

Ejecución y monitorización distribuida de procesos de negocios: una implementación con Bonita Open Solution

Leonardo Karabogolian¹, Patricia Bazán¹, Jose Martinez Garro¹

¹ Facultad de Informática – Universidad Nacional de La Plata,
50 y 120 – La Plata

leonardo.karabogolian@gmail.com, pbaz@info.unlp.edu.ar,
josemartinezarro@yahoo.com.ar

Abstract. BPM (*Business Process Management*) ha ido creciendo en relevancia dentro de las organizaciones y ha madurado tanto desde el punto de vista metodológico y como de la tecnología que la sustenta. Por otra parte, el avance tecnológico tanto en términos de comunicación como en poder de cómputo, han hecho que *Cloud Computing* sea una opción potencial para la reducción de costos y mejoras en el procesamiento. La combinación de las tecnologías asociadas a *Cloud Computing* y BPM modifica aspectos tanto de diseño como de ejecución de los procesos de negocios. Los ambientes distribuidos en el contexto de los procesos favorecen el rendimiento y proponen incorporar el concepto de descomposición de procesos, permitiendo que los mismos se ejecuten tanto en un entorno cloud como en uno embebido. Si bien la descomposición de procesos es un tema abordado en los últimos años, la monitorización de dichos procesos no ha sido demasiado explorado aún. Este trabajo propone una implementación de una arquitectura para un sistema de monitorización de procesos distribuidos utilizando Bonita Open Solution como motor de procesos, su API y el uso de conectores personalizados.

Keywords: BPM, BPMS, cloud computing, monitorización de procesos, descomposición de procesos, Bonita OS

1. Introducción

Las organizaciones actuales apuestan fuertemente a contar con procesos de negocio cada vez más eficientes y eficaces. La utilización de Sistemas de Gestión de Procesos de Negocio (BPMS - *Business Process Management Systems*) para identificar y diseñar, ejecutar, monitorear y optimizar los procesos de negocio ha sido de gran ayuda para poder disminuir los costos, aumentar la productividad, mejorar los servicios a los clientes, crear un marco de organización y coordinación de actividades para el personal de la empresa u organización.

Otra de las tecnologías que actualmente está teniendo un interés masivo en las empresas y organizaciones es la computación en la nube o *Cloud Computing*. Con este nuevo paradigma, las empresas buscan proporcionar servicios de computación bajo demanda con una alta fiabilidad, escalabilidad y disponibilidad en un entorno

distribuido, donde el poder de cómputo, almacenamiento de datos y servicios se subcontratan a terceros y se ponen a disposición de las empresas y clientes, lo que ocasiona que los riesgos económicos y técnicos disminuyan.

Indefectiblemente, hace unos años, estas dos potentes herramientas en el área de IT, hicieron que se comience con el estudio e implementación de BPMS en la “nube”. Esta combinación de técnicas de *clouding* y BPM ofrece un enfoque flexible y ágil, junto con las ventajas de ambos paradigmas. La gran capacidad computacional de los sistemas en el *cloud* y el “pago por uso” en lugar de enfrentar grandes inversiones en software y hardware, son dos grandes ventajas de la combinación de estas técnicas. Sin embargo, al utilizar un BPMS en la nube, se pierde el control sobre los datos sensibles del negocio, lo que conlleva a tomar un riesgo muy grande para las empresas de hoy en día.

Actualmente podemos encontrar BPMS que se encuentren en servidores locales o en servidores localizados en el *cloud* brindando Software como Servicio (SaaS – *Software as a Service*). La descomposición de procesos de negocios antes mencionada, permite que un proceso se ejecute en ambos ambientes, dependiendo de la lógica y modelo adoptado por el desarrollador del proceso al momento de la descomposición. Sin embargo, una vez que se realiza la descomposición del proceso, la monitorización y seguimiento del proceso de negocio original en este sistema híbrido se dificulta, haciendo que se deba recolectar toda la información de cada uno de los servidores en donde se encuentran las partes del proceso.

Si bien el estudio de la descomposición de procesos de negocios se encuentra en una etapa bastante avanzada, y podemos encontrar mucha información relacionada, el estudio de la integración de estos procesos descompuestos para la monitorización y seguimiento de los procesos de negocio distribuidos es un campo aún poco explorado. La motivación de este trabajo se basa en favorecer el seguimiento de procesos descompuestos en un sistema distribuido y cubrir de este modo la etapa de monitorización y optimización del ciclo de vida de los procesos de negocios. Para ello se propone una implementación de una arquitectura para un sistema de monitorización de procesos distribuidos utilizando *Bonita Open Solution* [6] como motor de procesos, su API y el uso de conectores personalizados.

El sistema de monitorización se basará en las siguientes etapas:

- Implementación de un conector en Bonita OS para la ejecución de un proceso de negocio descompuesto entre distintos servidores distribuidos.
- Investigación de la API REST de Bonita Open Solution para la instanciación de procesos remotos y el acceso a las propiedades de procesos ya instanciados y desplegados en el motor remoto.
- Confección de una base de datos local para la persistencia de la información relacionada a los procesos de negocios instanciados de forma local junto a la relación que tiene con los procesos de negocio remotos.
- Implementación de una aplicación Web para monitorización y seguimiento de procesos de negocios distribuidos. Utilización de Web Services y API REST para la integración de los procesos descompuestos y distribuidos, y la visualización unificada del proceso global para el correcto seguimiento del proceso de negocio.

Este trabajo se organiza de la siguiente manera: en la Sección 2 se plantea el enfoque de BPMS en un entorno distribuido. En la sección 3 se introducen conceptos de descomposición de procesos. En la Sección 4 se abordan aspectos propios de la

monitorización distribuida de procesos. En la Sección 5 se presenta la solución propuesta y su aplicación a un caso de estudio en la Sección 6. Finalmente, en la Sección 7 se describen las conclusiones.

2. BPMS en sistemas distribuidos

Si bien los BPMS basados en un entorno *cloud* pueden ayudar a las pequeñas y medianas organizaciones a mejorar la eficiencia de los procesos y reducir sus gastos de infraestructura, todavía existen algunas barreras al utilizar este servicio.

Los dos temas principales de estudio en la adopción y del uso de un BPMS en el *cloud* son:

- **Protección de la privacidad:** algunos datos gestionados por los procesos de negocios pueden ser confidenciales y las organizaciones no están dispuestas a correr el riesgo de un posible robo de información. Si bien los servicios de *cloud* se encuentra ocultos detrás de barreras de seguridad provistas por la infraestructura de red y tienen un avanzado mecanismo para mantener la división de información entre sus clientes, los dueños de la información sensible temen perder el control de estos datos, o puede ocurrir que por reglas contractuales con sus clientes no se les permita almacenar información de éstos en otros lugares que no sean servidores propios.
- **Actividades que no son altamente computacionales:** por otro lado, la eficiencia y efectividad de las actividades que no son altamente computacionales puede disminuir debido a que la transferencia de los datos puede tomar más tiempo que el procesamiento mismo de los datos. Además, los costos de la actividad pueden incrementarse debido a que la transferencia de los datos es uno de los elementos de facturación en un sistema de *cloud computing*.

Hoy en día, en la mayoría de las soluciones donde se utilizan BPMS, el motor de proceso, las actividades y los datos de los procesos se encuentran alojados en un mismo sitio, ya sea en un sistema embebido o en un ambiente de *cloud computing*, donde presentan las ventajas e inconvenientes de cada uno de estos. Para poder utilizar los principales beneficios de ambos sitios, esto es, privacidad de datos en un sistema embebido, y poder de cómputo y ubicuidad relacionado a un sistema de *cloud*, se ha investigado en [4] un modelo de distribución, denominado PAD (Proceso-Actividad-Datos), en el que el arquitecto de procesos de negocio puede separar uno de estos procesos de acuerdo a los beneficios que brindan cada una de estas ubicaciones. Los ambientes distribuidos en el contexto de los procesos de negocio favorecen el rendimiento y proponen incorporar el concepto de descomposición de procesos, permitiendo que los mismos se ejecuten tanto en un entorno *cloud* o embebido [1][2][3][4].

3. Descomposición de procesos

El modelo de distribución de procesos llamado PAD (Proceso-Actividad-Datos), describe las posibilidades de separación de los componentes de un BPMS entre nodos

cloud y embebidos alternativamente. El modelo PAD define cuatro posibilidades de distribución:

1. El primer patrón de solución es el enfoque tradicional de BPM donde todos los elementos están alojados en el usuario final.
2. El segundo patrón es útil cuando el usuario ya tiene un sistema BPM, pero las actividades con intensidad de cómputo se localizan en el *cloud* para incrementar su performance.
3. El tercer patrón es útil para los usuarios que aún no poseen un sistema BPM; en este caso podrían adoptar un esquema de *cloud* de manera de pago por uso, donde las actividades sin intensidad de cómputo y los datos sensibles se pueden localizar en el usuario final.
4. El cuarto patrón es el modelo basado en *cloud* donde todos los elementos se localizan en la nube.

Es posible generalizar la distribución de procesos e identificar un patrón en el cual el motor de procesos, las actividades y los datos se despliegan tanto en la nube como en el usuario final (en la Figura 1 se puede observar este caso).

Esta solución presenta dos beneficios potenciales:

- El motor de procesos regula el flujo de control y el flujo de datos. Una actividad recibe datos del motor de procesos y luego de su ejecución los datos que son producidos se pasan de nuevo al motor de procesos.
- Cuando la nube no se encuentra accesible, los usuarios pueden ejecutar los procesos de negocio en forma completa en el sistema embebido hasta que el primero vuelva a estar disponible.

Para observar la diferencia entre un motor de procesos situado en un solo lugar, como por ejemplo en el patrón 2 del esquema de distribución PAD, y motores de procesos situados en ambos sitios, veamos el siguiente ejemplo.

Consideremos que una secuencia de actividades se ubica en la nube, mientras que el motor de procesos se despliega en el usuario final. Cada actividad utiliza los datos de salida de la actividad previa como entrada. Los datos no se envían directamente de una actividad a la otra sino que son enviados al motor de proceso primero (Figura 1).

Debido a que la transferencia de datos es uno de los factores de facturación en el modelo de *cloud computing*, estas situaciones pueden volverse más caras cuando se transfieren grandes cantidades de datos entre actividades.

Para evitar este problema se puede agregar un motor de procesos al *cloud*, el cual regula el flujo de control y el flujo de datos entre las actividades situadas en él (Figura 1b). Cuando una secuencia de actividades se localiza en el *cloud*, los datos se regulan por el motor de procesos en el *cloud*, lo cual reduce la cantidad de datos a ser transferidos entre este y el sistema embebido.

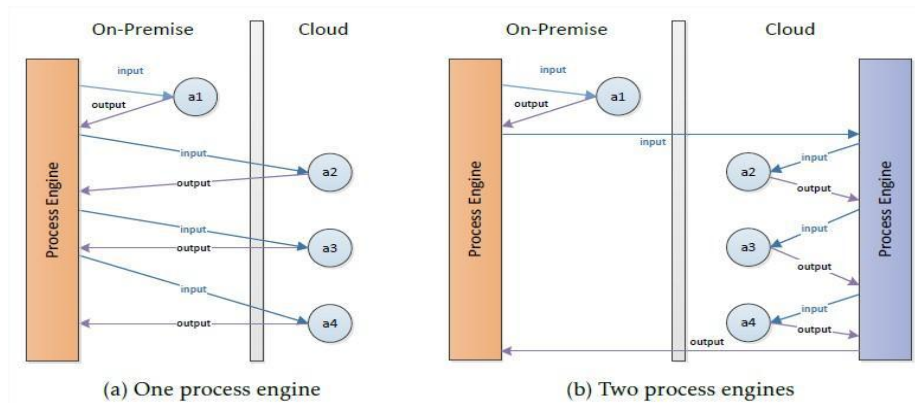


Figura 1. Datos enviados entre actividades coordinadas por motores de procesos [2]

Para poder correr un mismo proceso de negocio en dos motores de proceso separados, el mismo debe ser dividido en dos procesos individuales. Puede llegar a ser conveniente para los usuarios del BPMS tomar una lista de distribución del proceso de negocio y sus actividades, la cual puede ser automáticamente transformada en dos procesos de negocio, uno en el *cloud* y otro en el sistema embebido [1][2][3].

Una aproximación posible para manejar la descomposición del proceso es identificar la estructura y la semántica del mismo. Esto significa que se deben identificar aquellas actividades del proceso de negocio en donde la ejecución de las mismas dan la pauta de atomicidad, y que su ejecución, ya sea en el sistema embebido o en el *cloud*, no modificaría el comportamiento del proceso de negocio original. Una estrategia para detectar aquellas actividades atómicas es tratar de observar las dependencias de control y de datos que tienen con otras actividades y con el motor de procesos en el cuál se ejecutan. Por ejemplo, si existe una actividad que utiliza información confidencial que obtiene del sistema embebido, lo más probable es que no sea recomendable ejecutar la actividad en el motor de procesos situado en el *cloud*, ya que correríamos peligro de exponer tales datos. Al identificar las dependencias de control y de datos, se pueden investigar las consecuencias de mover ciertas actividades del sistema embebido al *cloud* y viceversa.

Cuando se conocen las consecuencias de la distribución de actividades, se puede crear una transformación de modelo en la cual un proceso de negocio y una lista con marcas se usan para crear dos procesos individuales, uno para el *cloud* y otro para el usuario final. Además, se puede generar una descripción de la coreografía para describir la comunicación entre ambos procesos de negocio utilizando algún lenguaje estándar.

4. Monitorización de procesos de negocio distribuidos

En las soluciones orientadas a procesos de negocios existen dos objetivos principales. Primero, la capacidad de proveer flexibilidad en los procesos de negocio, permitiendo una rápida reacción a los cambios del mercado. Y segundo, proveer una buena visualización de los procesos y permitir una optimización eficiente de estos.

Para poder optimizar los procesos de negocios tenemos que saber qué es lo que está funcionando mal o es ineficiente, por lo tanto, debemos comenzar a medir los procesos. Es aquí donde se puede utilizar una herramienta de monitorización llamada Monitorización de Actividades Empresariales (*BAM - Business Activity Monitoring*). Esta herramienta que se encuentra dentro de muchos BPMS nos permite medir varios aspectos de los procesos de negocio tales como tiempo, costos, rendimiento, estados y resultados de operaciones, procesos y transacciones.

Las personas encargadas de optimizar los procesos de negocio tienen que tener en cuenta varias tareas tales como identificar aquellas actividades que están teniendo poca eficiencia (ya sea en tiempo, costo o calidad); determinar los factores internos y externos que afectan la decisión de optimizar un proceso y como éstos influyen con el dueño del proceso, los dueños de las tareas y las áreas específicas de la organización; aplicar reingeniería, buenas prácticas o rediseño del proceso; definir nuevas medidas de rendimiento del nuevo proceso (*KPI - Key Performance Indicator*), realizar simulaciones y monitorearlos para confirmar que ha sido mejorado.

BAM provee acceso a información de los procesos en tiempo de ejecución, permite un análisis en tiempo real de los procesos de negocio, muestra los cuellos de botella en las tareas, mide el tiempo de cada tarea y provee herramientas para visualizar toda esa información [5]. Además, se utiliza para asegurar que los procesos de negocio funcionan como es esperado, simplificar la información compleja relacionada a los procesos y mostrarla oportunamente.

5. Solución propuesta

La solución que se propone para la ejecución y monitorización de los procesos de negocio distribuidos puede separarse en tres grandes fases: 1) el diseño del proceso de negocio distribuido, 2) la ejecución de cada uno de los subprocesos en los servidores en los que se encuentran, y 3) la recolección de la información perteneciente a los subprocesos o a las instancias, que servirán para realizar la monitorización.

El concepto de “subproceso” en la notación BPMN 2.0 tiene un significado diferente al que se utilizará en este trabajo. Mientras que en la bibliografía actual se utiliza el término para referirse a una actividad compuesta que es incluida dentro de un proceso y que contiene un conjunto de actividades, compuertas, eventos y flujos de secuencia [7], en esta sección se utilizará dicho término para referirse a las partes del proceso de negocio original que ha sido dividido a raíz de la descomposición del proceso.

El diseño del proceso de negocio distribuido, al igual que en el diseño de los procesos de negocio que se conocen comúnmente, es la fase previa para la ejecución de los procesos de negocio. Sin embargo, para realizar la ejecución de procesos de negocio de forma distribuida es necesario descomponerlos para crear los subprocesos que serán situados en los distintos motores de procesos que se encuentran en los servidores distribuidos, de acuerdo a si se desea aprovechar las características de pago por uso y la facilidad en la escalabilidad de sistemas basados en *cloud*, o la privacidad de los datos de los sistemas embebidos.

Como producto del proceso de negocio descompuesto se tendrán dos o más subprocesos que se deberán ejecutar en distintos motores de procesos encadenando el

flujo de control y de datos de las actividades pertenecientes al proceso original. Así, al finalizar una instancia en un servidor, esta inicia automáticamente una nueva instancia de la partición del proceso siguiente en el servidor que corresponda de acuerdo a la arquitectura de distribución. Para esto, cada servidor del esquema distribuido debe ser capaz de comunicarse con el servidor siguiente para poder iniciar instancias y continuar con la ejecución del proceso original. [1][3]

La ejecución del proceso de negocio distribuido comenzará con la generación de una instancia de uno de los subprocesos, y luego éste será el encargado de iniciar la ejecución del subproceso siguiente, o de los diferentes subprocesos en caso de que existiera más de uno. El seguimiento secuencial de la ejecución de los subprocesos, se implementa con un conector de Bonita que accede a la API REST del motor de procesos remoto. El conector se desarrolla para realizar dos tareas, la primera es la de instanciar procesos en motores de procesos remotos de Bonita, y la segunda, la de guardar en una base de datos local el identificador de las instancias generadas en otro motor, como así también el identificador de la nueva instancia en el motor de procesos remoto. Con estos datos, la aplicación de monitorización será capaz de interactuar con los diferentes servidores para recolectar la información perteneciente a las instancias ejecutadas en otros motores de procesos.

Para lograr un mejor seguimiento en la ejecución, el conector también genera datos de salida que sirven para controlar que se haya ejecutado correctamente o, y si la ejecución no fue exitosa, determinar el motivo de la falla. El uso de estos datos de salida servirá para controlar el flujo de trabajo del proceso, haciendo que la instancia no finalice hasta que se complete la instanciación del proceso remoto.

La fase de recolección de la información se lleva a cabo una vez que el proceso de negocio es ejecutado o está siendo ejecutado en los diferentes motores de procesos. Dada una instancia iniciada en un servidor de la arquitectura, debemos ser capaces de obtener, no solo los datos propios de la misma, sino también de cualquier instancia que esta haya desencadenado en otro servidor. Para poder realizar esto, será necesario asociar las distintas instancias de proceso iniciadas de manera encadenada, a los fines de poder recuperar información sobre las mismas accediendo a los distintos servidores intervinientes. [1][3]

Utilizando la API REST de Bonita y haciendo uso de la información almacenada en la base de datos, se puede obtener la definición de un proceso de negocio que se encuentra instalado dentro del motor de procesos, como así también la información de las instancias desplegadas dentro del motor de procesos. A través de las peticiones que realicemos a la API REST y con las respuestas obtenidas en formato XML, es posible desarrollar una aplicación web para recolectar la información referida al estado de los procesos y sus actividades en todo el camino de ejecución de los subprocesos de negocio en los diferentes servidores, unificar estos datos y visualizar al proceso de negocio distribuido como si se tratase de un proceso único. De esta manera, al unificar la información obtenida se le puede brindar al analista de procesos una visión integradora del proceso original, con el fin de monitorear las instancias, obtener valores de desempeño y determinar posibles fallas o mejoras que deban realizarse.

Si bien se ha hecho hincapié en la combinación del uso de un sistema *cloud* y uno embebido, en la solución propuesta se generaliza este modelo para lograr una visión

en la cual es posible adaptar la ejecución de procesos descompuestos a cualquier esquema de servidores distribuidos.

La arquitectura del sistema para la ejecución y monitorización de procesos de negocio distribuidos estará basada principalmente en tres componentes, estos son:

- El motor de procesos de Bonita Open Solution [6], que se ejecuta sobre un servidor Tomcat
- La aplicación de monitorización distribuida, desarrollada en PHP y que se ejecuta en un servidor Apache
- Un servidor de base de datos (MySQL) para almacenar la información de las instancias de los procesos de negocio

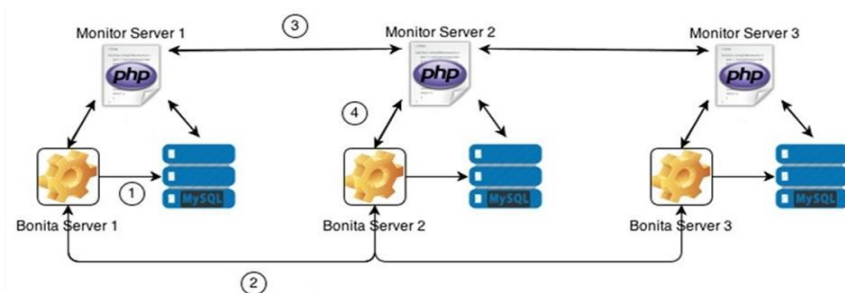


Figura 2. Arquitectura del Sistema de Ejecución y Monitorización con 3 servidores

6. Caso de estudio

El caso de estudio se basa en el Diseño 3D de un producto industrial, en donde el tiempo de procesamiento del modelo 3D del producto está dado por los detalles que contiene el mismo, y es por esto que el tiempo para efectuar la conversión del modelo 3D al diseño 3D depende del poder de procesamiento de la infraestructura de hardware.

El modelado es una técnica que se utiliza para ir dando forma a objetos. Por lo general, el modelo visual suele ser el modelo 3D que los diseñadores manejan, dejando las fórmulas a procesos computacionales. El proceso de transformación de un modelo en 3D hacia una imagen 3D es llamado renderización (rendering).

El propósito del caso de estudio, es utilizar el modelo de infraestructura como servicio de la nube, para realizar el renderizado del modelo 3D, siguiendo sus pasos a través de un proceso de negocio. Actualmente existe un servicio que brinda la empresa Autodesk llamado “Autodesk 360 Rendering”, en el cuál, al utilizar algunos de sus productos tales como AutoCAD, 3ds Max o Revit, le permite al usuario subir a la nube sus modelos y renderizarlos en ella.[8]

En este marco, suponemos una empresa que se compone de una Gerencia General, una oficina técnica, un departamento de mercadotecnia, un departamento de ingeniería, y un equipo de diseño, en donde la descomposición del proceso de negocio original (Figura 3) se hará en 3 servidores con sus respectivos BPMS de la siguiente manera: el primer servidor (en rojo), exclusivo de la Gerencia General, es el que ejecuta las actividades que manejan los datos sensibles de la empresa, alojado detrás

de un firewall o cuenta con una base de datos con acceso restringido, para asegurar la seguridad y confidencialidad de los proyectos; el segundo servidor (en verde), contiene las actividades que realizan los diferentes equipos de Ingenieros, de Mercadotecnia, de Diseñadores y la Oficina Técnica, para llevar a cabo los proyectos; y el tercer servidor (en azul), va a estar situado en la nube, ejecuta las actividades de renderizado de los modelos 3D.

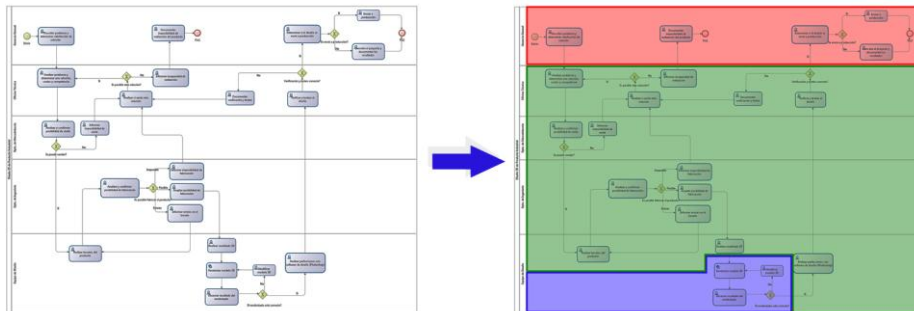


Figura 3. Descomposición del proceso de negocio original y división de servidores

El proceso de negocio original funciona de la siguiente manera: en lo más alto de la pirámide organizacional se encuentra la Gerencia General, encargada de plantear los problemas a solucionar, y tomar las decisiones de fondo de la empresa como así también de inicio y finalización de cada uno de los proyectos. Luego, se encuentra la Oficina Técnica que se encarga de analizar el problema y determinar posibles soluciones de acuerdo a factores como costos, competencia, entre otros, y funciona como nexo entre la Gerencia General y los equipos de especialistas. El Equipo de Mercadotecnia se encarga de analizar si el producto es capaz de comercializarse en el mercado, o si es necesario realizar cambios en las especificaciones que se han definido. El Equipo de Diseño realiza los bocetos del producto como una aproximación a la idea de modelo que tendrá dicho producto. El Equipo de Ingenieros verifica que es posible la fabricación del producto contenido en el boceto y se encarga de aprobar, enviar modificaciones o rechazar tales bocetos. En caso de que sean aprobados, los diseñadores deben realizar el modelo en 3D del producto, utilizando programas especializados basados en herramientas CAD (*Computer-Aided Design*), para enviar a renderizar al servidor que se encuentra en la nube y obtener una imagen real de producto final. La Oficina Técnica se encarga de realizar las pruebas necesarias y la aprueba de acuerdo a las especificaciones requeridas por la Gerencia General, que es la encargada de rechazar o aprobar el proyecto, y enviarlo a producción o al cliente que lo solicitó.

Este proceso de negocio es un claro ejemplo de la conjunción de las diferentes características tratadas en el punto 2, donde podemos ver que en el servidor 1 protegemos la privacidad de los datos de la empresa, mientras que en el servidor 3, situado en la nube, aprovechamos la capacidad de procesamiento otorgada por el servicio de infraestructura (*IaaS – Infrastructure as a Service*) que nos brinda *cloud computing*.

7. Conclusiones

Actualmente, las organizaciones que utilizan sistemas basados en BPMS como tecnología para conducir los procesos de negocios que modelan la actividad de dichas organizaciones, tienen sistemas embebidos con arquitecturas centralizadas donde deben estar sostenidas por una infraestructura muy costosa que a medida que pasa el tiempo se va dañando, teniendo que reparar los componentes, o va quedando obsoleta, en cuyo caso se debe reemplazar por una nueva seguramente más costosa que la anterior.

A partir de la descomposición de procesos de negocio, ejecución distribuida y con despliegue en la nube surgen las siguientes ventajas:

- Se aumenta la disponibilidad de las aplicaciones que interactúan a través del proceso.
- Se logra la integración de elementos dispersos en una arquitectura a través de servidores BPM
- Ubicuidad de los servidores, permitiendo la ejecución de los procesos en áreas más cercanas al lugar de los responsables de las tareas.
- Infraestructuras elásticas y adaptables a la necesidad del negocio, y con pago sólo por el uso, evitando grandes costos en infraestructuras propias.

Por otra parte, la distribución conlleva pérdida de la visión global del proceso tanto para la monitorización como para transparencia de la distribución en ejecución, siendo ambos aspectos resueltos por la propuesta de este trabajo.

Referencias

1. Karabogolian L., Bazán P., Martínez Garro J. "Ejecución y monitoreo de procesos de negocios distribuidos entre diferentes motores de Bonita OS". WICC 2014 XVI Workshop de Investigadores en Ciencias de la Computación. 2014
2. Evert F. Duipmans. "Business Process Management in the cloud with data and activity distribution". Faculty of electrical engineering, mathematics and computer science software engineering. University of Twente. EWI/SE - 2012-002. November 2012.
3. J. Martínez Garro, P. Bazán. "Monitoreo de procesos distribuidos en el cloud: Una propuesta arquitectónica". Jornadas Chilenas de Computación 2013.
4. Han YB, Sun JY, Wang GL. "A cloud-based BPM architecture with user-end distribution of non-compute-intensive activities and sensitive data". JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 25(6): 1157–1167 Nov. 2010. DOI 10.1007/s11390-010-1092-5
5. Jiri Kolar. "Business Activity Monitoring". Masaryk University Faculty of Informatics. Brno, spring 2009.
6. Bonitasoft - Open Source Workflow & BPM software. <http://es.bonitasoft.com>
7. Object Management Group, Inc. (OMG). "Business Process Model and Notation (BPMN) Version 2.0". January 2011
8. Brian Budge. Autodesk 360 Rendering - Scalable and Robust Rendering in the Cloud. http://www.highperformancegraphics.org/previous/www_2012/media/Hot3D/HPG2012_Hot3D_Autodesk.pdf