

Simulación de procesos de negocio utilizando técnicas de minería de procesos para estimar recursos computacionales necesarios para la apropiada implementación de un sistema BPMS/SIOP

Diego Cocconi*, Marisa Pérez*, Juan Pablo Ferreyra*, Claudia Verino*,
Guido Melano*, Noelia Cocconi*, Andrea Biasco*

Departamento de Ingeniería en Sistemas de Información
Universidad Tecnológica Nacional – Facultad Regional San Francisco
Av. de la Universidad 501 – San Francisco (2400) – Provincia de Córdoba – República
Argentina

*{dcocconi, mperez, jpferreyra, cverino, gmelano, ncocconi, [abiasco](mailto:abiasco@sanfrancisco.utn.edu.ar)}@sanfrancisco.utn.edu.ar

Resumen

La gestión de procesos de negocio (BPM, del inglés Business Process Management) implica la aplicación de un ciclo de mejora continua en las organizaciones, en el que intervienen diversas fases. El modelado de procesos de negocio abarca una duración considerable dentro de dicho ciclo, particularmente en las etapas tempranas, pero no suele hacerse énfasis inicialmente en la utilización, configuración y ejecución de los sistemas BPMS (del inglés Business Process Management Systems) y SIOP (Sistemas de Información Orientados a Procesos) que darán soporte a la ejecución de tales procesos, o en todo caso, en los sistemas ERP que actualmente posea la organización y demuestren contar con la flexibilidad suficiente como para especificar y ejecutar los procesos, verificar si todas las integraciones e interacciones con sistemas externos se encuentran disponibles, etc. De la misma manera, suele ser complicado en las primeras etapas estimar los recursos computacionales que se necesitarán para dar soporte a la ejecución de los procesos por parte de estos sistemas. Tal estimación sería útil, por ejemplo, para poder determinar si la organización tendría que lidiar con la adquisición de un servidor, un pequeño clúster de servidores o una solución en la nube para los sistemas. En este trabajo se pretenden considerar aspectos de implementación de los sistemas durante las etapas tempranas del ciclo de vida de BPM, haciendo uso de una simulación en dos etapas y

utilizando técnicas de minería de procesos (la cual se suele emplear recién en la etapa final del ciclo de vida), para determinar de manera aproximada los recursos de cómputo que permitirán ejecutar sin inconvenientes los sistemas BPMS y/o SIOP apropiados para la ejecución de los procesos en la organización.

1. Introducción

La aplicación de la Gestión de Procesos de Negocio (BPM, del inglés Business Process Management) en las organizaciones, teniendo en cuenta los nuevos modelos organizacionales y la tecnología de la información disponible actualmente, puede llevar a lograr mejoras significativas en su desempeño y nuevas oportunidades de negocio [1]. Aunque BPM está muy ligado a las tecnologías de información, no es una tecnología ni la aplicación de determinado software en particular, es una disciplina de gestión orientada a procesos [2].

BPM implica la aplicación de un ciclo de mejora continua, conocido como el ciclo de vida de BPM, en el que intervienen las fases de identificación, descubrimiento, análisis, diseño, configuración e implementación, ejecución y monitoreo, y evaluación de procesos [3-4]. Si bien el modelado de procesos de negocio abarca una duración considerable dentro de dicho ciclo, particularmente en las etapas tempranas, no debe entenderse que la aplicación de BPM se remite sólo a eso; el fin de BPM es la mejora

continua de procesos [5]. Por lo tanto, la aplicación de BPM en una organización no se remite solamente a llevar a cabo las etapas de diseño y análisis del ciclo de vida BPM, obteniendo los artefactos típicos de estas fases (*mapa de procesos*, macro-procesos, procesos principales detallados, descripción detallada de las actividades, etc.) y presentarlos como resultado final. Las fases de configuración e implementación, ejecución y monitoreo, y evaluación son importantes para permitir configurar los sistemas que realmente ejecutarán los procesos, los monitorearán y permitirán analizar su desempeño, con la finalidad de verificar la *performance* de la mejora continua.

Durante la fase de configuración e implementación se configuran y especifican aspectos necesarios para que los modelos de procesos puedan ser interpretados y orquestados por un *Sistema de Gestión de Procesos de Negocio* (BPMS, del inglés *Business Process Management System*) u otro tipo de *Sistema de Información Orientado a Procesos* (SIOP, del inglés *Process-Aware Information System*, PAIS) [3], convirtiéndolos en modelos ejecutables en términos de un lenguaje de especificación, como WS-BPEL (del inglés *Web Services Business Process Execution Language* [4]). Luego, en la fase de ejecución y monitoreo, el BPMS/SIOP crea instancias de los modelos de procesos de negocio de la organización para que puedan ser ejecutados. Durante esta fase también se realiza el monitoreo de diversos indicadores que se hayan definido previamente (*Key Performance Indicators*, KPI [6]) a fin de evaluar el desempeño. Finalmente, en la fase de evaluación se analizan los registros de ejecución (o *logs*) de las instancias de procesos para aplicar ajustes necesarios a futuro [4], cuando se realice una nueva iteración del ciclo de vida, que nunca se detiene como parte de la filosofía de mejora continua.

Respecto del modelado de los procesos de negocio, los modelos no sólo incluyen el conjunto ordenado de las actividades a ejecutar (perspectiva de *control de flujo*), sino que pueden dar soporte a la información que fluye entre ellos (perspectiva de *datos*) y a la asignación de recursos para garantizar su ejecución (perspectiva de *recursos*) [7]. Entonces, la implementación de BPM debería considerar estas diferentes perspectivas también. Durante las primeras etapas del ciclo de vida se suele poner mucho énfasis en la perspectiva de control de flujo y no tanto a las demás. Por lo tanto, no se suele pensar demasiado en los BPMS y/o SIOP que darán soporte a la ejecución de los procesos (o si se utilizarán los sistemas ERP que actualmente posea la organización, siempre y cuando demuestren contar con la flexibilidad suficiente como para especificar procesos, que garanticen todas las integraciones e interacciones con sistemas externos necesarios, etc.). De la misma manera, es difícil también en este punto estimar los recursos computacionales que se necesitarían para implementar tales sistemas. Tal estimación sería útil, por ejemplo, para poder determinar si la organización tendría que lidiar con la adquisición de un servidor, un pequeño clúster de servidores o una solución en la nube para los sistemas.

En este trabajo se pretenden considerar aspectos de implementación de los sistemas BPMS/SIOP durante las etapas tempranas del ciclo de vida de BPM, haciendo uso de una simulación en dos etapas y utilizando técnicas de

minería de procesos (la cual se suele emplear recién en la etapa final del ciclo de vida), para determinar de manera aproximada los recursos de cómputo que permitirán ejecutar sin inconvenientes tales sistemas para la ejecución de los procesos en la organización.

A tal fin, se motiva a realizar, durante las primeras etapas del ciclo de vida de BPM, modelos de procesos lo suficientemente detallados como para especificar la necesidad computacional de cada tarea, las probabilidades de los diferentes flujos de actividades, las distribuciones de probabilidad de la generación de instancias, etc. Con estos datos es posible realizar una simulación de procesos de negocio utilizando la herramienta apropiada para generar trazas de ejecución en forma de registro de eventos, como si hubieran sido producto de la ejecución real de los procesos de negocio por un sistema BPMS o SIOP. Además, si ya se cuenta con sistemas implementados para la ejecución de ciertas tareas, se podrían considerar también los registros de eventos que los mismos generen. De esta forma, se puede utilizar una segunda herramienta de simulación, esta vez para simular la ejecución de dichas trazas sobre hardware concreto (como la herramienta desarrollada para un trabajo previo descrita en [8]), con la finalidad de determinar la *performance* de la ejecución. Para esta segunda simulación se proponen utilizar técnicas de minería de procesos, con la finalidad de recuperar la estructura original de los procesos a partir de los registros de eventos, posibilitando la orquestación de la ejecución de las tareas durante la simulación.

El resto del artículo se organiza de la siguiente manera. Primero, en la Sección 2, se hará una breve descripción de la minería de procesos. A continuación, en la Sección 3, se detallará cómo realizar la estimación de los recursos computacionales mediante simulación de procesos, utilizando técnicas de minería de procesos. La Sección 4 ejemplificará la estimación mediante un caso de estudio concreto. En la Sección 5 se resumen trabajos relacionados. Finalmente, las Secciones 6 y 7 presentan las conclusiones y los próximos pasos a seguir, respectivamente.

2. Minería de procesos

La mayoría de los enfoques de la gestión de procesos de negocio utilizan modelos que parten de una interpretación de la realidad para su posterior ejecución. Al no contemplar en esa interpretación los datos de hechos fácticos relacionados a los procesos puede existir una brecha entre el modelo teórico y la realidad [9]. En el mismo sentido, la generación de mejores modelos no necesariamente produce automáticamente mejores procesos [10]. Por lo tanto, los esfuerzos se deben alinear con el objetivo original de BPM de mejorar los procesos de negocio, en lugar de mejorar los modelos de proceso.

La minería de procesos es una disciplina emergente que busca analizar la información que los sistemas registran sobre los procesos de negocio para poder entender, monitorear, analizar y mejorar dichos procesos, mediante métodos y técnicas para proveer información basada en hechos para dar soporte a las mejoras de procesos. Esta

nueva disciplina se construye sobre enfoques dirigidos por modelos de procesos y minería de datos [11].

El punto de partida para la minería de procesos son los registros de eventos, los cuales pueden almacenar información de recursos que ejecutan o inician una actividad, marcas de tiempo de un evento o elementos de datos grabados con un evento. Todas las técnicas de minería de procesos suponen que los sucesos pueden ser registrados secuencialmente de forma que cada evento se refiere a una actividad que está relacionada con una instancia de proceso particular [12]. Un formato de registro de eventos ampliamente utilizado en minería de procesos es MXML¹ (del inglés *Mining eXtensible Markup Language*).

Existen tres tipos de minería de procesos (Figura 1) [13]: (1) *descubrimiento*: parte de un registro de eventos y produce un modelo sin utilizar ninguna otra información *a priori*; (2) *conformidad*: compara un modelo de proceso existente con el registro de eventos del mismo, mostrando dónde el proceso real se desvía del modelado; (3) *mejora*: toma un registro de eventos y un modelo de proceso y extiende o mejora el modelo usando los eventos observados (mientras que el control de conformidad mide la alineación entre el modelo y la realidad, este tipo de minería de procesos cambia o extiende el modelo).

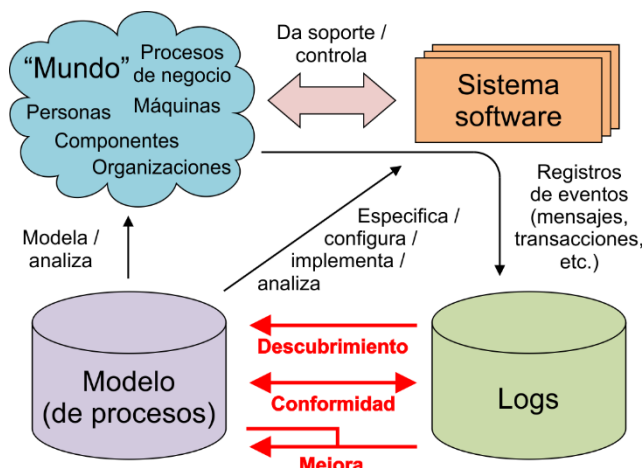


Figura 1: Los distintos tipos de minería de procesos [13]: descubrimiento, conformidad y mejora.

Adicionalmente, dependiendo de los factores que guían a un proyecto de minería de procesos, surgen tres tipos de proyecto [9]: (1) proyecto de minería de procesos *impulsado por los datos*: se dispone de registros de eventos, pero no existe una pregunta o necesidad concreta, por lo que el proyecto tiene un espíritu exploratorio y los interesados pretenden descubrir información valiosa a partir del análisis de los datos; (2) proyecto de minería de procesos *basado en preguntas*: busca responder una pregunta concreta, por ejemplo, ¿por qué los casos manejados por el equipo "x" toman más tiempo que los casos manejados por el equipo "y"?; (3) proyecto de minería de procesos *impulsado por*

objetivos: apunta a mejorar un proceso de negocio con respecto a un KPI particular (por ejemplo, reducción de costos o tiempos).

Como se mencionó previamente, el presente trabajo describe cómo ha sido aplicada la minería de procesos enfocada al descubrimiento para analizar trazas resultantes de una simulación de procesos de negocio, con la finalidad de conformar carga computacional para una segunda simulación que permita determinar la *performance* de la ejecución de dicha carga sobre hardware específico.

3. Estimación de recursos computacionales mediante simulación de procesos de negocio

En las etapas iniciales de la implementación BPM en una organización se realizan muchas actividades de análisis, diseño y modelado, que generan como salida diversos tipos de modelos, desde globales y abstractos hasta bien detallados y concretos, prácticamente ejecutables. Antes de la etapa de configuración debe definirse el sistema BPMS/SIOP que dará soporte a la ejecución de los procesos; sin embargo, algo que es un poco más complicado de determinar son los requerimientos de hardware para desplegar tales sistemas de acuerdo con el tamaño de la organización y su esperada ejecución de procesos. Por ejemplo, el mismo motor de procesos de *Camunda*² puede ser configurado en una sola computadora o incluso en clústeres para proporcionar un soporte *multi-tenant*, para cubrir diferentes necesidades de ejecución.

Este trabajo describe una alternativa para evaluar y seleccionar el perfil de hardware que mejor se adapte a un determinado despliegue de los sistemas BPMS/SIOP que ejecutarán los procesos, basada en una simulación en dos etapas: (1) *simulación de procesos de negocio* para la generación de la carga de trabajo que podrían esperar los sistemas BPMS/SIOP, sobre la base de los modelos que se determinen en la etapas de análisis y diseño de procesos; y (2) *simulación de la ejecución sobre hardware* de la carga de trabajo anterior para evaluar la *performance* de dicha ejecución y determinar la configuración más apropiada para alojar los sistemas.

Para comenzar, es válido aclarar que ninguna descripción de la arquitectura particular de los sistemas BPMS/SIOP es requerida como entrada, de ninguna naturaleza (lógica, despliegue, etc.), por lo que no es necesario pensar en algún producto software en particular. Pero sí será necesario llevar a cabo dos actividades de preparación previas. Primero, será necesario haber aplicado las primeras etapas del ciclo de vida BPM (identificación, descubrimiento, análisis y diseño de procesos) y contar con los modelos de procesos de la organización, con un alto nivel de detalle. Estos modelos serán la entrada para la primera etapa de simulación. Es por eso que los mismos deberían ser lo más detallado posibles, identificando todas las actividades a nivel de tarea y evitando construcciones abstractas. En segundo lugar, aunque los modelos sean lo

¹ <http://www.processmining.org/logs/mxml>

² <https://docs.camunda.org/manual/7.5/introduction/architecture/>

suficientemente detallados, es necesario realizar varios ajustes para poder generar una carga de trabajo computacional acorde para la segunda simulación. De alguna manera hay que adjuntarles información de recursos computacionales a las tareas de los procesos. Aunque esto puede realizarse de diversas maneras, como extendiendo la notación BPMN (del inglés *Business Process Model and Notation* [14]) utilizada para modelar los procesos, como en [15], donde se especifican recursos en la nube a disposición de cada tarea.

Sin embargo, sería conveniente adaptar los modelos para que puedan ser directamente simulados por una herramienta de simulación de procesos en general, como BIMP³, por ejemplo, sin requerir extensiones adicionales. BIMP es un simulador de procesos de negocio ampliamente utilizado, el cual permite realizar la simulación y adicionalmente, generar registros de eventos en formato MXML, como si los procesos se hubieran ejecutado realmente en un sistema BPMS/SIOP. Dichos registros consisten en archivos XML en el que se guardan marcas de tiempo de inicio de una instancia de proceso, evento o tarea; marcas de tiempo de fin para dichas instancias; y una descripción del recurso encargado de ejecutar las tareas (por ejemplo, un servicio automático para las tareas de servicio). Con estos datos es posible determinar las duraciones de las tareas que requieran el uso de recursos computacionales (como las tareas de servicio), las cuales serán utilizadas para determinar los recursos de hardware apropiados. Sin embargo, es importante que no se supriman del modelo las actividades humanas (como las tareas manuales) y las temporizaciones. Los tiempos de estas actividades y temporizaciones deben respetarse, representando esperas sin utilizar ningún recurso computacional.

Mediante varios experimentos previos se ha logrado determinar cómo los patrones más comunes en los modelos de procesos son interpretados por BIMP y registrados en su *log* de eventos, para recomendar las modificaciones apropiadas en los diagramas detallados, haciéndolos totalmente compatibles con la simulación.

En primer lugar, para simplificar, todos los subprocesos se deberían descomponer en procesos separados, hasta llegar en cada uno de ellos al nivel de tarea. Una vez conseguido esto, por lo expuesto anteriormente, cada tarea debería especificar su tipo. Las tareas de servicio y las manuales se dejan tal cual están y deberían registrar sus duraciones; las de servicio serán interpretadas como la ejecución de software, por lo que su duración tendrá en cuenta el tiempo que se utiliza hardware. La Figura 2.a muestra el resultado de ejecutar dichas tareas en BIMP; mediante los registros “*assign*” y “*complete*” es posible determinar la duración de las tareas, y haciendo uso de la etiqueta “*Originator*” es posible determinar el tipo de tarea. Por otro lado, las tareas de usuario, de envío y de recepción de mensaje deberían descomponerse en dos partes. Para las primeras, habría una tarea manual que represente el trabajo humano realizado y otra de servicio, que corresponda a la utilización del sistema (Figura 2.b); para las segundas, una tarea manual que represente el trabajo humano y una de servicio, que indique puramente el envío del mensaje

(Figura 2.c); finalmente, las tareas de recepción de mensaje pueden reemplazarse un evento temporizador que represente la espera por la recepción del mensaje, una tarea de servicio para la recepción del mensaje en sí y una tarea manual indicando el procesamiento humano (Figura 2.d).

Los eventos de inicio y fin genéricos pueden dejarse tal cual están, pues su duración esperada es cero. Los eventos de fin con envío de mensaje requerirán que se conecte a continuación el proceso que inicia con la recepción del mensaje (o la parte del mismo); la conexión entre los dos procesos se realizará insertando una tarea de servicio para el envío del mensaje, un evento de temporización para la espera de la recepción y otra tarea automática para la recepción del mensaje (Figura 2.e). Por su parte, la conexión entre un proceso que finalice con un envío de señal y aquella parte o proceso que comience con la recepción de la señal pueden unirse directamente, sin insertar ninguna tarea extra. Los eventos de envío y recepción de mensaje intermedios deberían ser reemplazados por tareas automáticas (el de recepción precedido por un evento temporizador) (Figura 2.f), pues para el simulador el evento de envío tiene duración cero generalmente (no siempre) y el de recepción una duración determinada; mediante el reemplazo se vuelven más deterministas. Por su parte, el evento de temporización puede dejarse tal cual está, ya que representa la duración de una espera sin hacer nada. En BIMP se generan los registros “*assign*” y “*complete*”, como para cualquier tarea manual.

Respecto de las actividades de bucle y multi-instancia, se observaron disparidades en la interpretación por parte de BIMP, ya que no se realiza la multi-instanciación, por más que se establezcan de manera apropiada los atributos en el modelo. Por lo tanto, aquellas tareas que tengan que utilizar marcadores de bucle y multi-instancia, se deben modelar como una tarea simple, con una duración equivalente al producto de la cantidad de instancias que son necesarias por la duración individual.

Finalmente, sobre los *gateways* hay también algunas consideraciones. Los *gateways* AND, OR exclusivo y OR inclusivo no requieren ninguna alteración, sin embargo, los dos últimos necesitan que sea especificada la probabilidad de ocurrencia de cada una de sus ramas. En el caso del OR exclusivo, la suma de las probabilidades debe ser exactamente 100%, mientras que cada rama del OR inclusivo puede tener una probabilidad independiente, de 0 a 100%. Por su parte, los *gateways* basados en eventos deberían convertirse a una estructura más simple, que consiste en un evento de temporización, el cual modela la espera por la ocurrencia de alguno de los eventos del *gateway*, y un *gateway* OR exclusivo en reemplazo del *gateway* basado en eventos (Figura 2.g).

Contando con los modelos BPMN adaptados para su simulación, se procede a definir los escenarios adecuados en cuanto a la cantidad de instancias que necesitan ser generadas, su distribución probabilística de arribos, la duración estimada de las tareas y las probabilidades de ocurrencia de eventos y bifurcaciones de flujo (todo ello puede anotarse en los diagramas de los modelos mediante

³ <http://bimp.cs.ut.ee/>

comentarios). Considerando estos escenarios y los modelos simulables, se procede a la primera simulación, en el simulador de procesos de negocio (como BIMP), para obtener las trazas de ejecución de las instancias que constituyen la carga de trabajo para la segunda simulación. Estas trazas deberían generarse de acuerdo a un estándar como MXML, por ejemplo.

Las trazas resultantes permiten planificar la segunda simulación. En este caso, sería requerido una herramienta de simulación de la ejecución de dichas trazas sobre

recursos computacionales directamente, considerando sólo las instancias de tareas que son puramente computacionales. En esta situación, se puede utilizar una herramienta que se ha desarrollado para trabajos relacionados anteriores, específicamente para realizar la simulación de la ejecución de procesos de negocio en la nube [16]. A tal fin, se ha debido extender la herramienta para ampliar su utilización a un rango que va desde la ejecución de los procesos en un solo PC o servidor, la ejecución en un clúster de servidores con escalamiento y la ejecución en la nube, que ya era

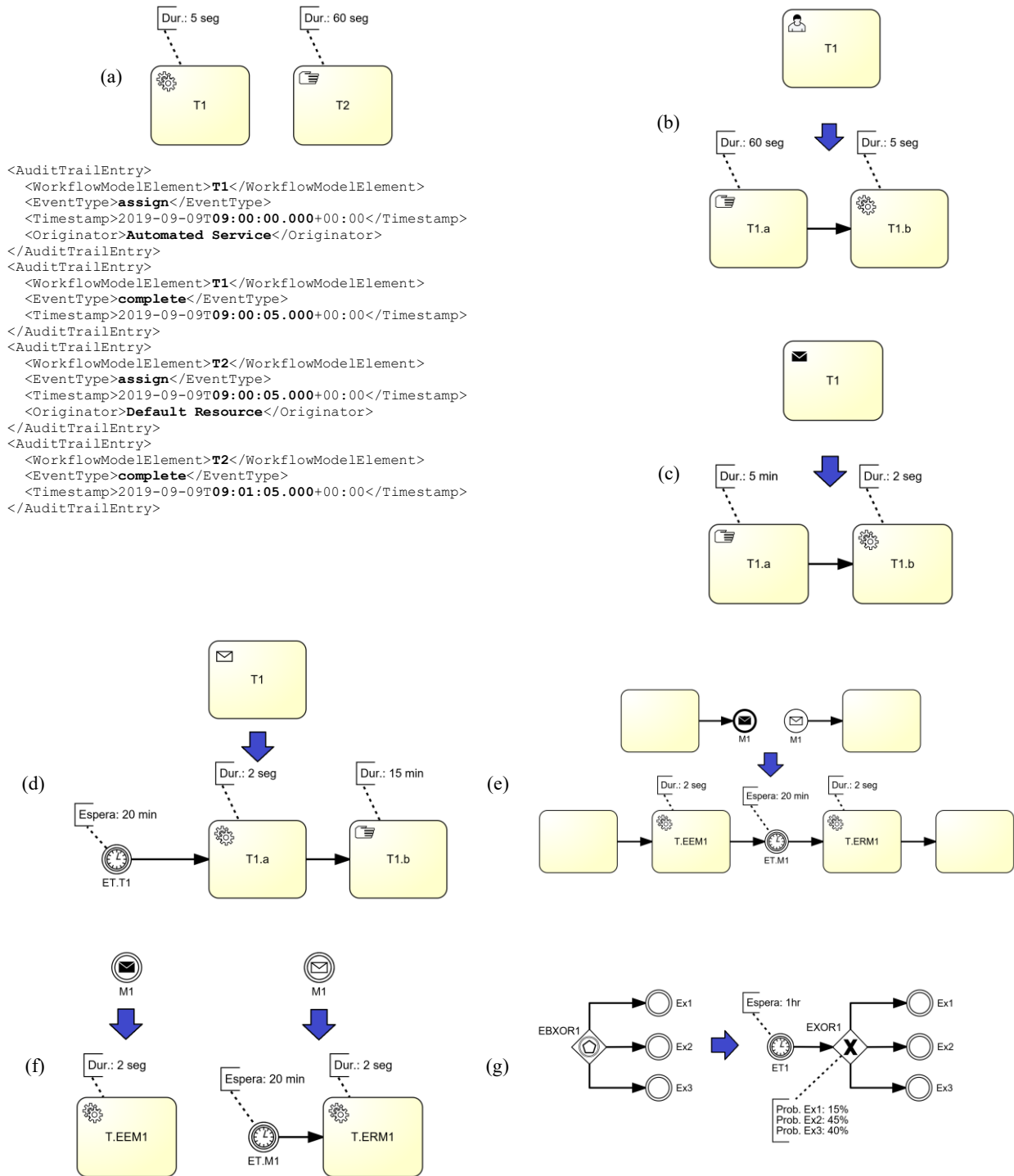


Figura 2: Adaptación de constructores y patrones BPMN para convertir el modelo detallado en un modelo más apropiado para la primera simulación (con BIMP).

realizada nativamente por la herramienta. De esta manera, es posible probar cada configuración y ver la que se desempeña más adecuadamente con la carga de trabajo proporcionada.

Lo único necesario para utilizar esta herramienta es la definición del perfil de hardware sobre el que se ejecutarán los procesos. Luego de la extensión es posible establecer las características de un servidor único, un clúster de servidores (físicos o virtuales) o un clúster elástico de máquinas virtuales para emular una plataforma en la nube.

A partir de este punto comienza a jugar un papel central la minería de procesos. Las trazas generadas después de la primera simulación deben importarse como trabajo computacional a ser realizado por el simulador de ejecución sobre recursos. Este proceso de importación debe considerar solamente las tareas que implican utilización de recursos computacionales, como las tareas de servicio, envío y recepción de mensajes. Los tiempos de inicio y duración de las mismas se pueden extraer directamente del registro de eventos. Para efectuar tal importación se ha propuesto la utilización de un algoritmo de minería de procesos, el cual reestablezca las relaciones de precedencia de las tareas (básicamente si están en secuencia o en paralelo). Se ha probado uno de los algoritmos más difundidos de minería de procesos, el algoritmo α [13]; sin embargo, se ha hallado una dificultad al intentar recomponer modelos a partir de trazas generadas por BIMP, principalmente en lo que respecta la detección de tareas paralelas, particularmente cuando se cuenta con pocas instancias. La Figura 3.a muestra un pequeño proceso con tareas paralelas, el cual fue simulado en BIMP considerando 100 instancias e importado luego en *ProM⁴*, una conocida herramienta para minería de procesos. Como resultado de aplicar el algoritmo α se obtiene el proceso de la Figura 3.b. Como se puede observar, el proceso no es adecuadamente reconstruido. Por lo tanto, se decidió utilizar una primera aproximación para tomar como caso testigo, la cual consiste en utilizar un algoritmo simple que reordene las trazas de manera secuencial. En la Sección próximos pasos se propone realizar mejoras a otros algoritmos (entre ellos, el propio algoritmo α), para aplicar y comparar con el caso testigo.

Finalmente, una vez importadas las trazas de ejecución de los procesos, puede simularse dicha carga de trabajo sobre los recursos de hardware. Luego de la simulación, pueden obtenerse varios indicadores e información gráfica para determinar la mejor alternativa entre los perfiles de hardware definidos. Básicamente, se pueden evaluar las alternativas sobre la base de las demoras en la ejecución de las tareas de los procesos de negocio, el porcentaje de las mismas que son afectadas por dichas demoras y el uso de CPU en cada perfil. En la siguiente Sección se muestra un ejemplo de esta estimación para un caso de estudio concreto.

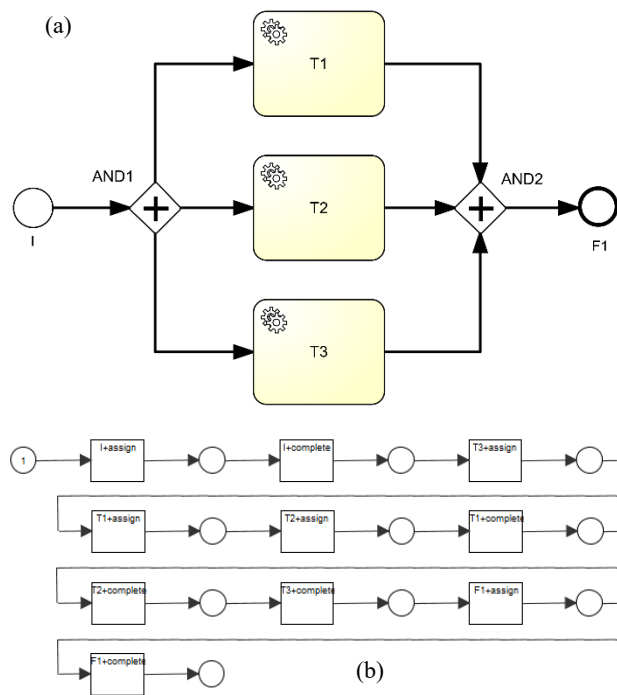


Figura 3: Resultado de aplicar el algoritmo α a un proceso simple con tareas en paralelo, partiendo de las trazas generadas por BIMP.

4. Caso de estudio

Para ilustrar la aplicación de la propuesta descrita se ha utilizado un caso de estudio presentado en un trabajo anterior [17]. El caso se refiere a la implementación de BPM en una secretaría de la UTN Facultad Regional San Francisco, la Secretaría de Extensión y Cultura. Entre las actividades que se desarrollan en dicha secretaría, se ha hecho foco en el servicio de cursos, el cual procura promover la transferencia de conocimiento a áreas productivas. Dicho servicio se brinda a un amplio espectro de destinatarios que comprende alumnos, docentes, graduados, administrativos, operarios, empresarios, técnicos, profesionales, miembros de distintas instituciones y la comunidad en general. Partiendo de este trabajo anterior, se dispone de los modelos de procesos detallados que se realizan para la gestión de cursos. En la Figura 4 se muestra el ejemplo del proceso para cancelar la inscripción previa a un curso. Así como este último, se dispone de 9 modelos de procesos detallados más que resumen las actividades que se llevan a cabo en la gestión de cursos. Cada uno de estos modelos es desagregado y configurado de acuerdo a los criterios detallados en la Sección 3, para obtener 14 nuevos modelos que se pueden simular sin problemas con un simulador como BIMP. A su vez, estos modelos se han ampliado con anotaciones para definir el escenario de ejecución que mejor describe los tiempos y las duraciones de las tareas que se llevan a cabo. La Figura 5 muestra el proceso de la Figura 4 después de las modificaciones.

⁴ <http://www.promtools.org/doku.php>

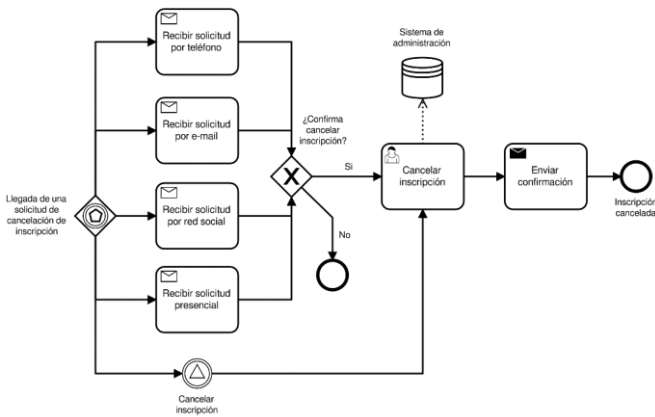


Figura 4: Ejemplo de modelo de proceso detallado para cancelar la inscripción previa a un curso.

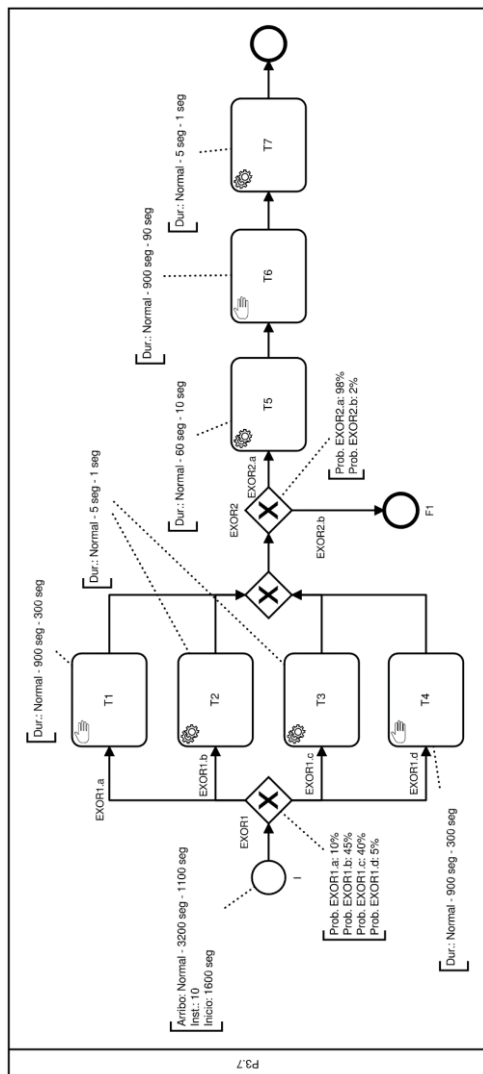


Figura 5: Ejemplo de modelo de proceso simulable, el cual corresponde al modelo detallado de la Figura 4.

Se puede apreciar que los nombres de los constructores no son importantes en esta etapa, sino que es necesario

definir la distribución de arribo de las instancias, la cantidad de las mismas, las distribuciones de las duraciones de las tareas y las probabilidades de los *gateways*. La cantidad de instancias se ha determinado en base a la simulación de un día completo de trabajo (aproximadamente 32500 segundos), considerando la máxima carga esperada (no el promedio), para ver cómo respondería el sistema en su máxima capacidad.

Con estos modelos y el escenario planteado, se realiza la primera etapa de simulación, para obtener las trazas de la ejecución de los procesos, en términos de registros de eventos de minería de procesos (archivos XML en formato MXML).

A partir de este punto, se definen los perfiles tentativos de hardware sobre los que se realizaría la segunda simulación, esta vez para la ejecución de la carga utilizando dicho hardware. A tal fin, se han definido dos posibilidades, que corresponden a servidores con las características detalladas en la Tabla 1.

Tabla 1. Perfiles de hardware tentativos

Perfil de hardware	Parámetros	Valores
<i>Servidor 1</i>	Cantidad de CPUs	2
	Velocidad de CPU [MIPS]	2200
	RAM [MB]	8192
	Almacenamiento [GB]	500
<i>Servidor 2</i>	Cantidad de CPUs	4
	Velocidad de CPU [MIPS]	2200
	RAM [MB]	8192
	Almacenamiento [GB]	500

Para la importación de la carga de trabajo de las trazas XML de este caso de estudio se utiliza un algoritmo simple que reordena las trazas de manera secuencial, aunque podría utilizarse cualquier otro algoritmo de minería de procesos. Luego, se procede a la simulación de la ejecución de los procesos sobre el hardware, considerando cada uno de los perfiles de la Tabla 1. Los resultados se muestran en la Tabla 2, y en la Figura 6 se pueden apreciar los gráficos de utilización de CPU para caso.

En la simulación sobre ambos perfiles, durante 32500 segundos se han ejecutado 534 tareas puramente computacionales. Para el caso del servidor 1, las tareas se han excedido en promedio 3,72 segundos en su ejecución, respecto de sus duraciones teóricas. Se observa también que si bien la mayoría de ellas (86,9%) se han ejecutado a tiempo o con un exceso no superior al 10% de su duración teórica, las restantes (13,1%) han presentado demoras que superan el 10% de la duración. Esto se ve reflejado en el gráfico del uso de la CPU (Figura 6.a), ya que se observa cierta saturación de la capacidad ofrecida, si bien puede manejarse considerablemente bien la ejecución. Por otro lado, el servidor 2 tiene una *performance* bastante superior. De las instancias de tareas que se ejecutan, el 100% no excede el 10% de su duración teórica, y el promedio del tiempo en que prolongan su duración es prácticamente cero. El gráfico de uso de CPU (Figura 6.b) muestra que puede manejar la ejecución mucho más holgadamente que el servidor 1. Debido a que ambos perfiles representan un

servidor físico, uno con 2 CPUs más que el otro, se concluye que la alternativa del servidor 2 es más conveniente para soportar la ejecución del sistema SIOP/BPMS, ya que el costo de un mejor procesador es razonable para cubrir un futuro crecimiento del sistema sin tener que recurrir a una arquitectura más escalable

Tabla 2. Resultados de la simulación de la ejecución de los procesos sobre el hardware

Perfil de hardware	Resultados de la simulación	Valores	
Servidor 1	Tiempo total de simulación [seg]	32500,00	
	Cantidad de instancias de tareas simuladas	534	
	Promedio de tiempo que las tareas han excedido su duración teórica [seg]	3,72	
	Porcentaje de tareas que exceden su duración teórica,	Rango	Porcentaje [%]
		Exceden entre 0 – 10%	86,9
Exceden entre 10 – 50%		4,2	
Exceden entre 50 – 100%		6,8	
Exceden más de 100%	2,1		
Servidor 2	Tiempo total de simulación [seg]	32500,00	
	Cantidad de instancias de tareas simuladas	534	
	Promedio de tiempo que las tareas han excedido su duración teórica [seg]	0,00	
	Porcentaje de tareas que exceden su duración teórica,	Rango	Porcentaje [%]
		Exceden entre 0 – 10%	99,8
Exceden entre 10 – 50%		0,2	
Exceden entre 50 – 100%		0,0	
Exceden más de 100%	0,0		

5. Trabajos relacionados

Actualmente existen numerosas herramientas de simulación de procesos de negocio. Suites BPM, como *Bizagi*⁵, suelen contar con entornos de simulación para sus modelos, generalmente con la finalidad de realizar la verificación y validación de los mismos, para evaluar su correctitud. Generalizando, lo que es llamado generalmente simulación de modelos BPMN se refiere a herramientas limitadas a ejecutar el flujo del proceso en términos de *tokens*, sin considerar la utilización de recursos [18]. Pocas herramientas de simulación, como la proporcionada por ADONIS⁶, cuentan con extensiones BPMN para modelar aspectos de simulación como recursos humanos.

En [19], los autores han realizado un análisis de diversas herramientas de simulación de procesos de negocio, haciendo énfasis en la falta de soporte para contemplar tareas humanas, lo que dificulta la predicción de

performance. Con la aparición de BPMN, es posible representar de manera apropiada llamadas a servicios de software y modelar tareas humanas, pero no representar adecuadamente el contexto de ejecución, en términos de arquitecturas y recursos de hardware. Aunque dichos autores sugieren la utilización de técnicas de análisis de eventos (como minería de procesos), los registros de eventos por sí solos tampoco aportan mucha información de los recursos (particularmente humanos).

Los autores de [18] proponen una herramienta de simulación que predice el impacto del diseño de un sistema de información sobre la ejecución de los procesos de negocio. Sin embargo, definen su propio metamodelo para representar carga de simulación, no asociándola directamente a cada tarea.

En contraste, la propuesta descrita en este trabajo aprovecha la definición de los modelos de proceso detallados provenientes de fases tempranas de implementación BPM, agregándoles información sobre duraciones, probabilidades de flujo, etc., que también puede obtenerse del trabajo en dichas etapas preliminares. Luego, es posible aplicar técnicas de minería de procesos que partan de registros de eventos obtenidos mediante la simulación estos modelos modificados, para simular su ejecución sobre hardware predefinido y evaluar su *performance*.

6. Conclusiones

Poder estimar los recursos computacionales que se necesitarán por parte de los sistemas BPMS/SIOP para dar soporte a la ejecución de los procesos de negocio en una organización durante las primeras etapas de implementación BPM sería útil, por ejemplo, para poder determinar si la organización tendría que lidiar con la adquisición de un servidor, un pequeño clúster de servidores o una solución en la nube para dichos sistemas. En este trabajo se han considerado aspectos de implementación de los sistemas durante las etapas tempranas del ciclo de vida de BPM, haciendo uso de una simulación en dos etapas y utilizando técnicas de minería de procesos, para determinar de manera aproximada los recursos de cómputo que permitirán ejecutar sin inconvenientes los sistemas BPMS y/o SIOP apropiados.

A tal fin, se parte de la definición de los modelos de proceso detallados provenientes de fases tempranas de implementación BPM, agregándoles información sobre

⁵ <https://www.bizagi.com/>

⁶ <https://www.adonis-community.com/en/>

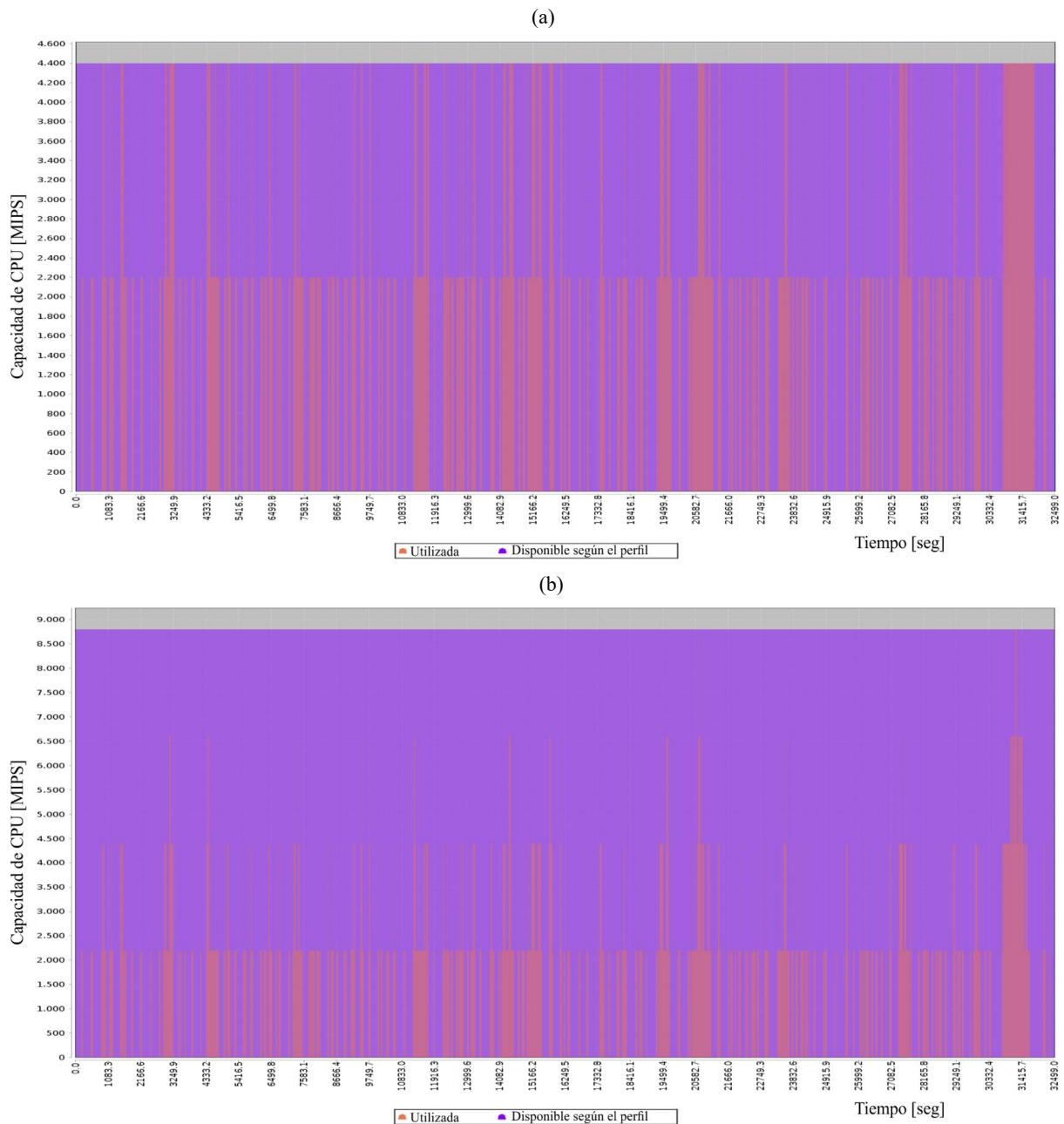


Figura 6: Gráficos de utilización de CPU para cada uno de los perfiles de hardware utilizados en la simulación de la ejecución de procesos

duraciones, probabilidades de flujo, etc., que también puede obtenerse del trabajo en dichas etapas. Con estos modelos se realiza una simulación de procesos de negocio para la generación de la carga de trabajo que podrían esperar los sistemas BPMS/SIOP y posteriormente dicha carga se importa en otra herramienta de simulación (de ejecución sobre hardware de la carga de trabajo), utilizando un algoritmo apropiado de minería de procesos. La segunda simulación permite evaluar la *performance* de dicha ejecución y determinar la configuración de hardware más apropiada para alojar los sistemas.

La estimación se ha probado en un caso de estudio descrito en un trabajo anterior, arrojando resultados

concluyentes, lo cual estimula a aplicarla en nuevas situaciones.

7. Próximos pasos

El trabajo futuro se centrará en perfeccionar el proceso de estimación mediante la evaluación y/o modificación de diferentes algoritmos de minería de procesos que puedan detectar mejor las relaciones entre las tareas a partir de los registros de eventos. También se puede considerar la posibilidad de automatizar la generación de los modelos simulables a partir de los detallados y de facilitar la primera simulación, ya que al utilizar un simulador de procesos de

negocio como BIMP, es necesario realizar la carga de los tiempos de arribo de las instancias, las duraciones de las tareas, las probabilidades de las ramificaciones de flujo, etc. en forma manual, lo cual insume un tiempo considerable y es propenso a errores.

Referencias

- [1] Lazarte, I. M., Tello-Leal, E., Roa, J., Chiotti, O., Villarreal, P. D. (2010, Octubre). "Model-driven development methodology for B2B collaborations". *2010 14th IEEE International Enterprise Distributed Object Computing Conference Workshops* (pp. 69-78). IEEE.
- [2] Ko, R. K. (2009). "A computer scientist's introductory guide to business process management (BPM)". *XRDS: Crossroads, The ACM Magazine for Students*, 15(4), 4.
- [3] Dumas, M., La Rosa, M., Mendling, J., Reijers, H.A. (2013). "*Fundamentals of Business Process Management*". Springer, Verlag Berlin Heidelberg.
- [4] Weske, M. (2007). "*Business Process Management. Concepts, Languages, Architectures*". Springer-Verlag Berlin Heidelberg.
- [5] He, G., Xue, G., Yao, S., Wu, Z. (2010). "*Business Process Modeling: A Survey*".
- [6] Harmon, P., 2014, "*Business process change*", 3ra edición. Morgan Kaufmann.
- [7] Stroppi, L. J. R., Chiotti O., Villarreal, P. D. (2015). "Defining the resource perspective in the development of processes-aware information systems". *Elsevier, Information And Software Technology*, 59, 3-2015, pp. 86-108.
- [8] Cocconi, D., Roa, J., Villarreal, P. (2018, Agosto). "A Platform Based on Cloud Computing for Executing Collaborative Business Processes". *CLEI Electronic Journal*, Vol. 21 No. 2, Paper 6, pp 6:1-6:26. ISSN: 0717-5000. DOI: 10.19153/cleiej.21.2 (<https://doi.org/10.19153/cleiej.21.2>).
- [9] van der Aalst, W. M. P. (2011). "Process Mining: Discovering and Improving Spaghetti and Lasagna Processes". *IEEE Symposium on Computational Intelligence and Data Mining, (CIDM 2011)*. Paris, France, IEEE.
- [10] van der Aalst, W. M. P., La Rosa, M., Santoro, F. M. (2016). "Business Process Management - Don't forget to improve the process!". *Business & Information Systems Engineering* 58(1): 1-6.
- [11] Lazarte, I., Acosta Parra, C. A., Vilallonga, D. G. (2016). "Mejora y gestión de procesos de negocio inter-organizacionales aplicando técnicas de minería de procesos". *XVIII Workshop de Investigadores de Ciencias de la Computación*. Bs. As., Argentina.
- [12] van der Aalst, W. M. P., Dustdar, S. (2012). "Process Mining Put into Context". *IEEE Computer Society*, vol. 16, IEEE, 2012, pp. 82-86.
- [13] van der Aalst, W. (2016). "*Process Mining. Data Science in Action*", 2da edición. Springer-Verlag Berlin Heidelberg.
- [14] Object Management Group, OMG. (2011). "Business Process Modeling Notation, V2.0" <https://www.omg.org/spec/BPMN/2.0/About-BPMN/>.
- [15] Hachicha, E., Assy, N., Gaaloul, W., Mendling, J. (2016, Junio). "A configurable resource allocation for multi-tenant process development in the cloud". *International Conference on Advanced Information Systems Engineering* (pp. 558-574). Springer, Cham.
- [16] Cocconi, D., Roa, J., Villarreal, P. (2019, Abril). "eBPSim: A Simulation Tool for Testing Elasticity Strategies in Cloud-based Business Process Solutions". *Proceedings of the 22nd Iberoamerican Conference on Software Engineering (CIBSE 2019)*, pp 377-390, Curran Associates, Inc. La Habana, Cuba. ISBN: 978-1-5108-8795-4. <http://www.proceedings.com/49253.html>.
- [17] Cocconi, D., Ferreyra, J. P., Verino, C., Pérez, M. (2018, Noviembre). "Optimización organizacional basada en la aplicación del ciclo de vida BPM completo para la mejora continua de los procesos de negocio". *Memorias del 6to Congreso Nacional de Ingeniería Informática – Sistemas de Información (CoNaiISI 2018)*. Mar del Plata, Argentina. ISSN: 2347-0372. <https://www.conaiisi2018mdp.org/memorias/memorias.html>.
- [18] Heinrich, R., Merkle, P., Henss, J., Paech, B. (2017). "Integrating business process simulation and information system simulation for performance prediction". *Software & Systems Modeling*, 16(1), 257-277.
- [19] Van der Aalst, W. M., Nakatumba, J., Rozinat, A., Russell, N. (2008). "Business Process Simulation: How to get it right". *BPM Center Report BPM-08-07*, BPMcenter.org, 285, 286-291.