

ANEXO F

Soportes para pruebas realizadas a BDMobIS

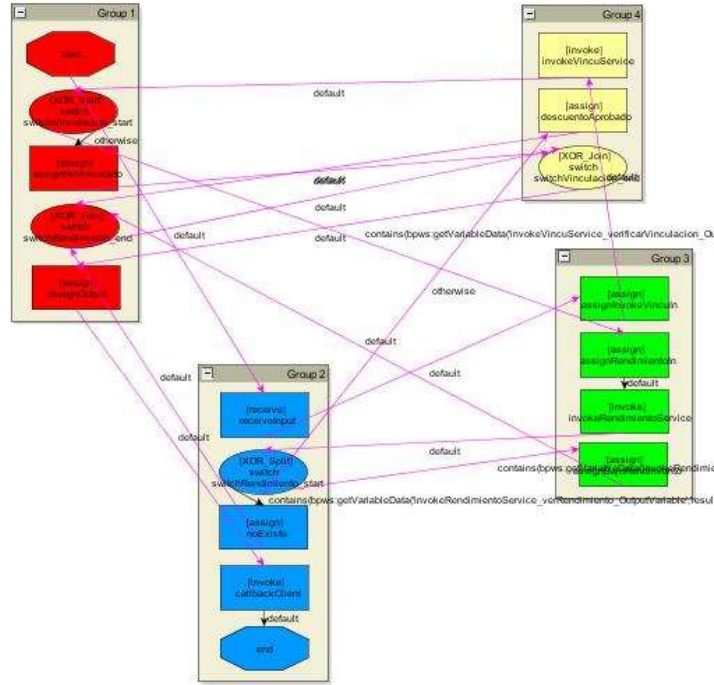
El anexo que se presenta a continuación tiene la finalidad de complementar el numeral 6.3 del capítulo 6 referente a pruebas, de la monografía escrita para el proyecto de grado “Arquitectura para la Distribución de Procesos de Negocio Empresariales en Sistemas Móviles de Información”.

1. Pruebas realizadas al algoritmo de particionamiento multiagente.

Las pruebas que se van a tratar a continuación hacen referencia a las realizadas únicamente sobre el algoritmo de particionamiento. Es importante realizar esta aclaración, ya que, numerales posteriores tratarán ejercicios realizados sobre toda la implementación de BDMobIS en su prototipo exploratorio.

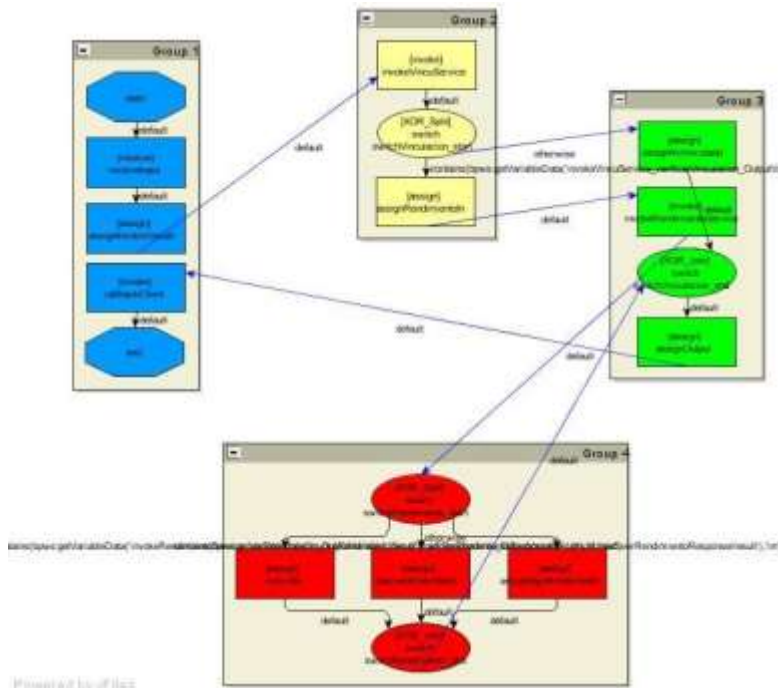
1.1. Reducción de aristas.

Prueba que se puede apreciar en las gráficas mostradas a continuación. En ellas se presenta el proceso de negocio apenas transformado en un grafo (proceso llevado a cabo por el módulo especificado por el numeral 5.2 del capítulo 5 de la monografía escrita para el proyecto de grado titulado “Arquitectura para la Distribución de Procesos de Negocio Empresariales en Sistemas Móviles de Información”) en el cual se pueden apreciar todas las aristas de comunicación que en un principio este tiene (ver figura 1). Posteriormente se presenta el grafo después de haberle aplicado el algoritmo de particionamiento multiagente, en él se puede ver la notable reducción de aristas de comunicación (ver figura 2). Se puede apreciar también el desempeño del algoritmo mediante la figura 3, en donde se muestra numéricamente el anterior proceso. Esto gracias al entorno integrado de desarrollo Eclipse.



Powered by yFiles

Figura 1. Grafo representativo del proceso de negocio. En el, se pueden apreciar 15 aristas de corte (color fucsia) debido a que no se ha ejecutado sobre este grafo el algoritmo de particionamiento multiagente.



Powered by yFiles

Figura 2. Grafo de aristas reducidas. En esta gráfica se puede apreciar el proceso de negocio reducido notablemente en sus comunicaciones. Esto debido a la aplicación del algoritmo de particionamiento multiagente el cual arroja finalmente solo 6 aristas (color azul) de comunicación entre los diferentes módulos.



```
<terminated> CristhianPrueba [Java Application] C:\Archivos de programa\Java\jre1.5.0_06\bin\javaw.exe (18/01/2008 11:59:23)
Cardinalidad del modulo vincuService 3
Cardinalidad del modulo rendimientoService 4
COSTO GLOBAL INICIAL 15.0
**** globalCost Final= 6.0
COSTO GLOBAL INICIAL 13.0
***** globalCost Actual= 6.0
```

Figura 3. Resultados numéricos resultantes de la ejecución del algoritmo de particionamiento multiagente. Imagen tomada del entorno integrado de desarrollo Eclipse.

Las cifras mostradas en la figura 3, muestran una reducción de aristas de comunicación de más del 50%. Cifra que alivia mucho las comunicaciones necesarias entre los diferentes trabajadores de la empresa. De 15 aristas de comunicación (ver figura 1 y ver figura 3 en recuadro rojo) quedan solo 6 (ver figura 2 y ver figura 3 en recuadro azul).

1.2. Tiempo destinado al particionamiento.

Artículos como los nombrados en [DPH95] y en [HLE95], proponen tiempos estimados para encontrar una solución al problema de particionamiento de grafos de 38 segundos, factor nombrado en la pagina 12 del artículo [HLE95] y en la página 5 del artículo [DPH95]. El algoritmo implementado para BDMobIS da soluciones encontradas entre los 1400 y 1800 milisegundos para un total de 50000 ejecuciones particionando grafos que tienen entre 13 aristas y 11 nodos. Si se quiere tener una mejor respuesta por parte del algoritmo se tendría que aumentar la cantidad de ejecuciones, factor que impactaría necesariamente en el tiempo destinado para encontrar una solución mas precisa.

A continuación, en la figura 4, se presentará un ejemplo de particionamiento de un grafo de 21 aristas y 19 nodos el cual muestra el tiempo destinado para esta operación.

Aquí se puede apreciar que el incremento en la cantidad de ejecuciones del algoritmo de particionamiento repercute necesariamente en el tiempo destinado para encontrar la mejor solución al problema planteado.

```

Problems @ Javadoc Console Debug Call Hierarchy
<terminated> BDMobIS [Java Application] C:\Archivos de programa\Java\jdk1.6.0_03\bin\javaw.exe (18/01/2008 13:28:14)
Transform file Descuentos1.kpel ...
El grafo tiene 19 nodos y 21 aristas
NUMERO DE MODULOS 5
NUMERO DE NODOS 19
Cardinalidad del modulo server 5 color Rojo
Cardinalidad del modulo client 3 color Azul
Cardinalidad del modulo FacultyService 5 color Amarillo
Cardinalidad del modulo CajaRecaudosService 3 color Verde
Cardinalidad del modulo RecursosHumanosService 3 color Gris
COSTO GLOBAL INICIAL 18.0
**** globalCost Final= 9.0
COSTO GLOBAL INICIAL 13.0
***** globalCost Actual= 9.0
FIN DEL ALGORITMO SOLCION ALCANZADA CON UN COSTO GLOBAL DE 9.0 Numero de ejecuciones = 100000
finished
Tiempo de ejecucion: 2984

```

Figura 4. Tiempo de ejecución. Este es el tiempo en milisegundos destinado para realizar el particionamiento multiagente en BDMobIS.

En el recuadro rojo de la figura 4 se puede apreciar el tiempo destinado por el algoritmo multiagente al particionamiento de un grafo de 21 aristas y 19 nodos. Con base en lo anterior, el algoritmo multiagente utilizado en BDMobIS realiza un particionamiento más rápido que el propuesto en [DPH95] y en [HLE95], los cuales se concentran en complejidades que son poco frecuentes en los procesos de negocios que hoy en día encontramos en la mayoría de las empresas.

2. Pruebas realizadas sobre toda la implementación de BDMobIS.

A continuación se hará referencia a las pruebas realizadas sobre toda la implementación de la arquitectura de BDMobIS. Estas pruebas se enfocaron en el tiempo necesario para concluir un objetivo de negocio. Los dispositivos móviles empleados para estas pruebas fueron (ver tabla 1):

MARCA	MODELO	MIDP
Sony Ericsson	W600	2.0
Sony Ericsson	Z-520	2.0
Sony Ericsson	Z-550a	2.0

Tabla 1. Dispositivos móviles BDMobIS. Esta tabla muestra los teléfonos celulares utilizados en el prototipo.

Igualmente se utilizó una computadora personal que contaba con las siguientes características: procesador AMD Athlon de 64 Bits, 1 GB de memoria RAM y Windows XP

como sistema operativo. Por otra parte, el software empleado para la implementación del prototipo fue: PostgreSQL pgAdmin III, Eclipse 3.1.2, Eclipse 3.2, Netbeans 6.0, Sun Java System Application Server 9.0, Sony Ericsson SDK, Wireless Toolkit 2.2 Sun Microsystems, Oracle BPEL Process Manager y Collaxa Server Oracle.

A continuación se explicarán las pruebas relacionadas con el tiempo empleado por BDMobIS para solucionar el caso de estudio escogido como proceso de negocio implementado en el prototipo.

Reducción del tiempo necesario para concluir un proceso de negocio

El proceso de negocio escogido para BDMobIS es la petición del descuento en la matrícula financiera para estudiantes que son hijos de docentes de la Universidad del Cauca. Dicho descuento consiste en la reducción del 100 % en los servicios básicos que tiene que cancelar todo estudiante de dicha institución educativa. Para ello se evalúan dos condiciones:

- Vinculación del padre del estudiante aspirante al descuento. Si el padre del estudiante en cuestión no se encuentra actualmente vinculado con la universidad del cauca, el descuento no puede ser otorgado.
- Buen rendimiento académico del estudiante. Si el estudiante se encontrara en bajo rendimiento académico, el descuento no puede ser otorgado.

Estas dos condiciones son evaluadas por dos entes de la Universidad del Cauca que son la división de recursos humanos, para la primera condición, y la Facultad a la cual pertenece el estudiante, para la segunda. Si ambas condiciones se cumplen, el descuento es otorgado, concluyendo así el proceso de negocio. Si alguna de las dos condiciones arriba nombradas no se cumple, el descuento no es otorgado, concluyéndose también así el objetivo de negocio. Según esto, en el proceso de negocio intervienen 3 entes que son el estudiante aspirante al descuento, la división de recursos humanos y la facultad a la cual pertenece el estudiante. Con el fin de alivianar el procesamiento necesario para este objetivo de negocio, se agrega a estas 3 partes, una cuarta que corresponde a un computador personal de escritorio. Este posee las características arriba nombradas.

Ahora, una vez hecho el particionamiento del proceso de negocio, del cual se habla en el capítulo 5 de la monografía escrita para el proyecto de grado titulado “Arquitectura Para La Distribución De Procesos De Negocio Empresariales En Sistemas Móviles De Información”, se procede a entregar a estas 4 partes los sub-flujos de trabajo obtenidos.

El proceso de negocio viene descrito de la siguiente manera. Inicialmente el estudiante solicita la otorgación del descuento llenando un formulario en el cual este pone sus datos y los del docente del cual es hijo. Seguidamente, este los envía, mediante mensajería corta de texto, al siguiente módulo que es la división de recursos humanos de la Universidad del Cauca. Este ente se encarga de verificar la vinculación del docente por el cual el descuento puede ser concedido. Seguidamente, se procede a verificar el rendimiento académico del estudiante. Finalmente, si toda la información es corroborada exitosamente, se avisa al estudiante acerca de la concesión del descuento, de lo contrario se le notifica su negación. Según lo anterior hay dos consultas importantes hechas en este proceso de negocio. Estas son:

- Verificación de la vinculación del funcionario. Esta se realiza mediante la consulta a una base de datos la cual tiene almacenada la actividad o inactividad de un docente en la Universidad del Cauca. Esta verificación se realiza gracias a un servicio web que es invocado, por la división de recursos humanos, con la cédula del funcionario a analizar.
- Verificación del rendimiento. Esta se realiza mediante la consulta a una base de datos la cual almacena las anotaciones que tiene el estudiante a analizar como lo son el estado de bajo rendimiento o su normal desempeño. Esta verificación se realiza gracias a un servicio web que es invocado, por la facultad a la cual el estudiante pertenece, con el documento de identidad del usuario del sistema.

Ahora, se extenderán las funcionalidades de cada ente participante del proceso de negocio.

- Estudiante. Módulo implementado por un dispositivo móvil W600 Sony Ericsson. La aplicación que tiene almacenado este dispositivo móvil es iniciada por el usuario. Se encarga de iniciar el sistema, suministrando la información necesaria para la otorgación o negación del descuento financiero en su recibo de matrícula. Esto se hace llenando un

formulario en el cual se piden los siguientes datos: Nombre de estudiante y su documento de identidad, Nombre del padre junto con su documento de identidad. Este dispositivo móvil envía esta información al sistema, haciendo uso del servicio de mensajería corta. Cabe aclarar que este módulo tiene actualmente un comportamiento síncrono, es decir, realiza la petición al sistema y espera por su respuesta.

- División de Recursos Humanos de la Universidad del Cauca. Módulo implementado por un dispositivo móvil Z-550a Sony Ericsson. La aplicación que tiene almacenada este dispositivo móvil es iniciada por la llegada de un mensaje de texto proveniente del usuario del sistema, en este caso, el estudiante. Esto se hace mediante el uso del Registro Push de la Edición Micro de Java y el registro estático de conexiones entrantes hecho en el archivo descriptor de la aplicación. Este módulo tiene la funcionalidad de verificar la vinculación del funcionario ingresado por el estudiante, en la Universidad del Cauca. Esto se hace gracias a un servicio web que consulta, con la cédula del docente, su actividad o inactividad al interior de la institución. Una vez se tiene el resultado de dicha operación, se procede a enviarlo, mediante un SMS, al siguiente módulo el cual corresponde a la Facultad a la cual pertenece el usuario del sistema.

- Facultad a la cual pertenece el usuario del sistema. Módulo implementado por un dispositivo móvil Z-520 Sony Ericsson. La aplicación que tiene almacenada este dispositivo móvil es iniciada por la llegada de un mensaje de texto proveniente de la división de Recursos Humanos. Esto se hace mediante el uso del Registro Push de la Edición Micro de Java y el registro estático de conexiones entrantes hecho en el archivo descriptor de la aplicación. Este módulo tiene la responsabilidad de verificar el rendimiento académico del estudiante. Una vez llega el mensaje de texto proveniente de la división de recursos humanos, se procede a extraer de su contenido el documento de identidad del estudiante. El dispositivo móvil, mediante SMS, se conecta con un servicio web, el cual consulta las anotaciones del usuario entre las que se encuentran su normal desempeño o bajo rendimiento académico. Una vez se conoce esta información se procede a enviarla, mediante el envío de SMS, al módulo Servidor, el cual, comunicará finalmente la otorgación o negación del descuento al usuario.

- Módulo Servidor. Implementado por un computador de escritorio el cual ejecuta constantemente un motor BPEL. Este se encarga de evaluar la información que le entrega la división de recursos humanos y la facultad a la cual pertenece el estudiante interesado en el descuento. Posee un programa BPEL que gracias a una actividad estructurada “Switch” realiza la simple clasificación entre otorgar o negar el descuento según los datos suministrados. Si el funcionario se encuentra vinculado actualmente en la Universidad y el estudiante no se encuentra en bajo rendimiento académico, este módulo se encarga de enviar un SMS al estudiante con la notificación de que se le ha otorgado el descuento en su matrícula financiera. Si el funcionario no se encuentra actualmente vinculado o el interesado en el descuento se encuentra en bajo rendimiento académico, este módulo se encarga de enviar un SMS al estudiante en el cual se le informa de la no otorgación del descuento financiero en su matrícula.

El proceso de negocio aquí descrito finaliza con la notificación hecha al estudiante de la otorgación o negación del descuento y no toma en cuenta el conocimiento que tengan los demás entes universitarios de ello, como por ejemplo, la división de Sistemas de la Universidad del Cauca.

Las pruebas realizadas para evaluar el tiempo necesario en concluir este objetivo de negocio fueron realizadas bajo condiciones ideales, en las cuales se tuvo un buen desempeño de los centros de mensajería corta de los operadores celulares y un excelente cubrimiento por parte de su red.

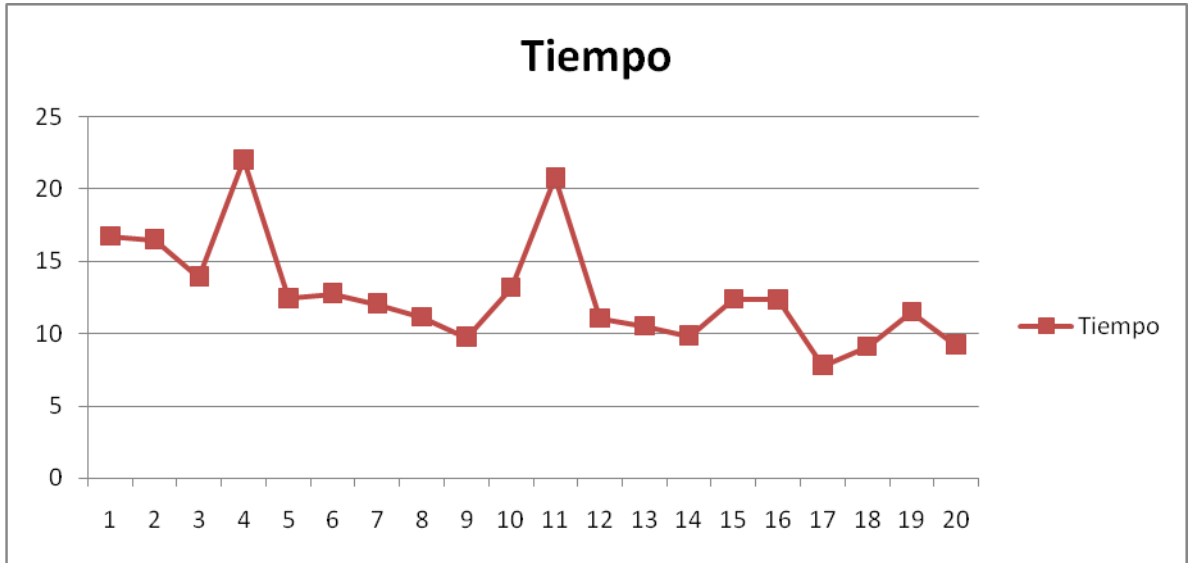


Figura 5. Pruebas de tiempo. Esta gráfica muestra el tiempo empleado por BDMobIS para concluir el objetivo empleado para la implementación. Para realizar esta gráfica se ejecutó 20 veces el proceso de negocio.

Ejecución	Tiempo [minutos]
1	16,75
2	16,49
3	13,91
4	22,05
5	12,43
6	12,75
7	12,07
8	11,12
9	9,74
10	13,19
11	20,78
12	11,03
13	10,52
14	9,85
15	12,37
16	12,34
17	7,77
18	9,08
19	11,47
20	9,23

Tabla 2. Pruebas de tiempo de ejecución realizadas sobre BDMobIS.

Estas mediciones (ver figura 5 y tabla 2) arrojan como promedio un tiempo de 12.747 minutos necesarios para alcanzar el objetivo de negocio propuesto para BDMobIS. En

total, se probó el sistema en 20 ocasiones las cuales arrojaron los tiempos arriba graficados. Estas mediciones fueron hechas desde el momento en que el estudiante ingresa la información necesaria para el descuento hasta que se le notifica el resultado del proceso. Los tiempos críticos del proceso se presentan cuando se realizan las conexiones SMS entre los dispositivos móviles y los Servicios Web y entre los entes participantes del proceso.

Referencias

[HLE95] B. Hendrickson and R. Leland, "A multilevel algorithm for partitioning graphs," en *Proceeding of Supercomputing '95*, ACM, Diciembre 1995.

[DPH95] Diniz, P., Plimpton, S., Hendrickson, B., Leland, R., "Parallel Algorithms for Dynamically Partitioning Unstructured Grids," 7th SIAM Conf. Parallel Proc. Sci. Comput., February 1995. Department of Computer Science, University of California at Santa Barbara; pedro@cs.ucsb.edu Sandia National Labs; {sjplimp, bahendr, leland}@cs.sandia.gov