = 19	Híper-Nodos = 30	Híper-Nodos = 34	Híper-Nodos = 36
3	9,6	9,6	12,2	34,6	49,8	75
4	9	9,4	12,6	37,4	74,6	84,4
5	15,6	18,8	31,4	43,6	106,2	140,6
6	16	25	31,4	90,8	118,8	175
7	15,6	15,8	22	93,8	146,6	175
8	31,2	40,6	46,6	149,8	231,4	262,6
9	15,6	22,2	31,2	140,6	184,2	218,8
10	21,6	24,8	31	265	212,2	250
11	25,2	41	41	250	297	350
12	40,6	40,4	28,2	187,6	250	331,2
13	40,6	40,6	28,4	203	274,6	387,6
14	40,6	40,6	31,4	222	296	446,8

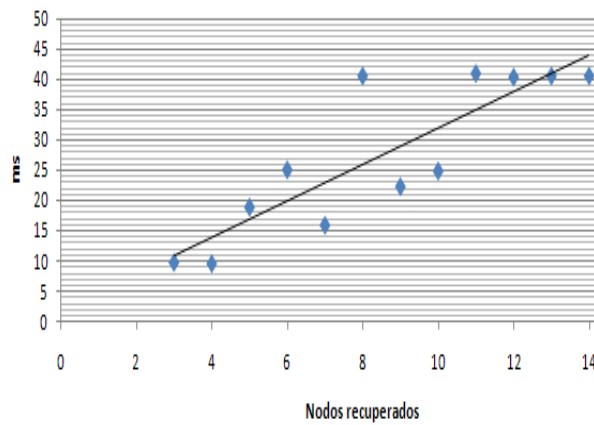
Nodos Hypergrafo= 11



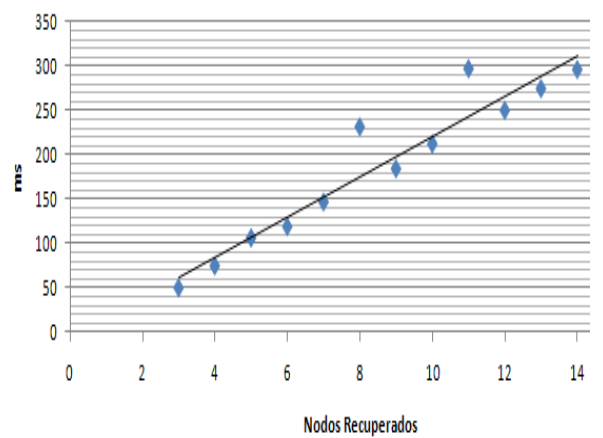
Nodos Hypergrafo 30



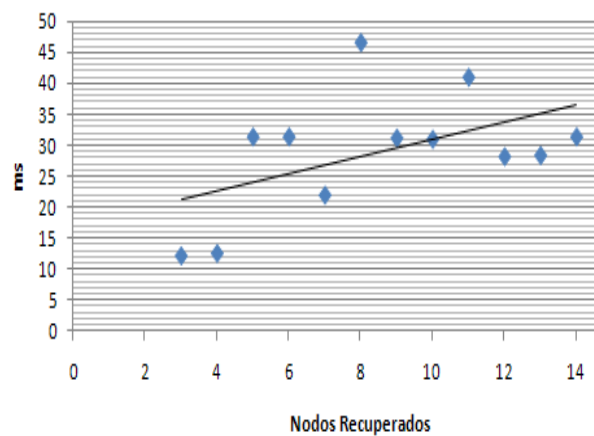
Nodos Hypergrafo 14



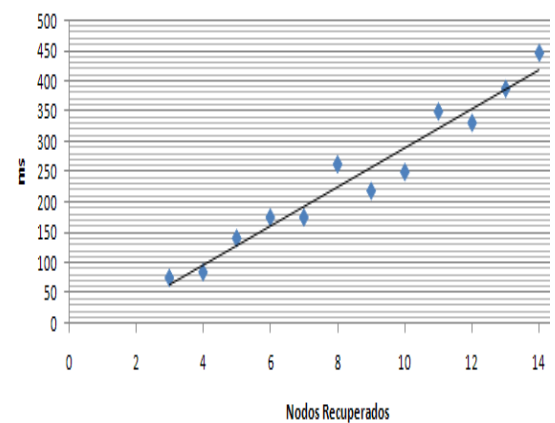
Nodos Hypergrafo 34



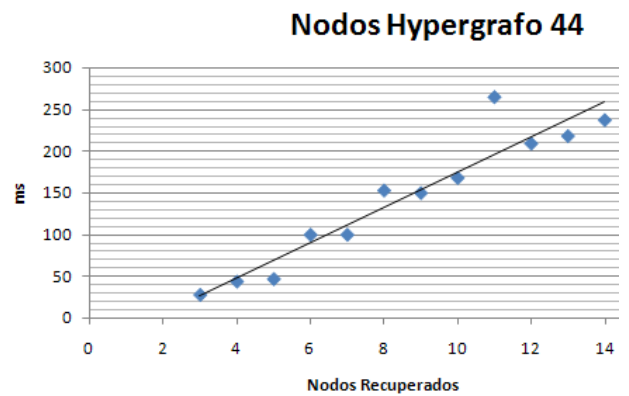
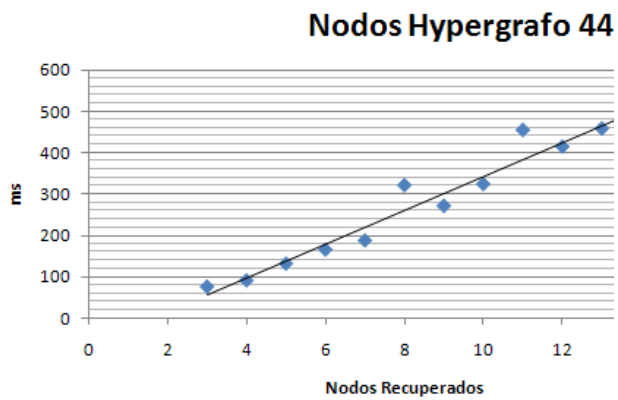
Nodos Hypergrafo 19

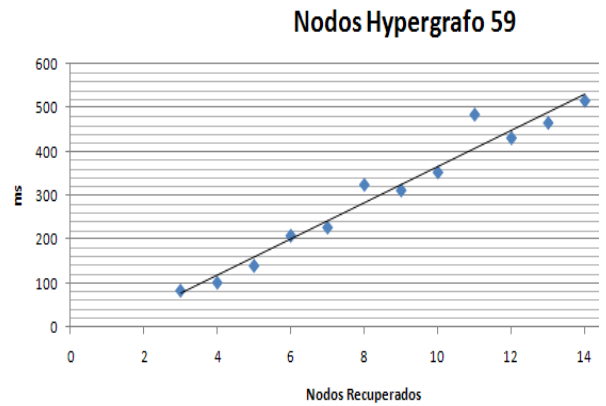
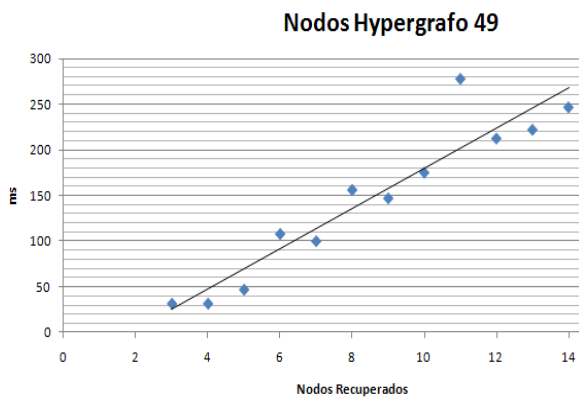
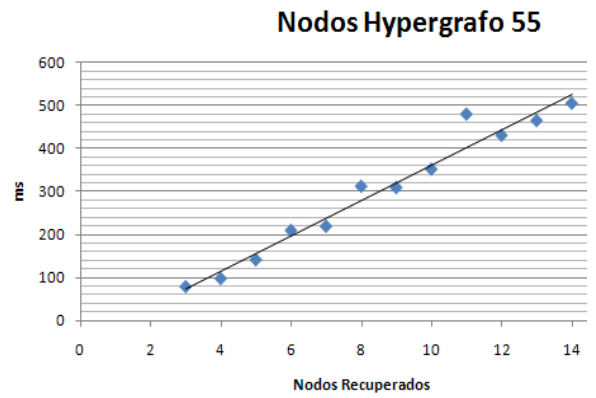
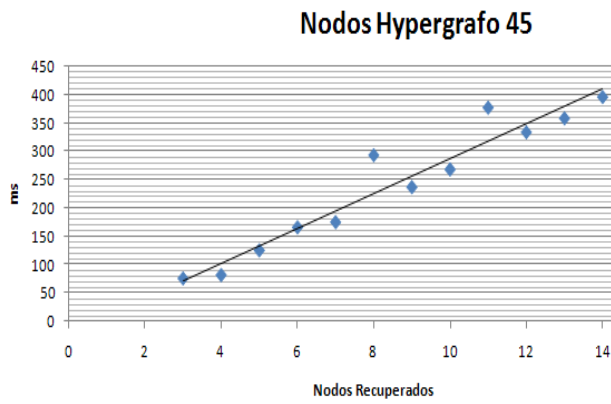


Nodos Hypergrafo 36



Actividades Recuperadas K	Tiempos Promedio de Composición (ms)					
	Híper-Nodos = 44	Híper-Nodos = 44	Híper-Nodos = 45	Híper-Nodos = 49	Híper-Nodos = 55	Híper-Nodos = 59
3	75,4	27,8	75,2	31,4	78	84,4
4	90,6	43,8	81,2	31,4	97	102,6
5	131,4	46,8	125	46,8	140,8	141
6	165,4	100	165,8	108	209,4	209,4
7	187,6	100	175	100	219	228
8	322	153,2	293,8	156,2	312,4	325
9	271,8	150,2	237,2	147	309,4	312,4
10	324,8	168,4	268,6	175,2	352,2	353,2
11	456,2	265,4	378,2	278	481	484,4
12	415,6	209,6	334,4	212,6	431,2	431,2
13	459,6	218,6	359,2	222,2	465,8	465,6
14	487,6	237,8	397	246,8	506,2	515,8

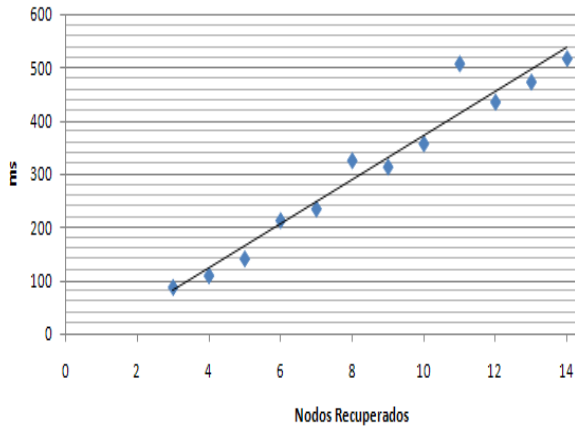




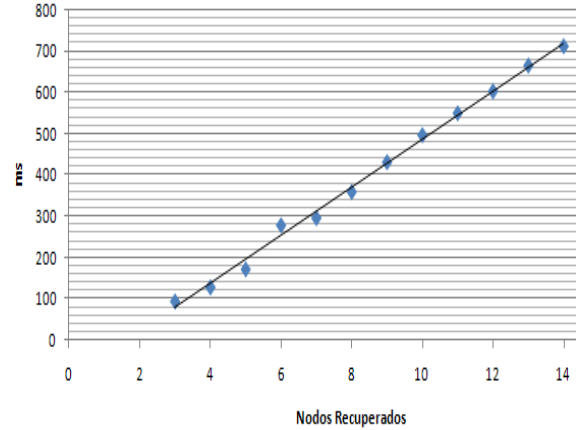
Actividades Recuperadas K	Tiempos Promedio de Composición (ms)					
	Híper-Nodos = 63	Híper-Nodos = 69	Híper-Nodos = 73	Híper-Nodos = 74	Híper-Nodos = 78	Híper-Nodos = 82
3	87,4	97	93,6	93,6	93,6	99,8
4	109,4	125	134,4	128,2	134,6	134,2
5	141	171,8	181,4	171,6	171,6	178
6	212,6	269	272,2	278,4	274,8	275,2
7	234,2	293,6	297	296	300	309,4
8	324,8	359,4	359,2	359,4	368,6	371,8
9	312,8	419	421,8	431,4	437,4	428,2
10	356,4	484,2	490,6	497	496,8	506
11	506,2	543,6	537,6	550	552,8	562,4
12	434,4	591,4	597	603,2	612,4	640,8

13	472	640,6	653,2	665,6	665,6	668,8
14	515,8	700	693,6	712,4	718,8	725

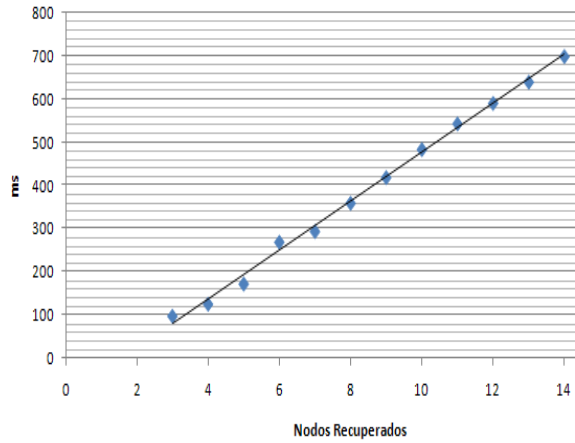
Nodos Hypergrafo 63



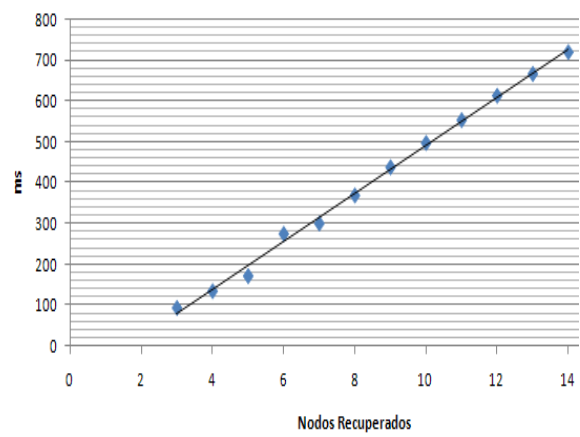
Nodos Hypergrafo 74



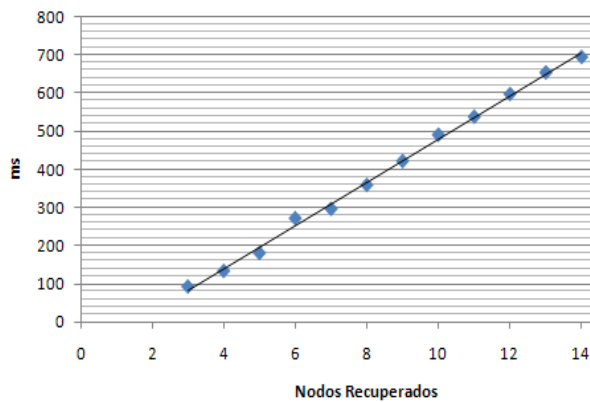
Nodos Hypergrafo 69



Nodos Hypergrafo 78



Nodos Hypergrafo 73



Nodos Hypergrafo 82

