LIBROS DE CÁTEDRA

FACULTAD DE INFORMÁTICA

ORQUESTACIÓN DE SERVICIOS

La orquestación de servicios y las aplicaciones actuales

Una visión por procesos de negocio

PERSPECTIVA POR PROCESOS DE NEGOCIO

Una presentación editorial limpia para una obra académica de informática.

Patricia Bazán (coordinadora)

Coordinación editorial · Libros de Cátedra

LA ORQUESTACIÓN DE SERVICIOS Y LAS APLICACIONES ACTUALES

UNA VISIÓN POR PROCESOS DE NEGOCIO

Patricia Bazán

(coordinadora)

Facultad de Informática

UNIVERSIDAD NACIONAL DE LA PLATA

edulp

EDITORIAL DE LA UNLP

A Dalila, por su lucha y valentía.

Índice

Prólogo 5

Capítulo 1
Procesos de Negocio 6
Patricia Bazán

Capítulo 2
Conceptos de servicios en los lenguajes de programación 26
Dalila Romero y Patricia Bazán

Capítulo 3
Integración de Aplicaciones 40
Patricia Bazán

Capítulo 4
Herramientas BPMS 54
Anahí Rodríguez y Patricia Bazán

Capítulo 5
Casos de éxito y escenarios posibles para implementación de BPM 70
José Martínez Garro y Patricia Bazán

Los autores 94

Prólogo

El presente libro ha sido concebido como una guía orientada a quienes incursionan por primera vez en los procesos de negocio como modelo de gestión de las organizaciones, tanto en el ámbito público como privado, y que buscan incorporar conceptos que ayuden a comprender el paradigma y sus herramientas asociadas.

El objetivo es presentar y describir los conceptos de BPM (*Business Process Management*) relacionados con la orquestación de servicios aplicados a las herramientas de software. En los siguientes capítulos, se analizan las herramientas para la aplicación de BPM como orquestador de servicios, se revisan las características de las herramientas de desarrollo para construir servicios, se estudia el desarrollo de soluciones desde la perspectiva del modelado de procesos y servicios y se enuncian problemáticas surgidas en casos de aplicación reales.

La obra fue concebida como libro de cátedra y busca plasmar la experiencia adquirida por los autores de este libro en la enseñanza de BPM, a lo largo de los últimos cinco años.

Asimismo, se describen en el último capítulo, una serie de casos de éxito en los cuales también estuvieron involucrados los autores y que fueron los disparadores del análisis de herramientas presentado en el capítulo 4.

Por último, queremos destinar un párrafo especial a nuestra colega y compañera Dalila Romero, quien ha sido activa protagonista de esta tarea e integrante de nuestra asignatura. Dalila nos dejó físicamente en Febrero 2020 luego de pelear con una dura enfermedad. A ella le dedicamos este trabajo, del cual fue hacedora e impulsora y en el cual participó hasta sus últimas horas.

CAPÍTULO 1
Procesos de Negocio

Patricia Bazán

En los orígenes de la ciencia de la computación todo el énfasis estuvo puesto en desarrollar sistemas que automatizan tareas que se hacían manualmente, siendo este suficiente desafío.

En la actualidad, se han automatizado muchas tareas y el gran desafío es cómo mejorar la capacidad de los sistemas para alcanzar nuevos requerimientos: agregar nuevas interfaces, combinar múltiples fuentes de datos en una sola, interactuar con dispositivos móviles y reemplazar viejas aplicaciones con nuevas.

La tendencia hoy es hacia un paradigma orientado a procesos, donde las aplicaciones cubren la actividad global de la empresa y las herramientas son los BPMS (*Business Process Management Systems* o Sistemas de Gestión de Procesos de Negocio).

Los modelos actuales resultan insuficientes porque son pobres en su integración y se orientan a describir datos y transacciones.

El cambio de enfoque en el modo de diseñar aplicaciones e implementar soluciones radica en:

- Explicitar el conocimiento de un proceso de negocio ayudando a documentarlo, a definirlo y a implementarlo.
- Proveer interoperabilidad de las soluciones
- Resolver la dinámica de los problemas en términos declarativos y cubriendo todas las etapas del ciclo de vida del software.

BPM (*Business Process Management* o Gestión de procesos de negocio) ha adquirido una atención considerable recientemente tanto por las comunidades de administración de negocios como las de ciencia de la computación. Los miembros de estas comunidades están identificados por diferentes soportes e intereses educacionales; entre ellos encontramos los analistas de negocios, que están interesados en mejorar las operaciones de las compañías. Hechos como incrementar la satisfacción de los clientes, reducir los costos de operación, y establecer nuevos productos y servicios a bajo costo, son aspectos importantes del manejo de procesos de negocio desde el punto de vista de un analista del negocio.

La comunidad de software, por su parte, está interesada en proveer sistemas robustos y escalables. Como los procesos de negocios son realizados en espacios de información tecnológi-

ca compleja, la integración de los sistemas de información existentes se convierte en una base importante para la implementación técnica de los procesos de negocio.

BPM se basa en la observación de cada producto que la compañía provee al mercado, lo cual genera como resultado un número de actividades ejecutadas. Los procesos de negocio son la clave para organizar estas actividades, y mejorar el entendimiento de sus interrelaciones.

La tecnología de la información en general y los sistemas de información en particular, merecen un rol importante en BPM, porque cada vez más actividades que realizan las compañías son soportadas por sistemas de información. Las actividades de los procesos pueden ser ejecutadas manualmente por empleados de la compañía, o con la asistencia de aplicaciones dedicadas a tal fin. También puede ocurrir que las actividades sean directamente ejecutadas por sistemas sin intervención humana.

Una compañía puede alcanzar sus objetivos de manera eficiente sólo si su gente y los sistemas de información se conducen en la misma dirección, siendo los procesos de negocio quienes facilitan esta colaboración.

En las compañías suele haber una brecha entre los aspectos organizacionales del negocio y la tecnología de información. Es importante hacer mínima esta brecha porque el mercado suele forzar a dar más y mejores productos a los clientes. Los productos que son exitosos hoy, pueden no serlo mañana. El mercado puede inclinarse hacia quien ofrezca el mejor producto y que sea más barato.

En un nivel organizacional, los procesos de negocio son esenciales para comprender cómo opera una organización. Aunque también son importantes para el diseño e implementación de sistemas de información flexibles. Estos sistemas proveen la base para la creación rápida de nueva funcionalidad que cree nuevos productos, y también para adaptar rápidamente funcionalidad existente a requerimientos del negocio.

BPM es entonces una estrategia para gestionar y mejorar el rendimiento de un negocio optimizando sus procesos a través de la modelización, ejecución y medida de rendimiento dentro de un ciclo de mejora continua.

Es una disciplina que va mucho más allá de la arquitectura de software que la implementa, pero éste es, sin lugar a dudas, un elemento muy importante.

El campo que abarca la gestión de los procesos de negocios incluye dos comunidades: la de la administración del negocio (analistas de negocio) y las de la IT (*Information Technology* o Tecnología de la Información).

Los profesionales del área de negocios tienden a considerar al área de tecnología como subordinados a los aspectos del negocio. Mientras que el área de tecnología considera que los objetivos de negocio y las regulaciones organizacionales no merecen mucho análisis ni un alto nivel de abstracción.

Uno de los principales objetivos que persigue el enfoque de resolución de problemas orientado a procesos de negocios, es conciliar estos puntos de vista y lograr modelar e implementar procesos de negocios correctos y robustos en un software adecuado, contribuyendo a la mejora continua de los procesos y a una ágil absorción de los cambios de la realidad.

Otro de los objetivos del enfoque BPM es aportar una visión transversal de la organización, es decir, si bien las organizaciones se organizan en torno a áreas funcionales, los procesos que la organización ejecuta atraviesan dichas áreas, para lo cual es necesario reconocer las tareas que se realizan, el tiempo que demora cada una y la manera en que se estructuran.

Este enfoque se contrapone al funcional clásico, pero no desconoce la organización por áreas funcionales. La Tabla 1, compara los enfoques entre organizaciones orientadas a funciones y las orientadas a procesos.

Tabla 1. Organizaciones orientadas a funciones Vs. orientadas a procesos

Orientadas a Funciones	Orientadas a Procesos
Los roles y las responsabilidades están alineadas por áreas.	Los roles y las responsabilidades están alineadas por procesos de negocio.
No hay una visibilidad clara del proceso a través de las áreas funcionales.	La visibilidad del proceso de negocio se tiene de principio a fin.
El control de costos se realiza por centros de costos alineados a las áreas funcionales.	Se pierde el valor del proceso al dar más peso a la búsqueda de eficiencia de las áreas funcionales.

En este capítulo se presentan definiciones básicas acerca de los procesos de negocios, su gestión y la manera de modelarlos. También se describe el ciclo de vida de los mismos, mostrando las etapas involucradas cuando se trabaja en el análisis y diseño de problemas con enfoque de procesos de negocios.

1.1 Definiciones y marco conceptual

Los procesos de negocio como concepto constituyen una idea ya existente en las organizaciones y que encontraron en las tecnologías de información, un terreno propicio para su automatización.

Se repasan a continuación una serie de definiciones de diferentes autores acerca de qué representa un proceso de negocio.

Definición 1 - Davenport (1993)

> Un conjunto estructurado y medido de actividades diseñadas para producir una salida específica para un cliente o mercado en particular. Esto implica un fuerte énfasis en cómo se hace el trabajo dentro de una organización, en contraste con el énfasis de un producto enfocado en qué. Un proceso es así un ordenamiento específico de las actividades de trabajo a través del tiempo y el espacio, con un principio y un fin, y entradas y salidas claramente definidas: una estructura para la acción. ... Tomar un enfoque de proceso implica adoptar el punto de vista del cliente. Los procesos son la estructura por la cual una organización hace lo necesario para producir valor para sus clientes [Sheina D.,2008].

Definición 2 – Johansson (1993)

Un conjunto de actividades vinculadas que toman una entrada y la transforman para crear una salida. Idealmente, la transformación que se produce en el proceso debe agregar valor a la entrada y crear un resultado que sea más útil y efectivo para el destinatario ya sea upstream o downstream [Johansson R., 1993].

Definición 3 - Hammer & Champy (1993)

Una colección de actividades que toma uno o más tipos de entrada y crea una salida que es de valor para el cliente [Hammer M. et al, 1993].

Definición 4- Rummler & Brache (1995)

Un proceso de negocios es una serie de pasos diseñados para producir un producto o servicio. La mayoría de los procesos (...) son de función cruzada, abarcando el espacio en blanco entre las casillas del organigrama. Algunos procesos resultan en un producto o servicio que es recibido por el cliente externo de una organización. Llamamos a estos procesos primarios. Otros procesos producen productos que son invisibles para el cliente externo, pero imprescindibles para la gestión efectiva del negocio. Llamamos a estos procesos de apoyo [Rummler, G. A. & Brache, A. P.,1995].

Definición 5 - Weske (2008)

Un proceso de negocio consiste de un conjunto de actividades realizadas en coordinación en un ambiente organizacional y técnico. Estas actividades alcanzan el objetivo del negocio. Cada proceso de negocio representa una única organización, pudiendo interactuar con otras organizaciones [Weske M., 2008].

Definición 6 - Kirchmer, M. (2017)

Un proceso de negocio es un conjunto de funciones, en una secuencia específica, que entrega valor a un cliente interno o externo de una organización [Kirchmer M., 2017].

Definición 7 - Von Rosing, M., Von Scheel, H., & Scheer, A. W. (2014)

Un proceso de negocio es una colección de tareas y actividades (operaciones de negocio y acciones) que consisten de empleados, materiales, máquinas, sistemas y métodos que se estructuran de tal manera para diseñar, crear y entregar un producto o servicio a un cliente [Von Rosing, M., Von Scheel, H., & Scheer, A. W., 2014].

Entre los aspectos más relevantes de estas definiciones, aparece el concepto de **tarea o actividad** y el de estructura o secuencia. Esas tareas o actividades constituyen unidades atómicas e indivisibles que, dependiendo de quién las ejecute serán automáticas o manuales y aquí surge otro elemento del proceso que es el **actor o participante** - quien ejecuta las activi-

dades-. Además, cada una de estas tareas tienen un objetivo y sus acciones modifican un recurso para alcanzarlo.

En la Figura 1, se observa la relación entre los elementos del proceso descriptos.

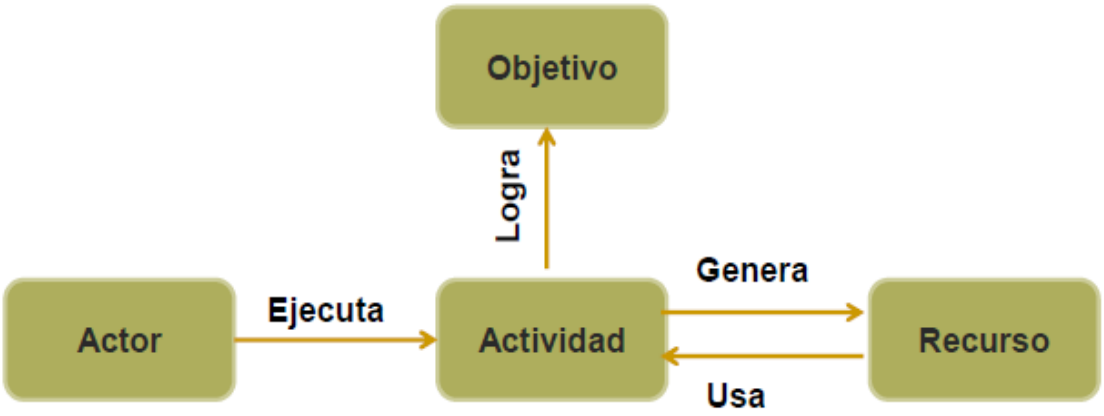


Figura 1. *Elementos de un proceso y su vinculación.*

Los elementos constitutivos de un proceso, por su parte, pueden resumirse en los siguientes:

- Un **mapa de proceso**, que es la representación gráfica de la organización en términos de su funcionamiento.
- Un **diagrama de proceso**, que describe en forma detallada las actividades, eventos secuencias y resultados del mismo.
- Una **instancia de proceso**, que representa un caso concreto de proceso (un diagrama de proceso puede producir varias instancias).

A su vez, los elementos básicos de un diagrama de proceso son: los flujos de entrada y salida, las actividades, los participantes y los puntos de decisión.

Estos elementos tendrán una representación distinta según el método de modelado que se elija para construirlo, tal como se describe en la próxima sección.

1.2 - Modelado de procesos. BPMN

El modelado de procesos consiste en obtener una representación de la realidad de manera abstracta y conceptual, utilizando una determinada simbología.

No hay una única manera de construir modelos y esto también aplica a los modelos de procesos. Los tipos de modelos más clásicos, utilizados para representar procesos de negocio son:

Diagramas de flujo

Los diagramas de flujo datan de los años 60 [Schriber T.J., 1969] y definen de manera gráfica una secuencia lógica de procesos de trabajo. Se trata de un formalismo muy flexible y si bien el estándar ofrece una simbología, quien diseñe el proceso será quien estructure los blo-

ques según el conocimiento que posea. Su característica vertical dificulta su lectura y oculta la visión general del sistema [Aguilar-Saven, 2004].

Un ejemplo de diagrama de flujo puede verse en la Figura 2.

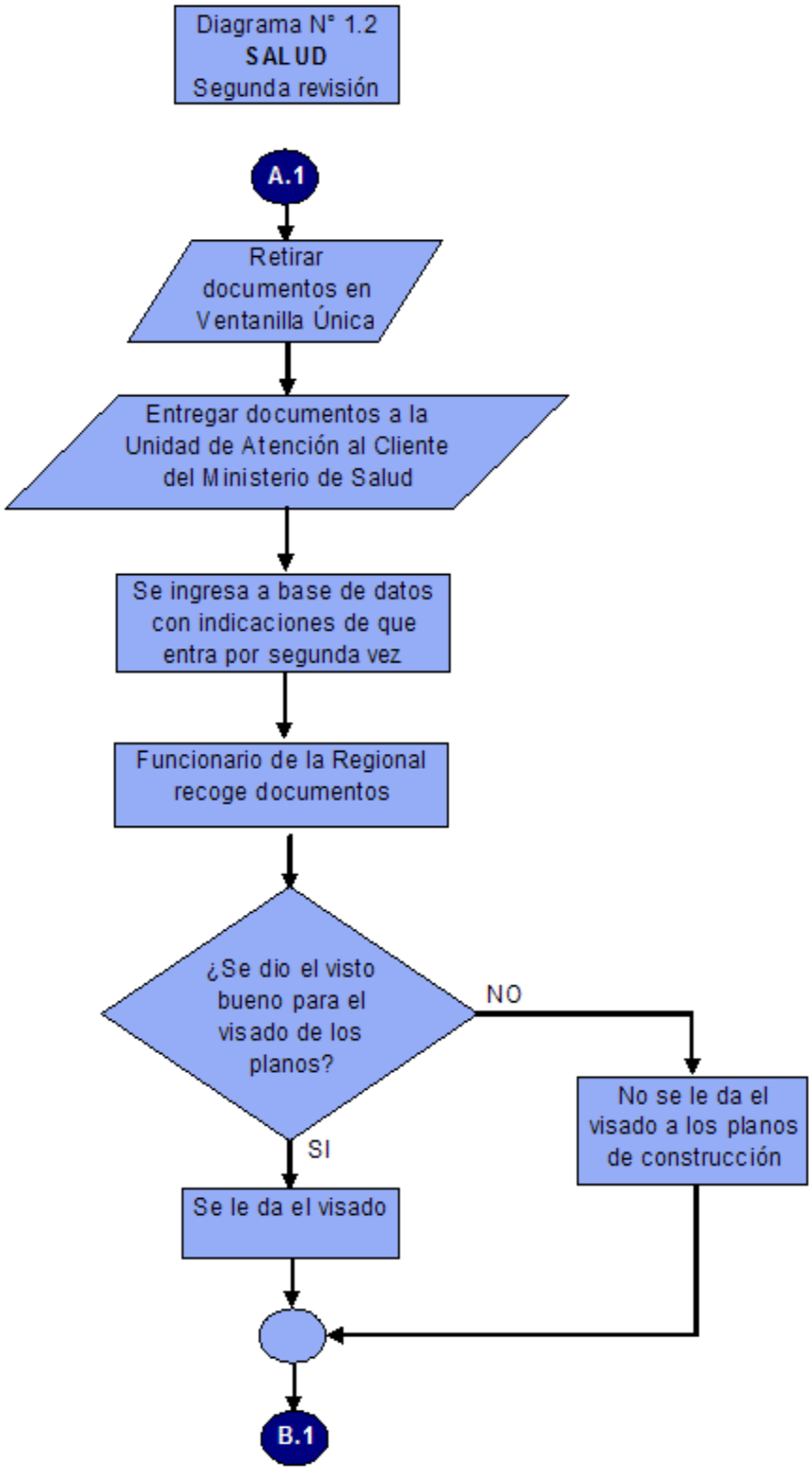


Figura 2. *Ejemplo de Diagrama de Flujo*

Diagramas de flujo horizontal o DFD

Los DFD, son representaciones de información a través de entidades externas, pasos internos de procesado y elementos de almacenamiento de datos de un proceso de negocio [**¡Error! No se encuentra el origen de la referencia.**] [Kettinger et al., 1997]. Estos diagramas permiten ver cómo fluyen los datos a través de la organización, los procesos, así como las transformaciones que sufren dichos datos y los diferentes tipos de salidas, aunque no modela representaciones de flujos de materiales, recursos humanos, y otros elementos relacionados con los procesos de negocio [Yourdon, E et al. 1988].

La Figura 3 muestra un ejemplo de DFD.

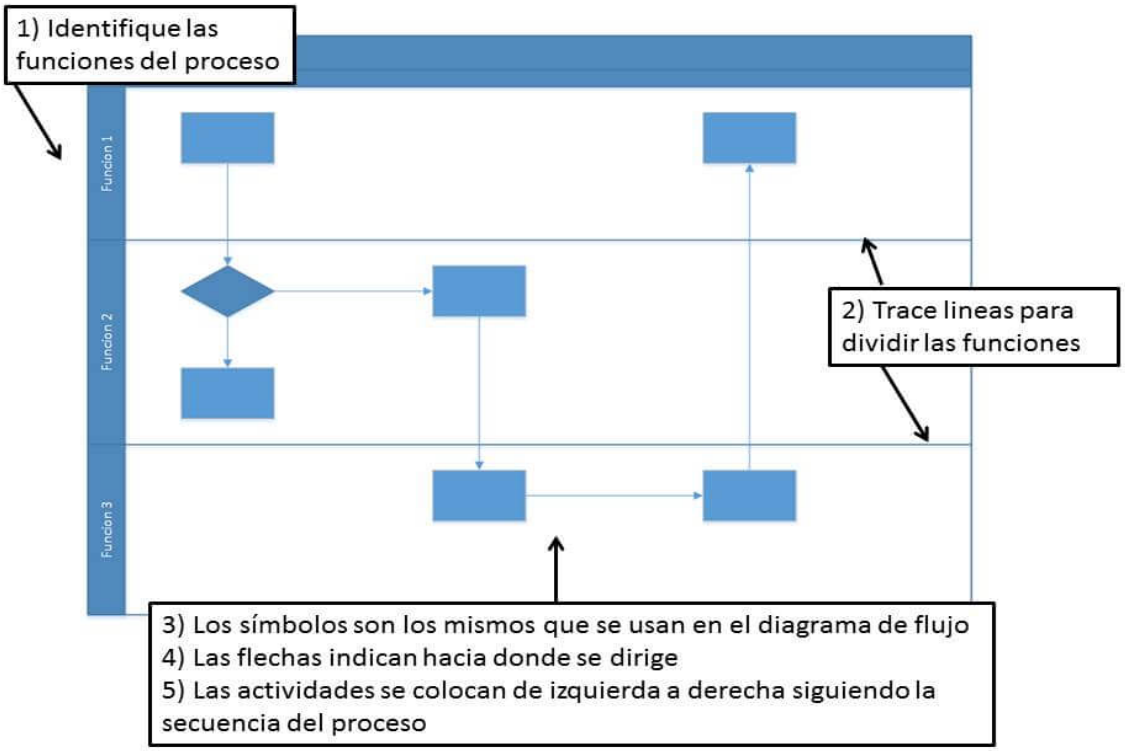


Figura 3. *Ejemplo de DFD*

Diagrama de transición de estados

Se originan para la descripción de la perspectiva dinámica de sistemas dependientes en el tiempo y consiste en círculos que representan los estados, definidos como el modo perceptible de comportamiento de un sistema, y flechas, que representan las transiciones entre estados. Son muy útiles ya que proporcionan información explícita acerca de la secuencia de tiempo relacionado con los diferentes eventos dentro del sistema. Las limitaciones las presenta en la descripción de la colaboración entre los objetos que causan dichas transiciones.

La Figura 4, muestra un ejemplo de diagrama de transición de estados de una central telefónica.

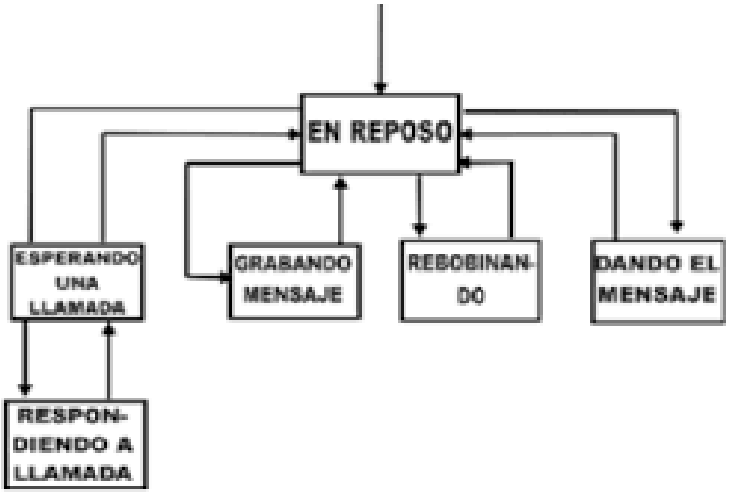


Figura 4. *Ejemplo de diagrama de transición de estados.*

IDEF Integrated Definition for Function Modelling

Es una familia de técnicas de modelado, que ofrecen una perspectiva integrada para representar y modelar procesos y estructuras de datos. Sus inicios se remontan a la necesidad de las Fuerzas Armadas Estadounidenses por mejorar sus operaciones de producción, iniciándose así el programa ICAM (*Integrated Computer-Aided Manufacturing*). La familia IDEF, consiste en un gran número de técnicas, entre las cuales se destaca IDEF0 e IDEF3, que son aquellas relacionadas con los procesos de negocio, aunque existen otras versiones como IDEF1, IDEF1X, IDEF2, IDEF4 e IDEF5.

La técnica IDEF0, está diseñada para modelar las decisiones, acciones y actividades de una organización u otro sistema, y representa la perspectiva funcional de modelado, es decir, el *qué* [Chedrese et al., 2014]. Es considerada una técnica sencilla pero poderosa, ampliamente usada en la industria durante la etapa de análisis en la reingeniería de procesos. Permite identificar apropiadamente los procesos y sus interfases así como elaborar los documentos que permitan su control en cualquiera de sus etapas de desarrollo. IDEF0 utiliza solo un tipo de anotación en sus representaciones gráficas conocido como ICOM (*Input-Control-Output-Mechanism*). La representación estática de sus diagramas no permite visualizar las perspectivas de modelado de comportamiento o informacional. Para vencer dichas limitaciones, se desarrolló IDEF3 (*Process Description Capture*), que describe a los procesos como secuencias ordenadas de hechos o actividades, representando el *cómo*, y mostrando la visión dinámica o de comportamiento.

La Figura 5 muestra un ejemplo de gestión de un departamento de ventas en IDEF0.

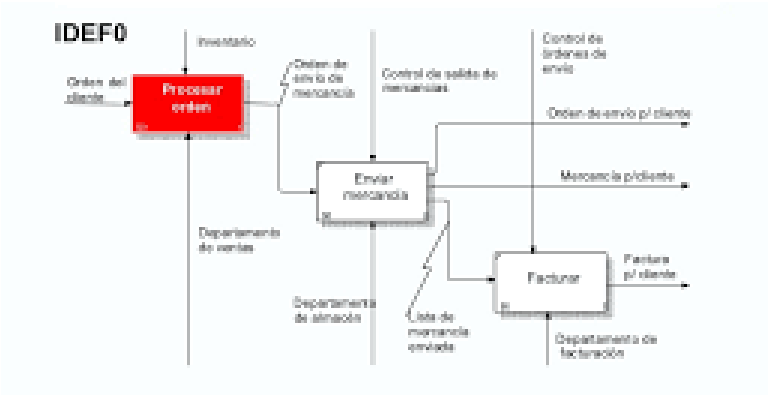


Figura 5. *Ejemplo IDEF0*

Diagramas de Actividad de UML

En UML un diagrama de actividades se usa para mostrar la secuencia de actividades. Los diagramas de actividades muestran el flujo de trabajo desde el punto de inicio hasta el punto final detallando muchas de las rutas de decisiones que existen en el progreso de eventos contenidos en la actividad. Estos también pueden usarse para detallar situaciones donde el proceso paralelo puede ocurrir en la ejecución de algunas actividades. Los Diagramas de Actividades son útiles para el Modelado de Negocios donde se usan para detallar el proceso involucrado en las actividades de negocio.

La Figura 6 presenta un ejemplo de diagrama de actividad para la compra de productos.

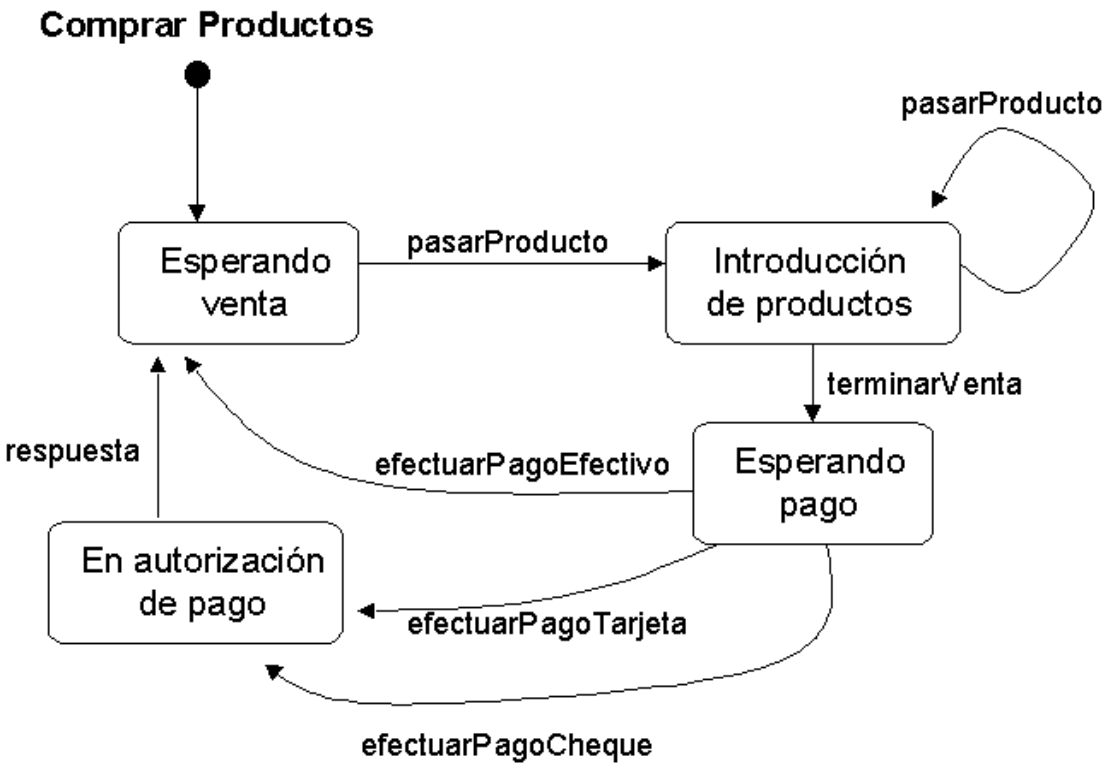


Figura 6. *Ejemplo de diagrama de actividad*

BPMN (Business Process Model and Notation)

El objetivo de BPMN es proveer una notación que sea legible y entendible para todos los usuarios de negocios, desde los analistas que realizan el diseño inicial de los procesos y los responsables de desarrollar la tecnología que ejecutará estos procesos, hasta los gerentes de negocios encargados de administrar y realizar el monitoreo de los procesos.

BPMN soporta un modelo interno que permite generar ejecutables creando un puente estandarizado para cubrir el hueco provocado por las diferencias entre el diseño de los procesos de negocios y su implementación. Se trata del lenguaje BPEL que será desarrollado en próximos capítulos.

BPMN define un diagrama de procesos de negocio (BPD) basado en una técnica adaptada de diagramas de flujo para la creación de modelos gráficos de operaciones de procesos de negocio. Un modelo de procesos de negocio, es una red de objetos gráficos que representan las actividades (por ejemplo, tareas) y los controles de flujo que definen su orden de ejecución [BPMN, 2019].

La Figura 7 muestra un ejemplo de diagrama de procesos de negocio en BPMN para la incorporación de personal en una organización.

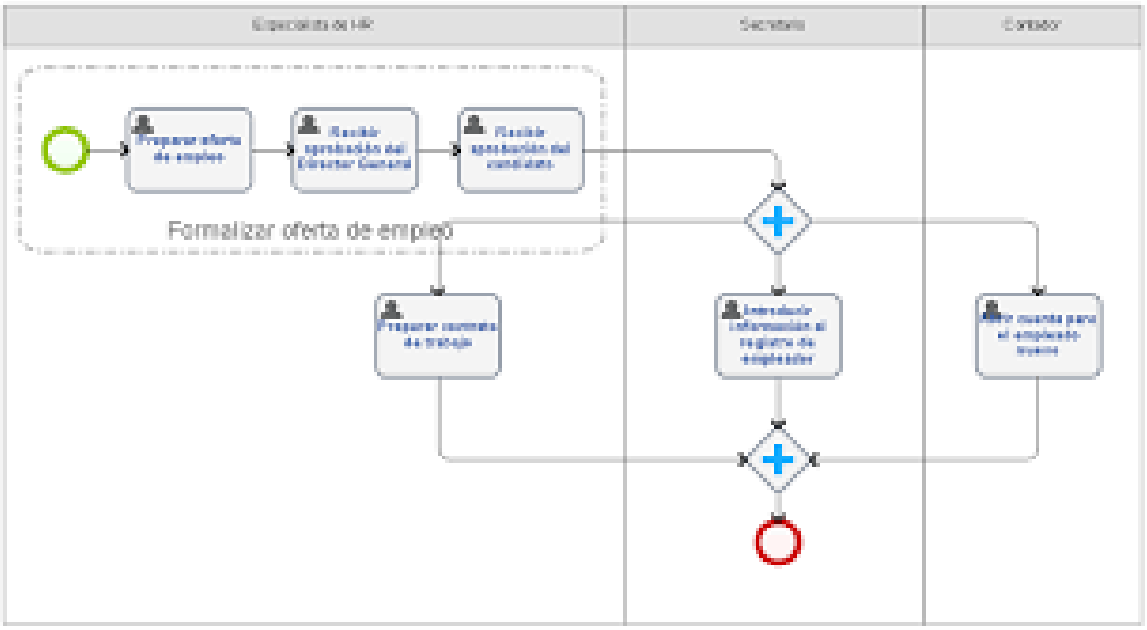


Figura 7. *Ejemplo de diagrama en BPMN.*

Tanto los Diagramas de Flujo como los Diagramas de Flujo Horizontal, constituyen representaciones sencillas y fáciles de utilizar por su acotada cantidad de símbolos, pero por esa misma razón se tornaron insuficientes para manejar la complejidad de algunos procesos de negocio.

En este sentido, y ante el surgimiento de UML [UML, 2019] con sus diferentes artefactos, se dio paso a la construcción de modelos de procesos un poco más complejos para luego llegar al BPMN [BPMN, 2019] como el estándar “de facto” del paradigma BPM.

El advenimiento de BPMN, BPMS y sus lenguajes de ejecución no deja obsoleta la necesidad de desarrollos de sistemas, como los que se logran utilizando UML (*Unified Modeling Language*). Los desarrollos de sistemas siguen teniendo un rol importante en la arquitectura de procesos en el ámbito empresarial.

UML es un lenguaje que facilita a los desarrolladores la especificación, visualización y documentación de modelos de sistemas de software. Está dirigido en líneas generales a los arquitectos de software e ingenieros de software. Fue desarrollado como un medio para mejorar el proceso de desarrollo de software, desde el diseño de la arquitectura hasta la implementación de la aplicación, para ser utilizado por personas con conocimientos técnicos (analistas de sistemas y programadores) [UML, 2019]

BPMN está dirigido a los analistas de negocio, arquitectos de sistemas e ingenieros de software. Fue desarrollado para mejorar todo el ciclo de vida del desarrollo de procesos desde el diseño de los mismos.

Por su parte, UML es un lenguaje desconocido para la mayoría de los analistas de negocio.

UML define un número de diagramas que se pueden clasificar en las siguientes categorías: estructura estática de la aplicación, comportamiento dinámico y administración y organización de soluciones de software.

De estas tres categorías, el comportamiento dinámico es el utilizado para modelar los procesos de negocio; los diagramas asociados son el de actividad UML y los de casos de uso. BPMN está emparentado con UML por el hecho que ambos definen una notación gráfica para los procesos de negocio similar a los diagramas de comportamiento de UML. Sin embargo, BPMN y UML usan enfoques muy diferentes para modelar procesos de negocio.

Si bien los diagramas de actividad constituyen la herramienta UML para modelar actividades de procesos, UML, en general, ofrece un enfoque orientado a objetos para modelar aplicaciones. Mientras que BPMN toma un enfoque centrado en los procesos. Este enfoque es mucho más natural e intuitivo para los analistas de negocios. Con BPMN, el control y los mensajes de flujo entre procesos son primeramente modelados. Luego, se definen implícitamente los modelos de objetos para los procesos en vez de hacerse explícitamente como en UML. BPMN también ofrece la opción de explicitar el modelado de objetos de negocio que pueden ser expuestos a través de servicios de negocio en el flujo del proceso.

Elementos de la notación BPMN

Un diagrama de proceso de negocio está compuesto de un conjunto de elementos gráficos. Los elementos utilizados para construir los diagramas fueron elegidos para ser distinguibles unos de otros y utilizar las figuras que son familiares a la mayoría de los diseñadores. Por ejemplo, las actividades se representan mediante rectángulos y las decisiones mediante rombos. Cabe destacar que uno de los objetivos del desarrollo de BPMN fue crear un mecanismo sencillo para la creación de modelos de procesos de negocio y al mismo tiempo ser capaz de manejar la complejidad inherente de los mismos. El enfoque adoptado para manejar estos dos requisitos fue la organización de la gráfica de los aspectos de la notación en

categorías específicas. Éste provee un pequeño conjunto de categorías de notación que permite al lector del diagrama de procesos de negocio reconocer fácilmente los elementos básicos y comprender el diagrama.

Dentro de las categorías básicas de elementos se pueden incluir variaciones adicionales o información para soportar requerimientos complejos sin tener un cambio drástico en la mirada y sentido básico del diagrama.

Las cuatro categorías básicas de elementos son las que se enumeran y se encuentran representadas en la Figura 8.

Diagramas BPM

La notación BPM permite construir diagramas fáciles de leer y que además manejen la complejidad traduciendo el diagrama en algún lenguaje de ejecución.

Para modelar un flujo sólo se modelan los eventos que ocurren. Las decisiones entre flujos se modelan con *gateways*.

El modelado está centrado en flujos y eventos, pero principalmente en eventos, que son los elementos disparadores de determinadas situaciones.

Un proceso en el flujo puede contener subprocesos que cuando son atómicos se denominan tareas.

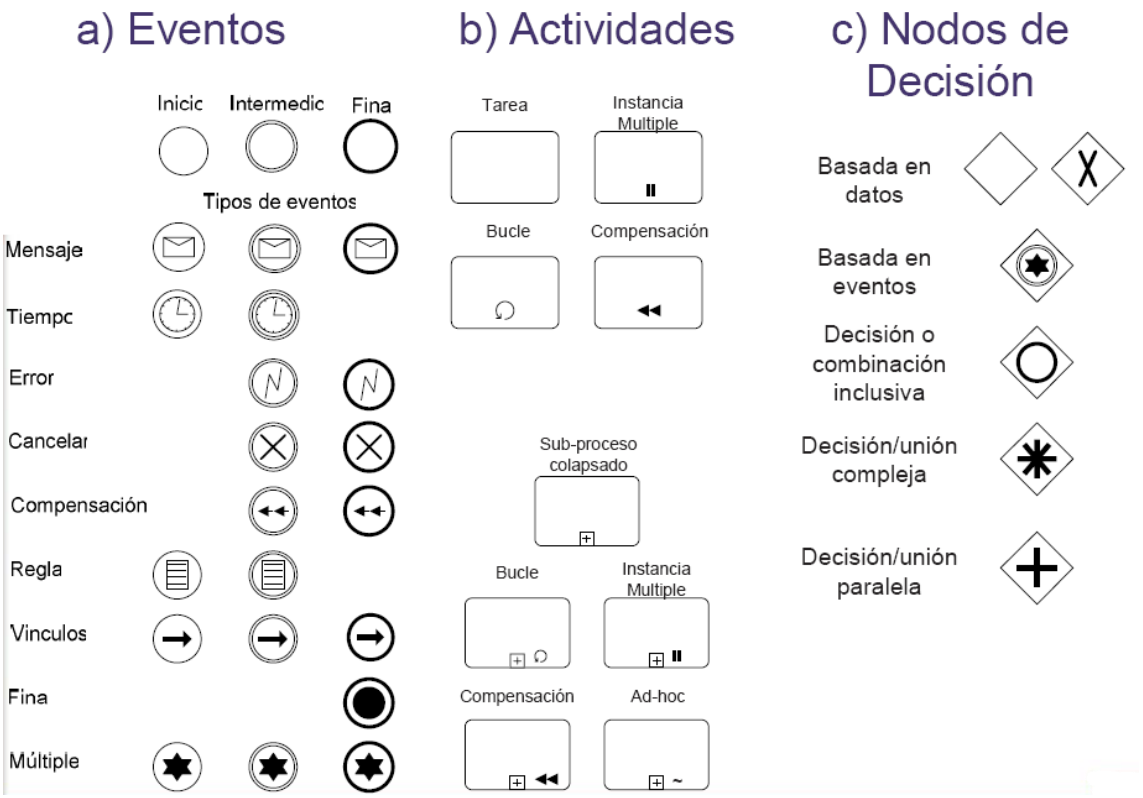


Figura 8. *Elementos Notacionales de BPMN*

Los eventos se dividen en iniciales, intermedios y finales según se desencadene al inicio, durante o al finalizar el flujo del proceso.

Además, se puede especificar "quién hace qué", ubicando los procesos en *pools* que denotan quién hace la tarea pudiendo particionar el *pool* en *lanes* o senderos. Típicamente un pool representa a toda la organización mientras que el *lane* representa un departamento dentro de la misma. Puede extrapolarse esto mismo a funciones, aplicaciones y sistemas.

Para mejorar el poder expresivo, a estos tres tipos de eventos básicos se los divide a su vez en eventos más complejos en función de la tarea que realizan. Esta sub-clasificación agrega restricciones al modelo. Ejemplo: un evento "*timer*" nunca finaliza un flujo.

Las Figuras 9 y 10 muestran el grado de detalle que puede obtenerse a través de un diagrama BPM en función del nivel de abstracción que los modeladores deseen reflejar.

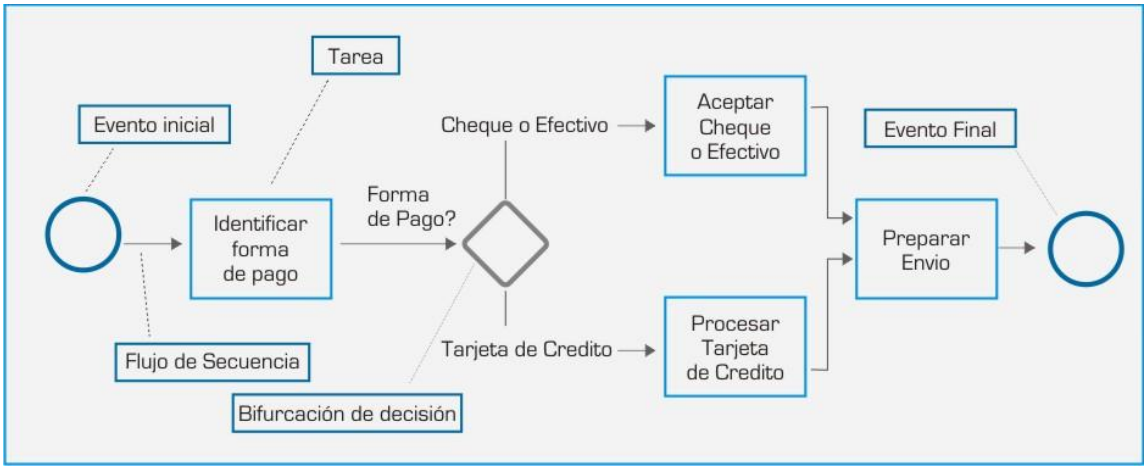


Figura 9. *Ejemplo de un proceso simple*

El proceso graficado en la Figura 9 representa el proceso de *recepción del pago de un cliente*. El proceso se inicia con la actividad de *Identificar la forma de pago*. Se prevén dos posibles formas de pago: en efectivo o con tarjeta de crédito. En cada caso se aplica la actividad de aceptar el pago según la forma del mismo y se pasa a la actividad de empaque de la mercadería, finalizando así el proceso.

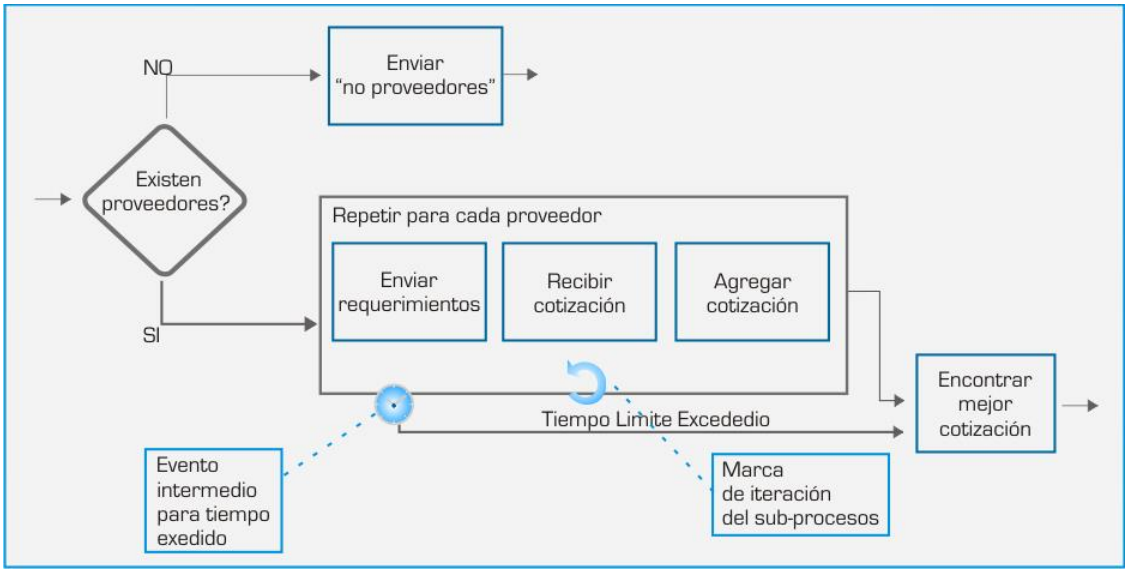


Figura 10. *Segmento de un proceso con detalle*

En la Figura 10, se representa una porción del proceso de "*evaluación de presupuestos"*, mostrando el lazo iterativo de análisis de cada proveedor. Como se ve, se detallan las sub-actividades consideradas dentro del proceso de *"análisis del proveedor".* Además, intervienen eventos intermedios que consideran el tiempo dentro de dicho proceso.

1.3 - Ciclo de vida de los procesos de negocio - BPM

La Gestión de Procesos de Negocio, en inglés, *Business Process Management* - BPM, incluye conceptos, métodos y técnicas para soportar el diseño, administración, configuración, representación y análisis de los procesos de negocio. Su objetivo es representar el proceso de negocio con sus actividades y las restricciones de ejecución entre ellas.

Además, el objetivo de BPM es mejorar los procesos de negocio y asegurar que se realicen, de la manera más eficiente y eficaz, todas las actividades críticas que afectan la satisfacción del cliente. Puede implicar pequeños pasos de mejora y continuo aprendizaje de mejores prácticas, o un rediseño radical de los procesos del negocio con el fin de lograr un mejor rendimiento [Zairi, M., & Sinclair, D., 1995]

En este sentido, los procesos de negocio, como concepto, poseen un ciclo de aparición, desarrollo y finalización, conocido como ciclo de vida.

El ciclo de vida de los procesos de negocio remite al conjunto de etapas o fases que atraviesa un proceso de negocio desde su definición hasta su finalización, con la particularidad que se trata de un ciclo cerrado, donde no existe un punto de finalización concreto, sino que conforma un ciclo de mejora continua. En este sentido, las fases son modelado (aparición del proceso), definición y despliegue (desarrollo del proceso) y monitoreo para volver a analizar el modelo (no existe finalización).

El diagrama de la Figura 11 representa un posible ciclo de vida, centrado en el concepto de mejora continua.

Figura 11. *Diagrama de ciclo de vida de los procesos de negocio centrado en mejora continua*

- **Definición.** Decidir su cometido, describir en forma gráfica el funcionamiento y evaluar el mismo antes de su implementación.
- **Medida.** Obtener información de los indicadores durante la ejecución.
- **Análisis**. Analizar, en función de los indicadores, el comportamiento del proceso según sus objetivos.
- **Control**. Controlar la ejecución interviniendo para asegurar eficiencia y eficacia.
- **Mejora**. Mejorar los procesos según sus rendimientos reales.

Por otra parte, Weske, propone un ciclo de vida centrado en la administración del proceso sin perder de vista la idea circular del mismo, tal como se observa en la Figura 12. De esta manera, sigue presente el concepto de mejora continua, pero otorga un rol importante a la administración como fase vinculada al software que soporta el ciclo de vida.

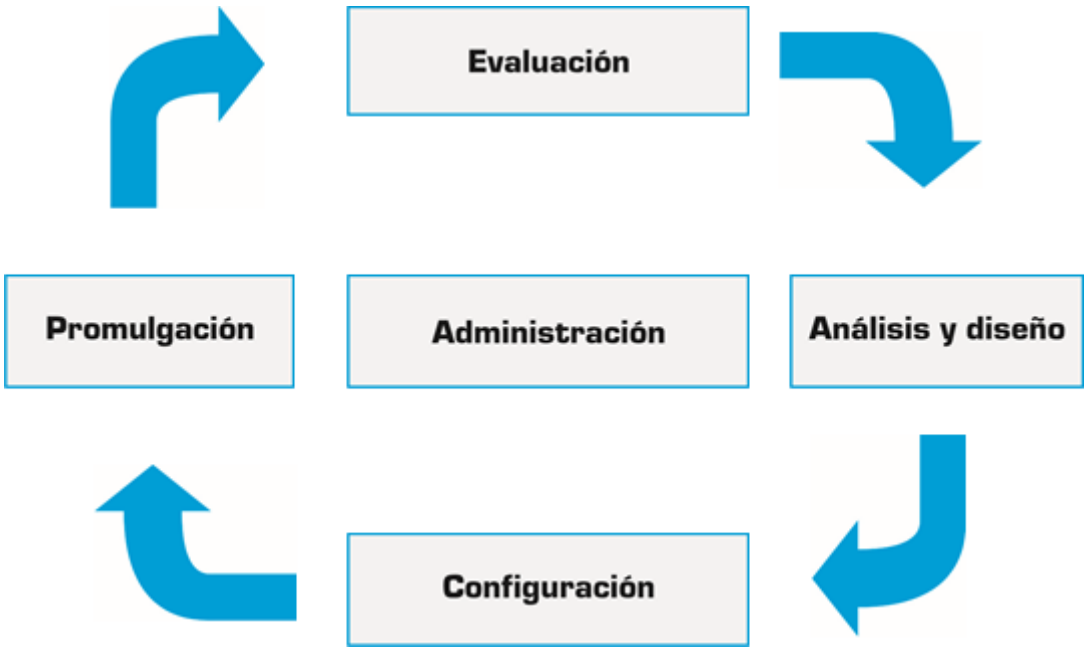


Figura 12. *Diagrama de ciclo de vida de los procesos de negocio centrado en la administración de procesos.*

- **Análisis y Diseño.** Estudio de la situación para identificar, revisar validar y representar los procesos de negocios en un modelo. Sobre la base de este modelo se valida, se simula y se verifica el proceso, siendo éstas las actividades de análisis.
- **Configuración.** Implementación de las políticas y reglas de negocio
- **Promulgación.** Consiste en "publicar" el proceso. Las instancias generadas con cada modelo de un proceso de negocio, se ejecutan de una manera única y repetitiva cumpliendo siempre las mismas restricciones y ejecutando la misma lista de actividades. El BPMS controla y monitorea la ejecución de cada instancia de proceso de negocio. Esta etapa requerirá de un BPMS necesariamente.
- **Evaluación.** Requiere disponer de los registros de ejecución de dichos procesos y la posibilidad de evaluar y simular cambios.

- **Administración.** Fase permanente, por lo tanto, se visualiza en el centro del ciclo ya que posee una mirada directa a cada etapa y su complejidad variará en función del soporte de software con que se cuente, el número de procesos que se modelen y las características propias de la organización.

1.4 - Clasificación de procesos de negocio

Los procesos de negocio pueden clasificarse según diversas taxonomías y criterios, propuestos por distintos autores. En particular Weske plantea a los siguientes criterios:

Según el nivel de granularidad

Desde este punto de vista, los procesos de negocios pueden calificarse como **organizacionales**, cuando describen en el ámbito global los procesos de la organización y marcan o delinean grandes objetivos, en contraposición con los procesos **operacionales** que presentan un mayor nivel de detalle y suelen concluir en un modelo completo del proceso de negocio.

Claramente los procesos organizacionales representan el primer nivel de abstracción posible en el análisis y los procesos operacionales son la explotación del nivel anterior.

La aplicación de metodologías iterativas y evolutivas sobre estos dos tipos de procesos constituye un modelo de procesos de negocio altamente detallado**.**

- *Procesos operacionales,* son aquellos que conducen directamente al cumplimiento de los objetivos o finalidad de la organización, dando por resultado un producto (bien o servicio) que es recibido por un cliente/ciudadano externo
- *Procesos organizacionales.* buscan monitorear y evaluar los avances y la eficiencia de la organización, vista como un conjunto de procesos interrelacionados para lograr un determinado objetivo. Facilitan y apoyan los procesos relacionados directamente con los clientes/ciudadanos. A diferencia de los procesos de apoyo, no están vinculados a la sustentabilidad operativa.

Según el alcance corporativo

Este aspecto permite clasificar a los procesos de negocios según se circunscriben a la organización en sí misma, o la trascienden hacia otras organizaciones.

Esta clasificación identifica procesos intraorganizacionales e interorganizacionales, marcando la diferencia existente entre orquestación y coreografía de procesos, como se describe en la próxima Sección.

Los procesos interorganizacionales son soportados generalmente por sistemas de gestión de workflow en su versión tradicional o, en versiones más modernas, implementados o desplegados como un conjunto de servicios ejecutados bajo un motor de orquestación.

Los procesos interorganizacionales requieren una coreografía de procesos donde se requiere establecer contratos entre las partes que interactúan.

En el caso en que se deba interactuar con otras organizaciones, estamos en presencia de coreografía de procesos donde se requiere establecer contratos con las partes con las que se interactúa.

Según el grado de automatización

El grado de automatización de un proceso de negocio permitiría clasificarlos en totalmente automatizados, parcialmente automatizados o manuales.

Este aspecto también marca el grado de interacción humana que requiere la promulgación del proceso.

Los Trabajadores del Conocimiento (actores involucrados en la etapa de administración) permiten marcar claramente el próximo paso a seguir para llevar a cabo un proceso, por lo tanto, a la hora de construir un software es el indicado para determinar el flujo de la interacción con el usuario.

Según el grado de repetición

Este aspecto permite tener una medida temprana del ROI (*Return Of Investment*) de la aplicación de metodologías con enfoque en los procesos de negocio.

Cuando el grado de repetición es alto, la inversión hecha en su modelización y promulgación está justificada ya que habrá muchas instancias que cumplen el mismo modelo.

En el caso en que no exista un alto grado de repetición, como puede suceder con procesos como el diseño de un avión, se duda acerca de la justificación de la inversión. En estos casos se puede poner el foco en modelizar la interacción entre personas mediante procesos de negocio colaborativos, donde el objetivo de modelar y promulgar no está en la eficiencia sino en obtener una traza de su ejecución para analizar los datos arrojados por la misma.

Según el grado de estructuración

Un proceso de negocio estructurado es el que prescribe las actividades a realizar y las restricciones de ejecución de una única manera. Las decisiones que se toman durante la promulgación del proceso fueron tomadas en tiempo de diseño. Los workflow de producción son un ejemplo de tales procesos.

Los procesos estructurados no permiten saltear actividades no requeridas o ejecutar concurrentemente actividades definidas como secuenciales.

Para dar soporte a estas ideas surge el concepto de actividades ad-hoc, donde el Trabajador del Conocimiento decide el orden y el momento de su ejecución dentro de un proceso. Es en estos casos que adquiere mayor relevancia el uso de BPMS, sobre todo aquellos que incluyen herramientas para realizar monitoreo de procesos (BAM *Business Activity Monitoring*).

1.5- Orquestación y coreografía de procesos de negocio

Los procesos de negocio atraviesan la estructura organizativa y definen sus reglas independientemente del proceso.

Los servicios resuelven funcionalidades concretas requeridas dentro de cada unidad organizativa y se componen para realizar los procesos de negocio a través de su orquestación y coreografía.

En la Tabla 2 se presenta una comparación entre ambos conceptos fijando como patrones a contrastar, el objetivo de cada uno, el modelo o metáfora que siguen, el enfoque que se le da y el fundamento para su uso.

Tabla 2. Comparación de Orquestación Vs. Coreografía

	Orquestación	**Coreografía**
Objetivo	Componer servicios para cumplir con un proceso de negocio dentro de una organización	Componer servicios para colaboración entre organizaciones
Modelo	Jerárquico. Pregunta-Respuesta	Peer – to –Peer
Enfoque	Componer servicios y el orden en que son ejecutados para alcanzar el objetivo de un proceso de negocio	Definir la manera en que múltiples partes colaboran para conformar una transacción de negocio
Fundamento	Constituye un servicio en sí mismo	Define la interacción del negocio

La orientación a procesos implica independizarse de la estructura organizativa, pensar las actividades según la manera en que se ejecutan en lugar de dónde se realizan.

Los servicios resuelven aspectos funcionales directamente vinculados a la ubicación de la unidad funcional dentro de la estructura.

Una buena resolución de procesos garantiza una buena solución orientada a servicios y no viceversa.

La noción de una aplicación o servicio compuesto se basa en la idea de la construcción de nuevas aplicaciones o servicios, interconectando las partes existentes. La orquestación juega un papel importante en esto, ya que es quien aglutina estas partes al coordinar la ejecución de cada servicio discreto.

La orquestación resuelve el problema de la ejecución de la aplicación de forma centralizada. En ella debe existir un mecanismo que dirige las actividades. Estas actividades son en realidad interacciones entre servicios, es decir, servicios que se invocan unos a otros, pero no de forma desordenada, sino de manera controlada por el orquestador que es quien conoce el detalle de todas las tareas que se deben llevar a cabo para completar el proceso.

La construcción del proceso de negocio se realiza en dos pasos: primero se publican los servicios y luego se orquestan, es decir, se integra cada servicio al proceso en su lugar y momento adecuado.

En la orquestación de servicios hay varios actores involucrados. Entre ellos encontramos la especificación del proceso de negocio, un motor de ejecución de procesos que contiene los procesos de negocios y sus reglas, y los consumidores de los servicios que se exponen.

A diferencia de la orquestación, la coreografía plantea un esquema en donde no hay un control centralizado del proceso, sino un control "declarativo" que sólo especifica cuáles son las interacciones permitidas entre dos pares. De esta forma, dadas las reglas correctas, las partes interactúan unas con otras en un estilo "peer-to-peer" y el proceso de negocio estará definido de forma implícita. De ahí su nombre (coreografía), ya que se asemeja a un estilo en donde cada parte hace su trabajo bajo ciertas reglas y se obtiene un resultado final conjunto.

Para implementar coreografías se puede usar BPEL aun cuando éste está pensado para orquestación. La diferencia reside en que se especifica una serie de procesos entre cada par que interactúa y cada uno de estos procesos especificados representa la interacción válida entre dichos pares [BPEL Tutorial, 2020].

Los estándares para orquestación de procesos incluyen:

- WSBPEL: cada proceso WSBPEL se expone como un Web Service usando WSDL que describe la entrada de datos y los puntos de salida del proceso.
- BPEL4People: es una extensión del estándar WSBPEL que inserta tareas humanas en la orquestación.
- BPMN: una notación visual para modelar procesos. Fue diseñado para ilustrar los procesos y mapearlos a los lenguajes de ejecución como BPEL

El estándar número uno para ejecutar procesos de negocios y controlar en forma centralizada (orquestar) servicios, es BPEL (*Business Process Execution Language*). BPEL es un lenguaje ejecutable que especifica la interacción entre Web Service. El estándar fue construido por un comité y hoy es mantenido por OASIS. Dicho comité se planteó ciertos objetivos, tales como usar Web Service, usar XML, poder administrar el ciclo de vida del proceso y poder manejar transacciones a largo plazo [Juric M. et al.,2007].

1.6 - Conclusiones del Capítulo

Los procesos de negocios trascienden la estructura organizativa y la atraviesan. Son las actividades que se realizan a través de las unidades organizativas de la empresa y comandan los procesos de cada área cuidando que éstas hagan lo que les corresponde.

El enfoque orientado a procesos de negocios implica un fuerte énfasis en "cómo" el trabajo es realizado dentro de una organización o entre organizaciones en contraste al acento en el "qué", en el enfoque orientado a producto.

Las actividades de los procesos son responsabilidad de personas o áreas de empresas, incluso si son automatizadas.

Las políticas empresariales y las reglas de negocio se establecen para determinar cómo debe actuar la empresa para cumplir sus objetivos, respondiendo a estrategias preestablecidas. Estas reglas son de aplicación en los distintos pasos de un proceso.

La definición de un proceso conlleva a identificar indicadores que puedan medir su rendimiento, favoreciendo el análisis y la mejora continua de dicho proceso.

Los objetivos de la gestión de los procesos de negocio pueden resumirse en los siguientes puntos:

- Explicitar los procesos de negocios para entender las operaciones realizadas en una organización y sus relaciones.
- Identificar actividades y representarlas en un modelo de procesos de negocio como medio de comunicación entre los participantes, además de permitir analizar el modelo y mejorarlo.
- Facilitar la mejora continua de los procesos de negocio, cuyo repositorio constituye un activo para la organización y su gestión.
- Reducir la brecha existente entre el proceso de negocio en sí mismo y su realización en un software.

CAPÍTULO 2
Conceptos de servicios en los lenguajes de programación

Dalila Romero y Patricia Bazán

El concepto de servicios y su uso dentro de los lenguajes de programación conlleva analizar la evolución tecnológica desde los objetos distribuidos, sus verdaderos antecesores. Se ha introducido este concepto en [Bazán P, 2017] y analizado los objetos en el marco de los sistemas distribuidos.

En este capítulo se amplían algunos de los conceptos allí planteados; se analizan en detalle los estándares y tecnologías habilitantes que precedieron a los servicios y de qué manera estos cobran protagonismo en los lenguajes de programación como piezas constructivas de software.

A medida que fue aumentando la complejidad de las soluciones distribuidas fue necesario incorporar conceptos, metodologías y buenas prácticas en la construcción de software y llevarlas al entorno distribuido.

El concepto de objeto en el sentido clásico de la Programación Orientada a Objetos y que tiene sus raíces principales en el lenguaje Smalltalk, se mantiene a nivel de diseño de aplicaciones, pero cambia algunas características cuando se lo lleva a un entorno distribuido.

Un objeto distribuido es un objeto (componente funcional con estado interno, comportamiento, heredable y encapsulado), que por estar en un entorno distribuido también debe ser:

- Transaccional: los cambios de estado que produce se deben ejecutar completamente o no ejecutarse.
- Seguro: debe evitar vulnerabilidades que corrompa el estado del sistema.
- Lockeable: debe fijar un cerramiento que garantice su ejecución correcta ante accesos concurrentes.
- Persistente: su estado interno debe conservarse más allá de su ciclo de vida.

Esta definición de objeto distribuido nos conduce a la necesidad de contar con una infraestructura que facilite la comunicación entre este tipo de componentes, logrando a su vez que logren esta comunicación a través de distintas plataformas de ejecución, sistemas operativos y lenguajes de programación.

Así surge el estándar CORBA (*Common Object Request Broker Architecture*)[1], definido por la OMG (*Object Management Group*) y que surge de la iniciativa de un consorcio de empresas interesadas en establecer una arquitectura interoperable sobre la base del paradigma orientado a objetos. CORBA se constituye así en un ejemplo de sistema distribuido basado en objetos.

Entre los elementos distintivos del estándar - y que se puede decir que sentaron las bases para lo que se conoce actualmente como web services - encontramos:

- Un lenguaje de especificación de interface, IDL *Interface Definition Language*, que define los límites de los componentes o sea las interfaces contractuales con los potenciales clientes.
- Un bus de objetos u ORB (*Object Request Broker*) que comunicar objetos independientes de su locación. El cliente se despreocupa de los mecanismos de comunicación con los que se activan o almacenan los objetos servidores.

Provee un vasto conjunto de servicios de middleware distribuido y tiene claramente un mecanismo de comunicación mucho más complicado que los clásicos RPC (*Remote Procedure Call*) y MOM (*Message Oriented Middleware*).

La Figura 13 muestra el mecanismo de interoperabilidad entre cliente y servidor a través de un bus de objetos. Por ejemplo, un componente escrito en C, expone sus interfases en IDL y se comunica con el cliente que genera un requerimiento al ORB. Por su parte, del lado del servidor, puede haber un componente escrito en Java que también expone sus interfases y se comunica con el ORB como el proveedor de un servicio. El ORB es un middleware que vincula cliente con servidor.

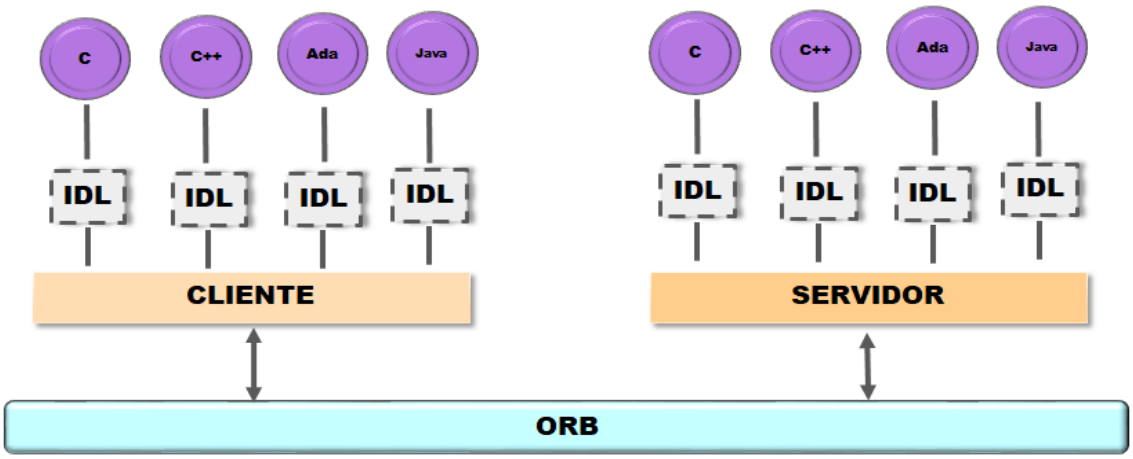


Figura 13. *ORB e interoperabilidad cliente/servidor*

[1] http://www.corba.org/

2.1 - CORBA - Arquitectura de Referencia

Hacia mediados de 1990 surge la arquitectura de referencia CORBA para gestionar objetos en un entorno distribuido. Los 4 elementos de la arquitectura incluyen: 1- ORB que define el bus de objetos de CORBA, 2- *Common Object Services* define los objetos a nivel sistema operativo que extienden el bus, 3- *Common Facilities* define aplicaciones horizontales y verticales usadas de manera directa por los objetos de negocio, 4- *Application Objects*, son los objetos de negocio y aplicaciones que usan la infraestructura.

La Figura 14 muestra gráficamente la arquitectura descripta.

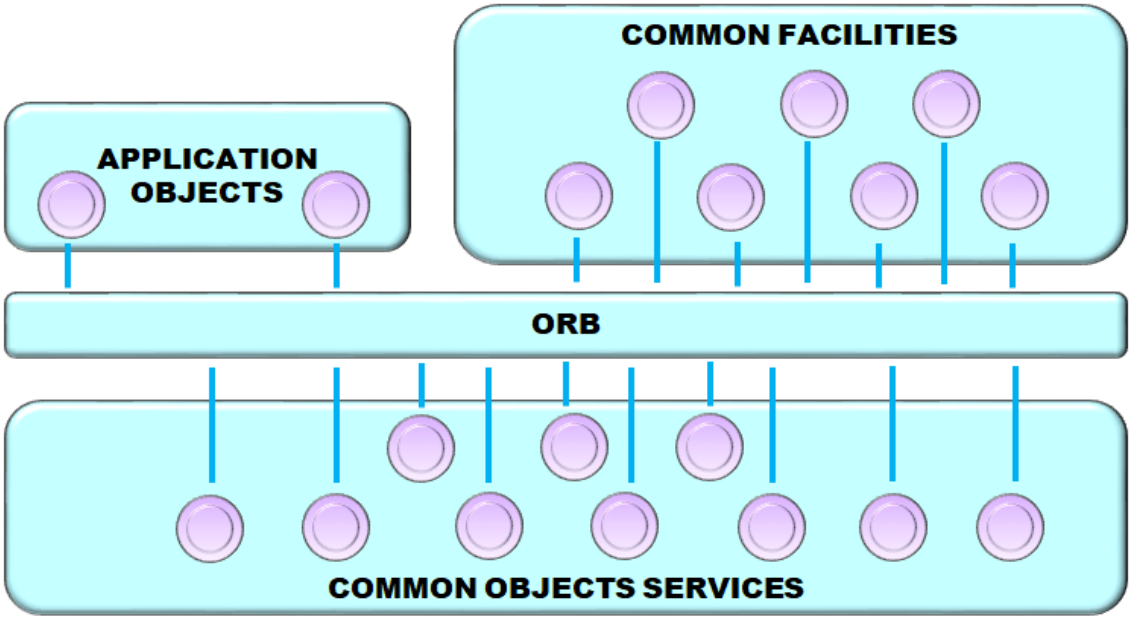


Figura 14. *Arquitectura de Gestión de Objetos Distribuidos*

El ORB en particular constituye un middleware basado en mecanismos de invocación remota similares al RPC. La diferencia entre ellos es equivalente a la diferencia entre el mecanismo *call-return* y *request-reply* específicos de la vinculación programa-subrutina y objeto-mensaje del paradigma procedural y orientado a objetos, respectivamente. En la Figura 15 se observa la diferencia entre la invocación a una función específica con RPC versus la invocación a un método de un objeto específico. En este sentido, el polimorfismo de los objetos permitirá que la respuesta sea diferente según el objeto que contenga el método en cuestión.

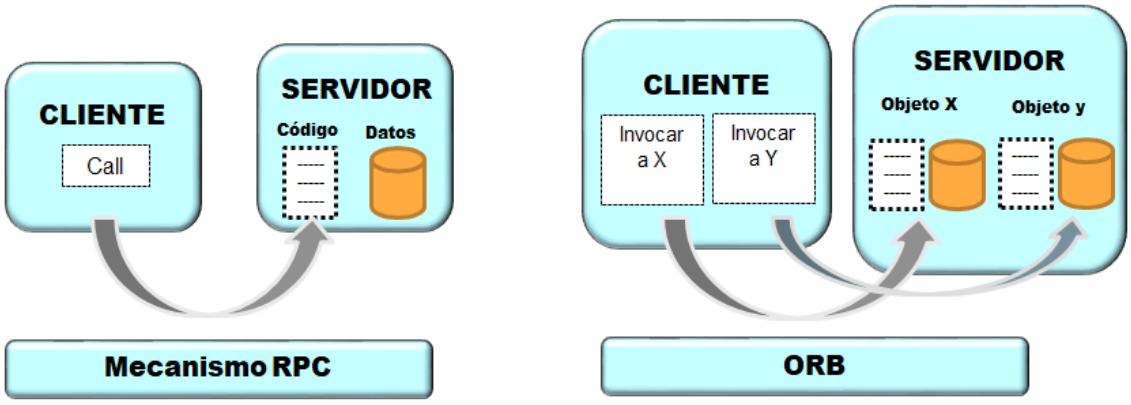


Figura 15. *RPC vs ORB*

Por su parte, un objeto de negocio se define como un activo de software en el dominio de aplicación, que posee, nombre, definición, atributos, comportamiento, relaciones y restricciones. En este sentido, y a la luz del concepto actual de "servicio" un objeto de negocio puede considerarse un antecedente del mismo.

La Figura 16 ayuda a comprender la ubicación del objeto de negocio en la pila de infraestructura de los objetos distribuidos.

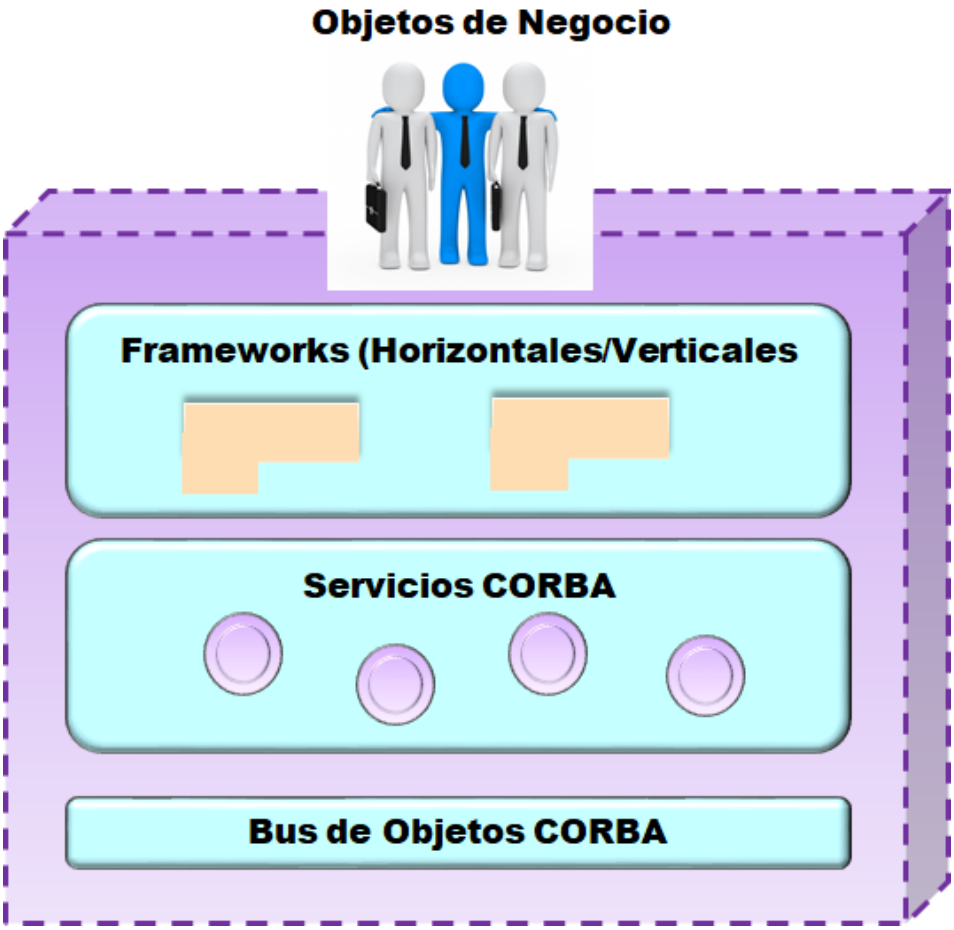


Figura 16. *Los objetos de negocio en la pila de CORBA*

2.2 - Servicios en los lenguajes de programación

Además de la construcción de estándares como CORBA para mejorar la interoperabilidad, el advenimiento de las plataformas basadas en Web favoreció ampliamente el desarrollo de aplicaciones distribuidas a gran escala y volvió a poner en evidencia la necesidad de integrarse e interoperar.

Como se ha desarrollado ampliamente en el Capítulo 3**,** la integración punto a punto, por adaptadores y por mediador de mensajes conforman una arquitectura accidental (se construye para cada caso), de mantenimiento costoso y lento y difícil de gestionar, monitorizar y extender.

El concepto de servicio y su arquitectura de referencia (SOA - *Services Oriented Architecture*) representa un cambio radical en la relación entre el mundo del negocio y el área de tecnología de la información. SOA constituye mucho más que un conjunto de productos aglutinados por una tecnología. Es un nuevo enfoque en la construcción de sistemas de IT que permite a las empresas aprovechar los activos existentes y abordar fácilmente los inevitables cambios en el negocio.

Si bien la industria del software ha venido enfocándose en una arquitectura orientada a servicios desde hace más de 20 años con la noción de reusabilidad y su aplicación a la construcción de software, lo cierto es que en los últimos años esto se ha fortalecido con la definición de estándares y la conformación de consorcios que participan en su definición.

Características de SOA	
Reuso de componentes de software existentes	Este concepto se encuentra vinculado fundamentalmente con la idea de federación de aplicaciones. Un entorno informático federado es aquel en el cual los recursos y aplicaciones se encuentran unidos manteniendo autonomía y autogobierno. La exposición de servicios mediante SOA facilita la federación y por ende el reuso.
Interoperabilidad entre aplicaciones y tecnologías heterogéneas	Este concepto con la capacidad de compartir información entre aplicaciones y también con la independencia de las plataformas. La exposición de servicios favorece la interoperabilidad y permite el uso de la funcionalidad que otorga un proveedor con independencia de la plataforma de implementación.
Flexibilidad para componer, integrar y escalar soluciones	La flexibilidad de una solución se vincula con un mayor alineamiento entre el dominio o negocio y la tecnología. Este alineamiento busca que el sistema reaccione rápidamente a los cambios del entorno. Los servicios de grano fino facilitan la evolución del sistema y otorgan agilidad a la organización.
Conceptos SOA	
Servicios	Piezas funcionales que resuelven un aspecto del negocio. Pueden ser simples (almacenar los datos de un cliente) o compuestos (el proceso de compra de un cliente).
Enterprise Service Bus (ESB)	Infraestructura que habilita una alta interoperabilidad entre servicios en un contexto distribuido.
Bajo acoplamiento	Reducción de dependencias entre sistemas.

Recapitulando, CORBA ha constituido una base conceptual muy importante en la aparición de los conceptos de SOA.

El concepto de objeto y el de servicio guardan muchas similitudes en lo que se refiere a su modularidad y capacidad de reuso, pero también se diferencian, fundamentalmente en el modelo de interacción que presentan.

Los objetos se definen con gran cohesión, en el sentido que existe gran asociación entre los métodos que atienden el objeto y una fuerte ligadura funcional, pero con bajo acoplamiento para minimizar el impacto de los cambios de una clase.

Los servicios, por su parte, necesitan un menor acople y no requieren que se conozca su nombre para utilizarlos porque poseen abundante meta-información.

Dentro de un servicio existen operaciones, no habiendo, a priori, asociaciones entre ellos, por lo tanto, existe menos cohesión que en los objetos.

Un servicio difiere de un objeto o un procedimiento porque se define en función de los mensajes que intercambia con otros servicios.

Los Web Service constituyen una manera apropiada (no la única) de implementar interfaces de aplicaciones basadas en servicios y permiten que diferentes aplicaciones, realizadas con diferentes tecnologías, y ejecutándose en una variedad de entornos, puedan comunicarse e integrarse.

En este sentido, los Web Services constituyen un primer punto para construir software basado en servicios, siendo estos últimos, escritos en cualquier lenguaje de programación.

Para lograr un análisis más específico de los lenguajes de programación en el mundo de los servicios, vamos a introducirnos en BPEL (*Business Process Execution Language*) es un lenguaje concebido por Oracle, BEA Systems, IBM, SAP y Microsoft entre otros y estandarizado por OASIS para la composición de servicios web. La especificación 2.0 de este lenguaje fue descrita por el consorcio OASIS a finales de marzo del 2007, el cual se encarga de definir estándares a nivel mundial.

2.3 – BPEL Business Process Execution Language

El lenguaje BPEL es un lenguaje de alto nivel que lleva el concepto de servicio un paso adelante, al proporcionar métodos de definición y soporte para flujos de trabajo y procesos de negocio. Está definido en XML (*eXtensible Markup Language*) y diseñado para orquestar procesos de forma automática.

El origen de BPEL se remonta a 2001 cuando IBM y Microsoft definieron WSFL (*Web Services Flow Language*) y XLANG, respectivamente. El posterior crecimiento de iniciativas de código abierto y el exitoso *BPMI.org* -que define estándares para la gestión de procesos de negocio que abarca varias aplicaciones y dominios de negocio-, guió a IBM y Microsoft a combinar sus soluciones en un nuevo lenguaje BPEL4WS (*BPEL for Web Services*).

En abril de 2003 surge BPEL4WS y es enviado a OASIS para su estandarización quien en setiembre de 2004 dio origen a WS-BPEL 2.0.

En junio de 2007, otro importante conjunto de empresas publicaron BPEL4People que describe cómo especificar tareas humanas en procesos BPEL.

Con estos antecedentes los objetivos principales que persigue BPEL se pueden resumir en los siguientes:

- Es una plataforma para ejecutar procesos BPEL que se exponen a sus clientes como servicios Web y que requieren la definición de su WSDL.
- Para comenzar el proceso, el cliente deberá invocar una operación del mismo que normalmente será una sola (único punto de entrada).
- Cada invocación genera una nueva instancia como mecanismo básico de ciclo de vida y que posee operaciones para su soporte como “suspender”, “continuar”, “terminar”.
- Provee funciones de manipulación simple de datos solo para mantener el flujo de datos y el flujo de control.
- Soporta un método de identificación de instancias de procesos que permita la definición de identificadores de instancias a nivel de mensajes de aplicaciones. Los identificadores de instancias deben ser definidos por socios (*partner*) y pueden cambiar [BPEL, 2020].

BPEL para orquestar servicios

Con respecto a la orquestación de servicios, es BPEL el encargado de consumir varios servicios en un orden especificado y realizar una función muy concreta, para entenderlo mejor, veamos un ejemplo:

Supongamos que estamos haciendo uso de un sitio de compras on line que nuclea diferentes proveedores y que también ofrece distinto tipo de servicios.

El cliente que hace uso del sitio solo debe indicar qué producto o servicio comprar y elegir un medio de pago. El sistema responde con un mail indicando el éxito de la compra y, eventualmente, brinda un número de rastreo. Cabe mencionar que se está considerando el caso de una compra exitosa, dado que se pretende mostrar el flujo de operaciones subyacentes a la acción de compra de un producto o servicio.

Más allá de si las validaciones y operaciones siguientes se realizan en el momento o de manera diferida, está claro que desde el sitio de compras se deben realizarlas y que no todas dependen de él.

El siguiente diagrama muestra la secuencia de operaciones que el sitio realiza ante la compra de un producto o servicio.

1. Registrar la compra
2. Reservar stock con el proveedor
3. Gestionar pago con el medio de pago
4. Confirmar la compra

Tanto la operación 2 como 3 no dependen del sitio de compras sino de servicios que deben ser expuestos tanto por el proveedor como por el medio de pago, a través de, por ejemplo, un Servicio Web. La Figura 16 muestra la interacción de las cuatro operaciones.

La reserva de stock y la gestión del pago son operaciones externas al sitio de compras y que deben ser resueltas por sus respectivos propietarios (proveedor o medio de pago). Eventualmente, ambos también requerirán una respuesta del sitio de compras para convalidar la reserva del stock y el pago, pero estos dos escenarios no están siendo considerados por este ejemplo para simplificarlo.

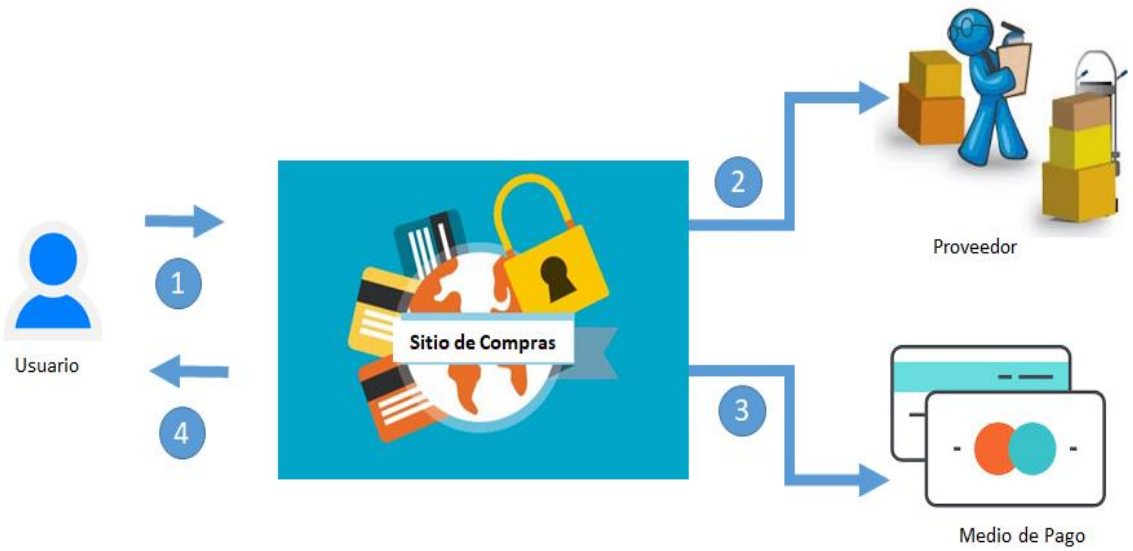


Figura 17. *Operaciones de compra en un sitio de compras con proveedores y medios de pago externos.*

¿Cómo encaja BPEL en este escenario?

Con BPEL se puede crear un solo servicio que reciba todos los parámetros necesarios para realizar la operación como los datos de la compra y los datos del medio de pago, para realizar la reserva de stock y la gestión del pago, guardar el registro de la compra y regresar al usuario la confirmación de su compra.

Si bien esto te puede parecer algo que cualquier lenguaje pueda realizar la realidad es que BPEL está diseñado para estos escenarios lo que permite una programación mucho más rápida.

BPEL como lenguaje de programación

Un proceso BPEL especifica el orden exacto en el que deben invocarse los servicios web participantes, tanto de forma secuencial como en paralelo. Con BPEL se puede expresar un comportamiento condicional, por ejemplo la invocación de un servicio web puede depender del valor de una invocación previa. También es posible construir bucles, declarar variables, copiar y asignar valores, y definir manejadores de fallos, entre otras cosas. Combinando todas estas construcciones se definen procesos de negocio complejos de una forma algorítmica. De hecho, debido a que los procesos de negocio son esencialmente grafos de actividades, podría ser útil expresarlos utilizando diagramas de actividad UML.

Para entender mejor el comportamiento de BPEL como lenguaje de programación, es necesario hacer hincapié en las siguientes características:

Transformar al XML en un lenguaje ejecutable

La máquina de servicios BPEL es un componente JBI (*Java Business Integration*) que satisface el estándar JSR 208 (*Java Specification Request*) y proporciona servicios para ejecutar procesos de negocio desarrollados con WS-BPEL 2.0. Para desplegar un proceso BPEL, es preciso añadirlo como un módulo JBI en un proyecto de composición de aplicaciones (*Composite Application Project*) [Orquestación BPEL, 2012].

La máquina de servicios BPEL se inicia juntamente con el servidor de aplicaciones, por lo que para desplegar y ejecutar procesos BPEL será necesario tener en marcha el servidor de aplicaciones. La máquina de servicios BPEL está representada como el componente JBI sun-bpel-engine [BPEL User Guide, 2007] del servidor de aplicaciones.

Una vez desarrollado el proceso BPEL, el último paso es su despliegue en una BPEL engine. Una alternativa mostrada en [Orquestación BPEL, 2012] es usar NetBeans como herramienta para desarrollar y desplegar procesos BPEL, por lo tanto, vamos a explicar cómo se realiza el proceso de despliegue y pruebas, utilizando NetBeans 7.1

Netbeans utiliza el entorno de ejecución JBI. Dicho entorno de ejecución incluye varios componentes que interactúan utilizando un modelo de servicios. Dicho modelo está basado en el lenguaje de descripción de servicios web WSDL 2.0. Los componentes que suministran o consumen servicios dentro del entorno JBI son referenciados como máquinas de servicios (*Service Engines*). Uno de estos componentes es la máquina de servicios BPEL (BPEL *Service Engine*), que proporciona servicios para ejecutar procesos de negocio. Los componentes que proporcionan acceso a los servicios que son externos al entorno JBI se denominan *Binding Components*. Toda esta descripción se grafica en la Figura 18.

Estructura de un programa BPEL

Un proceso BPEL puede ser síncrono o asíncrono. Un **proceso BPEL síncrono** bloquea al cliente (aquél que usa el proceso BPEL) hasta que finaliza y devuelve el resultado a dicho cliente. Un **proceso asíncrono** no bloquea al cliente. Para ello utiliza una llamada *callback* que devuelve un resultado (si es que lo hay). Normalmente utilizaremos procesos asíncronos para procesos que consumen mucho tiempo, y procesos síncronos para aquellos que devuelven un resultado en relativamente poco tiempo. Si un proceso BPEL utiliza servicios Web asíncronos, entonces el propio proceso BPEL normalmente también lo es.

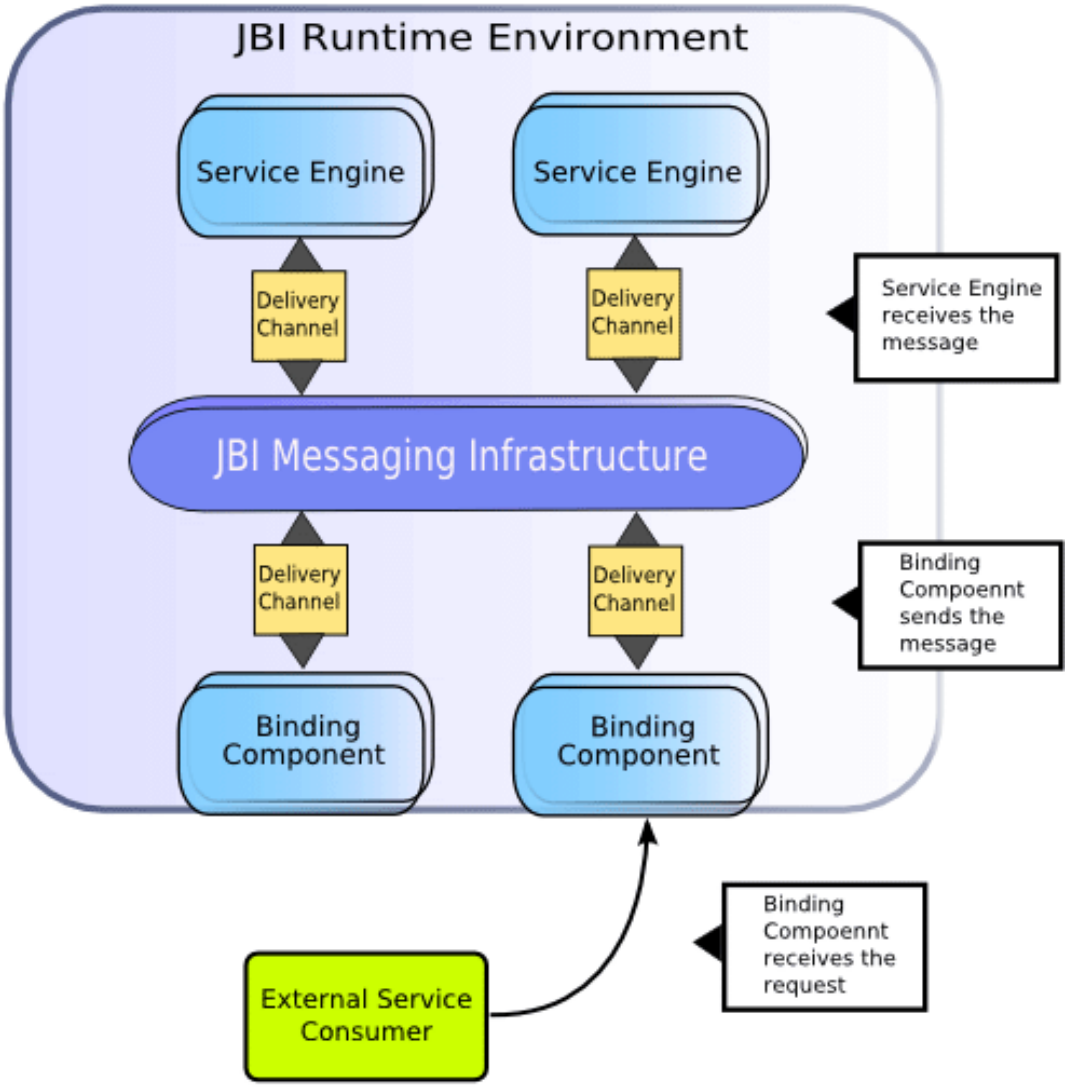


Figura 18. *Ejemplo de entorno de ejecución JBI*

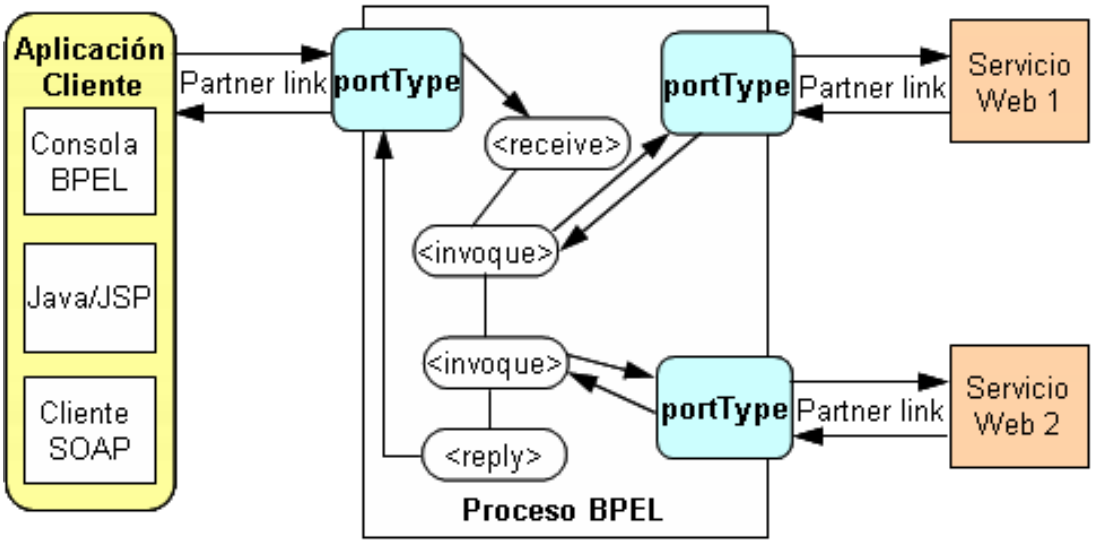


Figura 19. *Estructura de un Proceso BPEL*

La estructura básica de un documento (fichero con extensión **.bpel**) que define un proceso BPEL es la siguiente:

```
<process name="nameProcess" ... >

  <partnerLinks>
    <!-- Declaración de partner links -->
  </partnerLinks>

  <variables>
    <!-- Declaración de variables -->
  </variables>

  <sequence>
     <!-- Cuerpo principal de la definición del proceso BPEL -->
  </sequence>

</process>
```

En la Tabla 3 se resumen algunas de las etiquetas que estructuran un proceso BPEL, su sintaxis y su descripción -en [BPEL User Guide, 2007] se encuentra una guía completa-.

Tabla 3. Resumen de etiquetas principales que estructuran un proceso BPEL

Etiqueta	Sintaxis	Descripción
<process>	<process name="nameProcess" targetNamespace= "http://..." xmlns="http://schemas.xmlsoap.org/ws/2003/03/business-process/" xmlns:sw1="http://..." <!-- namespace del servicio Web sw1 --> xmlns:sw2="http://..." <!-- namespace del servicio Web sw2 --> ... > ... </process>	Incluyen los espacios de nombres, tanto el objetivo (**targetNameSpace**), como los *namespaces* para acceder a los WSDL de los servicios Web a los que invoca. También se declara el *namespace* para todas las etiquetas y actividades BPEL.
<partnerLinks>	<partnerLinks> <partnerLink name="ncname" partnerLinkType="qname" myrole="ncname" partnerRole="ncname"> </partnerLink> </partnerLinks>	Cada proceso BPEL tiene al menos un partner link cliente, debido a que tiene que haber un cliente que invoque al proceso BPEL. Por otro lado, un proceso BPEL tendrá (normalmente) al menos un partner link a quién invoque.
<partnerLinksType>	<!-- Extracto de Saludo.wsdl --> <partnerLinkType name="MyPartnerLinkType">	Especifica la relación entre dos servicios, definiendo el rol que cada

	<role name="ProveedorServicioSaludo" portType="SaludoPortType"/> </role> </partnerLinkType> <!-- Extracto de Saludo.bpel --> <partnerLinks> <partnerLink name="cliente" partnerLinkType="MyPartnerLinkType" myRole="ProveedorServicioSaludo"/> </partnerLinks>	servicio implementa. Es decir, declara **cómo** interaccionan las partes y lo que cada parte ofrece. Los nombres de los roles son cadenas de caracteres arbitrarias. Cada **rol** especifica exactamente un tipo *portType* WSDL que debe ser implementado por el servicio correspondiente.

<variables>	<variables>
	<variable name="nombreVar"
	messageType="qname" //tipo mensaje
	type="qname" //tipo simple
	element="qname"/> /tipo elemento
	</variables>
		Almacenan y transforman mensajes que contienen el estado del proceso. Sus tipos son:
		• Mensaje WSDL: indicado mediante el atributo *messageType*
		• Esquema XML: indicado mediante el atributo *type*
		• Elemento de esquema XML: indicado mediante el atributo *element*

IMPORTANTE

Es fácil confundir los *partner link* y los *partner link types*, sin embargo:

- Los *partner link types* y los roles son extensiones especiales de WSDL definidas por la especificación BPEL. Como tales, dichos elementos se definen en los ficheros WSDL, no en el fichero del proceso BPEL.
- *Partner link* es un elemento BPEL 2.0. Por lo que se define en el fichero del proceso BPEL.

Actividades en BPEL

Un proceso BPEL está formado por una serie de pasos. Cada uno de los pasos se denomina **actividad**. BPEL soporta dos tipos de actividades: primitivas y estructuradas.

Algunas de las actividades primitivas y su sintaxis, están descriptas en la Tabla 6 -en [BPEL User Guide, 2007] se encuentra una guía completa.

Tabla 4. Resumen algunas Actividades Primitivas

Actividad	Sintaxis	Descripción
receive	<receive partnerLink="ncname" portType="qname" operation="ncname" variable="ncname" createInstance="yes|no"> </receive>	Especifica un *partnerLink*, un *portType* y una operación que puede ser invocada. Permite que el proceso quede a la espera del arribo de un mensaje.

reply	<reply partnerLink="ncname" portType="qname" operation="ncname" variable="ncname"> </reply>	Responde a una actividad receive, se usa para interacción sincrónica y específica el mismo *partnerLink*, un *portType* y operación que invocó al proceso.
invoque	<invoque partnerLink="ncname" portType="qname" operation="ncname" inputVariable="ncname" outputVariable="ncname"> </invoque>	Invoca a otro servicio Web que se haya definido como *partner*. Puede ser ser síncrona (operación de tipo *request-response*) o asíncrona (operación de tipo *one-way*).

Las **actividades estructuradas** permiten combinar las actividades primitivas para especificar exactamente los pasos de los procesos de negocio. Entre las principales actividades estructuradas

- Ejecución secuencial (<*sequence*>), declara un conjunto de actividades que engloba se ejecutarán de forma secuencial.
- Ejecución paralela (<*flow*>) declara que las actividades que engloba se realizarán en paralelo.
- Selector múltiple (<*switch*>) implementa condicional sobre la siguiente actividad a ejecutar.
- Iterador (<*while*>) permite iterar sobre un conjunto de actividades.
- Selección de ejecución (<*pick*>) permite seleccionar un camino de ejecución.
- Fallas (<*throw*>) permite lanzar excepciones.
- Manejador de excepciones (<*faulthandler*>) permite tratar las excepciones que se producen.
- <*scope*> Determina el alcance de un grupo de actividades.
- <*terminate*> Finaliza la ejecución de un proceso de negocio.

2.4 - Conclusiones

El enfoque sobre procesos de negocios modernos más el bagaje de los lenguajes WSDL y XLANG, llevaron a BPEL a adoptar los Servicios Web como su mecanismo de comunicación externa.

Las facilidades de mensajería BPEL dependen del uso del WSDL para describir los mensajes entrantes y salientes y su mecanismo de sincronización se basa en las propiedades de los mismos.

Los contextos de ejecución de BPEL contemplan mecanismos de serialización para controlar el acceso a las variables garantizando la integridad ante la concurrencia.

Un proceso BPEL puede ser ejecutado sobre diferentes motores de ejecución, otorgándole portabilidad, lo cual es muy importante en el desarrollo B2B (*Business to Business*).

El diseño específico de BPEL para el desarrollo de procesos, lo hace eficiente en la ejecución de procesos de larga duración (ya sea minutos, horas, días).

BPEL soporta una característica de “deshacer” actividades previas del proceso de negocio. Es posible definir manejadores para estos casos indicando el alcance de los mismos dentro del proceso de ejecución. Estas características son conocidas como *Long-Running-Transactions* (LRT).

CAPÍTULO 3
Integración de Aplicaciones

Patricia Bazán

La integración de aplicaciones es un aspecto largamente abordado en la literatura y fueron varios los autores que acuñaron diversas definiciones.

Al referir el término "integración de aplicaciones" surge naturalmente la noción de EAI o *Enterprise Application Integration* como Uso de software y sistemas informáticos para integrar un conjunto de aplicaciones en una organización (Wikipedia)

Enterprise Application Integration emergió como una iniciativa de la industria de la computación con el advenimiento del año 2000 y la caducidad de muchos viejos sistemas no compatibles Y2K.

El objetivo de la EAI no es nuevo, comenzó ya con la distribución de datos desde los *mainframes* hacia los procesadores *front-end*, sin embargo, los datos atrapados entre distintas compañías y formatos tornan difícil la tarea [Juric M. et al., 2007].

EAI se enfoca en el desarrollo de sistemas que proveen funcionalidad de negocios fluida donde la integración está en la capacidad de intercambiar y compartir datos sin saber dónde están ubicados y qué formato tienen.

Las siguientes dos definiciones obtenidas de [Morgenthal JP., 2001] son las que mejor se adaptan a la idea de integración de aplicaciones que se quiere transmitir:

> La integración fluida de procesos de negocio con el propósito de conducirlos electrónicamente

> La integración es compartir y/o intercambiar datos entre los sistemas con el propósito de proveer una interfaz unificada

Este capítulo presenta la evolución de la integración de aplicaciones en términos tanto tecnológicos como metodológicos y tiene por objetivo mostrar que dicha evolución está guiada por los conceptos de reusabilidad, distribución y orquestación de servicios.

3.1. Evolución tecnológica de las aplicaciones

Los sistemas de información tradicionales que sustentan su modelado en un enfoque netamente dirigido a los datos han dominado la industria por largo tiempo. En los últimos tiempos

subyace, cada vez con más ímpetu, la idea de que los procesos son igualmente importantes y deben ser considerados de una manera sistemática.

Por otra parte, el paradigma de desarrollo orientado a servicios está llevando el mundo de la tecnología de información a un cambio de enfoque en la manera de resolver problemas y también a una nueva visión de integración de aplicaciones.

En la década del 70 el paradigma era el cálculo, las aplicaciones eran de estadísticas/contabilidad y balística y la herramienta los lenguajes de programación procedurales sin entornos de desarrollo.

En la década del 90 el paradigma era la construcción de los sistemas de información, aplicaciones de gestión generalmente de índole administrativa y las herramientas eran los RDBMS, TP, *Workflow* y *DataWarehouse*.

La tendencia actual es hacia un paradigma orientado a procesos cuyo enfoque se ha sustentado en los conceptos de *workflow* pero que encontró en BMP un soporte de la idea de procesos mucho más amplio incluyendo simulación, verificación y monitoreo.

Además, el enfoque orientado a procesos actual ordena la arquitectura de integración accidental originada por la integración punto a punto, proveyendo un mayor nivel de abstracción donde existen procesos que orquestan servicios, que a su vez, resuelven la funcionalidad.

La tecnología de Servicios Web representa el estándar de computación distribuida más adoptado de la historia de la industria y son la plataforma ideal para la Arquitectura Orientada a Servicios (SOA).

Asimismo, la orquestación y coreografía de los Servicios Web como *workflow* son partes esenciales de las definiciones de trabajo actuales.

La orquestación impone el orden y el ritmo individual a un conjunto de Servicios Web. La coreografía define el comportamiento variado entre los Servicios Web.

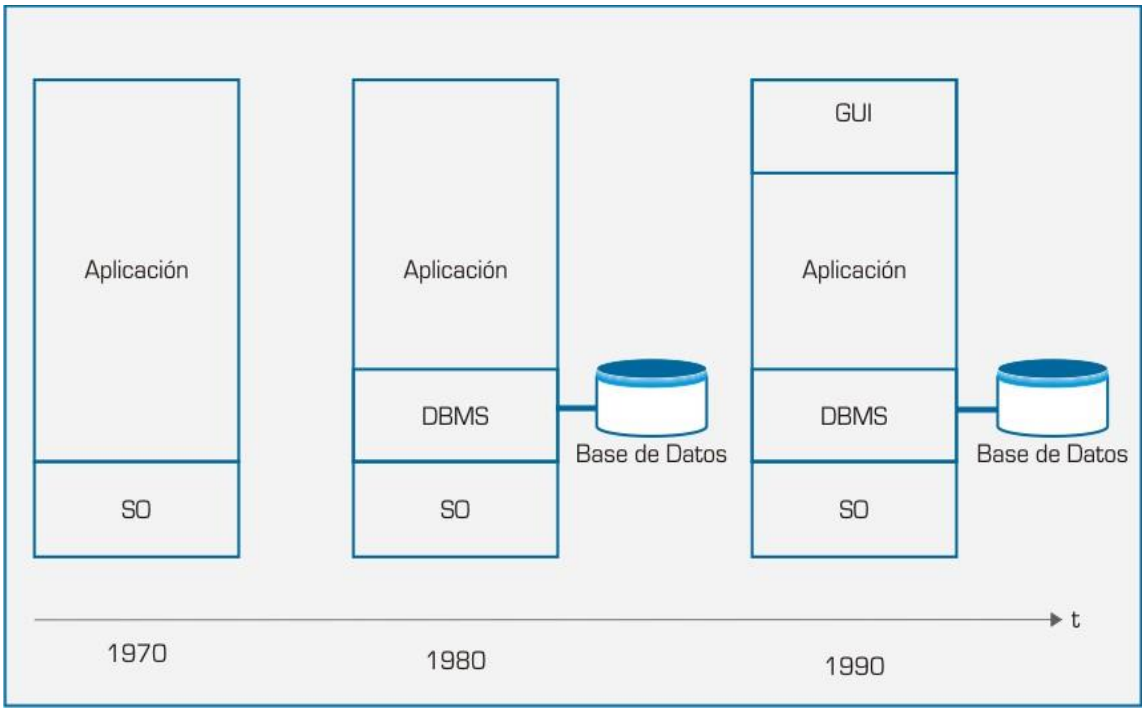


Figura 20. *Evolución de tecnológica de aplicaciones*

Como se muestra en la Figura 19 extraída de [Weske M., 2008], la evolución marca la tendencia de otorgarle primero una gran independencia al componente de manipulación de datos con la aparición de los DBMS. Luego de ello se dio mayor relevancia a la interfaz con el usuario, dejando la lógica de la aplicación como un elemento o componente separado y que debería tender a no mezclarse con los otros dos elementos.

3.2. Alternativas de integración de aplicaciones

La integración de aplicaciones presenta una evolución descripta parcialmente en [Bazán P, 2017] cuando se menciona la evolución desde los objetos distribuidos hacia los servicios. Allí se menciona la integración punto a punto como una de las primeras aproximaciones donde (existe) la misma se encontraba embebida en el código con serias dificultades para su modificación y gestión y haciendo imposible su administración y monitoreo.

Esto provocó enormes frustraciones tanto en las áreas de IT como de negocios de las organizaciones y comenzó a esbozar las definiciones de EAI enunciadas más arriba.

Surge así la EAI tradicional como primera alternativa que dará lugar a los modelos basados en *workflow* y más tarde al uso de modelos de procesos que utilizan servicios dentro de una SOA.

EAI tradicional

Este modelo se basa en la construcción de un nodo central y un conjunto de repetidores directamente asociados a éste, aunque los repetidores no se encuentran conectados entre sí. De esta manera, el middleware está representado por la integración centralizada de aplicaciones, y las aplicaciones que deben ser integradas son reflejadas por los repetidores. Las mismas interactúan entre ellas por medio de la integración centralizada de aplicaciones [Weske M., 2008].

Las aplicaciones pueden interactuar de formas muy diversas, desde invocaciones simples hasta interacciones complejas entre múltiples aplicaciones. Estas últimas consisten en una serie de actividades representadas por una invocación a una aplicación, además de existir restricciones de ejecución entre las mismas.

Este esquema de agentes y mensajes tiene ciertos inconvenientes. El primero de ellos es que el agente contiene cierta lógica, oculta en las reglas. La programación de éstas puede volverse una labor compleja debido a las dependencias que pueden darse entre las mismas, y el hecho de cambiar una regla puede tener implicancias en el comportamiento global del sistema.

La razón principal de estos problemas es la “pérdida” conceptual que se da en la integración de aplicaciones, ya que la integración de datos y de procesos requiere gran actividad de programación y configuración de bajo nivel, tanto de adaptadores como de los agentes de mensajes.

La integración de datos suele darse mediante actividades de mapeo, lo cual requiere un modelo de datos acordado entre todas las aplicaciones y que reside en los agentes. Este mo-

delo global suele no ser explícitamente desarrollado, pero es común encontrarlo oculto en las reglas de mapeo de datos efectuadas por los adaptadores.

Sistemas de Workflow y la integración de aplicaciones

El término *workflow* consiste en la automatización de un proceso de negocio, en su totalidad o en parte, en el cual se intercambian documentos, información o tareas de un participante a otro, para provocar la acción de acuerdo a un conjunto de reglas procedimentales [Weske M., 2008].

Un *sistema de manejo de workflow* es un sistema que permite definir, crear y manejar la ejecución de flujos de trabajo a través del uso de software, que corre en uno o más motores, y que es capaz de de interpretar la definición del proceso, interactuar con los participantes del *workflow* y, donde sea requerido, invocar el uso de herramientas y aplicaciones IT.

La integración de aplicaciones es efectuada por el sistema de manejo de *workflow*, usando adaptadores similares a los que se usan en un ambiente tradicional de aplicaciones empresariales.

La tecnología de *workflow* es capaz de soportar procesos de negocio dentro de un sistema dado o dentro de un conjunto de aplicaciones, lo que permite efectivamente integrar estos sistemas. Sin embargo esta tecnología posibilita también representar procesos en los que hay seres humanos activamente involucrados, y así mejorar la colaboración entre los trabajadores con conocimiento.

La tecnología de manejo de *workflow* puede ser utilizada para facilitar la modificación de la lógica del proceso realizado por aplicaciones. Las funciones de una aplicación son pasos en el *workflow*, y cada componente usa un modelo de *workflow* para representar las funciones. Por la modificación de la lógica del proceso especificada en los modelos de *workflow*, se puede modificar el comportamiento de las aplicaciones sin codificar.

Hoy en día, la mayor cantidad de aplicaciones empresariales, como las aplicaciones de planeamiento, poseen un componente *workflow* que facilita la adaptación flexible de los procesos de negocio dentro de estos sistemas. Observar que usamos el término “componente de *workflow*” además de “sistema de manejo de *workflow*”, debido a que un componente no es un sistema aislado sino que está embebido en la aplicación.

Podemos concluir que una aplicación de *workflow* única consiste de actividades y su correspondiente ordenamiento causal y temporal, donde las mismas son realizadas por un sistema común (tal como se muestra en la Figura 20 [**¡Error! No se encuentra el origen de la referencia.**] [Weske M., 2008]). Los *workflow* de aplicación múltiple contienen actividades que son realizadas por sistemas de múltiples aplicaciones, proveyendo así una integración de las mismas (tal como se muestra en la Figura 21 [Weske M., 2008]).

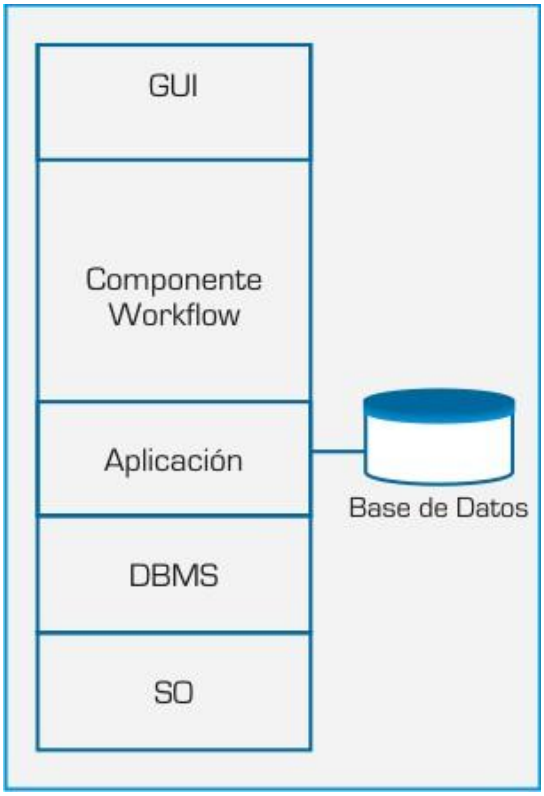


Figura 21. *Componente Workflow de aplicación simple*

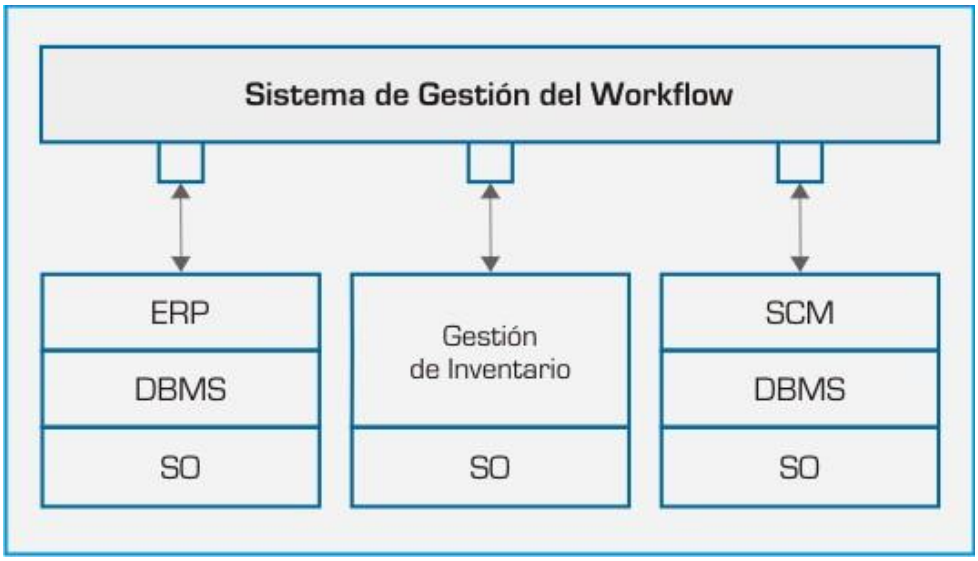


Figura 22. *Componente Workflow de aplicación múltiple.*

Integración basada en SOA y BPM

El software de w*orkflow* utilizado para integrar aplicaciones, puede cubrir cuatro aspectos:

1. Ser un componente de aplicaciones verticales. Un software de *workflow* es ágil a la hora de adaptarse a los cambios de procesos y cambios organizacionales. Esta es una realidad en muchas aplicaciones verticales.

2. Son adecuados para utilizar como APIs. Esto es así si cuenta con soporte para Java que permite integrarse tanto para aplicaciones Web como para otras aplicaciones de IT.
3. Constituye el elemento unificador para aplicaciones colaborativas, tanto desde el punto de vista de aplicaciones que se construyen como composición de otras bajo la filosofía de *Web Services*, como del de la automatización de procesos basados en reglas.
4. Se ajustan a la implementación de *Web Services.*

Pero en una aplicación donde el proceso de negocio sea realmente un conjunto de tareas cuyos participantes son Servicios Web, provoca inevitablemente un desorden dentro del *workflow* y surge la necesidad de interoperar y describir procesos ejecutables.

La interoperabilidad se logra a través de la adopción de estándares como XML y WSDL, mientras que la descripción de procesos ejecutables se puede llevar a cabo a través de BPEL [Kettinger et al., 1997] [Yourdon, E et al. 1988].

Los desarrollos en arquitectura de software empresarial y en BPM están relacionados con el manejo de *workflow*. El logro principal que se busca alcanzar aquí es la representación explícita de las estructuras de los procesos a través de modelos, y la representación controlada de los procesos basada en los modelos creados con anterioridad [Weske M., 2008].

Para la realización de aplicaciones de composición en un ambiente de orientación a servicios, se utilizan técnicas de composición de servicios.

El *middleware* de integración de aplicaciones en general, y el *middleware* de bus de servicios empresariales en particular, proveen una base técnica aceptable para realizar composición de servicios, debido a que proveen interfaces estándar que pueden ser utilizadas en desarrollos de composición. El *middleware* típico para la integración de aplicaciones empresariales presenta un componente de *workflow* de sistema, que puede o bien usar un código propietario o bien usar código BPEL (Lenguaje de ejecución de procesos de negocio para *Web Services*).

La composición de servicios es una idea de especial interés para el desarrollo de nuevas aplicaciones, basándose en funcionalidad ya existente. Así, la composición describe la forma en que se relacionan los distintos servicios, es decir que se están describiendo estructuras de proceso. Como resultado, una composición posee un conjunto de servicios, cada uno de los cuales realiza una actividad.

La Figura 22 muestra la analogía entre aplicaciones tradicionales, que conllevan un mecanismo de integración tradicional, con las aplicaciones basadas en servicios [Oracle, 2007] [Scott McKorkle, 2007].

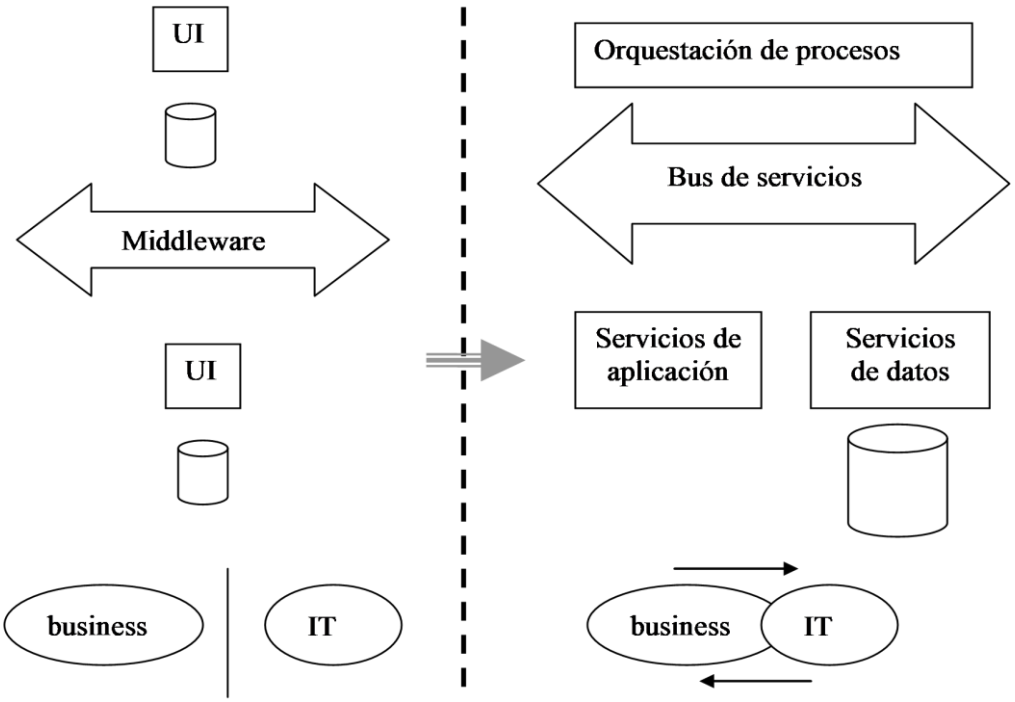


Figura 23. *SOA alinea negocios con IT, llevando las aplicaciones tradicionales hacia las que usan servicios*

BPM como la nueva metodología para satisfacer los objetivos de una organización a través de la gestión de procesos de negocio, no plantea una visión completamente renovadora y de empezar de cero, sino que es necesario contemplar todo el trabajo realizado con anterioridad dentro de la empresa, de manera de apuntar a la integración. Aquí es donde se insertan los conceptos de SOA, *Web Services* y BPEL, de manera tal de lograr construir una aplicación real, integrada e insertada en un entorno B2B, y por sobre todo, orientada a procesos.

3.3. Elementos de un modelo de integración

El modelo de integración tiene por objetivo lograr una integración completa, segura y confiable de un conjunto de sistemas de software existentes, maximizando la reutilización de código, manteniendo un bajo acoplamiento y favoreciendo el mantenimiento ágil y a bajo costo.

Los elementos de este modelo de integración llevan a analizar los tipos de integración posible, los métodos aplicados para llevar a cabo la integración, los componentes de infraestructura requerida y los actores que participan. Este análisis realizado tiene por finalidad aportar criterios a la hora de decidir cuáles de todos los elementos se elegirán para componer un modelo de integración.

Tipos de integración

Si bien existen diversas taxonomías acerca de los posibles tipos de integración existentes [Juric M. et al., 2007], los más frecuentes, teniendo en cuenta el enfoque que se abordará, se refieren a la integración desde dos aspectos posibles:

- Integración a nivel de datos: se enfoca en el movimiento de datos entre aplicaciones con el objetivo de compartirlos. Es una integración relativamente simple si se comparten formatos y estructuras, de lo contrario se establecen protocolos o acuerdos entre las partes para poder realizar la integración. (Ejemplo: integración por XML)
- Integración a nivel de aplicaciones: se basa fundamentalmente en compartir funcionalidad. Es una integración basada en APIs que exponen su funcionalidad a través del uso de interfaces que serán tanto más portables dependiendo del lenguaje utilizado para definirse. (Ej: IDL de CORBA o WSDL de los *Web Services*)

La integración de datos dirigida por procesos [Bazán P.,2009] ayuda a enriquecer los servicios de negocios SOA y los procesos BPM a través de una secuencia de servicios de datos combinados de manera reusable que incorpora la intervención de tareas humanas transformando la información en exacta, consistente y oportuna.

La integración de aplicaciones tiene por objetivo entender y usar las interfaces para acceder a la funcionalidad requerida y enmascarar u ocultar las diferencias tecnológicas usadas por cada interfaz en su acceso. Esto último se lleva a cabo con servicios que exponen sus interfaces [Juric M. et al., 2007].

Escenarios de integración de datos dirigida por procesos

Entre las particularidades que presenta un escenario de integración dirigido por procesos podemos encontrar: mantenimiento de la integridad transaccional, manejo de tareas de larga duración o múltiples pasos, colaboración entre las personas y las actividades de integración y el soporte para integración de datos analítica.

- *Manejo del rechazo o datos inválidos, para mantener la integridad transaccional.* Los procesos o aplicaciones deben tener en cuenta cómo responden los procesos de integración de datos ante entradas erróneas o respuestas recibidas fuera de tiempo. Sin duda el manejo de excepciones consume cerca del 80% del tiempo en la construcción de los procesos de negocios.
- *Manejo de tareas de integración de datos de larga duración y múltiples pasos.* La integración no puede hacerse en un único paso y necesita ser modelada como un workflow orquestado.
- *Promover una mejor colaboración entre las personas y las actividades de integración de datos.* La integración de datos basada en procesos puede ayudar a coordinar esfuerzos a través de múltiples grupos. La orquestación junto con el workflow humano puede asistir con la revisión y aprobación de ciertos datos que caen bajo la responsabilidad de varias personas.

- *Soporte para BI e integración de datos analítica.* Cuando un KPI que se monitorea en una consola de BI atraviesa ciertos umbrales, se requiere la intervención de humanos que tomen acciones correctivas.

Método para la integración basada en servicios

La orientación a servicios es una filosofía de diseño que hace que los recursos de IT están disponibles sobre una red en forma independiente de su ubicación [Oracle, 2007]. Es posible delinear dos maneras de introducir esta filosofía de orientación a servicios en una organización.

La orientación top-down o dirigida por procesos, parte a partir organización y va refinando hacia sus áreas funcionales identificando actividades, quién las realiza y bajo qué restricciones.

La orientación bottom-up o dirigida por aplicaciones, se inicia a partir de las aplicaciones para identificar componentes funcionales que puedan combinarse para dar respuesta procesos de grano más grueso.

La Tabla 7 enuncia las actividades de cada método y la Figura 23 muestra un ejemplo acerca de cómo asociar servicios a actividades, procesos y sub-procesos.

Tabla 5. Integración top-down y bottom-up

Método	Actividades
Top - down o dirigida por procesos, la organización se modela a sí misma desde lo general a lo particular.	Identificar procesos de negocio y eventos.
	Reunir requerimientos de negocio de cada proceso.
	Descomponer procesos hasta un grado de detalle atómico (actividades y servicios).
	Coordinar la descomposición dentro de un flujo de procesos transversales a la organización.
Bottom-up o dirigida por aplicaciones, a partir de ellas se detectan áreas debajo acoplamiento y reusabilidad para luego componer servicios de grano más grueso	Identificar la funcionalidad dentro de las aplicaciones para ser publicados como servicio.
	Exponer y publicar los servicios.
	Orquestar los servicios expuestos de modo que logren alcanzar la funcionalidad de las aplicaciones existentes.
	Reevaluar la cartera de aplicaciones para eliminar redundancias.

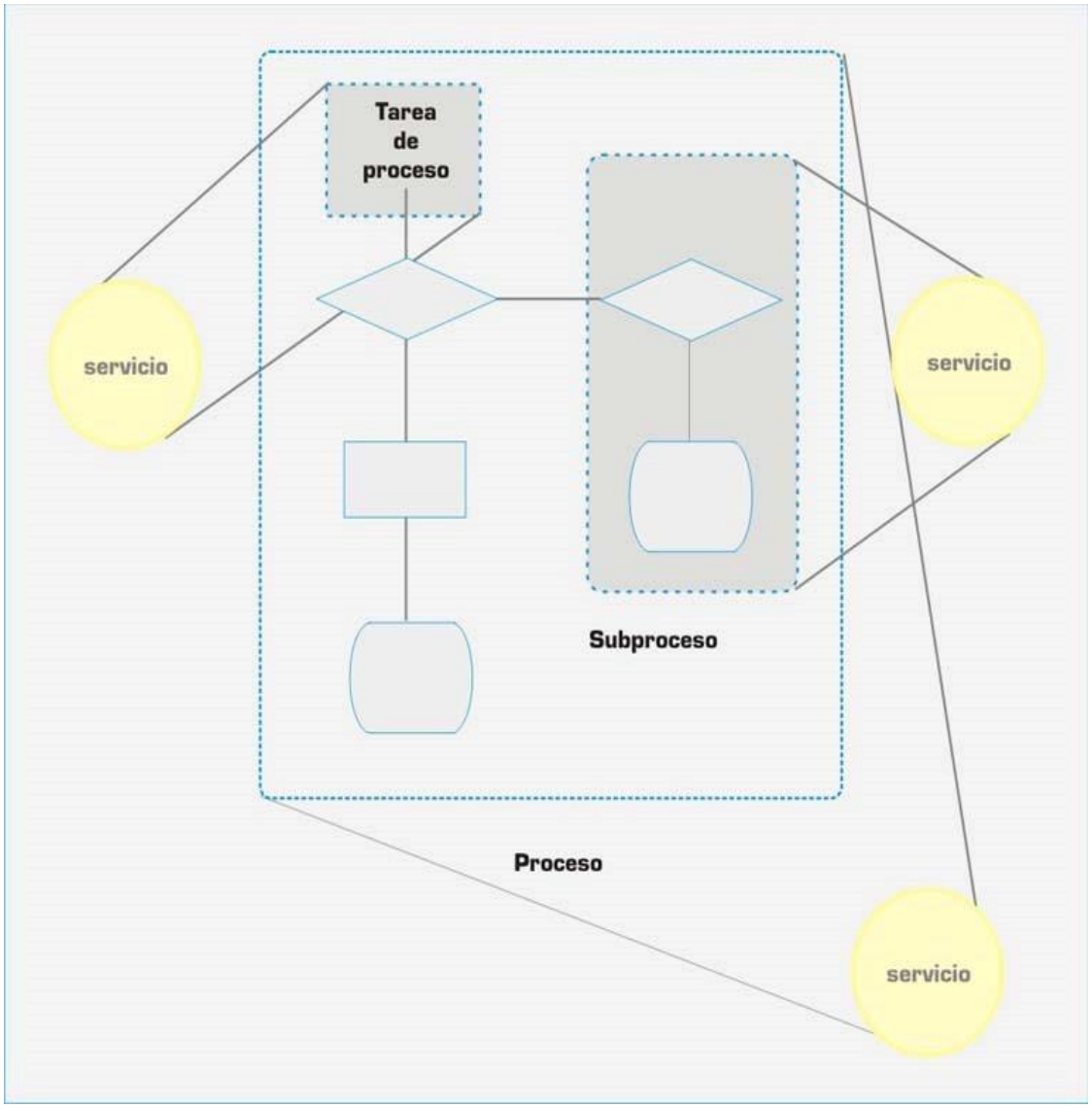


Figura 24. *Asociación de servicios a actividades, procesos y sub-procesos*

Componentes de la infraestructura para la integración

Otro de los elementos de un modelo de integración de aplicaciones, aplicables cada uno al escenario o enfoque de integración abordado, se enuncian en esta sección como los elementos que componen la infraestructura de la integración. Entendiendo por infraestructura al *"conjunto de medios técnicos e instalaciones necesarias para llevar a cabo un actividad".*

Comunicación

La comunicación asegura a los desarrolladores un nivel de abstracción tal, que pueden independizarse de los detalles de bajo nivel asegurando que proveedor y consumidor de servicio puedan encontrarse.

Pueden utilizarse sistemas de comunicación asincrónicos basados en mensajería o bien sincrónicos vía un broker de objetos o bien un ESB.

Estos *middleware* usan protocolos estándar como SOAP (*Simple Object Access Protocol*), HTTP, TCP/IP, IIOP (*Internet Inter-ORB Protocol*) [Kettinger et al., 1997].

Ruteo y mediación

El ruteo y mediación es un nivel que adapta el nivel de comunicación entre aplicaciones de tal modo que las mismas puedan interoperar [Juric M. et al., 2007]. Entre las responsabilidades que tiene este nivel está la de lograr que datos provenientes de distintas fuentes representan un concepto de negocio.

Transformación

La transformación de las estructuras de datos ha sido siempre un problema resuelto de manera puntual, escribiendo código ad-hoc que leía y transformaba al formato destino de manera puntual.

La aparición de los lenguajes de marcado y en particular el XML como estándar de facto para el intercambio de datos, otorgaron un mayor nivel de madurez a la transformación de los datos [Juric M. et al., 2007].

La transformación hoy puede considerarse como un servicio provisto por las máquinas de transformación basadas en XSLT (*EXtensible Stylesheet Language for Transformations*) que producen transformaciones independientes del lenguaje y la plataforma.

Coordinación y Orquestación

En un enfoque dirigido por procesos es preciso sincronizar las actividades de integración definiendo formalmente los servicios que requieren los procesos y las aplicaciones, secuenciándolos en una orquestación.

En este sentido, una infraestructura de integración debe contar con algún mecanismo de coordinación inter-procesos o intra-procesos que ordene los pasos a seguir para conducir los servicios.

La orquestación ha evolucionado desde el enfoque manual al automatizado. En el primer caso, se reducía a código inyectado en las aplicaciones que resolvía integración punto a punto y que tenía embebida la lógica del *workflow*. Esto resultaba difícil de mantener y de modificar. El enfoque automático provee una mayor flexibilidad, desacoplando procesos de datos y produciendo un servicio de datos independiente que inicia una tarea de integración de datos de bajo nivel como parte del proceso de *workflow*.

Este último enfoque, el automático, es solamente posible si los servicios están construidos de modo adecuado como módulos autocontenidos sin las dependencias típicas de la programación procedural.

Transacción

Uno de los principales aportes al definir una infraestructura de integración es la de proveer un mecanismo para llevar a cabo las operaciones de modo transaccional, esto es, la capacidad de invocar operaciones sobre diferentes sistemas soportando la semántica del modelo transaccional bajo las propiedades ACID (*Atomicity, Consistency, Isolation, Durability*) [Juric M. et al., 2007].

Estas propiedades garantizan la preservación de la consistencia del sistema, aíslan las operaciones entre sí, otorgan persistencia a los cambios y además son atómicas.

Seguridad

Desde el punto de vista de la integración, la seguridad conlleva los mismos conceptos tradicionales: autorización, autenticación y auditoría. En este sentido, la infraestructura de la integración debe proporcionar los medios para limitar el acceso al sistema, hacerlo de una manera unificada y dejar rastros de dichos accesos.

Actores que participan en la integración

Los actores que participan en un escenario de integración de aplicaciones, tienen la finalidad de instanciar los componentes del modelo, elegir un tipo de integración posible, identificar el escenario y aplicar el método elegido.

Los actores pueden o no estar representados por seres humanos, pudiendo reemplazarse actores no humanos por humanos cuando la solución tecnológica del modelo de integración así lo requieran. La Tabla 8 describe los diferentes actores o roles y su función.

Tabla 6. Actores/Roles participantes en la integración de aplicaciones

Actor/Rol	Función
Analista de negocio	Identificar los proceso que deben llevarse a cabo para alcanzar el objetivo de integrar sistemas de software en el marco un una visión de procesos de la organización. Eventualmente, modela los procesos en alguna herramienta de diseño basada en BPMN.
Arquitecto del sistema	Identificar funcionalidades existentes y transformarlas en insumos para los procesos a cubrir. Refinar los modelos realizados por el analista mejorando los detalles que se requieren para identificar componentes.
Coordinador de servicios	Aplicar los flujos y condiciones que deben cumplirse para que los servicios y componentes identificados por el arquitecto trabajen en forma coordinada (orquestada) para alcanzar el objetivo del negocio. Este actor puede ser reemplazado por un software.
Desarrollador de componentes y servicios	Codificar las interfaces necesarias y desarrollar los componentes o servicios que se requieran según lo han delineado los procesos identificados y descriptos dentro del modelo de integración.

Como se muestra en la Figura 24, los actores del modelo de integración trabajan de manera colaborativa en cada nivel.

Los Analistas de Negocios definen el modelo de procesos y sus instancias, en el contexto de la composición de servicios y transformaciones de datos.

Los Arquitectos del Sistema trabajan sobre los servicios a nivel de composición y luego articulan con el Desarrollador de Componentes al momento de realizar la orquestación BPEL y definir el transporte y publicación de los servicios.

El Coordinador de Servicios realiza y da soporte al despliegue a nivel de Operación y Gestión y también trabaja junto al Desarrollador de Componentes a nivel de orquestación y transformación XML.

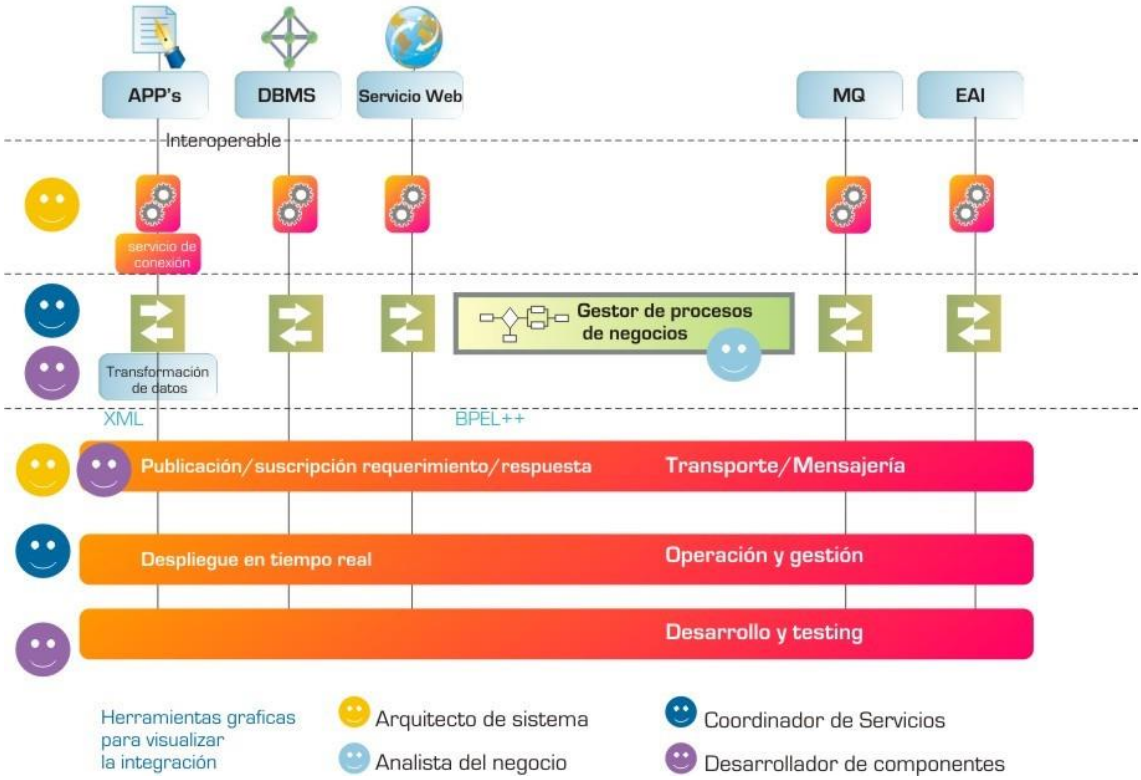


Figura 25. *Escenario de un modelo de integración y sus actores*

3.4 Conclusiones

En escenarios típicos de integración de aplicaciones, la funcionalidad de los sistemas de integración está organizada en forma de un proceso. Este proceso consiste en la ejecución de un conjunto de actividades con restricciones de ejecución, apuntando a la obtención de un objetivo. En estos escenarios la representación de los procesos está embebida en las reglas que los agentes manejan, y parece ser más apropiado alcanzar una representación explícita del mismo.

La integración por línea de base siempre es posible, pero hacer una verdadera integración significa reconocer que la orquestación de servicios y la gestión de procesos de negocios no son lo mismo y que usar adaptadores para integrar a modo de APIs directamente desde el modelo de procesos, es diferente a usar servicios de negocios gestionados por tecnología SOA, como lo es un bus de servicios o la registración de servicios. También significa reconocer

que la tarea humana juega un importante rol en muchos procesos de negocios y que cualquier plataforma de BPM debe incluir soporte para *workflow* e interacción humana.

El desarrollo de aplicaciones con enfoque orientado a procesos comparte la mayoría de los conceptos vertidos en este modelo de integración, pero aplicado a la definición, especificación, desarrollo, puesta en marcha y seguimiento de proyectos, cuyo objetivo es modelar una organización en términos de los procesos que ejecuta.

Un modelo con enfoque a procesos y a servicios cuenta con el valor agregado de que sus conceptos se pueden utilizar para el desarrollo de soluciones computacionales nuevas o bien como mecanismos de integración de aplicaciones existentes, además de permitir capturar el conocimiento subyacente en la manera en que se ejecutan los procesos de negocio.

De acuerdo a investigaciones de Gartner, la integración de aplicaciones representa un 35% del costo total en diseño, desarrollo y mantenimiento de software en una organización típica, sea esta grande, mediana o pequeña. Por lo tanto, el aspecto crucial ya no es si se debe integrar sino directamente cómo hacerlo.

CAPÍTULO 4
Herramientas BPMS

Anahí Rodríguez y Patricia Bazán

La gestión de los procesos de negocio se realiza a través de un sistema de gestión de procesos de negocio (BPMS). Según Weske [Weske M., 2008] un BPMS puede definirse como "un sistema de software genérico que se basa en diseños de procesos explícitos para promulgar y administrar procesos de negocios".

Las herramientas BPMS dan soporte a todo el ciclo de vida de los procesos de negocio y proporcionan mecanismos para mantener el ciclo de vida de los procesos de negocio y obtener mejora continua. Por ejemplo, en la etapa de configuración se puede elegir un BPMS para dar soporte a la implementación y despliegue del proceso de negocio. Luego con ayuda del BPMS se puede monitorear, realizar un seguimiento de la ejecución de cada una de las instancias del proceso de negocio y poder recolectar rastros de ejecución para retroalimentar el ciclo de vida del proceso y así tener una mejora continua.

El papel de los analistas de negocio y el personal técnico de TI queda bien determinado dado que los analistas desarrollan sus análisis y construyen los modelos de procesos sin saber cómo se realiza su ejecución. Luego el personal de TI son los encargados de diseñar y ejecutar los sistemas de información. Contar con una herramienta informática para dicho ciclo de modelado, despliegue y monitorización permite recolectar los rastros de ejecución para retroalimentar el ciclo de vida del proceso y así tener una mejora continua.

La elección de una herramienta BPMS no es una tarea sencilla, dado que la oferta de dichas herramientas es muy variada. Existen varias propuestas para la evaluación de software algunas generales, como ser la evaluación de características de calidad de software definido en la ISO/IEC 9126. Esta evaluación no es específica de los BPMS por eso es indispensable una definición de características propias de estas herramientas para poder realizar una evaluación más correcta o acertada con las necesidades de la organización. En el trabajo propuesto en [Delgado A. et al 2015] propone un total de 94 aspectos a ser evaluados entre los cuales se definen características técnicas y no técnicas.

Otros criterios bien fundamentados para evaluar tecnologías en particular, y BPMS en particular, lo constituyen Gartner, TEC y Forester.

En el caso de Gartner, se trata de una empresa pública estadounidense fundada en 1979 que se dedica a la investigación y análisis de TI para profesionales, empresas de tecnología y la comunidad de la inversión. Sus resultados los genera a partir de varios formatos como

reuniones informativas, servicios de pares en red (*peer networking service*) y programas de socios diseñados explícitamente para CEOs.

Uno de los resultados más esperados en torno a las herramientas de software generadas por Gartner, es el "cuadrante mágico" que consta de una serie de informes de investigación de mercado que se basan en métodos de análisis de datos cualitativos patentados para demostrar las tendencias del mercado, como la dirección, la madurez y los participantes. En [Gartner, 2019] se encuentra el cuadrante mágico 2019 para los BPMS.

En el caso de TEC, Technology Evaluation Centers, es una empresa imparcial de análisis de software que no representa, implementa ni comercializa soluciones de software específicas. Su misión es colaborar con organizaciones de todos los tamaños para identificar las soluciones particulares de software empresarial que mejor se alinean con sus requisitos individuales En [TEC, 2020], TEC realiza una evaluación de BPMS.

Por último, Forrester es una empresa estadounidense de investigación de mercado que brinda asesoramiento sobre el impacto actual y potencial de la tecnología, a sus clientes y al público. En particular, en torno a los BPMS, Forrester produjo el informe que se encuentra en [Forrester, 2013]

4.1. Aspectos funcionales de un BPMS

Además del trabajo de Delgado A. en [Delgado A. et al 2015] y de los aportes mencionados por TEC, Gartner y Forrester, es posible evaluar y elegir herramientas BPMS según aspectos funcionales clásicos y también no clásicos, como se propone en [Rodríguez, A. S., Bazan, P., & Díaz, F. J., 2015].

Los aspectos clásicos son aquellos se pueden vincular de manera directa con las distintas etapas del ciclo de vida de los procesos y que las herramientas BPMS deberían cubrir de manera evidente.

Los aspectos no clásicos se refieren a prestaciones funcionales que no están presentes en muchos BPMS y que aportan una ventaja adicional a los mismos.

Entre los aspectos funcionales clásicos provistos por los BPMS se pueden mencionar:

1. Obtener un gráfico que puede ser de utilidad para los dueños del negocio, como a los analistas para conocer el flujo de trabajo.
2. Simular el proceso de negocio, pudiendo utilizarlo como prueba con datos actuales e históricos.
3. Proporcionar la facilidad de crear automáticamente interfaces y reportes.
4. Proporcionar la facilidad de crear reglas de negocio, pudiendo ser utilizadas para conducir el flujo del proceso y la toma de decisiones.

5. La capacidad para la integración con otras herramientas externas, dado que muchas veces el BPMS no provee todo lo necesario para el ciclo de vida de los procesos de negocio.
6. La capacidad de enviar o recibir mensajes de eventos del sistema o del negocio.
7. La capacidad de reconocer indicadores de desempeño, que se pueden obtener de la ejecución de los procesos de negocio.

Considerando las componentes de un BPMS que llevan a cabo las características funcionales antes mencionadas, se puede concluir que los BPMS deben contemplar, al menos, las siguientes fases en la ejecución del proceso de negocio [BPEL User's Guide, 2007]

- **Modelado del proceso** - El modelado de proceso implica la definición de tareas, ordenarlas, su ramificación, definición de recursos, y otros aspectos del proceso. Con esto podemos tener un modelado del proceso. Este modelo es representado por una gráfica de un flujo y un ordenamiento de tareas involucradas.
- **Instanciación del proceso** - La instancia de un proceso de negocio es cada despliegue del modelo de procesos en tiempo de ejecución. Cada modelo puede tener diferentes instancias una por cada vez que el proceso es iniciado con un caso en particular.
- **Ejecución del proceso** – La ejecución del proceso se lleva a cabo dentro del entorno de ejecución donde se interpreta el modelo del proceso, lo inicializa y lo ejecuta de acuerdo a la información de entrada. Generalmente el entorno de ejecución incluye un motor de procesos, y posiblemente un repositorio para los mismos.
- **Monitorización del proceso** – Al monitorizar el proceso de negocio permite recolectar rastros de ejecución para proponer una mejora en el rendimiento del proceso de negocio.

Entre los aspectos no clásicos que se abordan en [Rodríguez, A. S., Bazan, P., & Díaz, F. J., 2015], se encuentran:

1. **Cobertura de las Etapas del Ciclo de Vida de los Procesos:** para el análisis de los componentes de las herramientas tendremos en cuenta las etapas de un ciclo de vida clásico cíclico, considerando las siguientes fases: Diseño y análisis - Configuración (Definición) - Promulgación (Ejecución) - Evaluación (Monitorización) [Weske M., 2008] [Bazán P.,2009].
2. **Mecanismos de Actualización del Modelo de Procesos de Negocio**: sería útil que la herramienta tenga un mecanismo colaborativo durante el modelado por parte de varios usuarios para el cambio de algún proceso o flujo de trabajo. Si la herramienta utiliza una tecnología WEB 2.0 podría mejorar el trabajo colaborativo independientemente del lugar donde se encuentre el usuario, en esto hay que tener en cuenta algunos riesgos

que debemos afrontar con respecto a la seguridad de los archivos compartidos y el acceso a los mismos [Documentación Bizagi, 2015].

3. **Mecanismos de Actualización de Instancias de Proceso**: ante las distintas instancias que se pueden crear de la ejecución del proceso de negocio, del mismo pueden surgir diferentes subconjuntos de tareas y pueden tener diferentes caminos de ejecución, por lo que los cambios en el mismo pueden ocasionar inconsistencias en el mismo. En los cambios del esquema de flujos de trabajo, hay dos cuestiones a tener en cuenta: se deben analizar la nueva situación excepcional y ver si los cambios se deben propagar hacia abajo en la implementación del proceso de negocio. Los procesos pueden respetar una estructura determinada o variar con el tiempo, ya sea porque la definición de los mismos presenta cambios por naturaleza, o los requerimientos aún no están tan definidos completamente o simplemente porque modelan una realidad que presenta cambios continuos. Los cambios a realizarse en las instancias del proceso, por cambios en el modelo por ejemplo, se pueden realizar una vez terminada la ejecución del mismo, o bien durante la ejecución de las mismas. Hacer los cambios una vez que termine la ejecución de los casos puede ser un inconveniente cuando el tiempo de ejecución de las instancias es muy largo. En ese caso, es importante considerar cuál sería el mecanismo de actualización más adecuado. Los cambios en la ejecución del proceso, en las instancias del mismo, pueden ser causados por eventos no planificados, excepciones, o información errónea en la entrada a las tareas. Ante los cambios en las instancias, la herramienta BPMS debe tener en cuenta: ¿Cómo se resuelven los cambios? ¿Qué estrategias de cambios siguen? Existen patrones de diseño que tienen en cuenta cambios en las instancias como se muestran en [Documentación Bizagi, 2015]
4. **Capacidad de Distribución de Procesos en Varios Motores**: hoy en día es muy común la colaboración entre procesos de negocio como servicios, lo que se lleva a pensar en flujos de trabajo más dinámicos y flexibles, así como la distribución de los procesos de negocio. La necesidad de escalar y la naturaleza netamente colaborativa que subyace a la ejecución de procesos de negocio, ponen en juego la necesidad de contar con entornos descentralizados que permitan optimizar el uso de la tecnología y dar respuesta a la necesidad de crecimiento de las organizaciones. En la actualidad hay varios métodos para descomponer un proceso de negocio en múltiples procesos que se despliegan en las instalaciones propias y en la nube en función del rendimiento y requisitos de sensibilidad, expresándose los mismos como anotaciones sobre las actividades y los datos. Este modelo de distribución da cuenta de la factibilidad de contar con un modelo de distribución de procesos.
5. **Integración con Portales de Autenticación SSO (*Single Sign On*):** Los sistemas SSO permiten a los usuarios utilizar los servicios de varios sitios web en diversas aplicaciones o servicios web sin identificación y contraseña si los usuarios se autentican en un sitio web. Por lo que SSO ofrece la reducción de costos y comodidad a los administradores de sitios web y a los usuarios [Dae-Hee Seo et al., 2003].

6. **Cumplimiento del Estándar BPMN**: BPMN ha sido desarrollado para proveer a los usuarios de una notación estándar, fue definido por *Object Management Group* (OMG). BPMN está dirigido a usuarios y proveedores de servicios que requieren comunicar los procesos de negocio de una forma estándar [BPEL Tutorial, 2020]. El estándar BPMN v.1.x se definió en el año 2010 luego tuvo mejoras con la versión v.2.0., algunas mejoras son [BPMN, 2019]: 1- formaliza la semántica de la ejecución para todos los elementos de BPMN, 2- agrega nuevos eventos que no interrumpen (*Non-interrupting Events*) y eventos subprocesos entre procesos, 3- genera un metamodelo formal mostrado mediante diagramas de clases, 4- define el modelo de colaboración entre procesos y 5- define el modelo de coreografía.

4.2. Comparación de BPMS

Como hemos visto, las herramientas BPMS cuentan con características funcionales particulares y constituyen un recurso muy importante en el ciclo de vida de los procesos de negocio.

En este capítulo seleccionamos un conjunto de herramientas para compararlas en cuanto al tipo de licencia que utilizan y también en cuanto a los componentes funcionales que componen su arquitectura.

La selección fue realizada teniendo en cuenta el grado de popularidad y el volumen de la comunidad que conforman sus usuarios, al momento de la redacción de este libro. También se tuvieron en cuenta herramientas de software libre o que, al menos, cuenten con una edición con licencia abierta.

En la Tabla 7 se muestran una selección de herramientas sus licencias de uso.

Tabla 7. Herramientas seleccionadas para analizar

Nombre de la herramienta	URL	Licencia
Bonita BPM	https://es.bonitasoft.com/	La edición Community tiene licencia GNU GPL v2
BizAgi	https://www.bizagi.com/	*Módulo Modeler*: FreeSoftware, *Módulo Studio:* FreeSoftware, solo permite un máximo de veinte (20) Usuarios en ambientes de desarrollo *Módulo Engine:* Propietaria
Activiti	https://www.activiti.org/	Apache 2.0
jBPM	https://www.jbpm.org/	Apache 2.0
ProcessMaker	https://www.processmaker.com/	ProcessMaker (GNU General Public License version 3.0) Nayra (Apache 2.0)

BonitaBPM

Es una herramienta con una comunidad muy activa, con contribuciones de códigos fuentes, ayudas en el foro, documentación muy completa, videos de tutoriales, etc. Además, es una herramienta muy popular y con constantes actualizaciones. Esta herramienta está desarrollada en el lenguaje JAVA Consta de cuatro ediciones:

- *Community:* Es la edición Open Source, la cual permite entre las funcionalidades más importantes: modelado, conectores, generación de aplicaciones, manejos de los procesos, etc.
- *Teamwork*: esta edición está basada en los ambientes colaborativos, y entre las funcionalidades más importantes: un repositorio compartido, productividad avanzada para desarrolladores y personas de negocio, etc.
- *Efficiency:* esta edición presenta un ambiente avanzado, además de contar con las características de la edición anterior, también presenta plantillas de procesos y perfiles personalizados, entre otras características.
- *Performance:* esta edición es para un ambiente más crítico, monitorización, tareas de gestión, entre otras características.

BizAgi

Es una herramienta muy popular también. En su sitio web se pueden encontrar cursos, foros y tutoriales para poder utilizar la herramienta de una manera fácil y guiada. Consta de una suite con tres componentes:

- *Modeler:* permite el desarrollo de modelos de procesos de manera visual (*drag and drop*) minimizando la brecha digital entre los grupos de TI y los diseñadores de procesos, colaboración en la nube del diseño. Se rige por el estándar BPMN 2.0.
- *Studio:* permite el despliegue del proceso de negocio modelado, sin necesidad de programación. Incluye toda la información necesaria para realizar la ejecución del proceso, dado permite la definición de los datos de proceso, interfaz de usuario, reglas de negocio, etc.
- *Automation:* Es el componente pago de la suite. Permite la ejecución de los procesos creados con Bizagi Studio. Bizagi Engine permite generar un portal de trabajo web. Contiene varios motores tales como Motor de Workflow, Motor de Reglas de Negocio, Motor de Asignaciones entre otros. Los usuarios finales acceden a cada uno de ellos a través de una aplicación web.

Activiti

Es una herramienta liviana, escrita en lenguaje Java. Permite la extensión de sus funcionalidades dado que el código está disponible por su licencia Apache 2.0, fácilmente integrable con aplicaciones externas. El motor de procesos se puede utilizar como servicio externo dado que tiene una API REST, pudiendo extenderlo con nuevas operaciones.

jBPM

Es una herramienta para el desarrollo y despliegue de un proceso de negocio escrita íntegramente en lenguaje JAVA, se ejecuta en cualquier JVM, posee un editor en Eclipse para la definición y creación del proceso de negocio de manera gráfica. Provee una consola desde la cual permite la gestión de las instancias del proceso, listas de tareas, y gestión de formularios. JBPM puede usarse como servicio. La herramienta realiza un registro de historial, lo que permite luego realizar consultas, monitoreos y análisis de los resultados de la ejecución

ProcessMaker

Es un proyecto que provee tanto módulos de código abierto como ediciones pagas, los productos de código abierto que proporcionan son:

- ProcessMaker BPM 4: es un paquete de software BPM compatible con BPMN 2.0. Provee una RESTful APIs para poder acceder de manera más flexible y escalable a las funcionalidades de ProcessMaker y poder realizar una integración con otras aplicaciones ya existentes.
- *Nayra:* Es un motor de flujo de trabajo BPMN 2.0 construido en PHP.
- *Conectores PM*: provee conectores ya predefinidos con los cuales se puede realizar una integración para lograr una plataforma de integración como servicio.

4.3. Mecanismos de Integración de los BPMS

Como se ha desarrollado extensamente en el Capítulo 3, la integración de aplicaciones constituye un concepto ampliamente abordado por la literatura y de gran utilidad, debido a que es muy frecuente que las organizaciones cuenten con software de distinto origen y requieran ser integrados.

Los BPMS, como herramientas de software, también tienen que poder integrarse, más aún por la función que desempeñan: administran los procesos de negocio que, como se mencionó

también en el Capítulo 3, conforman un mecanismo altamente recomendado para orquestar servicios e integrar funcionalidades prexistentes.

Entre los distintos mecanismos o tecnologías que los BPMS utilizan para resolver la integración se encuentran:

- **Conectores Nativos.** Algunas herramientas BPMS proveen conectores nativos, tales como conectores a servidores de bases de datos o a servidores de mail, con los cuales se pueden realizar las conexiones definiendo simples parámetros de configuración. Además, permiten la posibilidad invocar comandos a ejecutar en dichos servidores con la posibilidad de guardar los resultados en variables de proceso para luego su manipulación dentro del proceso de negocio.
- **Web Services.** En el Capítulo 2 se introdujo el concepto de servicio como componente funcional y se define un web service como una aplicación con la cual se puede interactuar a través de una red. Este tipo de tecnología puede ser utilizada para poder conectar distintas aplicaciones existentes. Esto permite gran interoperabilidad entre distintos sistemas dado que productor y consumidor del servicio solo requieren conocer su interface de comunicación y no la manera en que está implementado.

  En el caso de los BMPS, la integración por web service permite invocar componentes funcionales que den respuesta a la ejecución de las actividades, aunque las mismas estén desarrolladas en otra plataforma o tecnología.
- **API (Application Programming Interface).** Una interfaz de programación de aplicaciones es un conjunto de acciones o simplemente mensajes en los cuales se definen las operaciones que se pueden acceder del software o herramienta. Crear una interfaz de comunicación permite que los productos y servicios establezcan una comunicación, sin importar cómo está implementado. También se lo puede considerar como un "contrato" entre aplicaciones, dado que se establece un acuerdo entre las partes en las cuales se determina como es la comunicación.

  Las API son un medio simplificado para conectar su propia infraestructura a través del desarrollo de aplicaciones nativas de la nube, pero también le permiten compartir sus datos con clientes y otros usuarios externos. Las API públicas representan un valor comercial único porque simplifican y amplían la forma en que se conecta con sus partners y, además, pueden rentabilizar sus datos (un ejemplo conocido es la API de Google Maps)[2].

  Actualmente los BPMS, presentan una API de programación con la cual se puede acceder externamente a distintas funcionalidades, como por ejemplo, iniciar procesos o acceder a las variables, entre otros.

[2] https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces

4.4. Uso de API en BonitaBPM. Un ejemplo

El motor de procesos de BonitaBPM proporciona una API REST Web. Esta API permite el acceso a todos los objetos del motor, como por ejemplo, procesos, tareas, usuarios o conectores, entre otros, y permiten ejecutar operaciones con ellos - crear, recuperar, actualizar, eliminar-.

El componente de Bonita Engine es el encargado de ejecutar la lógica del flujo de trabajo. Los usuarios pueden gestionar procesos y tareas, y realizar actividades administrativas. La Figura 26 muestra la comunicación entre aplicaciones externas y el Bonita Engine, mediante API REST.

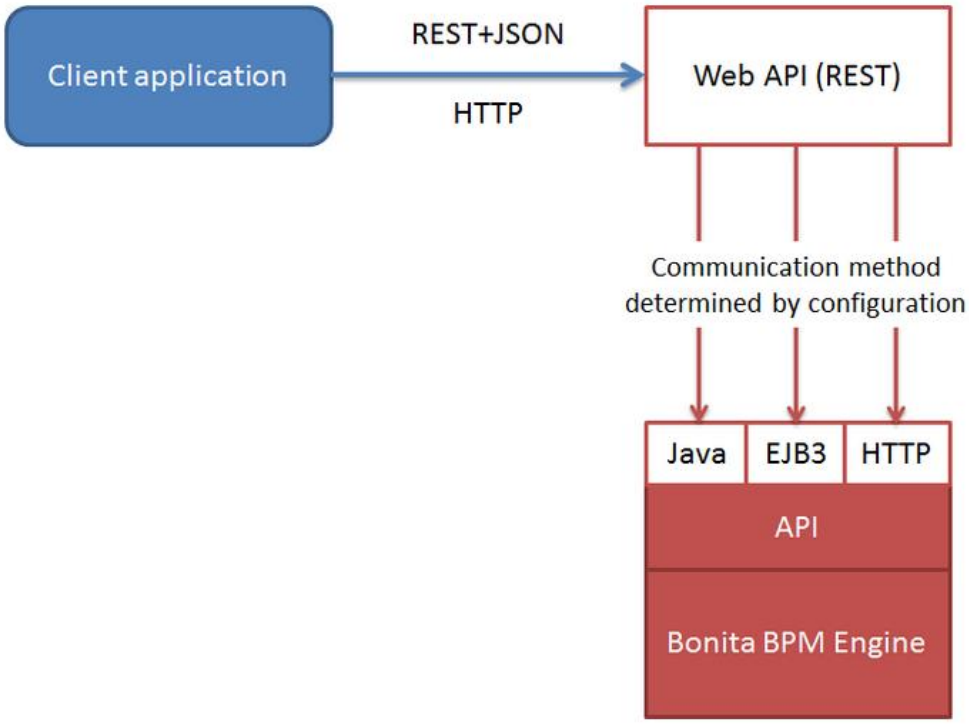


Figura 26. *Comunicación entre un cliente y Bonita Engine por API REST*

Ejemplos de peticiones y uso de API de BonitaBPM

Para poder utilizar la API REST, primero se debe iniciar sesión en el Bonita BPM Engine, de la siguiente manera:

- Request URL: http://host:port/bonita/loginservice
- Request Method: POST
- Content-Type: application/x-www-form-urlencoded
- Form Data:
  - username: a username
  - password: a password

- redirect: true or false.
- redirectURL: la URL de la página que se mostrará después de iniciar sesión

La respuesta a esta llamada genera cookies y el JSESSIONID debe transferirse con cada llamada posterior.

La creación de un Caso es de la siguiente manera:

- URL
  `/API/bpm/case`
- Método
  POST
- Cuerpo de la petición
  Id de proceso en JSON

Obtener una variable de un caso

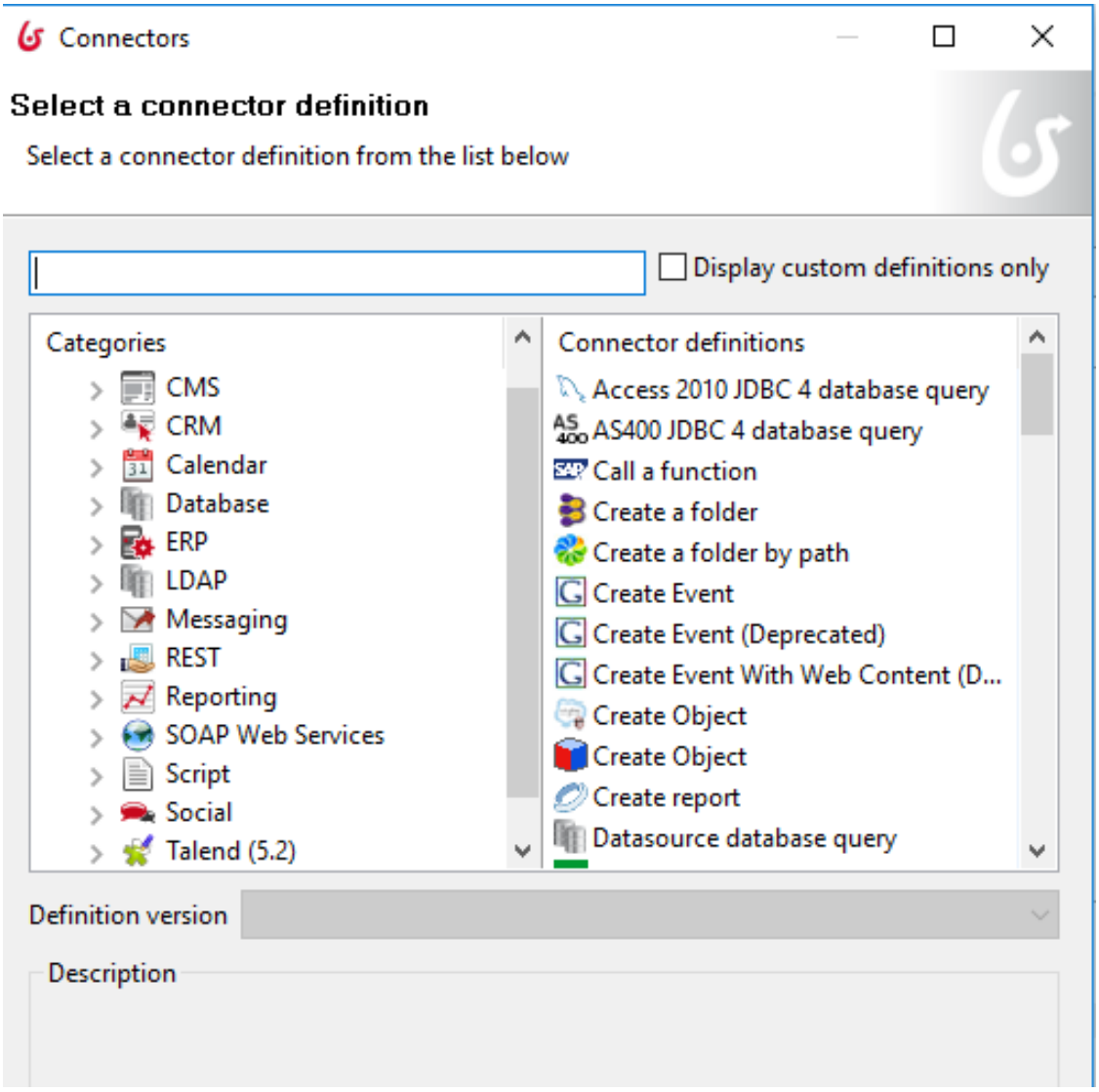
- URL
  `/API/bpm/caseVariable/:caseId /:variableName`
- Método
  GET
- Respuesta exitosa
  Código: 200

```
{
   "descripción":"",
   "name": "myInvoiceAmount",
   "value": "14.2",
   "case_id": "1",
   "type": "java.lang.Float"
}
```

Estos y más métodos se pueden encontrar en la página oficial de Bonita https://documentation.bonitasoft.com/bonita/

Conectores definidos en BonitaBPM

Los conectores definidos en la herramienta, presentan una forma *wizard* para la configuración de los mismos. En la siguiente imagen podemos ver los conectores que están disponibles en la herramienta:



Algunos de ellos son:

- Conectores a Bases de Datos: los cuales permiten realizar conexiones a varios motores populares de base de datos.
- Conectores para envío o recepción de mails: permite hacer conexiones POP3 a los servidores de mail que lo permitan.
- Conectores REST y SOAP: permiten realizar conexiones a servicios REST o SOAP.
- Conectores para Script: permite la creación de Script para su ejecución.

Utilización de conectores nativos - Invocando servicios REST

A continuación, se muestra el wizard que permite la configuración de una petición REST:

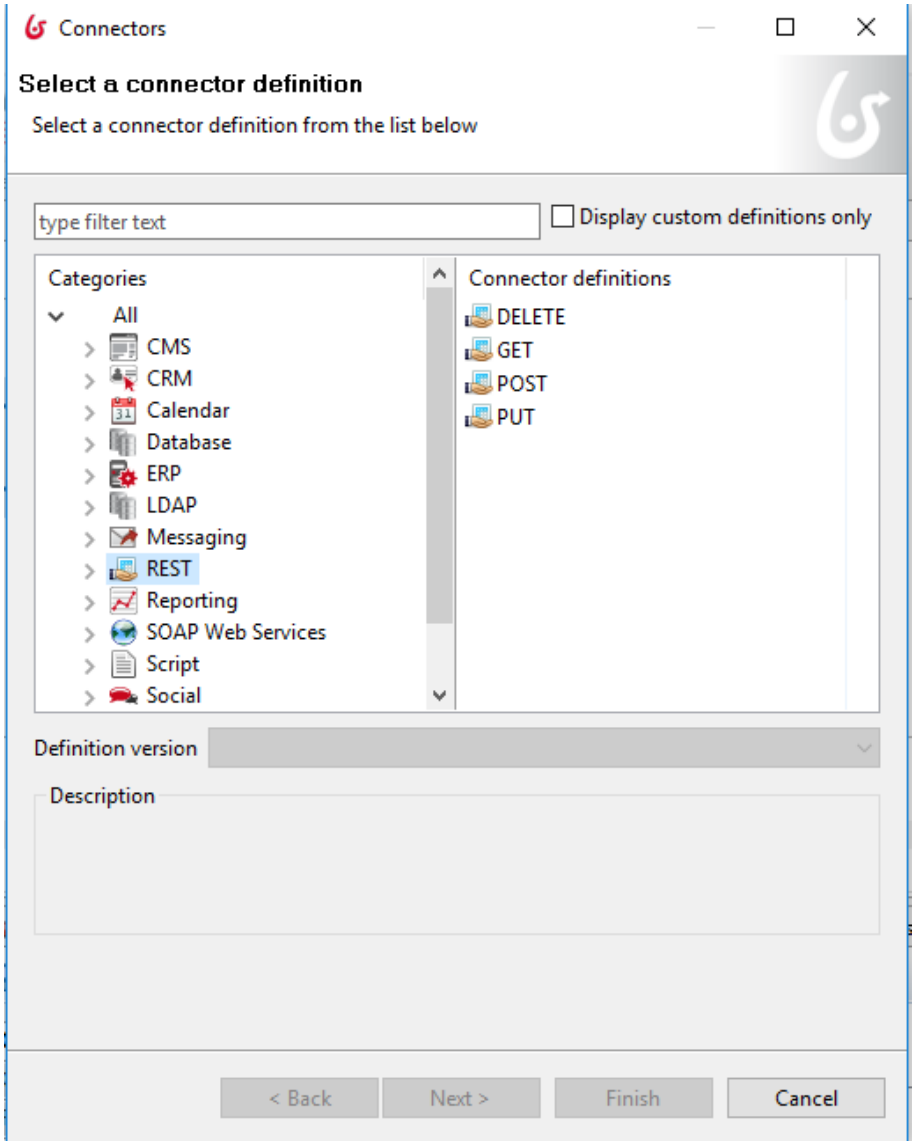


Figura 27. *Selección del tipo de petición.*

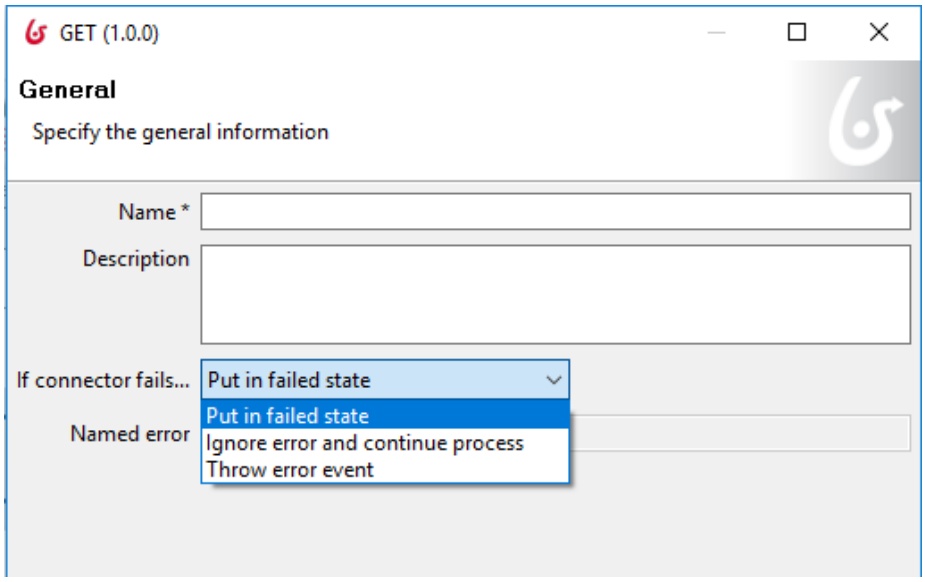


Figura 28. *Configuración del nombre y de seteos por errores de conexión.*

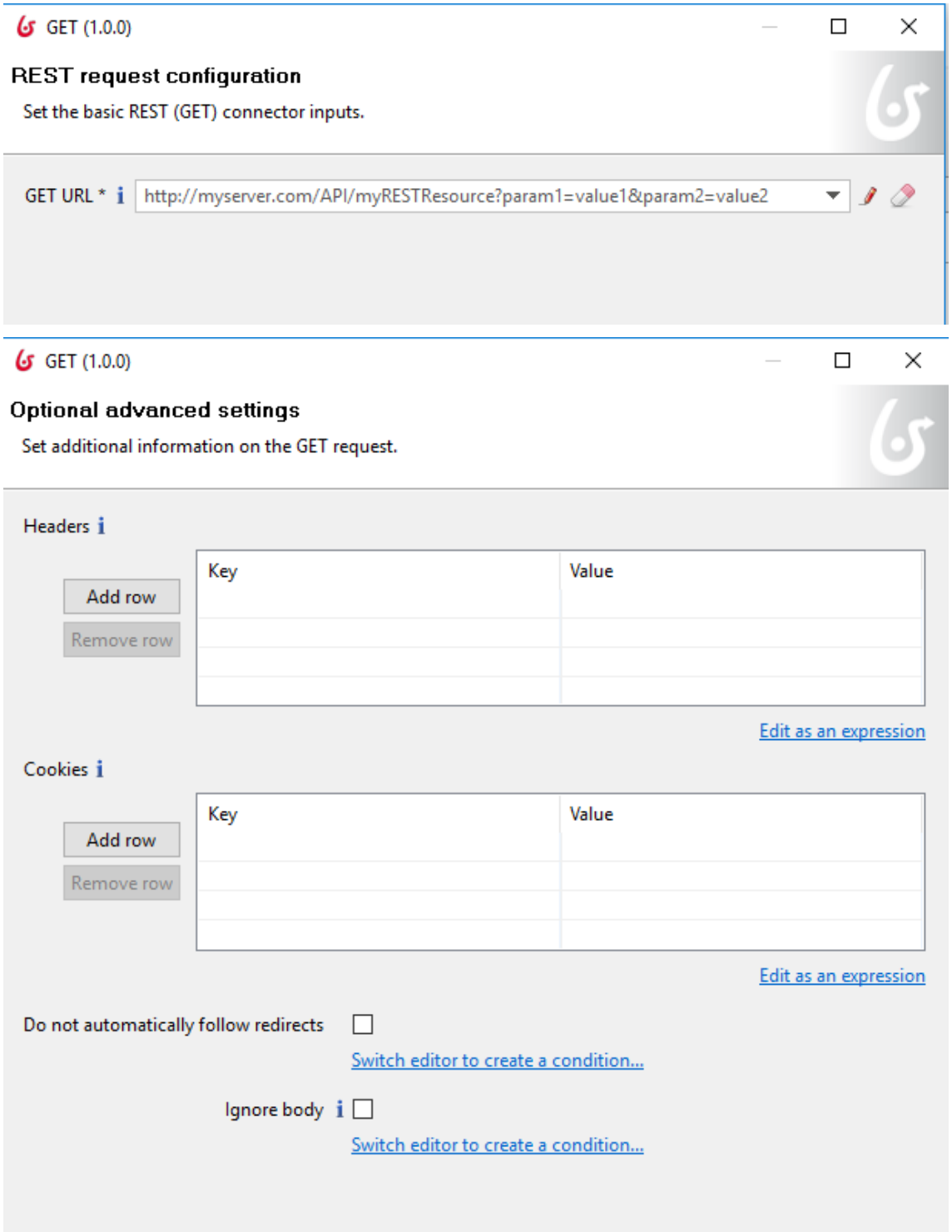


Figura 29. *Configuración del headers y cookies*

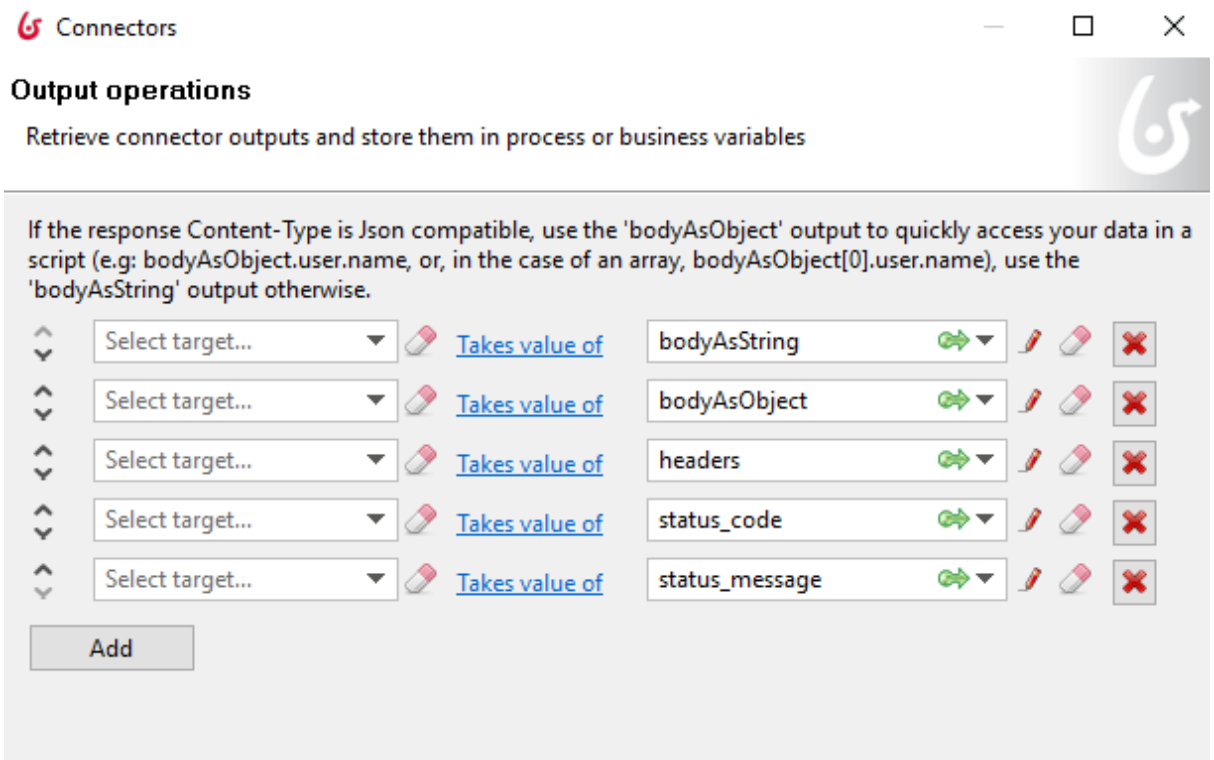


Figura 30. *Seteos de las salidas de la petición realizada, por ejemplo, a través de variables del proceso.*

Creación de nuevos conectores - Definición en una clase de JAVA

Los conectores pueden ser implementados por una clase Java. A través de este mecanismo se amplía la funcionalidad del gestor de procesos, a continuación, se puede ver un ejemplo de conexión a una Base de datos:

Los conectores pueden ser implementados por una clase Java. A través de este mecanismo se amplía la funcionalidad del gestor de procesos:

- Desde la línea de código 2 a la 13 inclusión de dependencias.
- Desde la línea de código 14 a la 21 seteo de parámetros de conexión a las variables
- Desde la línea de código 22 a la 28 manejo de excepciones para la conexión.
- Desde la línea de código 29 a la 42 seteo de parámetros para realizar la conexión.
- Desde la línea de código 44 a la 58 validación de los parámetros.
- Desde la línea de código 60 a la 67 genera la conexión.

A continuación, se puede ver el código desarrollado.

```
1. package org.bonitasoft.connectors.database.jdbc;
2. import java.sql.ResultSet;
3. import java.sql.SQLException;
4. import java.util.ArrayList;
5. import java.util.Collections;
6. import java.util.HashMap;
```

```
7. import java.util.List;
8. import java.util.Map;
9. import java.util.StringTokenizer;
10. import org.bonitasoft.connectors.database.Database;
11. import org.bonitasoft.engine.connector.Connector;
12. import org.bonitasoft.engine.connector.ConnectorException;
13. import org.bonitasoft.engine.connector.ConnectorValidationException;
14. public class JdbcConnector implements Connector {
15     public static final String USERNAME = "username";
16.    public static final String PASSWORD = "password";
17.    public static final String SCRIPT = "script";
18.    public static final String SEPARATOR = "separator";

19.    private String script;
20.    private Database database;
21.    private ResultSet data;


    @Override
22.   public Map<String, Object> execute() throws ConnectorException {
23.        if (separator != null) {
24.            return executeBatch();
25.        } else {
26.            return executeSingleQuery();
27.        }
28.    }

29.    @Override
30.    public void setInputParameters(final Map<String, Object> parameters) {
31.         userName = (String) parameters.get(USERNAME);
32.        final String paswordString = (String) parameters.get(PASSWORD);
33.        if (paswordString != null && !paswordString.isEmpty()) {
34.            password = paswordString;
35.        } else {
36.            password = null;
37.        }
38.        script = (String) parameters.get(SCRIPT);
39.        separator = (String) parameters.get(SEPARATOR);
40.        driver = (String) parameters.get(DRIVER);
41.        url = (String) parameters.get(URL);
42.    }

43.    @Override
44.    public void validateInputParameters() throws ConnectorValidationExcep-
tion {
45.       final List<String> messages = new ArrayList<String>(0);
```

```
46.         if (url == null || url.isEmpty()) {
47.             messages.add("Url can't be empty");
48.         }
49.         if (driver == null || driver.isEmpty()) {
50.             messages.add("Driver is not set");
51.         }
 52.        if (script == null || script.isEmpty()) {
 53.            messages.add("Script is not set");
 54.        }

 55.        if (!messages.isEmpty()) {
 56.            throw new ConnectorValidationException(this, messages);
 57.        }
 58.    }
 60.   @Override
 61.     public void connect() throws ConnectorException {
 62.        try {
 63.            database = new Database(driver, url, userName, password);
 64.        } catch (final Exception e) {
 65.            throw new ConnectorException(e);
 66.        }
 67.    }
```

CAPÍTULO 5
Casos de éxito y escenarios posibles para implementación de BPM

José Martínez Garro y Patricia Bazán

En este capítulo se describen algunos casos de éxito en organizaciones que han llevado adelante algún aspecto de gestión por procesos de negocio, cada una de ellas con mayor o menor grado de adhesión, pero en todos los casos, aplicando una metodología y las herramientas tecnológicas habilitantes, presentadas en este libro.

Para cada caso presentado mostraremos el objetivo y alcance del mismo, es decir el campo del dominio al que aplica, qué resuelve y hasta dónde abarca la solución. Luego se presenta la solución propiamente dicha, incluyendo la arquitectura tecnológica y proceso/s involucrado/s y finalmente abordaremos la discusión y resultados de la solución.

5.1. Caso Seguimiento de Proceso Licitatorio y Ejecución de Proyectos UEC

Sistema basado en BPM para el Seguimiento del Proceso Licitatorio y la Ejecución de Proyectos del Programa PMGM-UEC Ministerio del Interior y Transporte de la Nación Argentina

En el año 2013, los autores de este libro participaron en el desarrollo de este sistema y en 2014, el trabajo finalizado dio origen a la publicación en el 8vo SIE (Simposio de Informática en el Estado), en el marco de las 43a JAIIO (Jornadas Argentinas de Informática e Investigación Operativa) [Chedrese V. et al, 2014]

Objetivo

Es un sistema basado en BPM para el seguimiento del proceso licitatorio en una organización pública y muestra las mejoras obtenidas como consecuencia de dicha implementación.

El objetivo general del Programa de Mejora de la Gestión Municipal PMGM (BID 1855 OC-AR) es el de mejorar la capacidad de gestión de los gobiernos municipales, para que éstos respondan de forma más efectiva a las necesidades locales. Sustentado por el BID (Banco Interamericano de Desarrollo), el Programa de Mejora de la Gestión Municipal PMGM (BID

1855 OC-AR) tiene como objetivo general el de mejorar la capacidad de gestión de los gobiernos municipales, para que éstos respondan de forma más efectiva a las necesidades locales. Para ello, el BID ha asignado un monto de u$s 80.000.000, por única vez, para el desarrollo e implementación de proyectos.

El Programa propone desarrollar e implementar mecanismos replicables de fortalecimiento municipal en las áreas de administración interna, finanzas, tributación, catastros, servicios de atención al ciudadano, gobierno electrónico, planificación urbana y promoción económica local, entre otras [3],[4]

Este programa contempla intervenciones coordinadas en los tres ámbitos del gobierno (nacional, provincial y municipal), tendientes al fortalecimiento de las capacidades de los gobiernos municipales. Las actividades a ser financiadas son de dos tipos:

(i) Proyectos de iniciativa provincial, orientadas a mejorar el marco de las políticas de descentralización, ampliar el alcance de los sistemas de gestión pública desarrollados por la provincia, implementar proyectos de apoyo a las municipalidades y desarrollar instrumentos de gestión que requieren soluciones de escala.

(ii) Proyectos de desarrollo institucional identificados por los propios municipios, en temas como planificación urbana, centros de atención de trámites, preparación de proyectos y otros. Adicionalmente, el Programa incluye un componente de apoyo a actividades de promoción, fortalecimiento y seguimiento del sector municipal en general, a ser ejecutado por el Ministerio del Interior.

La Unidad Ejecutora Central-UEC es la encargada de coordinar y supervisar la ejecución de los proyectos pertenecientes al Programa. La misma está constituida por varios sectores funcionales y se relaciona con entidades externas como las Provincias y el BID, que deben interactuar fluidamente para que se cumplan los objetivos de la Unidad.

Alcances

Analizando las actividades de cada sector y la forma en que éstos se comunican, aparece una estructura basada en los procesos funcionales. Si bien al momento de iniciar este trabajo no existía un buen grado de formalización de los procesos dentro de la UEC, el Programa busca dentro de este contexto, implementar su funcionamiento en base a procesos de negocio.

Es posible considerar entonces al Programa constituido por cada uno de sus sectores, que se relacionan entre sí, que intercambian documentación e información con las otras áreas de la UEC, con la participación de entidades externas como las Provincias y el Banco, consolidando

[3] http://www.uec.gov.ar/uec2009/web/index.php
[4] http://xgob.blogspot.com.ar/2013/06/programa-de-mejora-de-la-gestion.html

distintos procesos operativos. Este funcionamiento integral del Programa, con el aporte de valor de todos los participantes y basado en procesos operativos, se encuadra en lo que se denomina una gestión por procesos.

Luego de un primer análisis, resultó evidente que la gestión por procesos permitiría a la UEC mejorar la gestión de sus actividades, de los recursos utilizados, del tiempo de respuesta de los servicios que presta y del control en general. Para comenzar este trabajo, fue necesario identificar, estudiar y adecuar los procesos del programa con todos los participantes, internos y externos, para que pudieran implementarse operativamente en una herramienta informática de características colaborativas y especialmente diseñadas para la operación y administración de los procesos.

La premisa principal fue la utilización de herramientas de software libre, tanto para la gestión de los procesos, la administración de la documentación asociada a los mismos y el desarrollo de las aplicaciones que, a manera de servicios, abastecieron a las actividades de los procesos.

Solución propuesta y metodología

La metodología aplicada se muestra en la Tabla 8 que enuncia las etapas y en la Figura 31 que representa a las mismas agrupadas en cuatro grandes pasos y donde se puede observar la retroalimentación en las fases de Descubrimiento de Procesos y Definición de los procesos buscando una versión mejorada de los procesos que sea entrada para las fases de codificación de procesos y actividades, al modelado de los datos y a la especificación e implementación de servicios y formularios. Estas tres etapas también se retroalimentan entre sí, aunque de una manera secuencial y no cíclica. El resultado de estas tres etapas, llevan a la fase de pruebas y finalmente al test, puesta a punto y puesta en marcha.

Tabla 8. Etapas de desarrollo del proyecto

Etapa	Descripción	Producto resultante
Descubrimiento de procesos en la organización	Relevamiento con los actores principales a fin de detectar los pasos que ellos realizaban cotidianamente para cumplir sus funciones y derivar desde allí un diseño preliminar de procesos	Mapa de Procesos
Definición de los procesos	Definición de actividades y participantes, identificación de actores externos, la información que se intercambia y las variables de proceso involucradas.	Diseño global de los procesos
Codificación de procesos y actividades	Construcción del diccionario de procesos, actividades y reglas de negocio permitiendo así catalogar los mismos e identificar su transversalidad.	Diseño de detalle de los procesos,

Modelo de datos	Identificación de entidades, relaciones y diccionario de datos a fin de generar un modelo de datos relacional y su despliegue en una base de datos relacional.	Modelo de entidad/relación, casos de uso funcionales.
Servicios	Diseño de servicios se circunscribió a definir y codificar formularios, definir su interfaz, establecer las variables de conexión con las actividades y conformar un diccionario de formularios.	Modelo de Servicios
Prueba preliminar	Revisión final del diseño de proceso, integración de los servicios (en este caso formularios) y el despliegue y ejecución de los procesos en el BPMS.	Solución desplegada
Test, puesta a punto y puesta en marcha	Etapa clásica de cualquier desarrollo de software	Solución en producción

.

Figura 31. *Metodología aplicada en UEC*

El desarrollo de la solución se inició con la realización de un diagnóstico de la situación vigente en la UEC, orientado a detectar y aplicar una reingeniería de los procesos de negocio ejecutados informalmente en el Programa. Para ello, se llevaron a cabo un conjunto de tareas técnicas de relevamiento, análisis y diagnóstico de los circuitos administrativos y de gestión del Programa, destinadas a determinar el estado actual y proponer las medidas necesarias para el procesamiento de los requerimientos de los productos internos y finales, que son objeto del PMGM.

Junto a los responsables del Programa se intentó determinar mejoras o rediseños en los procesos ya detectados por ellos y definir explícitamente los objetivos a lograr por cada proceso bajo estudio, dentro del plan global del PMGM.

Asimismo se buscó seleccionar el personal clave de cada proceso, para obtener los conocimientos y el compromiso de los participantes que conocieran acabadamente los distintos pasos de los procesos.

Se elaboró un detallado diagrama preliminar de relaciones entre procesos (mapa de procesos) y de cada proceso en particular, utilizando el lenguaje BPMN que suministra el modelador de la herramienta BPMS. A través de los diagramas de procesos, se describieron secuencialmente los pasos que los departamentos, áreas o sectores ejecutan para convertir los insumos en productos tal como se desarrollaba en el momento de encarar la tarea de mejora, para facilitar su comprensión y análisis por parte de los analistas.

En la Figura 32 se observa el mapa de los procesos que estuvieron involucrados dentro del alcance de este trabajo. Cada uno de estos sub-procesos fue explotado y refinado dentro de las fases de Descubrimiento y Definición de procesos.

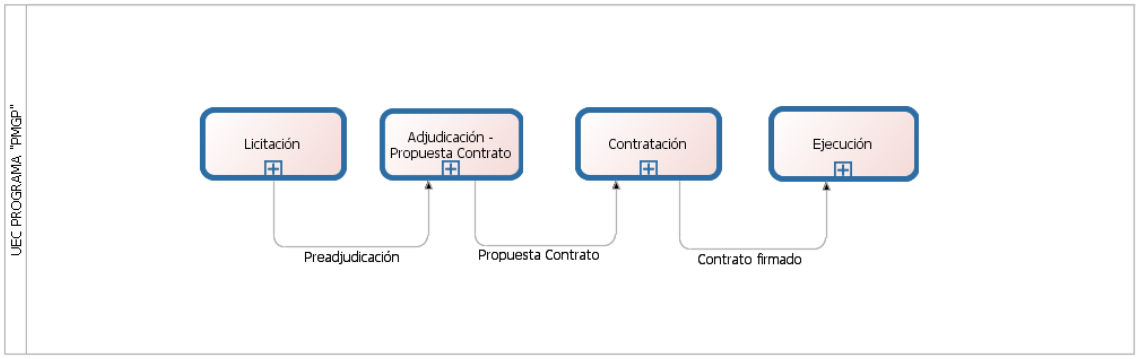


Figura 32. *Mapa de Procesos del PMGM*

Se analizó la relación entre los procesos y la información, tratando de determinar la manera de combinar la tecnología de información con el diseño de los procesos y la organización de las personas. Ello permitió detectar las disfunciones con respecto al objetivo principal y trazar escenarios posibles, orientados a mejorar el análisis de la información, optimizar la toma de decisiones, definir o cambiar actividades en forma radical, o suprimir flujos de trabajo que gracias a la tecnología son superfluos o innecesarios.

Finalmente, se elaboró una propuesta de diseño o reingeniería con modificaciones o ajustes, explicitando claramente los objetivos y metas a alcanzar en los procesos. Estas propuestas de cambios se realizaron verificando que no estuvieran contrapuestas con normativas legales y de auditoría que rigen el funcionamiento de los procesos. Asimismo se incluyó una descripción de los posibles resultados de este rediseño en términos de mejora tales como reducción de tiempos de proceso, eliminación de pasos innecesarios, definición de puntos de control, notificaciones a los participantes clave y tiempos límite.

En la Figura 33 se observa el detalle del Proceso de Consulta de Oferentes obtenido como resultado de la reingeniería.

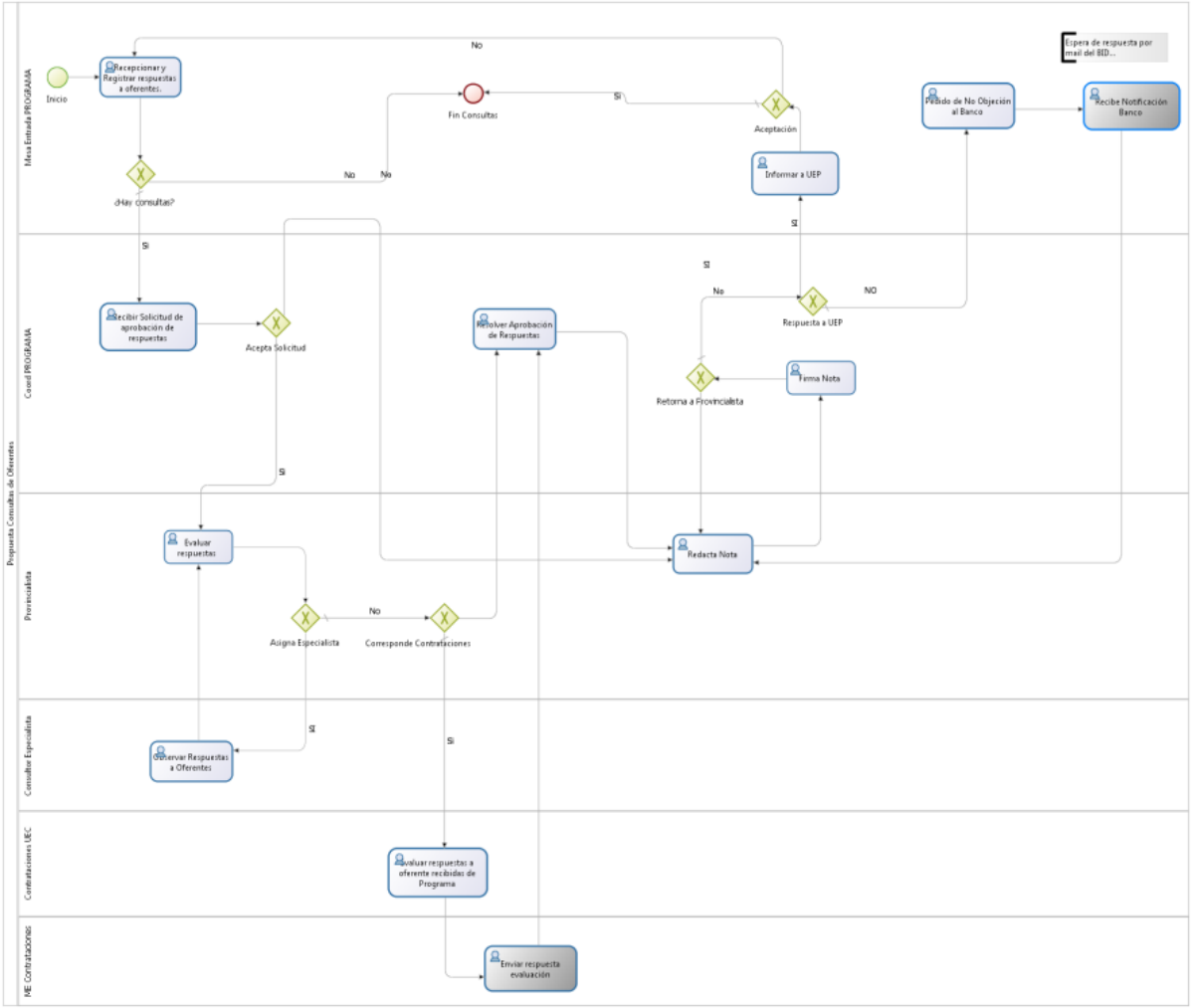


Figura 33. *Proceso de Consulta de Oferentes*

Arquitectura Tecnológica de la Solución

La solución incluye la instalación del software de base sobre el hardware que puso a disposición la UEC para este proyecto. Dicha instalación se realizó en base a las indicaciones y supervisión directa de los responsables del Data Center de la UEC, de manera que las instalaciones de software necesarias para el proyecto, no interfieran en el funcionamiento operativo de la UEC.

Se instaló un ambiente tecnológico (de desarrollo, pruebas y producción) para la herramienta BPMS. Este ambiente permitió realizar las pruebas operacionales de los procesos desarrollados.

Las herramientas utilizadas para dar soporte a la solución y permitir llevar a cabo la metodología planteada son: Bonita Open Solution: suite BPMS (Sistema de gestión de procesos de negocio).de código abierto que combina el diseño gráfico de los procesos de negocio siguiendo el estándar BPMN (Notación para el modelado de procesos de negocio), un motor de ejecución y una interfaz de usuario sencilla y configurable. El usuario puede crear

conectores que permiten comunicar Bonita con otras aplicaciones y compartirlos en forma colaborativa a través de la gran comunidad de usuarios de Bonita. Alfresco Community Edition - On-Premise: gestor documental de código abierto. Proporciona todos los servicios de gestión de documentos, permitiendo capturar y guardar archivos electrónicos, previsualizar los mismos y gestionar su versionado en distintos repositorios, administrando usuarios y roles. Lenguaje PHP: lenguaje de programación orientado al desarrollo de aplicaciones web dinámicas, que puede ser usado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. Tiene capacidad de conexión con la mayoría de los motores de base de datos, especialmente con MySQL. MySQL: sistema de gestión de bases de datos de código abierto y uso gratuito. Apache Tomcat: como servidor Web y servidor de aplicaciones

Como se observa en la Figura 34, la ejecución y compleción de las actividades se resolvió por medio de la invocación de servicios web vinculados al motor de procesos a través de variables de proceso, conectores y validadores apropiados. El acceso al gestor documental y al servidor de correos se realizó íntegramente desde los formularios PHP, resultando transparente para los usuarios que sólo deben autenticarse al ingresar a la aplicación a través del motor de procesos.

Figura 34. *Solución propuesta en UEC*

Discusión de Resultados y Conclusiones

El PMGM es sólo uno de los programas federales de distintos organismos financieros internacionales que lleva adelante la UEC, abarcando 24 provincias y 250 municipios. El trabajo

llevado a cabo, servirá indudablemente como prueba piloto para la futura generalización en esa organización de la gestión por procesos.

Por otra parte, el uso de software libre se vislumbra como doblemente ventajoso en un organismo del estado, tanto desde el punto de vista de los costos como de la independencia tecnológica, entre otros beneficios.

Sin lugar a dudas, la representación y coordinación de las actividades de los procesos, junto con su despliegue en una versión ejecutable realizada íntegramente en herramientas de software libre y orientada a servicios, constituye en el PMGM un cambio tecnológico y operativo que permitirá verdaderamente instrumentar una gestión por procesos posibilitando llevar a cabo una tarea más ágil, mejor supervisada y más eficaz, que permita responder rápidamente a los cambios y desafíos que se presenten en el futuro.

Los procesos instrumentados en esta solución constan de la ejecución de varias actividades que involucran a distintos participantes. Estos son, la totalidad de dependencias del PMGM, consultores especialistas, los solicitantes de los préstamos (Provincias y Municipios) y el mismo BID. Todos participan efectuando actividades para alcanzar el objetivo final del Programa.

El uso de BPMS ha posibilitado adelantos importantes en cuanto a la velocidad y agilidad con que las organizaciones mejoran el control y rendimiento de sus procesos, logrando instrumentar metodologías que permiten alinear esfuerzos entre la conducción y los participantes de las distintas actividades, mejorando la productividad y el rendimiento personal.

La implementación de esta solución con enfoque de procesos ha tenido un fuerte impacto en la organización que se puede evaluar en torno a las siguientes mejoras obtenidas:

1. Se eliminaron, en la tarea de reingeniería de los procesos, tareas que resultaban superfluas o solapadas. Algunas entradas/salidas que no justificaban su soporte en papel, fueron reemplazadas por notificaciones electrónicas. Se alivianó la labor del Coordinador del programa, que centralizaba una serie de tareas que fueron reasignadas a otros participantes, canalizando el trabajo de control por medio de procesos de notificación.
2. Se utilizó al pliego licitatorio como hilo conductor de una única instancia de proceso, por lo que sólo con disparar el proceso que registra el alta, la secuencia de actividades queda administrada íntegramente por el motor de procesos hasta el final de la ejecución del pliego. Cada actividad completada genera un movimiento sobre el pliego que permite en todo momento conocer la evolución sufrida por el mismo y los actores intervinientes.
3. Toda la gestión documental se realiza dentro del sistema, registrándose en el gestor documental tanto los documentos digitales incorporados manualmente, como los generados desde los distintos servicios, tales como notas o correos electrónicos, permitiendo asociar metadatos a los documentos. Desde cualquier formulario, el usuario tiene la

posibilidad de consultar en línea la documentación necesaria para la toma de decisiones y agregar la que considere oportuna.

4. Se incorpora la redacción de notas desde los formularios, en los que se ofrecen al usuario modelos predefinidos almacenados en la base de datos, clasificados de acuerdo a la actividad de proceso que los invoca. El participante puede completar el texto según su necesidad en un editor que suministra cada formulario. Asimismo, se provee la firma digital de notas, en la que el responsable adjunta al documento la imagen con su firma, asociada previamente a su cuenta de usuario de la aplicación.
5. El sistema permite la lectura, la redacción en base a modelos predefinidos y el envío de correos electrónicos a cuentas prefijadas o agregadas por el usuario, ofreciendo la posibilidad de adjuntar a los mismos cualquier documentación residente en el gestor documental. Esta funcionalidad se encuentra embebida en los formularios asociados a la actividad respectiva.
6. Las notas y emails enviados, se incorporan automáticamente al gestor documental.
7. Se incorpora un sistema de notificaciones automáticas que se muestran al usuario como actividades pendientes en una bandeja especial y que se guardan en un repositorio de notificaciones con un estatus de enviadas o leídas. Estos avisos, además de informar distintas situaciones, evitan la emisión y circulación de memorandos impresos.
8. Es posible definir fechas límite (deadlines) para la compleción de las actividades. Las tareas que se encuentran atrasadas o en riesgo aparecen en bandejas de entrada específicas y se notifica de esta situación a los responsables correspondientes.
9. El usuario tiene acceso desde cualquier formulario al diccionario de reglas de negocio, construido a partir de la normativa vigente para el Programa.
10. Se provee un proceso de consulta, que permite conocer en todo momento y en forma centralizada, la marcha de los proyectos del Programa ya sea en su faz administrativa como en la operativa.

La instrumentación de esta solución permite gestionar y monitorear 84 proyectos con sus respectivos pliegos licitatorios.

El sistema cuenta con cerca de 20 usuarios simultáneos, participando de las actividades de los procesos que se desplegaron.

La solución incluye el despliegue y ejecución de 18 procesos y subprocesos, cada uno de los cuales cuenta con 8 actividades en promedio.

Estos números permiten tener una medida de magnitud de las actividades que se encontraran centralizadas y conducidas solamente por el Coordinador del proyecto y que ahora se han distribuido entre varios participantes, además de contar con un conocimiento explícito de cuáles son los pasos que la organización lleva a cabo, efectivamente, para el seguimiento de los pliegos licitatorios.

5.2. Caso Solicitud de Compras

En el año 2013, los autores de este libro participaron en el desarrollo de un proceso de negocio que cubra el circuito de compras en un organismo de la administración pública nacional, en Argentina.

Objetivo

El objetivo del proyecto fue definir e implementar una arquitectura para el ambiente de desarrollo, prueba y producción, con componentes de software libre, integrado principalmente un BPMS y utilizando como repositorio la base de datos Oracle del organismo.

Asimismo, se realizó la reingeniería de cinco procesos seleccionados, su posterior desarrollo en el BPMS y la implementación de los mismos en el ámbito del organismo.

En este libro se describe el resultado del diseño e implementación de uno de los procesos seleccionados.

Alcances

Los alcances que se plantearon para el proyecto fueron los de seleccionar una herramienta BPMS de software libre, que mejor se adapte para resolver los problemas del organismo y particularmente de los cinco procesos objeto del proyecto.

En base a dicha selección, trabajar sobre la instalación de un ambiente tecnológico (de desarrollo, pruebas y producción) para la herramienta BPMS seleccionada, dentro del data center que utiliza el organismo para su funcionamiento.

Asimismo, se realizó el relevamiento de los cinco procesos seleccionados, la aplicación de una ingeniería de procesos sobre los mismos, el diseño, implementación y puesta en marcha de la solución, para tales cinco procesos.

Por último, se desarrolló la capacitación, a las áreas pertinentes del organismo en el nuevo modelo de gestión y en las herramientas tecnológicas utilizadas.

Los procesos involucrados en la solución y que se identificaron como prioritarios en su momento para el organismo fueron: 1 - Solicitud de Compra, 2 - Compras por Licitación Pública, 3 - Compras por Licitación Privada, 4 - Otorgamiento de Licencias para Pilotos y 5 - Transferencia de Aeronaves.

Se describe en este capítulo el proceso de Solicitud de Compra.

Solución Propuesta - Solicitud de Compra

El proceso de Solicitud de Compra comienza, luego de una serie de tareas previas, en el Área requirente quien confecciona las Especificaciones Técnicas para el Pliego de Bases y

Condiciones Particulares del bien o servicio requerido, así como la Solicitud de Gasto necesaria para la reserva de crédito por parte de Presupuesto.

Pueden ser Requirentes cualquiera de las Direcciones del organismo que son quienes pueden caratular expedientes en el Sistema de Gestión de Expedientes propio del organismo.

El Requirente deberá adjuntar la documentación necesaria para iniciar la compra. Para ello tendrá acceso al gestor documental, donde podrá incorporarse la documentación asociada al expediente de la compra. De la misma manera, en cualquier actividad del proceso podrán adjuntarse informes o redactarse observaciones antes de completar la misma.

De corresponder la adquisición de productos informáticos, las Especificaciones Técnicas deberán ser aprobadas por la ONTI.

Se deberá requerir la autorización de la Administración Nacional y de la Dirección Nacional o General de la cual dependa el originante primario de la necesidad.

El área de Suministros analizará las Especificaciones Técnicas y, de ser necesario, solicitará al Área requirente las correcciones pertinentes.

Una vez puestas a punto y autorizadas las Especificaciones Técnicas y la Solicitud de Gasto, el Área requirente inicia el expediente de compra y remite el pedido de autorización del gasto a Presupuesto.

Si Presupuesto autoriza el gasto, el proceso finaliza para comenzar la confección del pliego. De lo contrario, vuelve al inicio para que el Área requirente quede en espera de crédito.

A continuación, describimos cada una de las actividades:

Confección ET y SG

El Área requirente redacta las Especificaciones Técnicas y la Solicitud de Gastos. Ingresa el objeto de la contratación, el monto preventivo, la unidad requirente y realiza una fundamentación y necesidad de tramitación excepcional, si correspondiera. Adjunta los documentos redactados al gestor documental.

Recepción de ET y SG

En esta instancia, es posible cancelar el proceso si por alguna razón no se continuara con la compra; esta situación será indicada por el participante, quien deberá ingresar el motivo, quedando registrado el mismo junto al usuario que ordenó la cancelación. Si el proceso continúa normalmente y no había sido ingresado anteriormente, se incluye el objeto de la contratación, el monto preventivo y la unidad requirente. Asimismo, se indica si la Administración Nacional ya ha autorizado la compra y si la misma requiere la intervención de ONTI. De corresponder, se indican las direcciones intervinientes, distintas al Área Requirente.

Autorizar

En el caso en que la compra esté relacionada con una Regional o requiera la intervención de una o más Direcciones Centrales -distintas del Área requirente-, se requerirá la autorización de la misma por parte de cada una de ellas, pudiendo adjuntarse en cada caso los documentos

relacionados. Si alguna de las áreas no autorizara la compra, el expediente retorna al área Requirente para que ésta realice las correcciones indicadas.

Inicio de expediente

El Área requirente inicia el expediente en el sistema de expedientes e ingresa los datos del mismo en el sistema.

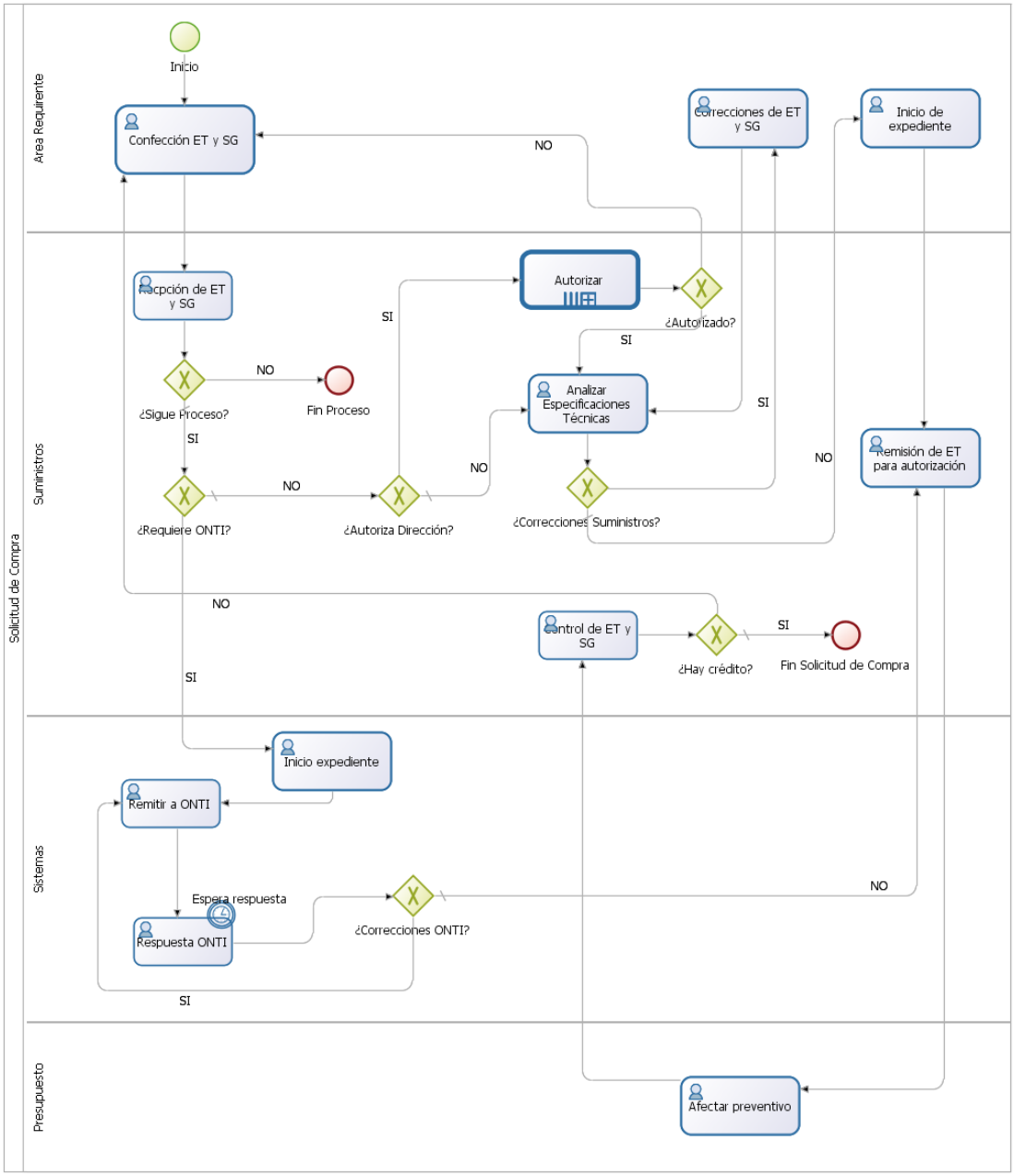


Figura 35. *Diagrama del Proceso de Solicitud de Compra*

Remitir a ONTI

Cuando el objeto de la contratación está referido a Sistemas Informáticos, el área de Sistemas debe remitir las Especificaciones Técnicas y la Solicitud de Gastos a la ONTI para su evaluación. Envía documentos a la ONTI vía email.

Respuesta ONTI

El sector de Sistemas espera la recepción de un email por parte de la ONTI con el resultado de la evaluación sobre las Especificaciones Técnicas y la Solicitud de Gastos. Si la respuesta es positiva, el proceso continuará a pedir la autorización de C35. De lo contrario, volverá al Área de Sistemas para que ésta realice las correcciones determinadas por el organismo.

Analizar especificaciones técnicas

Una vez autorizada la compra –si hubiera correspondido- Suministros continuará con la actividad de evaluación de las especificaciones técnicas. Consulta las Especificaciones Técnicas y la Solicitud de Gasto en el gestor documental y realiza el análisis de las mismas. Deberá verificar si hay desdoblamiento, si se efectuó la catalogación, el encuadre legal y si debía intervenir la ONTI. Ingresará número de catálogo. Indicará si corresponden correcciones o datos complementarios. Si corresponden modificaciones, puede elaborar un informe o ingresar observaciones al respecto y el proceso volverá al Área Requirente para que ésta realice las correcciones indicadas. Si el análisis es positivo, informará la autorización para que el Área Requirente inicie el expediente.

Correcciones de ET y SG

El Área requirente realiza las correcciones indicadas por Suministros a las Especificaciones Técnicas y/o la Solicitud de Gasto, adjuntando las nuevas versiones de los documentos al gestor documental.

Remisión de ET para autorización

Con el número del expediente iniciado, Suministros ingresa la fecha estimada de pedido de autorización de C35, la fecha estimada de adjudicación y el número de Solicitud de Gasto. Redacta nota a Presupuesto para pedido de autorización del gasto.

Afectar preventivo

Presupuesto verifica la existencia de crédito y de ser suficiente, afecta el presupuesto preventivo autorizando la Solicitud de Gasto. Indica si hay crédito o no y si el mismo está sujeto a una modificación.

Control de ET y SG

Luego de recibir el expediente físico, Suministros controla que las Especificaciones Técnicas y la Solicitud de Gasto que se encuentran en el gestor documental coincidan con las con-

tenidas en el expediente físico. Controla foliatura, inicialado y firmas. Si Presupuesto indicó que hay crédito o el mismo está sujeto a una modificación presupuestaria, el proceso finaliza para continuar con la confección del pliego. De otra manera, el expediente permanece en el área requirente a la espera de crédito.

Arquitectura Tecnológica de la Solución

La solución planteó una arquitectura basada en servicios desarrollados en tecnología .NET con persistencia de datos en una base de datos Oracle. Ambas, tecnologías ya existentes en la organización.

El motor de procesos de negocio utilizado fue Bonita BPM, elegido como resultado de un análisis de herramientas que se realizó como parte del proyecto (en la próxima Sección se describe dicho análisis).

Se utilizó un gestor documental como repositorio de la información asociada con los trámites y actividades de los procesos. Dicho gestor fue OpenKM.

El portal de procesos se construyó utilizando la consola de Bonita BPM con una personalización de los colores y el módulo de monitoreo utilizado fue el provisto por el mismo gestor de procesos, Bonita BPM.

La Figura 36 muestra la arquitectura descripta.

Discusión de Resultados y Conclusiones

Los resultados del desarrollo de este proyecto, así como las principales conclusiones han girado en torno a dos aspectos: 1 - la definición de criterio de flujo de los procesos y 2 - el análisis y elección del BPMS.

Definición de flujo de los procesos

Una característica de la gestión por procesos es que obliga a establecer un criterio de flujo de proceso que sea utilizado para la crear las distintas instancias del mismo.

Se define como instancia de un proceso a la secuencia de pasos que deben cumplirse desde que el mismo se inicia hasta que finaliza.

En el caso de este organismo, se ha definido al expediente como dato director de las instancias de procesos.

La instancia de proceso asociada a un expediente comienza cu-ando se registran los datos del mismo en el sistema, y finaliza con la salida o archivo del expediente.

La secuencia de actividades a completar por los distintos participantes dependerá del tipo de expediente.

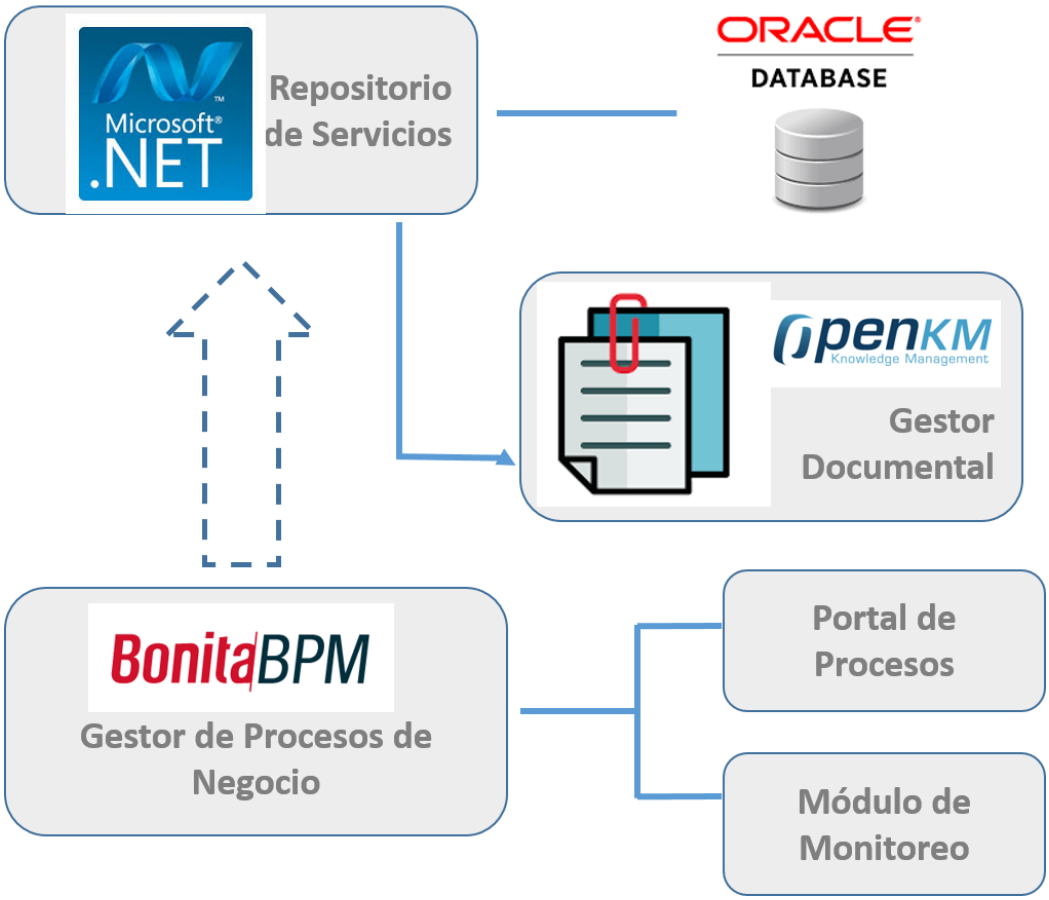


Figura 36. *Arquitectura tecnológica en el organismo*

Los movimientos sobre los expedientes se generan cada que vez que un participante completa una actividad, quedando registrada la información tanto del movimiento como de la información adjunta al expediente. También se actualiza el estado del expediente que coincidirá con el resultante de la operación efectuada.

Análisis y elección de BPMS

El análisis realizado para poder elegir el BPMS adecuado tuvo en cuenta, inicialmente, las características de la organización y también ciertas pautas acordadas con dicha organización, para la selección.

Entre las características organizacionales que influyeron en la elección podemos enumerar:

- Se trata de una organización moderna y comprometida con el concepto de gobierno digital.[5]
- Es un organismo descentralizado que requiere funcionalidad para atención al ciudadano.
- La tecnología utilizada por los activos de software están basadas en .net, Oracle 11g y Microsoft Windows.

[5] Gobierno electrónico https://es.wikipedia.org/wiki/Gobierno_electr%C3%B3nico

En relación a las pautas, acordadas con la organización que se debían tener en cuenta en la selección se encuentran:

- Tipo de organización: gubernamental, descentralizada.
- Uso de software libre
- Modelador de procesos incluido en la herramienta
- Escalabilidad a versión comercial
- Comunidad muy activa tanto de desarrolladores como de usuarios.
- Interfaz ágil para conexión con otros productos.

De acuerdo al análisis realizado sobre las características de la organización y de las prestaciones de las distintas herramientas de tipo Open Source disponibles, se eligió Bonita BPM es el BPMS más apropiado para su utilización en el organismo.

Bonita Open Solution es un software de código abierto editado bajo licencia GPLv2. Combina tres herramientas en una: un Studio para el diseño de procesos, un motor de ejecución de procesos y una interfaz de usuario sencilla.

Las características que influyeron en su elección fueron que se trate de una herramienta gratuita, adherente al estándar BPMN2, totalmente basada en web, con API Web Services (REST y SOAP) para conectarse con otras aplicaciones, más de 100 conectores incorporados y una comunidad amplia y activa. Asimismo, es escalable a versiones pagas.

En el marco del proyecto desarrollado en este organismo, se realizó una comparación exhaustiva entre Bonita y JBPM (herramienta que, al momento de desarrollo del proyecto, contaba con la popularidad necesaria para ser considerada como opción).

La Tabla 9, muestra algunos de los resultados de la comparación.

Tabla 9. Comparación Bonita BPM y jBPM

Aspecto	Bonita BPM	JBPM
Sitio Web	http://es.bonitasoft.com/	http://www.jboss.org/jbpm/
Versión estable (2013)	6.0	5.4
Lenguaje de modelado	BPMN2	BPMN2
Motor de procesos	Bonita Execution Engine (Java API)	jBPM Engine
Diseño de procesos	Bonita Studio basado en Eclipse	Plugin jBPM Eclipse
Motor de reglas	Tablas de decisión. Integración con Drools a través de conector.	Integración con Drools a través de conector.
Editor de formularios	Bonita Studio (Bonita Web Application Builder)	No tiene un editor de formularios. Sí un plugin con JBPM Form Builder.

Desarrollo de aplicaciones (API, depuración y test)	Bonita Studio	Plugin jBPM Eclipse
Integración para la administración de contenidos	Más de 100 conectores integrados (ERP, CRM, ECM, entre otros) y la posibilidad de crear los propios con Bonita Studio. API Java y Rest	Se deben construir todos los conectores.
Nivel de madurez	Bonita es un proyecto maduro (desde 2001) con una gran comunidad. Amplia documentación en español con ejemplos funcionales.	El primer lanzamiento importante de jBPM se produjo con la edición 2.0 (2004). Documentación en inglés.
Idioma	Configurable	Inglés

5.3. Caso Solicitud de Préstamos

La solución a describir fue implementada para una Caja Profesional de la provincia de Buenos Aires, la cual entre sus diferentes beneficios otorga préstamos a sus afiliados, beneficiarios y empleados. Dicha Caja no poseía aplicaciones modernas tanto para la gestión de sus trámites en general, como así tampoco en particular para los préstamos. El trabajo, entonces, incluyó no solamente el desarrollo de una solución tecnológica para el circuito de otorgamiento de préstamos, sino también aspectos de integración vinculados con aplicaciones ya existentes en la organización y que debían ser incluidas en el desarrollo propuesto.

Objetivo

El objetivo del proyecto fue definir e implementar una arquitectura para el ambiente de desarrollo, prueba y producción, con componentes de software libre, integrado principalmente por un BPMS y utilizando como repositorio la base de datos Oracle de dicha Caja Profesional, accediendo a la misma a través de servicios de integración.

Asimismo, se realizó la ingeniería de tres procesos seleccionados, su posterior desarrollo en el BPMS y la implementación de los mismos en el ámbito de la Caja.

En este libro se describe el resultado del diseño y desarrollo de uno de los procesos seleccionados.

Alcances

Los alcances que se plantearon para el proyecto fueron los de seleccionar una herramienta BPMS de software libre, que mejor se adapte para resolver los problemas de la Caja y particularmente de los tres procesos objeto del proyecto.

En base a dicha selección, se trabajó sobre la instalación de un ambiente tecnológico (de desarrollo, pruebas y producción) para la herramienta BPMS seleccionada, dentro del data center utilizado por la Caja para su funcionamiento, siempre bajo un contexto de orientación a servicios, ya que la base de datos de la organización corresponde a una versión antigua, en general ya no soportada para desarrollo en los lenguajes estándar del mercado.

Asimismo, se realizó el relevamiento de los tres procesos seleccionados, la aplicación de una ingeniería de procesos sobre los mismos y el diseño e implementación de la solución para los mismos.

Los procesos involucrados en la solución e identificados como prioritarios en su momento para el organismo fueron:

1 - Otorgamiento de préstamos personales para Beneficiarios
2 - Otorgamiento de préstamos personales para afiliados
3 - Otorgamiento de préstamos hipotecarios.

Se describe en este capítulo el proceso de Otorgamiento de Préstamos Personales para afiliados.

Solución propuesta y metodología

La metodología aplicada es la misma a la que se muestra en la Tabla 8 de este capítulo, con las correspondientes etapas de la Figura 31. El resultado de las etapas que se retroalimentan llevan a la fase de pruebas y finalmente al test, puesta a punto y producción.

El proyecto se inició con la detección del proceso interno de la Caja destinado a ser el puntapié inicial de la Gestión por procesos, resultando determinante la decisión de la propia institución al inclinarse por el sector de préstamos. Una vez establecido este sector, se llevaron a cabo un conjunto de tareas técnicas de relevamiento y análisis sobre el mismo, destinadas a determinar el estado actual y proponer las medidas y el esquema de solución necesarios para obtener el resultado deseado sobre sus procesos principales.

El responsable de la oficina de préstamos fue determinado como el dueño del proceso, presentándose durante todo el proyecto una intervención fuerte del área de sistemas, sobre todo en la determinación y viabilidad de los requerimientos a incluir.

Durante el análisis se evidenciaron una serie de procedimientos internos de la oficina propios de la gestión no digital de los expedientes generados. Esto provocó una revisión de los mismos, siendo que en algunos casos se realizaron las modificaciones necesarias, así como en otros se optó por aplazarlas para un futuro cercano, de manera de no impactar negativamente en el proyecto con tanta divergencia.

El proceso que se describe en la Figura 37 corresponde al Otorgamiento de Préstamos Personales. Dicho proceso considera las siguientes etapas:

A grandes rasgos, proceso incluye las siguientes etapas:

Ingreso de solicitud

El afiliado se presenta en una delegación o ingresa por el sitio oficial de la Caja y completa una solicitud indicando una serie de datos obligatorios. El sistema evalúa si la persona reúne los requisitos de aptitud para el mismo y en caso contrario lo informa. Se presenta toda la documentación respaldatoria del solicitante, así como la requerida por el tipo de préstamo seleccionado y se avanza.

Control de la solicitud

Varias oficinas internas de la Caja realizan una serie de controles pertinentes sobre el afiliado y la documentación presentada, de manera de detectar algún tipo de inconveniente. Pueden ocurrir una serie de interacciones entre ambos hasta que dichos requerimientos se cumplimenten de manera completa. Una vez finalizados todos los pedidos, se completan los análisis.

Análisis final y recomendación de estado

Una vez cumplimentados todos los controles la oficina dueña del proceso realiza el análisis de viabilidad final y genera una recomendación de estado, la cual será tomada por el Directorio para su decisión final.

En todas estas etapas, la solución implementada permite evaluar la etapa del proceso en que el expediente se encuentra, de manera de poder detectar demoras y/o inconvenientes en la circulación del mismo. Por otro lado, se permite en cada una de las etapas el acceso a la documentación digital del expediente, ya sea tanto la presentada por el solicitante, así como la generada por cada una de las oficinas internas en sus actuaciones correspondientes. De esta manera se permitió a la Caja comenzar a gestionar por primera vez un expediente electrónico tendiente al proceso de despapelización, quedando para un futuro la implementación de una estructura de firma digital que permita la realización de una política de expediente digital propiamente dicha.

Arquitectura Tecnológica de la Solución

Los productos de software que se consideraron para el presente proyecto son todos Open Source, permitiendo de esta manera a la Caja reducir los costos en el licenciamiento de software, así como su apertura hacia estándares aceptados en el mercado.

Se consideraron en un principio tres entornos, uno para desarrollo y pruebas primarias, otro para test y otro finalmente para producción. En cada uno de estos se reprodujo de manera exacta la misma arquitectura, integrada formalmente por los siguientes productos:

Bonita Open Solution

BPMS seleccionado para la gestión de todos los procesos desarrollados. La versión utilizada fue la 7, y se instaló en su versión *bundle* con un servidor de aplicaciones Tomcat

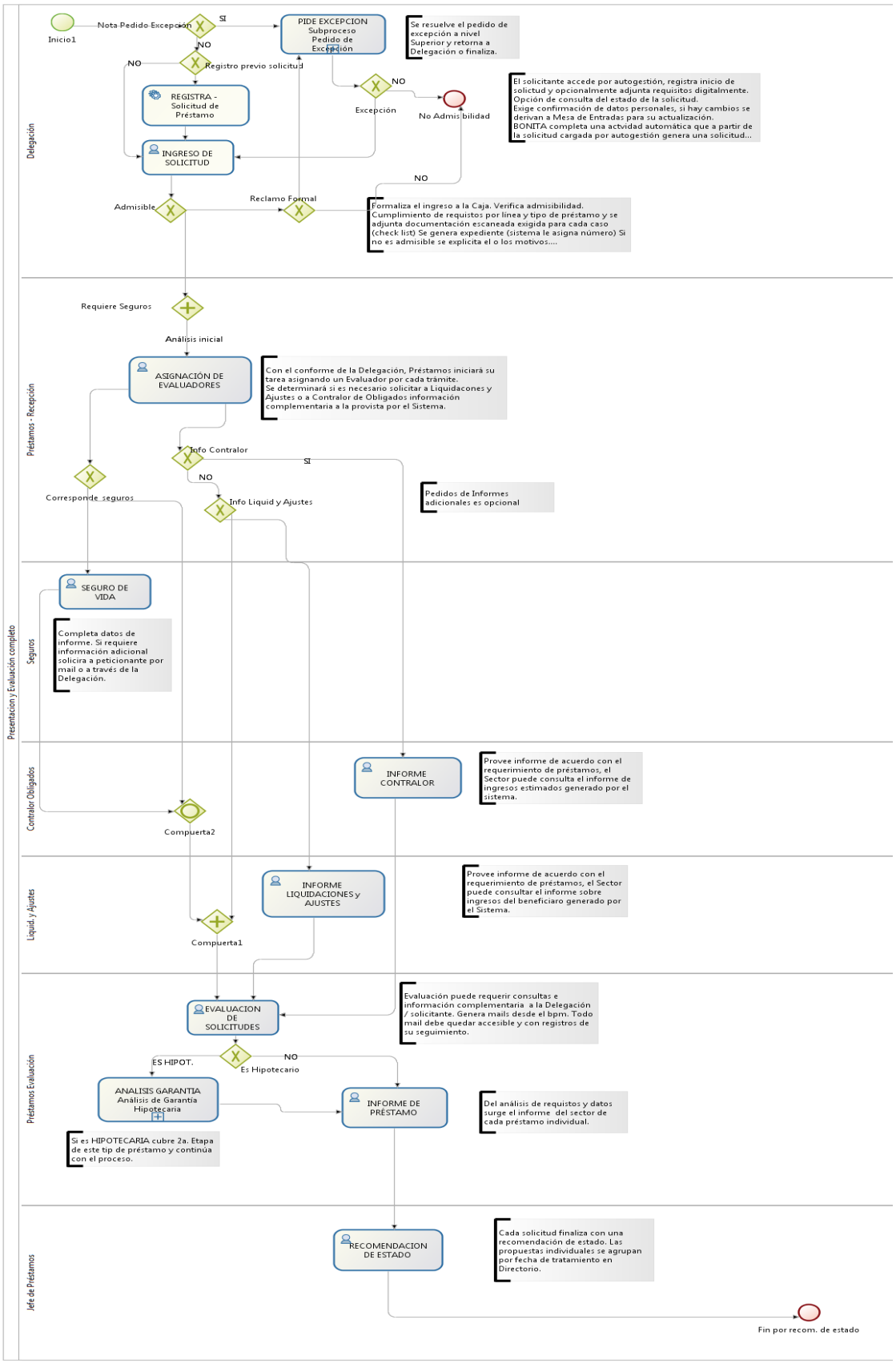


Figura 37. *Proceso de Otorgamiento de Préstamos Personales*

Servicios

Los servicios fueron integrados en varias APIs, de las cuales podemos distinguir:

1. API de servicios de la Caja: dicha institución posee una BD Oracle de una versión antigua, que si bien ha demostrado una amplia robustez presenta el inconveniente de no tener capacidad para ser accedida de manera estándar por los lenguajes actuales de programación (se necesitan librerías muy específicas y discontinuadas). Por lo cual se desarrolló una capa de servicios implementados en Java con REST para permitir el acceso a distintas funcionalidades provistas por aplicaciones legacy de la institución relacionadas con la BD. Esta visión por servicios permitió por un lado acceder a los datos y funcionalidades, así como independizar a las aplicaciones consumidoras de la versión de la BD utilizada, permitiendo una portabilidad necesaria para migraciones futuras.
2. API de servicios del proceso: los servicios que el proceso necesita consumir (formularios web, consultas, entre otros) fueron implementados en PHP con Angular. Todo esto fue desplegado en un servidor de aplicaciones Apache.
3. API de servicios de Bonita: el BPMS presenta una API REST para acceder a cada una de sus funcionalidades. Por razones de usabilidad, se construyó un portal de acceso para los usuarios, de manera de personalizar los menúes y accesos sin depender de la consola provista por el BPMS. De esta manera, se construyó una capa de servicios que acceden a dicha API Rest permitiendo tanto al portal como a los servicios PHP acceder a la funcionalidad necesaria del BPMS.

Servidor de aplicaciones

Se optó por un servidor Apache para aplicaciones web dentro del cual fueron desplegadas las API desarrolladas y el portal de acceso de los usuarios.

Gestor Documental

Se seleccionó Alfresco como servidor de archivos. Este gestor posee una API Rest que permite la gestión de cada uno de sus componentes, otorgando de esta manera a los servicios del proceso la capacidad de gestionar el expediente electrónico junto con toda su documentación.

Discusión de Resultados y Conclusiones

Este proyecto resultó un gran aprendizaje, no sólo desde el aspecto técnico y tecnológico propiamente, debido a la implementación completa de una capa de servicios que permitieran acceder a una base de datos propietaria y de versión antigua, sino también desde el punto de vista profesional sobre la gestión del proyecto y de las contrapartes involucradas.

Desde el punto de vista tecnológico, el proyecto introdujo en la Caja una serie de mejoras:

1. La institución no poseía hasta el momento ningún tipo de gestión electrónica organizada sobre los trámites realizados. Si bien se poseía aplicaciones legacy que cubrían is-

las dentro del circuito, no había una implementación clara y controlada de un proceso que permitiera efectuar una gestión digital de dicho trámite, así como de su documentación. Mediante el presente proyecto la Caja logró formalizar por primera vez dentro de un entorno actualizado y normalizado uno de sus circuitos de trámite.

2. Manejo de la documentación del trámite mediante un gestor documental: se permitió digitalizar la documentación propia del circuito, y acceder a la misma desde cualquiera de los puntos de acceso del proceso.
3. Se implementó por primera vez en la Caja el control de admisibilidad para los distintos tipos de préstamo, cuestión que era realizada de manera manual por los empleados de la Caja a partir de reportes y archivos Excel.

Se implementaron consultas que permiten efectuar un seguimiento en tiempo real de los lugares por los que pasó el expediente, así como el punto donde se encuentra actualmente el mismo.

4. Se implementó un sistema de notificaciones por email que permite a los participantes del proceso enterarse de las acciones que se van realizando respecto de un trámite determinado, que podría eventualmente involucrar su labor.
5. Se desarrollaron formularios para la generación automática de documentación que antes era elaborada de manera manual por los participantes del proceso. Esto incluye el guardado automático de dicha documentación en el gestor documental para su posterior descarga y consulta.

Debemos notar también que este tipo de proyectos, donde no sólo se implementa una innovación tecnológica sino que se accionan mecanismos de renovación en el devenir cotidiano del trabajo de la organización, muchas veces generan gran resistencia interna. Los usuarios se resisten a la incorporación de nuevos hábitos ya que no solamente creen ver incrementada su labor cotidiana por los nuevos sistemas, sino que a su vez se resisten a la incorporación de nuevos paradigmas y buenas prácticas en pos de conservar tradiciones.

Es en este sentido donde se debe hacer un hincapié fundamental en el alineamiento de las organizaciones con sus objetivos reales, más que con sus resultados. Muchas veces los resultados diarios reflejan una lejanía con objetivos tales como la modernización, la despapelización, la gestión ordenada de los procesos internos, entre otros objetivos fundamentales. Seguir defendiendo internamente los paradigmas tradicionales sólo genera fricción con estas nuevas tendencias, y es allí donde se nota la necesidad de liderazgos claros, que conduzcan a alinear las acciones reales de la organización con sus objetivos.

Referencias

Aguilar-Saven, R. S. (2004). Business process modelling: Review and framework. International Journal of production economics, 90(2), 129-149.

Bazán, P. (2009). Un modelo de integrabilidad con SOA y BPM (Doctoral dissertation, Facultad de Informática). http://sedici.unlp.edu.ar/handle/10915/4181

Bazán, P. (2017) "Aplicaciones, servicios y procesos distribuidos." Series: Libros de Cátedra. https://libros.unlp.edu.ar/index.php/unlp/catalog/book/779 (enlace a Marzo 2020).

BPEL (2020) https://www.ecured.cu/BPEL (Abril 2020)

BPEL Designer and Service Engine User's Guide (2007) https://docs.oracle.com/cd/E19182-01/821-0539/index.html (Abril 2020)

BPEL TUTORIAL (2020) https://searchapparchitecture.techtarget.com/tutorial/BPEL-tutorial?track=NL-130&ad=714364&Offer=mn_eh070809sSOAUNSC_bpel&asrc=EM_USC_8638480 (a Abril 2020)

Business Process Modeling Notation (BPMN) Version 2.02 OMG https://www.omg.org/spec/BPMN. 2019. (al 23/08/2019)

Chedrese, V., Parra, J., Martínez, J., Pelitti, M., Lorenzon, E., Bazán, P., & Alvarez, M. (2014). Sistema basado en BPM para el Seguimiento del Proceso Licitatorio y la Ejecución de Proyectos del Programa PMGM-UEC-Ministerio del Interior y Transporte de la Nación Argentina. In Simposio Informática en el Estado, en el marco de las 43a JAIIO (Vol. 43).

Dae-Hee Seo, Im-Yeong Lee, Soo-Young Chae, Choon-Soo Kim (2003) Single sign-on authentication model using MAS (multiagent system). Communications, Computers and signal Processing, 2003. PACRIM. 2003 IEEE Pacific Rim Conference on (Volume: 2).

Davenport, T. H. (1993). Process innovation: reengineering work through infomation technology. Harvard Business Press.

Delgado, A., Calegari, D., Milanese, P., Falcon, R., & García, E. (2015, May). A systematic approach for evaluating BPM systems: case studies on open source and proprietary tools. In IFIP International Conference on Open Source Systems (pp. 81-90). Springer, Cham,

Documentación Bizagi (Julio 2015). Bizagi BPM Suite Descripción Funcional - https://www.bizagi.com/docs/BizAgi%20Descripcion%20Funcional.pdf.

Hammer, M., & Champy, J. (1993). A manifesto for business revolution. Reengineering the Corporation.

Johansson, R. (1993). System modeling and identification (Vol. 1). Englewood Cliffs, NJ: Prentice Hall.

Juric Matjaz B., Loganathan Ramesh, Poornachandra Sarang, Frank Jennings (2007). SOA Approach to Integration XML, Web services, ESB, and BPEL in real-world SOA projects. Packt Publishing. ISBN 978-1-904811-17-6

Kettinger, W. J., Teng, J. T., & Guha, S. (1997). Business process change: a study of methodologies, techniques, and tools. MIS quarterly, 55-80.

Kirchmer, M. (2017). Business process management: what is it and why do you need it? In High Performance Through Business Process Management (pp. 1-28). Springer, Cham.

Magic Quadrant for Intelligent Business Process Management Suites. (2019 https://www.gartner.com/en/documents/3899484/magic-quadrant-for-intelligent-business-process-manageme

Mayer, Richard J., et al. (1992). IDEF Family of Methods for Concurrent Engineering and Business Reengineering Applications. Knowledge-Based Systems, Inc.

Morgenthal JP (2001). Enterprise Application Integration with XML and Java. Prentice Hall. 1-86. ISBN 0-13-085135-3.

Oracle Whitepaper (2007). SOA Governance: Framework and Best Practices.

Orquestación de Servicios BPEL (2012) http://www.jtech.ua.es/j2ee/publico/servc-web-2012-13/sesion03-apuntes.html (Abril 2020)

Rodríguez, A. S., Bazan, P., & Díaz, F. J. (2015). Características funcionales avanzadas de los BPMS: análisis comparativo de herramientas. In XXI Congreso Argentino de Ciencias de la Computación

Rummler, G. A., & Brache, A. P. (1995). Improving performance. San Francisco: Jossey-Bass.

Schriber, T. J. (1969). Fundamentals of flowcharting. John Wiley & Sons, Inc.

Scott McKorkle. (2007). Model-Driven SOA. Achieve Higher Productivity Gains throughout the Design Process. Telelogic White Paper.

Sheina Dana. (2008). Realising the promise of SOA and BPM. Ovum. SearchCIO.

TEC – Technology Evaluations Center https://www3.technologyevaluation.com/store/software-evaluation-report/business-process-management-bpm-software-evaluation-report.html (al 26/10/2020)

The Forrester Wave: BPM Suites https://www.forrester.com/report/The+Forrester+Wave+BPM+Suites+Q1+2013/-/E-RES88581. 2013

Unified Modeling Language (UML), Version 2.5.1 OMG https://www.omg.org/spec/UML. 2019. (al 23/08/2019)

Von Rosing, M., Von Scheel, H., & Scheer, A. W. (2014). The complete business process handbook: body of knowledge from process modeling to BPM (Vol. 1). Morgan Kaufmann.

Weske Mathias (2008). Business Process Management: Concepts, Languages, Architectures. Springer. 3-67. ISBN 978-3-540-73521-2.

Yourdon, E., & De Marco, T. (1988). Structured analysis. Prentice Hall.

Zairi, M., & Sinclair, D. (1995). Business process re-engineering and process management: a survey of current practice and future trends in integrated management. Business Process Re-engineering & Management Journal, 1(1), 8-30.

Los autores

Bazán, Patricia

Es Doctora en Ciencias Informáticas graduada en la Facultad de Informática UNLP en 2015. Es Magister en Redes de Datos graduada en la misma unidad académica en 2010. Se desempeña como Profesor Titular con Dedicación Exclusiva en la Facultad de Informática UNLP en la asignatura Desarrollo de Software en Sistemas Distribuidos de la carrera de Licenciatura en Sistemas. Las tareas de investigación las desarrolla en el Laboratorio de Investigación y Formación en Informática Avanzada (LINTI) desde 1995, dirigiendo becarios y tesistas. Ha sido Consultor en proyectos financiados por el Banco Mundial y el BID.

Romero, Dalila (1975-2020)

Licenciada en Informática, graduada en la Facultad de Informática UNLP en 2007. Ha obtenido un postítulo en formación docente en 2004. Se ha desempeñado como Profesor Adjunto con Dedicación Simple en la Facultad de Informática UNLP en la asignatura Desarrollo de Software en Sistemas Distribuidos de la carrera de Licenciatura en Sistemas y de la asignatura Conceptos y Paradigmas de Lenguajes de la carrera de Licenciatura en Informática. Fue miembro del CeSPI (Centro Superior para el Procesamiento de la Información), desempeñándose como coordinadora de la Dirección de Sistemas Académicos hasta su fallecimiento en 2020. Su labor docente cuenta con una destacada trayectoria en el dictado de cursos de capacitación y talleres de formación para usuarios. Ha sido una profesional comprometida con los proyectos en los que se ha desempeñado y una excelente colaboradora tanto para sus pares como para las autoridades.

Martínez Garro, José

Es Licenciado en Sistemas graduado en la Facultad de Informática UNLP en 2010. Se desempeña como Jefe de Trabajos Prácticos y ayudante en la Facultad de Informática UNLP, en las asignaturas Desarrollo de Software en Sistemas Distribuidos, Conceptos y Paradigmas de Lenguajes de Programación, y Sistemas y Organizaciones de la carrera de Licenciatura en Sistemas. Colabora en tareas de investigación en dicha unidad académica, asesorando profesionalmente varias tesinas de grado. Se ha desempeñado como Consultor Informático en proyectos en Argentina y Latinoamérica, así como Líder Tecnológico en proyectos financiados por el Banco Mundial y el BID.

Rodríguez, Anahí Soledad

Es Licenciada en Sistemas graduada en la Facultad de Informática UNLP en 2012. Se desempeña como Ayudante Diplomado con Dedicación Exclusiva en la Facultad de Informática UNLP en la asignatura Desarrollo de Software en Sistemas Distribuidos de la carrera de Licenciatura en Sistemas y de la asignatura Conceptos y Paradigmas de Lenguajes de la carrera de Licenciatura en Informática. Las tareas de investigación las desarrolla en el Laboratorio de Investigación y Formación en Informática Avanzada (LINTI) desde 2006.

Libros de Cátedra

La orquestación de servicios y las aplicaciones actuales : una visión por procesos de negocio / Patricia Bazán ... [et al.] ; coordinación general de Patricia Bazán. - 1a ed. - La Plata : Universidad Nacional de La Plata ; EDULP, 2021.
Libro digital, PDF - (Libros de Cátedra)

Archivo Digital: descarga
ISBN 978-950-34-2020-1

1. Tecnología Informática. I. Bazán, Patricia, coord.
CDD 004.028

Diseño de tapa: Dirección de Comunicación Visual de la UNLP

Universidad Nacional de La Plata – Editorial de la Universidad de La Plata
48 N.º 551-599 / La Plata B1900AMX / Buenos Aires, Argentina
+54 221 644 7150
edulp.editorial@gmail.com
www.editorial.unlp.edu.ar

Edulp integra la Red de Editoriales Universitarias Nacionales (REUN)

Primera edición, 2021
ISBN 978-950-34-2020-1
© 2021 - Edulp

exactas

EDITORIAL DE LA UNLP

UNIVERSIDAD NACIONAL DE LA PLATA