

# **Modelado de procesos de Negocio**

---



# Índice

Presentación	5
Red de contenidos	7
<b>Unidad de Aprendizaje 1</b>	
<b>MODELADO DESCRIPTIVO</b>	9
<b>1.1 Tema 1 : Conceptos generales</b>	11
1.1.1 : Definiciones básicas de negocios: giro de negocio, macroproceso, proceso, actividad, tarea, procedimiento, método	11
1.1.2 : OMG: Object Management Group	13
1.1.3 : ¿Qué es un Modelo? – Utilidad y características de un modelo de procesos de negocio	15
1.1.4 : Importancia del modelado	21
<b>1.2 Tema 2 : BPMN</b>	24
1.2.1 : BPM y BPMN – Importancia del estándar BPMN	24
1.2.2 : Lo que es y lo que no es BPMN	26
1.2.3 : BPMN en las Organizaciones – El Ciclo BPMN	26
1.2.4 : Herramienta CASE: introducción y descripción de su interfaz principal	30
1.2.5 : Elementos organizativos del BPMN: swimlanes, pools, lanes y groups.	31
1.2.6 : Elementos básicos: tareas, conectores y compuerta básica	32
1.2.7 : Elementos del Flujo de trabajo básico: tipos de tareas, objetos de datos y tipos de eventos iniciales y finales	34
<b>1.3 Tema 3 : Modelado a nivel Descriptivo</b>	38
1.3.1 : Niveles de Modelado: descriptivo , analítico y ejecutable	38
1.3.2 : Elementos de modelos descriptivos: tipos de eventos iniciales y finales, eventos intermedios simples, artefactos, conectores y elementos de legibilidad.	39
1.3.3 : Categorías de Procesos: Orquestación, Coreografía y Colaboración.	43
1.3.4 : Uso de Fases en un entorno CASE	52
1.3.5 : Elementos de modelos descriptivos: compuertas exclusivas y compuertas inclusivas basadas en datos.	56
1.3.6 : Desarrollo de aplicaciones	57

**Unidad de Aprendizaje 2**

<b>MODELADO ANALÍTICO</b>	62
<b>2.1 Tema 4 : Modelado a nivel Analítico</b>	63
2.1.1 : Compuertas paralelas.	63
2.1.2 : Compuertas basadas en eventos	63
2.1.3 : Compuertas complejas	66
2.1.4 : Técnicas de simplificación de procesos	67
2.1.5 : Metodología de diseño top-down y modelado BPMN	69
2.1.6 : Identificación de Subprocesos: Reusables y embebidos	71
2.1.7 : Uso de Temporizadores, manejo de errores	75
2.1.8 : Transacciones: envío y captura de compensaciones	79

**Unidad de Aprendizaje 3**

<b>PATRONES DE DISEÑO</b>	84
<b>3.1 Tema 5 : Patrones de Control básico de Flujo</b>	85
3.1.1 : Secuencia – División paralela	85
3.1.2 : Sincronización.	86
3.1.3 : Decisión exclusiva – Unión simple.	87
3.1.4 : Elección múltiple – Unión sincronizada.	89
3.1.5 : Unión múltiple – Discriminador estructurado.	91
<b>3.2 Tema 6 : Patrones basados en Eventos y de Iteración</b>	95
3.2.1 : Estructura y bloqueo parcial de unión	95
3.2.2 : Unión y división de caminos – otros.	97
3.2.3 : Decisión implícita – ejecución paralela intercalada.	100
3.2.4 : Milestone – Sección crítica	102
3.2.5 : Ciclos arbitrarios – Bucle estructurado.	104

**Unidad de Aprendizaje 4**

<b>FUNDAMENTOS DE UML</b>	108
<b>4.1 Tema 7 : UML – Modelado de Negocios</b>	109
4.1.1 : Nociones generales de UML	109
4.1.2 : Definiciones básicas y diagramas principales	111
4.1.3 : Introducción al Modelo de Casos de Uso de Negocio	119
4.1.4 : Elementos del Modelo de Negocio: Simbología	121
<b>4.2 Tema 8 : UML – El Modelo de las Actividades</b>	126
4.2.1 : Actividades y Decisiones	126
4.2.2 : Rutas concurrentes e indicaciones	128
4.2.3 : Marcos de responsabilidad - Ejemplos	130
Anexos	136

# Presentación

Todos los enfoques y teorías acerca de la calidad y su importancia para aumentar el valor de negocio de una organización, apuntan al mejoramiento continuo de sus procesos. Si partimos de la premisa que lo que generalmente busca el cliente o consumidor final es, la mejor calidad al más bajo precio; entonces los procesos deben ser los más eficientes, para asegurar la Calidad y hacer que la magnitud del valor de negocio, aumente para la empresa conforme aumente la satisfacción del usuario.

El análisis y mejoramiento continuo de los procesos de Negocio en las organizaciones, permite alcanzar estándares de eficiencia que son indispensables para su automatización. La implementación de sistemas de software debe realizarse sobre la base de procesos eficientes y bien diseñados, las cuales magnificarán su valor de negocio; pero también podrían magnificar el caos si los procesos son ineficientes.

**La comprensión de los procesos proviene de la interpretación de la realidad;** pero dado que las personas tenemos diferentes percepciones acerca de una misma realidad, se hace necesario objetivarla para tener una lectura correcta de la situación analizada.

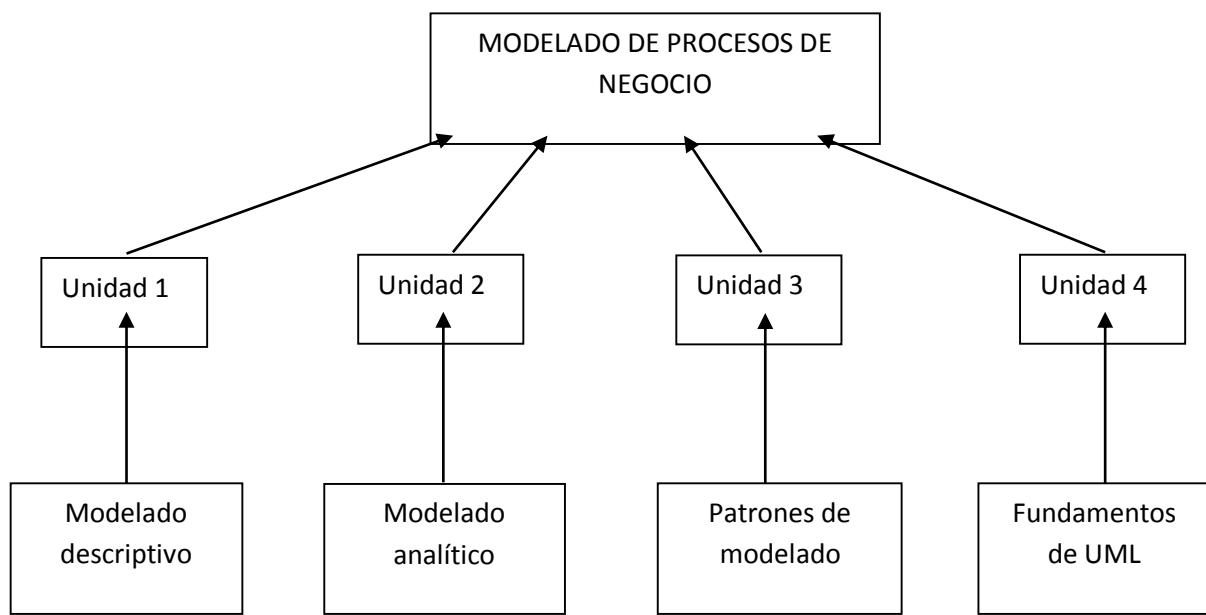
Es aquí donde intervienen los modelos, los cuales deben usar lenguajes estandarizados como BPMN o UML, que ofrecen una simbología estándar. Tales modelos permiten analizar y discutir la coherencia y pertinencia del modelo elaborado.

**En resumen, aquí radica la naturaleza del curso de Modelado de Procesos de Negocio, entender los procesos, modelarlos bajo una notación estándar y desarrollar habilidades para su mejoramiento constante; lo cual le permitirá posteriormente, implementar sistemas informáticos de calidad.**

El curso se desarrolla presencialmente en Laboratorio de cómputo y parcialmente en formato Blended learning con la plataforma Moodle. En la primera Unidad de aprendizaje, se explica el modelado de procesos a nivel descriptivo para capturar la lógica de los procesos; en la segunda Unidad, se estudia el modelado de procesos a nivel analítico empleando la mayor parte de la semántica de BPMN. En la tercera Unidad de aprendizaje, se hace énfasis en los patrones de modelado que son estructuras estereotipadas prestablecidas; y en la cuarta Unidad, se hace una introducción al Modelado de Negocios con UML.



# Red de contenidos







# MODELADO DESCRIPTIVO

## LOGRO DE LA UNIDAD DE APRENDIZAJE

Al término de la unidad, el alumno usa la notación básica BPMN para construir modelos a nivel descriptivo regido por la OMG, a partir de la comprensión de la importancia de modelar procesos de negocio.

## TEMARIO

### 1.1 Tema 1 : Conceptos generales

- 1.1.1 : Definiciones básicas de negocios: giro de negocio, macroproceso, proceso, actividad, tarea, procedimiento, método
- 1.1.2 : OMG: Object Management Group
- 1.1.3 : ¿Qué es un Modelo? – Utilidad y características de un Modelo de procesos de negocio
- 1.1.4 : Importancia del modelado

### 1.2 Tema 2 : BPMN

- 1.2.1 : BPM y BPMN – Importancia del estándar BPMN
- 1.2.2 : Lo que es y lo que no es BPMN
- 1.2.3 : BPMN en las Organizaciones – El Ciclo BPMN
- 1.2.4 : Herramienta CASE: introducción y descripción de su interfaz principal
- 1.2.5 : Elementos organizativos del BPMN: swimlanes, pools, lanes y groups.
- 1.2.6 : Elementos básicos: tareas, conectores y compuerta básica
- 1.2.7 : Elementos del flujo de trabajo básico: tipos de tareas, objetos de datos y tipos de eventos iniciales y finales

### 1.3 Tema 3 : Modelado a nivel Descriptivo

- 1.3.1 : Niveles de modelado: descriptivo , analítico y ejecutable
- 1.3.2 : Elementos de Modelos descriptivos: tipos de eventos iniciales y finales, eventos intermedios simples, artefactos, conectores y elementos de legibilidad
- 1.3.3 : Categorías de Procesos: Orquestación, Coreografía y Colaboración
- 1.3.4 : Uso de Fases en un entorno CASE
- 1.3.5 : Elementos de Modelos descriptivos: compuertas exclusivas y compuertas inclusivas basadas en datos
- 1.3.6 : Desarrollo de Aplicaciones

## 1.1. CONCEPTOS GENERALES

### 1.1.1. Definiciones básicas de Negocios

a) **Giro de Negocio:** Actividad específica de comercio, industrial, artesanal, de servicios y/o profesional que se desarrolla en un establecimiento, para cuya realización se solicita

Autorización Municipal de Funcionamiento (Ordenanza Municipal Nº 857 – MML).

Para entender mejor la definición, a continuación algunos ejemplos:

Giro principal	Giro afín	Giro complementario
Bazar	Bazar - regalos, Bazar - Tienda, Bazar - Bodega, Bazar - Boutique, Bazar - Librería	Perfumería, venta de artículos para fiestas infantiles, regalos, decoraciones.
Cafetería	Fuente de soda, heladería, jugería, salón de té.	Venta de sandwichs y postres, venta de pasteles, helados.
Confecciones	Sastrería, lencería.	Bordaduría, confección de ropa deportiva.
Ferretería	Venta de pinturas y matizados, cerrajería.	Venta de repuestos.
Servicios profesionales	Estudio Jurídico, Estudio Contable, Oficina Técnica.	Fotocopiado, anillados, ploteos.
Estación de servicios	Grifo	Minimarket, venta de lubricantes.

Figura 1: Ejemplos de giro o esencia del negocio

Fuente.- <http://www. http://thesmadruga2.blogspot.com/>

b) **Macroproceso:** Son los procesos críticos para cumplir los requerimientos del cliente, así como el soporte a dichos procesos. Por ejemplo: Macroproceso de venta.

Es el menor grado de segregación de la operación básica de la organización. Por ejemplo, en el instituto Cibertec, se tiene el proceso de gestión académica dentro del cual se puede encontrar otros procesos, como proceso de matrícula, retiro de curso, justificación de inasistencias, etc.

**c) Proceso:** Conjunto de actividades mutuamente relacionadas o que interactúan, los cuales transforman elementos de entrada en resultados

Los elementos de entrada para un proceso son generalmente resultados de otros procesos.

Los procesos de una organización son generalmente planificados y puestos en práctica bajo condiciones controladas para aportar valor.

Un proceso, en el cual la conformidad del producto resultante no pueda ser fácil o económicamente verificada, se denomina habitualmente "proceso especial".

**d) Actividad:** Es el menor número de partes en que es posible descomponer un proceso, por ejemplo pago de recibo por matrícula o llenado de solicitud.

**e) Tarea:** Es un mayor nivel de segregación de la actividad. Es la que implica una acción de verificación, tal es el caso, por ejemplo, cuando un alumno desea matricularse, se debe verificar que no tenga cursos pendientes

**f) Procedimiento** forma especificada para llevar a cabo una actividad o un proceso.

Nota 1: Los procedimientos pueden estar documentados o no.

Nota 2: Cuando un procedimiento está documentado, se utiliza con frecuencia el término "procedimiento escrito" o "procedimiento documentado". El documento (3.7.2) que contiene un procedimiento puede denominarse "documento de procedimiento".

**g) Método:** Es el procedimiento adoptado en una organización, con el cual se hace diferente de las demás.

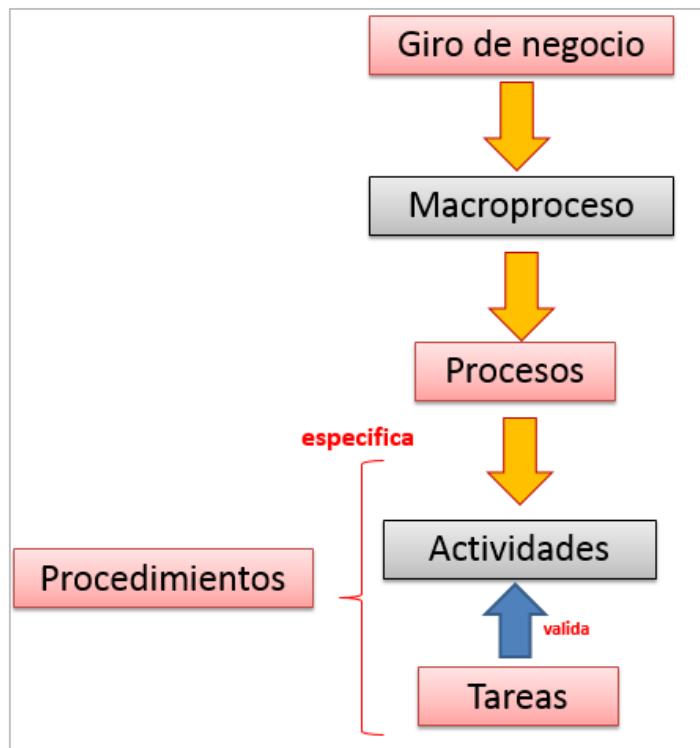


Figura 2: Jerarquía de elementos básicos

Fuente.- Elaboración propia.

### 1.1.2. OMG: Object Management Group

El Object Management Group u OMG es un consorcio formado en 1989 dedicado al cuidado y el establecimiento de diversos estándares de tecnologías orientadas a objetos, tales como UML, XMI, CORBA y **BPMN**. Es una organización sin fines de lucro que promueve el uso de tecnología orientada a objetos mediante guías y especificaciones. El grupo está formado por diversas compañías y organizaciones con distintos privilegios dentro de la misma.

([https://es.wikipedia.org/wiki/Object\\_Management\\_Group](https://es.wikipedia.org/wiki/Object_Management_Group))

BPMN es un estándar administrado por la OMG (Object Management Group) y ahora se ha convertido además a partir de mediados del año 2013 en una norma para el modelamiento e implementación de procesos definida por la ISO (International Organization for Standardization) que lleva el código ISO / IEC 19510:2013.

La primera versión de la Business Process Modeling Notation (BPMN) fue desarrollada por el instituto Business Process Management Initiative (BPMI) principalmente bajo la tutela de Stephan A. White profesional de la IBM en 2004.

Desde un principio, fue el principal objetivo de disponer de una notación gráfica, estandarizada, que permitiera automatizar los procesos a partir del diseño gráfico.

En el año 2005, fue trasladado el proyecto a la Object Management Group (OMG), debido a que el BPMI no era un instituto que administra estándares. La OMG es muy conocida en el mundo informático, porque administra entre otros el estándar del lenguaje para el diseño de software llamado Unified Modeling Language (UML). A través de la OMG, de la cual son miembros la mayoría de los proveedores más importantes de TI, BPMN se difundió rápidamente a nivel mundial y casi todos los proveedores sean grandes o pequeños, académicos o consultores empezaron a adoptar este estándar.

La última versión oficial 1.2 fue publicada en enero 2009 [Obj09]. La versión 2.0, completamente nueva y ampliada, se terminó a mediados del año 2010 y, a finales de este, el equipo de la OMG encargado de revisar y finalizar la nueva versión, llamada Finalization Task Force (FTF), dio la recomendación al gremio de la decisión de oficializar la versión 2.0.

A partir de la versión 2.0 la sigla BPMN cambia levemente de nombre a: Business Process Model and Notation.



Figura 3: Página Web de la OMG

Fuente.- Tomado de <http://www.omg.org/bpmn/index.htm>

El estándar no forma parte de la iniciativa de algún fabricante. Está regulado por la **OMG**.

La OMG es un consorcio sin fines de lucro a nivel internacional establecido en 1989 que define estándares de “Modelado” para lo que conocemos como “**Tecnologías de Negocio**”.

El objetivo de la OMG es el desarrollo, con el apoyo de sus miembros, de estándares integrales corporativos de gran valor. Son promotores de ideologías como Tecnología de Negocio y Optimización para Innovación mediante su programa BEI (Business Ecology Initiative) y su Asociación de Comunidades de Práctica.

([http://www.netec.com.pe/blog/120/bpmn-business-process-modeling-notation#.Vap6-l\\_Oko](http://www.netec.com.pe/blog/120/bpmn-business-process-modeling-notation#.Vap6-l_Oko))

### 1.1.3. ¿Qué es un Modelo? – Utilidad y características de un Modelo de Procesos de Negocio.

La realidad única, concreta y objetiva no puede ser captada como tal. Aun cuando pudiésemos asumir que esta realidad única existe, cada uno de nosotros la modifica a través del filtro de su percepción. La percepción de cada persona es algo bastante complejo, que está influido, entre otros posibles factores, por el tiempo, espacio y estado de ánimo al momento de realizar la percepción, además del impacto de experiencias previas, factores ambientales, estructura neuronal y el código genético del individuo.

Lo relevante es que para “n” observadores de un fenómeno, es posible obtener al menos “n” percepciones distintas (aunque posiblemente no “radicalmente” distintas).

Las **herramientas** que utilizamos para poder **comunicar** y **plasmar** nuestras **percepciones de realidades**, se denominan: **Modelos**.

Los **Modelos** son representaciones de algún fenómeno o hecho del mundo, que nos interese. **Ej.:** Modelos de Organizaciones, datos o procesos de negocios.

Para expresar estos Modelos es que se necesitan los **Lenguajes** (como UML o BPMN).

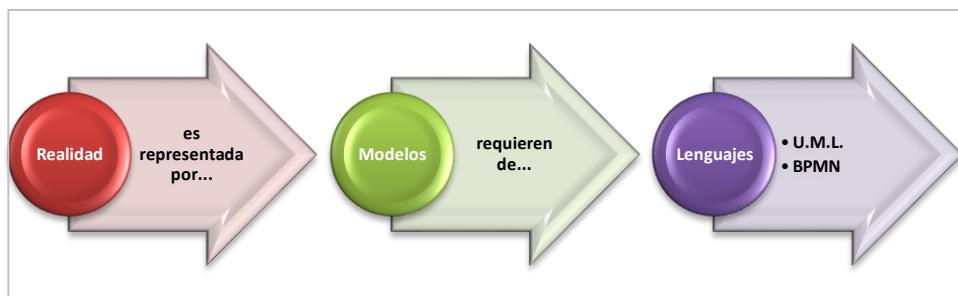


Figura 4: Realidades, modelos y lenguajes

Fuente.- Elaboración propia

Los Lenguajes son herramientas creadas por el hombre con el fin de comunicarse y son imprescindibles para poder concebir los modelos, pues uno expresa a lo más, lo que el lenguaje le permite.

Los **Lenguajes** son los que permiten comunicar los **modelos** a otros, validarlos, discutirlos y ampliar la percepción del otro sobre un mismo fenómeno.

### 1. Modelo de Procesos de Negocio:

El **proceso de negocio** es una colección de actividades diseñadas para producir una salida específica para un cliente o mercado particular. Implica un fuerte énfasis en '**cómo**' se hace el trabajo en una organización, en contraposición al enfoque en '**qué**' de producto. Así, un proceso es un ordenamiento específico de actividades de trabajo a través del tiempo y del espacio, con un comienzo, un fin, entradas y salidas claramente identificados: una estructura para la acción.

El **modelo de negocios** es el estudio de la organización.

Durante el proceso de modelado del negocio, se examina la estructura de la organización y se observan los roles en la compañía y cómo estos se relacionan.

También, se examina el flujo de trabajo de la organización, los procesos principales dentro de la compañía y cómo ellos trabajan. Además, se deben examinar las entidades externas, cualquier individuo u otras compañías, y cómo interactúan con el negocio, y observar las implicaciones de esas interacciones.

El "**Modelado de Negocios**" se define como un proceso de representación de uno o más aspectos o elementos de una empresa u organización.

Se busca modelar las actividades que se hacen dentro de la empresa, la cual va a tener un fin que es de alguna forma lucrativa, los aspectos que se desean modelar son:

1. **Su propósito**
2. **Su estructura**
3. **Su funcionalidad**
4. **Su dinámica**
5. **Su lógica de negocios**
6. **Sus componentes:**
  1. **Fines**
  2. **Procesos de negocio**
  3. **Reglas de negocio**
  4. **Objetos de negocio**
  5. **Actores**
  6. **Unidades organizativas**
  7. **Etc.**

Un Modelo del Negocio es una descripción de los elementos que constituyen una organización, o una parte de ella, así como de las relaciones entre estos elementos. Un Modelo del Negocio es una conceptualización de una empresa u organización, es la caracterización de los aspectos más significativos de la empresa o de una parte de ella. Para ello, se debe tener claro cuál es el fin que se busca con ese modelo y así tener los elementos del negocio que se deseen representar.

## 1. ¿Qué se usa para modelar el negocio?

Dentro de las notaciones para modelar el negocio y/o sistemas de negocio, se utilizan varias notaciones y lenguajes de modelado, entre las que se puede mencionar:

- a)** Diagramas de Flujo de Datos (DFD) – Análisis estructurados –
- b)** UML (diferentes versiones) – Son los usados actualmente para diagramar modelos de negocios –
  - Diagramas de Casos de Uso • Diagramas de clases y objetos • Diagramas de Actividades • Diagramas de estados
- c)** BPMN (*Business Process Modeling Notation*) 4) UML Business extensiones de UML para modelado de procesos del negocio • Diagramas de procesos

## 2. Utilidad y características:

La utilidad del Modelo de Procesos de Negocio se puede apreciar a través de lo siguiente:

### a) Entender el Negocio

Uno de los principales motivos para desarrollar cualquier modelo es incrementar el conocimiento del negocio y facilitar la comunicación. Un modelo visual es más fácil de comprender y discutir que una descripción textual. El modelo es una fotografía actual de cómo los modeladores ven actualmente al negocio. El modelo cambiará y evolucionará conforme los modeladores mejor entiendan el negocio o conforme el negocio cambie.

Una vez que los modelados sean estables, debido principalmente a que ya dan una imagen clara de los roles y tareas a través de toda la organización, entonces estos podrán ser usados tanto para entrenar al personal como para desarrollar las herramientas tecnológicas que apoyen a los procesos que realiza dicho personal (Cauvet, 2008).

### b) Soporte del Sistema de Información

Hoy en día, la mayoría de los negocios usan algún tipo de sistema de información. A decir verdad, puede decirse que la tecnología de información es una parte integral de la operación diaria de muchas compañías. Con la Internet como infraestructura técnica para comunicación y transacciones financieras, una riqueza de nuevas oportunidades de negocios, está emergiendo. Los modelos de negocio existentes necesitan ser adaptados con las nuevas posibilidades que provee la Internet.

Aun siendo tan grande esta tendencia, sin embargo, muchas compañías están insatisfechas con la calidad de sus sistemas de información, mencionando que ofrecen el soporte de negocio insuficiente o inútil, difícil de usar, no son fiables, y no están integrados con otros sistemas. En muchos casos, esto se debe al hecho de que los sistemas no están siendo desarrollados con un entendimiento correcto del negocio. Este es un efecto común en las empresas que emplean numerosos sistemas informáticos pequeños, que aunque apoyan ciertas acciones del negocio, funcionan de manera aislada de forma que es frecuente que la información producida por dichos sistemas sea inconsistente (Laudon, 2004).

La solución es crear un modelo global de negocio que puede ser usado para determinar qué sistemas de información son requeridos, cómo deben ser desarrollados y qué funcionalidad deben tener. Si la especificación de requerimientos está basada en un buen modelo de negocio, hay una gran probabilidad de que el sistema de información soportará las operaciones del negocio adecuadamente. Existen varias ventajas para basar todos los sistemas de información en un mismo modelo básico de negocio:

1. Los sistemas de información se vuelven una parte integral del negocio global, soportando las operaciones, fortaleciendo el trabajo y la obtención de resultados.

2. Los sistemas se integran fácilmente unos con otros y pueden compartir o intercambiar información.
3. Los sistemas son más fáciles de actualizar y modificar como dicten los cambios en el modelo de negocio, como resultado del ambiente que le rodea, las metas de la organización, o por las mejoras e innovaciones del modelo de negocio. Esto reduce el costo de mantener actualizados los sistemas de información y de poner al día continuamente los procesos del negocio.
4. La lógica de negocio puede ser reusada en algunos sistemas reduciendo los costos y el esfuerzo de desarrollo.

Idealmente, los objetos presentados en los modelos de negocio traducen o mapean objetos en el sistema de información. Normalmente, esto no es un mapeo uno a uno.

Un objeto o proceso en el modelo de negocio no siempre puede ser traducido a un objeto dentro del sistema de información y viceversa, hay objetos en el sistema de información que no están presentes en absoluto en el modelo del negocio, como son por ejemplo los elementos técnicos que apoyan la correcta operación del sistema, tales como ciertos controles de las bases de datos y de los que soportan las telecomunicaciones.

Usar modelos de negocio como base para los sistemas de información también presenta una oportunidad para el reuso de software. Si son varios sistemas de información que apoyan el mismo negocio, usualmente, tendrán un conjunto de objetos que coincidan. Esos objetos tienen que ser implementados solo una vez y pueden ser reusados en otros sistemas de información (**Eriksson, 2000**).

La ventaja del reuso también es aplicable a los modelos. Si el mismo modelo de negocio, puede actuar como la base de varios sistemas de información. Este puede ser reusado como la entrada básica para definir los requerimientos de cada sistema. Sin un modelo de negocio común, cada equipo de desarrollo de sistemas crearía su propio modelo de análisis para entender el mundo real. No solo es trabajo redundante, sino que aumenta el riesgo de que los equipos interpreten la realidad de manera diferente y; por lo tanto, desarrollarán sistemas incompatibles.

### c) Mejora:

Un modelo de negocio puede ser usado para mejorar la operación actual de la organización. Esta técnica, a veces llamada Mejora de Procesos de Negocio (Business Process Improvement, BPI), es usada para identificar las posibles formas de hacer el negocio más eficiente. El negocio actual es modelado y luego analizado para oportunidades de perfeccionamiento o mejora. **La mejora de procesos de negocio sugiere que el negocio sea cambiado incrementalmente en lugar de hacerlo inmediatamente y radicalmente. Cuando una oportunidad de mejora es identificada, es producido un nuevo modelo de negocio para demostrar cómo este debe verse después de que esos cambios sean implementados (Cauvet, 2008).**

Varias actividades deben ser completadas en orden para cambiar el negocio e implementar un nuevo modelo de negocio:

1. Describir nuevas rutinas y crear soporte administrativo para esas rutinas.
2. Entrenar al personal afectado por el cambio; enseñarles los nuevos procesos y motivarlos a volverse parte del cambio.
3. Cambiar los sistemas de información que participan en el negocio para mejorar el soporte y perfeccionar su operación.
4. Negociar con subcontratistas y personal de la empresa que necesitaran adaptarse a los cambios.

Dependiendo de la extensión de los cambios y del cuidado en el proceso de modelado, mejorar el negocio puede ser trabajo simple o complicado. Con frecuencia un modelado superficial y poco reflexivo, genera resultados rápidos, pero de escaso o nulo impacto, incluso a la larga suelen ser contraproducentes.

#### **d) Innovación:**

La innovación de negocio involucra el análisis del negocio actual y la búsqueda del modelo para nuevas formas de hacer las cosas. El modelo de negocio y sus procesos son cambiados significativamente para crear diferentes y mejorados procesos. A menudo, pueden existir rutinas en un negocio por razones históricas o, porque la infraestructura demanda que se hagan de cierta manera.

La innovación implica un premio mucho más grande que la mejora, pero si la innovación tiene éxito, el resultado puede lograr grandes ganancias en eficiencia. La innovación es por consiguiente usada en compañías que requieren un cambio radical incitado por un pobre rendimiento, falta de presupuesto, y productividad ineficiente (Ruiz, 2008).

Una forma extrema de innovación de negocio es la Reingeniería de Procesos de Negocio (BPR por sus siglas en inglés). La BPR define los cambios radicales para los procesos de negocio; esto significa que todas las cosas acerca de la manera actual en que funciona el negocio son cuestionadas y a menudo substancialmente cambiadas.

Conseguir tal innovación es mucho más difícil y tiene un alto riesgo de fracaso. Por consiguiente, ha habido una fuerte resistencia para la BPR.

#### **e) Diseñar Nuevos Procesos:**

El modelado de negocio puede ser usado para crear nuevos modelos. Los modelos son usados para determinar si la organización actual, los recursos y los sistemas de información pueden ser fácilmente usados o adaptados a los nuevos procesos. Los

modelos de negocio también son usados para hacer benchmarking de un negocio, esto es, para copiar o estudiar procesos de negocios usados por los competidores para medir nuestros propios procesos contra los de la competencia.

A menudo, nuevos procesos son diseñados sobre la base de una visión de nuevas oportunidades. Modelando esta visión o idea se crea un “primer intento” que prueba la viabilidad del proceso. Obviamente, otras actividades tienen que ser llevadas a cabo antes de implementar el proceso, incluyendo cálculos de ganancia, análisis de costo y estudio de mercado. Estas actividades pueden usar el nuevo modelo de proceso como una especificación de las metas proyectadas y de los recursos necesarios, para determinar cómo el nuevo proceso es implementado dentro del negocio actual (**Sinogas, 2001**).

#### 1.1.4. Importancia del Modelado.

Un negocio es un sistema complejo. Está constituido por una organización, en la cual generalmente, se distinguen tanto elementos tangibles como intangibles tales como grupos de trabajo formales y las funciones de dichos grupos. Algunas de estas funciones, sin embargo, no están restringidas a un solo departamento; cruzan horizontalmente a través de varios departamentos.

El método tradicional para documentar un negocio es dibujar un mapa de la organización, que divide la empresa en un número de departamentos o secciones como por ejemplo, producción, mercadotecnia, ventas, investigación y desarrollo entre otros.

Esta técnica, pese a ser simple, no ofrece en su propia naturaleza una visión global de la organización, de manera que la documentación desarrollada con frecuencia cae en los extremos de ser redundante, contradictoria o inexistente.

La Notación para el Modelado de Procesos de Negocios (Business Process Modeling Notation) es una iniciativa mantenida actualmente por la Object Management Group, Inc. (OMG), la cual ha sido usada para el modelado es el proceso de negocio, el cual describe las actividades clave de la organización y cómo se relacionan e interactúan con los recursos del negocio para lograr la meta establecida para el proceso (OMG, 2008).

Sin embargo, este ideal de abstraer la realidad mediante un modelo no está exento de ciertas consideraciones.

Un modelo de negocio nunca puede ser totalmente exacto o completo, simplemente, porque ninguno de los posibles observadores de un negocio tendrá una percepción idéntica o estará de acuerdo con un modelo exacto. El modelo debe concentrarse en las tareas y mecanismos clave principales del negocio. Determinar con precisión las tareas principales e identificar qué debe plasmarse en el modelo es responsabilidad del modelador, lo que implica una cierta proporción de subjetividad. Igualmente, el

modelo de una vista futura de un negocio no necesariamente va a realizarse tal como se planeó.

Los cambios en el mundo real pueden afectar la base sobre la que el modelo fue creado quedando este como algo incompleto (Hernández, 2005). Pese a estas limitaciones, los siguientes argumentos para producir modelos de negocio apoyan su existencia:

1. **Permiten comprender mejor los mecanismos clave de un negocio existente:** Se debe proveer una imagen clara de sus roles y tareas en la organización global, los modelos pueden ser usados para entrenar a las personas. Pueden ser usados tanto en una organización jerárquica como en una organización orientada a procesos.
2. **Actúan como base para crear sistemas de información:** Las descripciones de negocio son usadas para identificar el apoyo de sistemas de información a los principales procesos de la organización. Los modelos también son usados como una base para especificar los requerimientos clave de esos sistemas.
3. **Facilitan la identificación de ideas para mejorar la estructura actual del negocio y su operación:** Los modelos permiten identificar situaciones susceptibles de ser mejoradas, la construcción de un modelo implica un proceso reflexivo del porqué se hacen las cosas como se hacen, de manera que pueden visualizarse cambios en el negocio actual que son necesarios para implementar el modelo mejorado.
4. **Para experimentar con un nuevo concepto de negocio:** Un modelo es una entidad conceptual de bajo costo sobre, la cual pueden hacerse ciertas pruebas para validar su operación, lo que los hace ser un medio para la adopción de mejores prácticas inspiradas por otros modelos de negocios exitosos. También, permite tomar ventaja mediante la adopción de nuevas tecnologías, tales como las relacionadas con Internet.
5. **Para identificar oportunidades de Outsourcing:** Los elementos del negocio no considerados como parte central, son delegados a proveedores externos. Los modelos son usados como especificación para los proveedores.
6. **Para mostrar la estructura de un negocio innovado:** Los modelos sirven para presentar ante la gerencia la nueva propuesta de trabajo, de manera tangible y concreta. A partir de este punto, es posible definir nuevas acciones, entonces los modelos se vuelven la base para los planes de acción que apoyarán la transformación del negocio.

# Resumen

1. El modelo de negocios es el estudio de la organización y se le define como un proceso de representación de uno o más aspectos o elementos de una empresa u organización.
2. La importancia de modelar los procesos de Negocios puede apreciarse por lo siguiente:
  1. Permiten comprender mejor los mecanismos clave de un negocio existente.
  2. Actúan como base para crear sistemas de información.
  3. Facilitan la identificación de ideas para mejorar la estructura actual del negocio y su operación.
  4. Para experimentar con un nuevo concepto de negocio.
  5. Para identificar oportunidades de Outsourcing.
  6. Para mostrar la estructura de un negocio innovado.
7. El Modelado de procesos de negocios tiene diversos estándares normados por la OMG, entre ellos tenemos a BPMN, UML, XMI, CORBA, entre otros.
8. BPMN es un estándar administrado por la OMG (Object Management Group) y ahora es una norma ISO (International Organization for Standardization) que lleva el código ISO / IEC 19510:2013

Pueden revisar los siguientes enlaces para ampliar los conceptos vistos en esta unidad:

1. [https://es.wikipedia.org/wiki/Object\\_Management\\_Group](https://es.wikipedia.org/wiki/Object_Management_Group)
2. [http://futstrat.com/books/book\\_images/Guia%20de%20Referencia%20y%20Modelado%20BPMN%20chap1-2.pdf](http://futstrat.com/books/book_images/Guia%20de%20Referencia%20y%20Modelado%20BPMN%20chap1-2.pdf)
3. <http://www.omg.org/spec/BPMN/2.0/>

## 1.2. BPMN

### 1.2.1. BPM y BPMN – Importancia del estándar BPMN

1. **BPM: (Business Process Management = Gerencia de Procesos de Negocios)** es un enfoque sistemático para identificar, levantar, documentar, diseñar, ejecutar, medir y controlar tanto los procesos manuales como automatizados, con la finalidad de lograr a través de sus resultados en forma consistente los objetivos de negocio que se encuentran alineados con la estrategia de la organización. BPM abarca el apoyo creciente de TI con el objetivo de mejorar, innovar y gestionar los procesos de principio a fin, que determinan los resultados de negocio, crean valor para el cliente y posibilitan el logro de los objetivos de negocio con mayor agilidad.

El concepto de BPM, como disciplina de gestión por procesos, es amplio; tiene objetivos claros y bien definidos:

1. **Lograr o mejorar la agilidad de negocio en una organización:** Concepto que se entiende como la capacidad que tiene una organización de adaptarse a los cambios del entorno a través de los cambios en sus procesos integrados.
  2. **Lograr mayor eficacia:** Se entiende como la capacidad de una organización para lograr, en mayor o menor medida, los objetivos estratégicos o de negocio.
  3. **Mejorar los niveles de eficiencia:** Eficiencia es la relación entre los resultados obtenidos y los recursos utilizados, es decir, el grado de productividad de un resultado. El término está relacionado con los indicadores de productividad en cuanto a calidad, costos y tiempos.
- 
1. **BPMN: (Business Process Model and Notation = Modelo y Notación de Procesos de Negocio)** es una notación gráfica estandarizada que permite el modelado de procesos de negocio, en un formato de flujo de trabajo (**workflow**).

BPMN fue inicialmente desarrollada por la organización *Business Process Management Initiative (BPMI)*, y es actualmente mantenida por el *Object Management Group (OMG)*, después de la fusión de las dos organizaciones en el año 2005. Su versión actual es la **2.0.2**.

El principal objetivo de BPMN es proporcionar una notación estándar que sea fácilmente legible y entendible por parte de todos los involucrados e interesados del

negocio (**stakeholders**). Entre estos interesados, están los analistas de negocio (quienes definen y redefinen los procesos), los desarrolladores técnicos (responsables de implementar los procesos) y los gerentes y administradores del negocio (quienes monitorizan y gestionan los procesos).

En síntesis, **BPMN** tiene la finalidad de servir como lenguaje común para cerrar la brecha de comunicación que frecuentemente se presenta entre el diseño de los procesos de negocio y su implementación.

([https://es.wikipedia.org/wiki/Business\\_Process\\_Model\\_and\\_Notation](https://es.wikipedia.org/wiki/Business_Process_Model_and_Notation))

## 2. Importancia del estándar BPMN:

El papel del **BPM**, exige un nuevo lenguaje común en el mundo de los negocios y de la **TI** (Tecnología de la información), que es **BPMN** (Business Process Model and Notation) el estándar de **OMG**.

BPMN es lo suficientemente simple y fácilmente comprensible para ser aplicado en cualquier negocio, sin embargo, es suficientemente rico como para apoyar la aplicación ejecutable, sin cambiar el metamodelo subyacente. Como resultado, se ha convertido en el estándar de facto o la dirección del futuro de BPM Suite los vendedores que van de Lombardi, Savvion, y Apia, a TIBCO, Oracle, IBM y SAP. Por lo tanto, la comprensión de cómo modelar procesos con eficacia en BPMN se ha convirtiendo en una habilidad que cualquier experto en el tema de procesos de negocio debe conocer.

Durante años las herramientas de BPM que tenían alguna noción de cajas y flechas, estaban apoyadas en BPMN, y no hay duda de que muchos Suites BPM ahora dicen apoyar BPMN 2.0 en sus herramientas de diseño ejecutable.

En resumen, la importancia de modelar con BPMN radica en lo siguiente:

1. BPMN es un estándar internacional de modelado de procesos aceptado por la comunidad.
2. BPMN es independiente de cualquier metodología de modelado de procesos.
3. BPMN crea un puente estandarizado para disminuir la brecha entre los procesos de negocio y la implementación de estos.
4. BPMN permite modelar los procesos de una manera unificada y estandarizada permitiendo un entendimiento a todas las personas de una organización.

### 1.2.2. Lo que es y lo que no es BPMN

BPMN fue desarrollado para modelar procesos. Esta afirmación suena banal, pero en muchas ocasiones se critica que BPMN no puede representar las siguientes estructuras:

1. Mapas de procesos
2. Estructuras organizacionales
3. Estructuras de datos
4. Estrategias y modelos de negocio
5. Reglas de negocio
6. Infraestructura de TI

BPMN se concentra en el modelamiento de los procesos y no de otras estructuras organizacionales. BPMN no fue concebida como una notación para modelar otras estructuras de la arquitectura empresarial. Nos parece bien que así sea; imagínese la complejidad que tendría, fuera de la que tiene, si pretendiese abarcar la metodología y las reglas sintácticas para todos los otros modelos que describen una organización.

También, nosotros estamos conscientes que no es suficiente aplicar solo BPMN para introducir BPM en una organización. Muchos expertos de BPM, sobre todo aquellos que vienen del mundo de ARIS y han utilizado la técnica de «event process chain (EPC)» se quejan que BPMN no es suficiente. Esta falta de entendimiento se debe principalmente a que no comprendieron los objetivos del estándar de BPMN:

Modelos de BPMN pueden relacionarse con otros modelos de una arquitectura empresarial. BPMN ofrece la posibilidad de ampliarse, por ejemplo de incluir símbolos propios o de relacionarse con otros objetos de una arquitectura empresarial.

Admitimos que sería mejor si es que existiera una convención que fuera más allá de solo contemplar la vista de los procesos. La integración a una arquitectura empresarial no lo considera el estándar, pero tampoco lo impide.

Así algunas empresas han hecho esta integración en sus plataformas de arquitectura empresarial.

### 1.2.3. BPMN en las Organizaciones – El Ciclo BPMN

Una Organización puede iniciar un Proyecto de BPM en las siguientes situaciones:

- a) Para el rediseño y/o mejoramiento del rendimiento de procesos existentes y actuales con apoyo de tecnología.

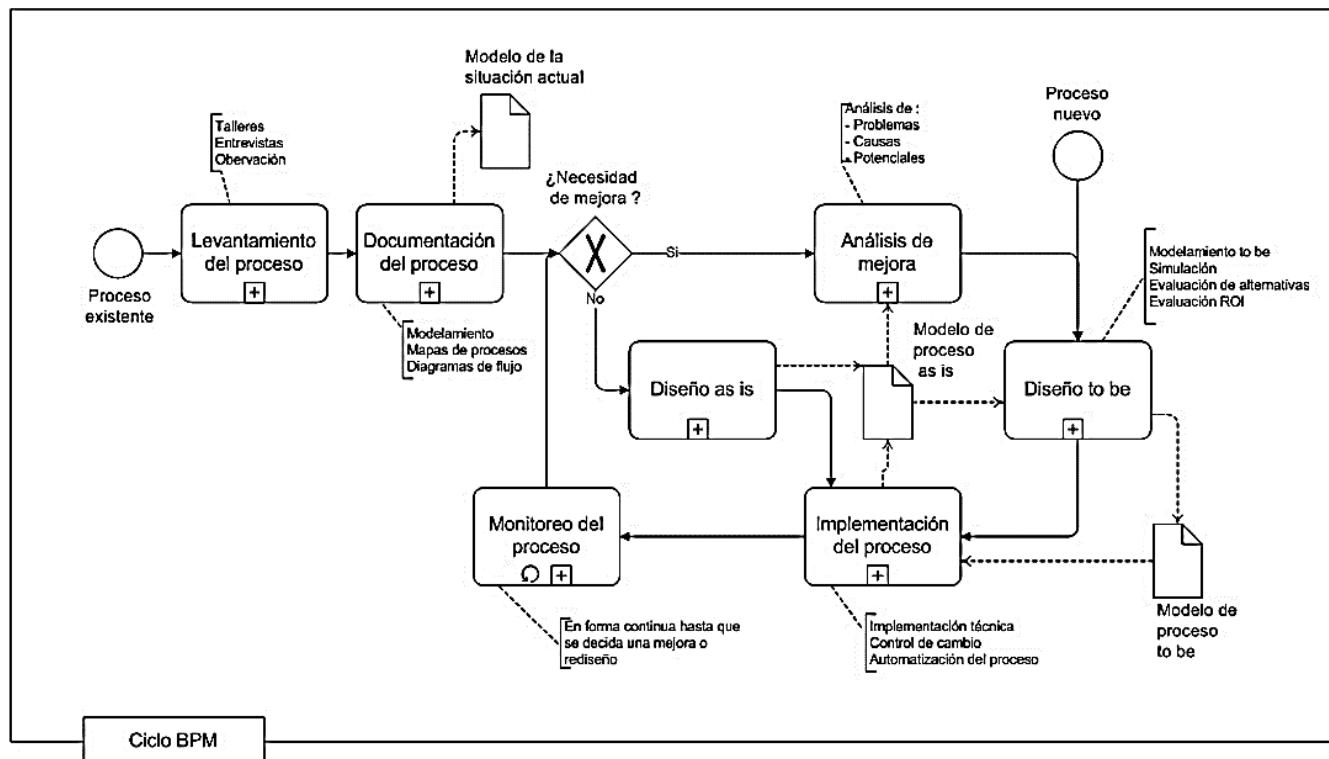
b) Levantar y documentar procesos actuales, con la finalidad de automatizarlos u otros fines como por ejemplo preparar una certificación de ISO 9000.

c) Introducir un nuevo proceso en la organización.

En la mayoría de los casos, nos encontramos con la primera situación en proyectos de BPM, ya que se espera una mejora por medio de un rediseño y una posterior implementación por medio de TI.

Por lo general los modelos de BPM, son muy simples o muy complejos. Si son muy simples, contienen solo procedimientos banales y sirven a lo más para presentaciones de marketing. Mientras que si son modelos muy complejos tratan de captar todas las ocurrencias y eventualidades, amarrando u obligando al usuario en un plan de trabajo demasiado intensivo, que generalmente no es aplicable en la práctica.

Por otra parte, si no contamos con ningún modelo nos faltaría una carta de navegación para orientarnos en nuestros proyectos de BPM. A continuación, presentamos un modelo que representa el ciclo de BPM, ni muy simple ni muy sofisticado, pero que ha dado muy buenos resultados en la práctica.



**Figura 5: Ciclo de Vida de BPMN**  
 Fuente.- Freund et.al. - BPMN 2.0  
 Manual de Referencia y Guía Práctica

El ciclo está pensado para ser aplicado para cada proceso por separado o en forma independiente. Cada proceso puede encontrarse en un estado diferente del ciclo. El ciclo comienza a partir de dos posibles constelaciones:

Un proceso actual que debe levantarse y documentarse y/o rediseñarse. Se debe introducir un nuevo proceso, no existente en la organización.

En la fase de “**Levantamiento del Proceso**” primero, se debe recoger la información sobre cómo está organizado el flujo de trabajo. Esto se realiza con la ayuda de técnicas de moderación, talleres, entrevistas, recolección de documentación, etc.

1. Para esto, en el proceso a levantar, se debe:
2. Delimitar claramente de procesos anteriores o posteriores.
3. Describir los servicios que produce para los clientes y qué prioridad tiene desde el punto de vista de los objetivos empresariales
4. Representar tanto el flujo de trabajo como los roles que intervienen en cada uno de los pasos, los recursos que se utilizan y los sistemas de información que lo apoyan

En la etapa de “**Documentación del Proceso**” el conocimiento adquirido en la etapa de levantamiento, se documenta en un modelo de procesos que refleja la situación actual.

La documentación resultante comprende los diagramas de los flujos, fichas de descripción, políticas de negocio y procedimientos que se utilizan para ejecutar el trabajo.

Las debilidades identificadas, en la fase de “**Análisis de mejora**” o las desviaciones que muestra el “**Monitoreo del Proceso**”, son, por lo general, el punto de partida para un rediseño de procesos. Eventualmente, se pueden evaluar diferentes variantes o escenarios con ayuda de simuladores. Esto aplica también si se está diseñando un proceso nuevo. En ambos casos, el resultado o entregable, es un modelo de procesos deseado (**To be**).

La etapa de “**Implementación del Proceso**” abarca tanto la implementación técnica como también las adaptaciones organizacionales que se requieren. La gestión del cambio (en inglés: Change Management) y la estrategia de comunicación constituyen elementos fundamentales a considerar para el éxito del proyecto. El modelo técnico puede implementarse por medio de un Process Engine o una Suite de BPM (en inglés: Business Process Management Suite, BPMS) o a través de un clásico desarrollo de software. El resultado final de la implementación técnica del proceso en la situación actual (**As is**) automatizado y documentado, corresponde con el modelo de proceso deseado (**To be**).

En la literatura y en el mercado, se utilizan varios términos para sistemas que implementan procesos: sistema de workflow (WfM), Business Process Management Suite (BPMS), motor de workflow y Process Engine. De ahora en adelante, vamos a utilizar preferentemente el término «Process Engine» en forma genérica que en la práctica puede ser cualquiera de ellos. Por lo general, la Suite de BPM (BPMS) es el sistema más completo que trae todas las componentes integradas (modelador técnico, motor de workflow, panel de control, interfaz de usuario, APIS de integración y en algunos casos Enterprise Service Bus (ESB).

Las fases desde el **“Levantamiento del Proceso”** hasta la **“Implementación del Proceso”** se administran, por lo general, por medio de la organización de un proyecto, mientras que el **“Monitoreo del Proceso”** (en inglés: Process controlling) se concibe como un proceso continuo y forma parte de todas las operaciones. Las actividades más importantes de **“Monitoreo del Proceso”** son el control constante de las operaciones (técnicalemente hablamos del control de instancias de los procesos reales) y su respectiva evaluación de los indicadores. De acuerdo a la escuela de BPM, si se detectan problemas puntuales debieran corregirse de inmediato o en línea. Si hay recursos disponibles, es posible solucionar problemas estructurales sin necesidad de formular un proyecto, pero si sus causas no están claras o son complejas, se hace necesario planificar e implementar un proyecto de mejora y rediseño. La decisión sobre si es necesario formular un proyecto nuevo o instalar un equipo de trabajo en operaciones, debiera tomársela el responsable del proceso de común acuerdo con los participantes.

Con esta breve explicación de cómo funciona el ciclo de BPM, el lector se dará cuenta de la importancia que tienen los modelos de procesos en BPM y junto a ello la importancia que puede adquirir un estándar de modelamiento como BPMN. Usted puede constatar también que el modelamiento de procesos no es una etapa del ciclo de BPM, sino que es más bien una actividad transversal, porque de facto se aplica en todas las fases del ciclo, sobre todo en las fases de **“Documentación del Proceso”**, **“Diseño As is”** y **“Diseño To be”**. Desgraciadamente, siempre nos volvemos a encontrar con gente que confunden la **“Documentación del Proceso”** con el modelamiento del proceso y lo incluyen como una fase en el ciclo; esto es una equivocación.

El ciclo BPM muestra en sus principales fases cómo funciona el círculo virtuoso de mejora continua de los procesos. Para aplicarlo es necesario:

Asignar responsabilidades a los procesos y a cada uno de sus pasos Emplear métodos de análisis y gestión en el contar con el apoyo de soluciones adecuadas de TI.

Lograr una coordinación fluida entre estas tres componentes es tarea de gestión por procesos (BPM-Governance). Gestión por procesos se encuentra por sobre cualquier proyecto de BPM y tiene por consiguiente la misión de introducir la «Gestión por Procesos de Mejora Continua».

El primer objetivo de BPMN fue desarrollar una notación gráfica que permitiera automatizar en forma más rápida los procesos. Esta es la razón del porqué usted tiene que entender los fundamentos de TI si quiere diseñar buenos modelos en BPMN. Si ha logrado entender los conceptos detrás de BPMN, habrá construido fundamentos sólidos para diseñar buenos modelos y de esta forma poder cerrar la brecha entre la capa de negocio y la capa de TI. (**Freund et.al., 2014**).

#### **1.2.4. Herramienta CASE: Introducción y descripción de su interfaz principal.**

**Bizagi Modeler** es una herramienta que le permite modelar y documentar procesos de negocio basado **100%** en el estándar de acepción mundial conocido como **Business Process Model and Notation (BPMN)**.

Usted puede crear documentación de procesos en **Word, PDF, SharePoint o Wiki**, e importar o exportar la información de los mismos desde y hacia **Visio** o **XML** entre otros.

Con su comportamiento intuitivo y su amigable interfaz gráfica, usted podrá diagramar y documentar procesos de manera más rápida y fácil sin necesidad e esperar por alguna rutina de validación.

Bizagi guarda sus procesos en una archivo **.bpm**. Cada archivo se denomina **Modelo** y puede contener uno o más diagramas. Un modelo puede ser toda su organización, su área o un proceso específico según sean sus necesidades. Cada pestaña es un diagrama y allí usted dibuja sus procesos. Usted puede navegar entre diagramas seleccionando la pestaña del diagrama en la parte inferior del Modelo.

El Modelado de Procesos es un ejercicio en el cual usted diseña y diagrama un flujo de proceso. El proceso debería ser auto explicativo de manera que cualquier persona pueda entenderlo fácilmente. Modelar un proceso le permite a usted y su equipo entender y analizar los procesos con el fin de proponer mejoras de manera iterativa.

BPMN es una notación gráfica creada para proveer un lenguaje unificado de acepción mundial, utilizada para la especificación de procesos de negocio. El estándar ha sido mantenido por el Object Management Group (OMG). Bizagi es un miembro activo dentro del OMG.

Bizagi Modeler soporta la versión actual, BPMN 2.0. Para mayor información, vea <http://www.omg.org/spec/BPMN/2.0>

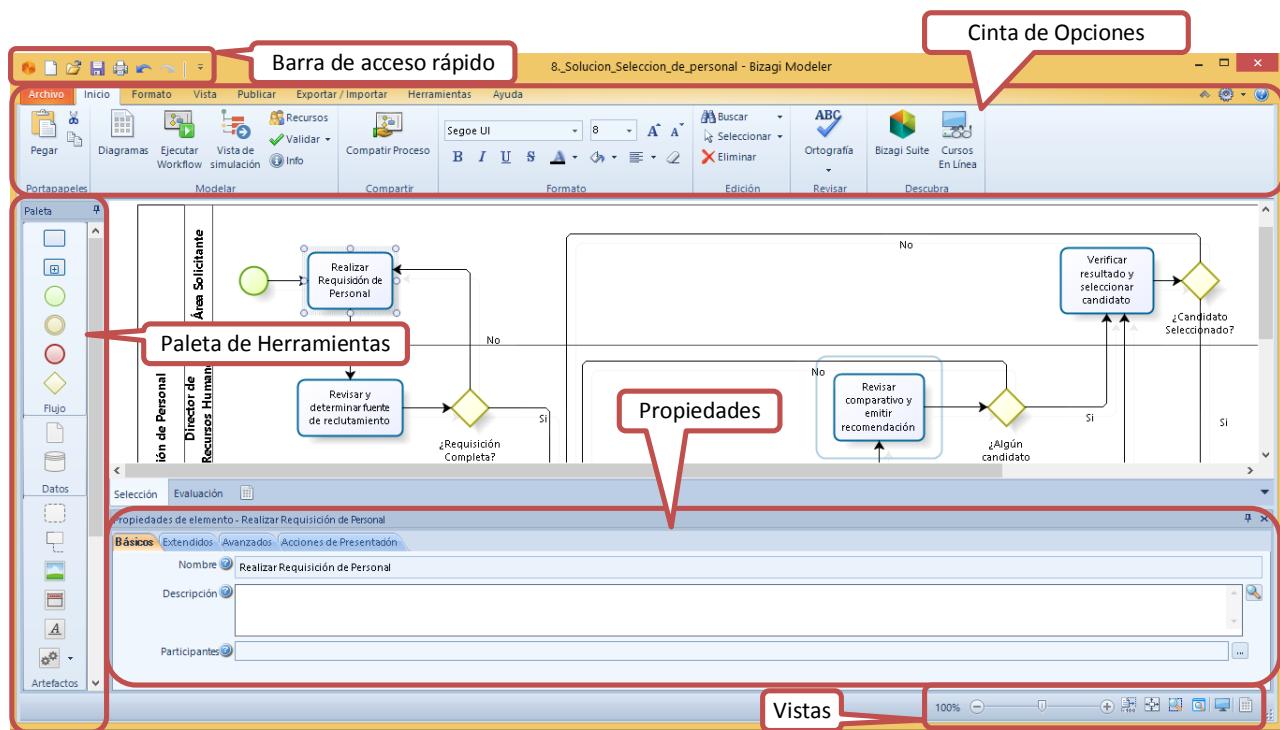


Figura 6: Interfaz principal de Bizagi Modeler

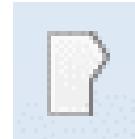
Fuente.- elaboración propia

### 1.2.5. Elementos organizativos del BPMN: Swimplanes, pools, lanes y groups.

ELEMENTO	DESCRIPCIÓN	NOTACIÓN
<b>Pool</b>	<p>Un pool es un contenedor de procesos simples (contiene flujos de secuencia dentro de las actividades).</p> <p>Un proceso está completamente contenido dentro de un pool. Siempre existe por lo menos un pool.</p>	
<b>Lane</b>	<p>Es una sub-partición dentro del proceso. Los lanes se utilizan para diferenciar roles internos, posiciones, departamentos, etc.</p>	

**Fase**

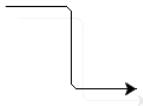
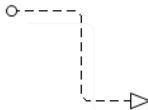
Es una sub-partición dentro del proceso. Puede indicar diferentes etapas durante el mismo.



**Group** Es un artefacto que provee un mecanismo visual para agrupar elementos de un diagrama de manera informal.



### 1.2.6. Elementos básicos: tareas, conectores y compuerta básica

ELEMENTO	DESCRIPCIÓN	NOTACIÓN
<b>Tarea</b>	Es una actividad atómica dentro de un flujo de proceso. Se utiliza cuando el trabajo en proceso no puede ser desglosado a un nivel más bajo de detalle	
<b>Asociación</b>	Se utiliza para asociar información y artefactos con objetos de flujo. También, se utiliza para mostrar las tareas que compensan una actividad.	
<b>Flujo secuencia</b>	Un flujo de secuencia es utilizado para mostrar el orden en el que las actividades se ejecutarán dentro del proceso.	
<b>Flujo mensaje</b>	Se utiliza para mostrar el flujo de mensajes entre dos entidades que están preparadas para enviarlos y recibirlos.	

**Compuerta exclusiva**

De **divergencia**: Se utiliza para crear caminos alternativos dentro del proceso, pero solo uno se selecciona.



De **convergencia**: Se utiliza para unir caminos alternativos.

**Ejemplo 1:** Para hacer un “**Café con leche**”, primero debe hervir el agua y luego combinarla con la leche, a continuación echar el café y decidir si desea azúcar o edulcorante. En cualquiera de los casos, echará dos cucharadas del complemento elegido. Finalmente, mezclará y servirá.

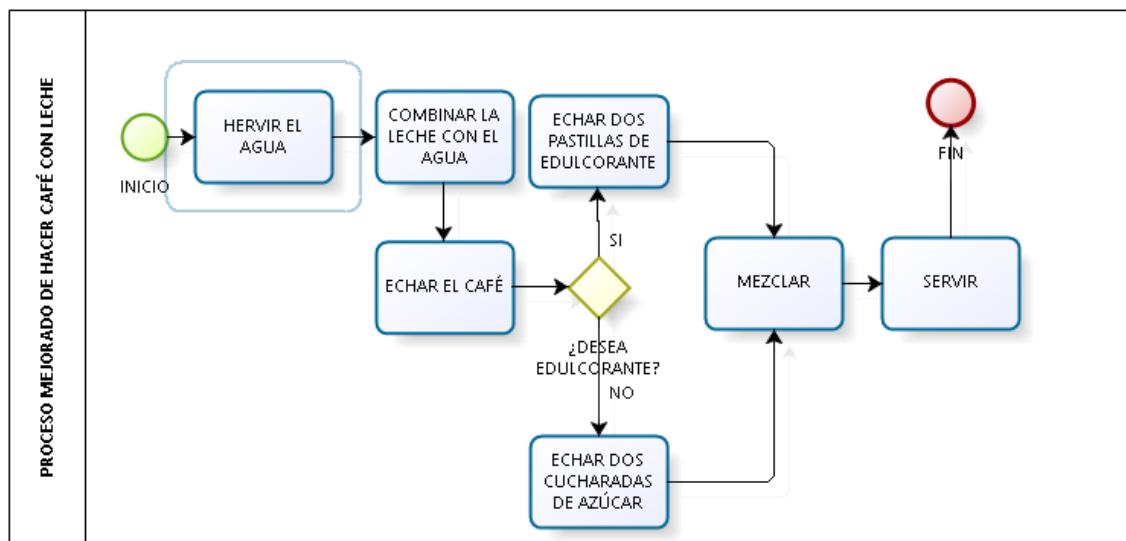


Figura 7: Diagrama de flujo básico

Fuente.- elaboración propia

**Ejemplo 2:** El proceso de atención de entrega “**delivery**”, comienza por recoger los productos que debe entregar, luego realiza una verificación en su lista de productos para comprobar si están completos los productos a entregar. Si están completos, procede a acomodar el pedido en su caja transportadora y determina la ruta más corta para llegar a dirección de entrega.

Busca la dirección y si la encuentra entrega el pedido, cobra el importe y entrega el comprobante de pago.

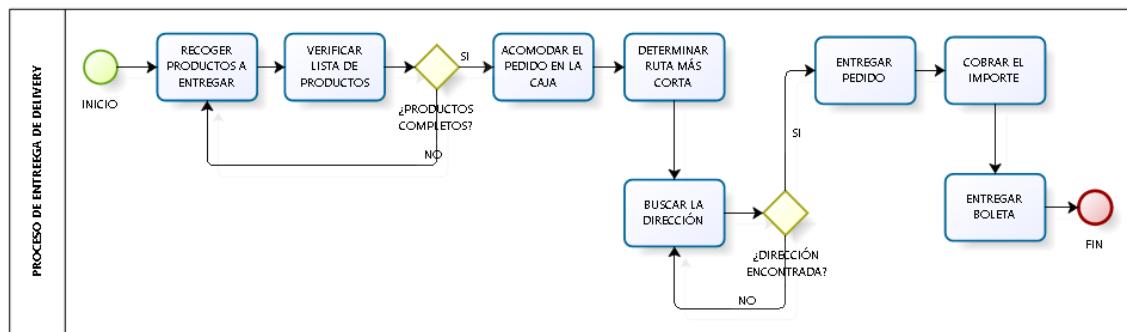


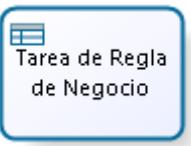
Figura 8: Diagrama de flujo con dos compuertas

Fuente.- elaboración propia

**Ejemplo 3:** Modifique el ejemplo anterior incluyendo la tarea de controlar el tiempo. Si tarda más de 30 minutos la entrega, es gratis. También, incluya la variante, si el pago realizado es mayor al importe a cobrar, por lo cual deberá calcular el vuelto y modelar la situación en el diagrama.

### 1.2.7. Elementos de Flujo de trabajo básico: tipo de tareas, tipo de eventos iniciales y finales

ELEMENTO	DESCRIPCIÓN	NOTACIÓN
Tarea de usuario	Es una tarea de workflow típica donde una persona ejecuta con la asistencia de una aplicación de software.	
Tarea de Servicio	Es una tarea que utiliza algún tipo de servicio que puede ser Web o una aplicación automatizada.	
Tarea de recepción	Es una tarea diseñada para esperar la llegada de un mensaje por parte de un participante externo (relativo al proceso).	
Tarea envío	Es una tarea diseñada para enviar un mensaje a un participante externo (relativo al proceso).	

<b>Tarea de script</b>	Es una tarea que se ejecuta por un motor de procesos de negocio. El usuario define un script en un lenguaje que el motor pueda interpretar.	 Tarea de Script
<b>Tarea manual</b>	Es una tarea que espera ser ejecutada sin la asistencia de algún motor de ejecución de procesos de negocio o aplicación.	 Tarea Manual
<b>Tarea de Regla de Negocio</b>	Ofrece un mecanismo para que el proceso provea una entrada a un motor de reglas de negocio y obtenga una salida de los cálculos que realice el mismo.	 Tarea de Regla de Negocio
<b>Ciclo multiinstancia</b>	Las tareas pueden repetirse secuencialmente comportándose como un ciclo. El ciclo multi-instancias permite la creación de un número deseado de instancias de actividad que pueden ser ejecutadas de forma paralela o secuencial.	 Ciclo multiinstancia
<b>Ciclo estándar</b>	Las tareas pueden repetirse secuencialmente comportándose como un ciclo. Esta característica define un comportamiento de ciclo basado en una condición booleana. La actividad se ejecutará siempre y cuando la condición booleana sea verdadera.	 Ciclo estándar

### Objetos de datos

ELEMENTO	DESCRIPCIÓN	NOTACIÓN
<b>Objeto de datos</b>	Provee información acerca de cómo los documentos, datos y otros objetos se utilizan y actualizan durante el proceso.	
<b>Depósito de datos</b>	Provee un mecanismo para que las actividades recuperen o actualicen información almacenada que persistirá más allá del alcance del proceso.	

## Eventos

ELEMENTO	DESCRIPCIÓN	NOTACIÓN
<b>Evento de inicio simple</b>	Indica dónde se inicia un proceso. No tiene algún comportamiento particular.	
<b>Evento intermedio simple</b>	Indica que algo sucede en algún lugar entre el inicio y el final de un proceso. Esto afectará el flujo del proceso, pero no iniciará (directamente) o finalizará el mismo.	
<b>Evento de finalización simple</b>	Indica que el flujo finaliza.	

# Resumen

1. **BPM:** (Business Process Management = Gerencia de Procesos de Negocios) es un enfoque sistemático para identificar, levantar, documentar, diseñar, ejecutar, medir y controlar tanto los procesos manuales como automatizados.
2. **BPMN:** (Business Process Model and Notation = Modelo y Notación de Procesos de Negocio) es una notación gráfica estandarizada que permite el modelado de procesos de negocio, en un formato de flujo de trabajo (workflow)
3. La importancia de modelar con BPMN radica en:
  1. BPMN es un estándar internacional de modelado de procesos aceptado por la comunidad.
  2. BPMN es independiente de cualquier metodología de modelado de procesos.
  3. BPMN crea un puente estandarizado para disminuir la brecha entre los procesos de negocio y la implementación de estos.
  4. BPMN permite modelar los procesos de una manera unificada y estandarizada permitiendo un entendimiento a todas las personas de una organización.
  5. El Ciclo de BPMN constituye carta de navegación para orientarnos en nuestros proyectos de BPM. Si el proyecto, es complejo su uso es muy recomendable.
  6. Una de las herramientas Case de mayor uso para proyectos BPMN es Bizagi Modeler. Su interfaz contiene las herramientas necesarias para realizar el modelamiento de procesos de negocios.

Pueden revisar los siguientes enlaces para ampliar los conceptos vistos en esta unidad:

1. <http://www.omg.org/spec/BPMN/2.0>
2. <http://bpmn-bayard.blogspot.com/2011/06/41-bpmn-framework.html>
3. <http://www.club-bpm.com/Observatorio-y-Recursos-BPM.htm>

## 1.3. Modelado a nivel Descriptivo

### 1.3.1. Niveles de Modelado: Descriptivo, analítico y ejecutable

La OMG establece tres niveles de modelado que se describen a continuación:

**NIVEL DESCRIPTIVO:** usado para comunicar procesos a través de la organización. Esta descripción de alto nivel, describe los flujos esenciales del proceso de negocio ignorando complejidades e incluso las reglas de Negocio.

En este nivel, los diagramas de Modelado del flujo del proceso son parecidos a los diagramas de flujo tradicionales, centrándose en el “camino correcto” o “happy path” y utiliza un subconjunto de toda la paleta de símbolos de BPMN. Se pueden omitir algunos flujos excepcionales

**NIVEL ANALÍTICO:** describe todas las actividades, decisiones y excepciones necesarias para analizar un proceso completamente y crear requerimientos detallados. Expresa toda la lógica de negocio y las reglas de negocio.

Se utiliza la paleta completa de BPMN para describir de manera precisa el flujo del proceso. Se incluyen cualquier posible camino excepcional en el mismo.

Aún no son procesos ejecutables (se omiten los detalles técnicos) como las estructuras de datos y expresiones, por ejemplo:

En este nivel, se refleja una perspectiva orientada al negocio, entendible tanto por analistas como por técnicos. Existen 2 posibles utilizaciones a este nivel:

1. Reflejar y documentar a nivel de análisis el proceso con todo detalle, aunque no sea la intención “automatizarlo” con un BPMS
2. Ser el punto de partida para su automatización en un BPMS

**NIVEL EJECUTABLE:** modelar a este nivel depende mucho de las capacidades del BPMS que utilicemos. Actualmente, para ejecutar se utiliza el lenguaje BPEL.

Son procesos que se pueden ejecutar directamente en el BPMS. Actualmente, es el propio producto el que dice como “parametrizar” los procesos para hacerlos ejecutables.

(BPM-Cómo alcanzar la agilidad y eficiencia operacional a través de BPM y la organización orientada a procesos – José ramón País.)

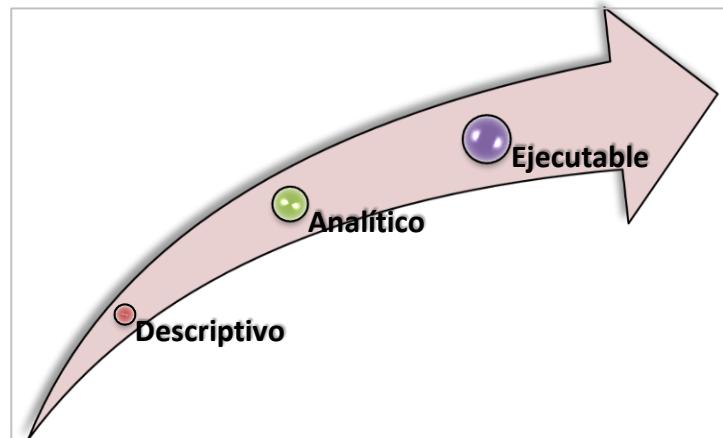


Figura 9: Niveles del BPMN  
Fuente.- elaboración propia

### 1.3.2. Elementos de Modelos descriptivos: Tipos de Eventos iniciales y finales, eventos intermedios simples, artefactos, conectores y elementos de legibilidad.

#### EVENTOS DE INICIO:

ELEMENTO	DESCRIPCIÓN	NOTACIÓN
Evento de inicio de mensaje	Se utiliza cuando el inicio de un proceso se da al recibir un mensaje de un participante externo.	
Evento de inicio de temporización	Se utiliza cuando el inicio de un proceso ocurre en una fecha o tiempo de ciclo específico. (ej.: todos los viernes).	
Evento de inicio condicional	Este tipo de evento dispara el inicio de un proceso cuando una condición se cumple.	
Evento de inicio de señal	El inicio de un proceso se da por la llegada de una señal que ha sido emitida por otro proceso. Tenga en cuenta que la señal no es un mensaje; los mensajes tienen objetivos específicos, la señal no.	
Evento de inicio múltiple	Significa que hay múltiples formas de iniciar el proceso. Solo se requiere una de ellas.	
Evento de inicio paralelo múltiple.	Indica que se requieren múltiples disparadores para iniciar el proceso. TODOS los disparadores deben ser lanzados para iniciar el proceso.	

**EVENTOS DE INTERMEDIOS:**

ELEMENTO	DESCRIPCIÓN	NOTACIÓN
<b>Evento intermedio de mensaje</b>	<p>Indica que un mensaje puede ser enviado o recibido.</p> <p>Si un proceso, está esperando por un mensaje y este es capturado, el proceso continuará su flujo.</p> <p>El evento que lanza un mensaje se identifica con una figura sombreada. El evento que capta un mensaje se identifica con una figura sin relleno.</p>	 Mensaje envío  Mensaje recepción
<b>Evento intermedio de temporización</b>	Indica un retraso dentro del proceso. Este tipo de evento puede ser utilizado dentro de un flujo secuencial para indicar un tiempo de espera entre actividades.	
<b>Evento intermedio escalable</b>	El evento indica un escalamiento a través del proceso.	
<b>Evento intermedio de compensación</b>	Permite el manejo de compensaciones. El uso de este tipo de evento dentro del flujo de proceso indica que una compensación es necesaria.	
<b>Evento intermedio condicional</b>	Este evento se activa cuando una condición se cumple.	
<b>Evento intermedio de enlace.</b>	<p>Este evento se utiliza para conectar dos secciones del proceso. Los eventos de enlace pueden ser utilizados para crear ciclos o evitar líneas de secuencia de flujo largas. Si en un proceso, hay dos enlaces (uno que lanza y otro que recibe) el modelador entenderá que están unidos.</p> <p>Si hay dos que lanzan y uno que recibe, el modelador entenderá que los que lanzan están unidos al que recibe.</p> <p>Si hay varios que lanzan y que reciben los nombres de las 'parejas', deben ser iguales para que el Modelador sepa cuál corresponde a cuál.</p>	 Enlace de envío  Enlace de recepción

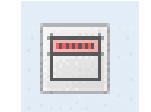
<b>Evento intermedio de señal.</b>	<p>Estos eventos se utilizan para enviar o recibir señales dentro o a lo largo del proceso. Una señal es similar a una bengala que se dispara al cielo para cualquiera que pueda estar interesado en ella y reaccionar.</p> <p>El evento que lanza una señal se identifica con un triángulo sombreado.</p> <p>El evento que recibe una señal se identifica con un triángulo sin relleno.</p>	 <b>Señal de envío</b>  <b>Señal de recepción</b>
<b>Evento intermedio múltiple.</b>	<p>Significa que hay múltiples disparadores asignados al evento.</p> <p>Cuando se utiliza para captar el disparador, solo uno de los disparadores asignados es requerido y el marcador del evento se mostrará vacío.</p> <p>Cuando se utiliza para lanzar el disparador, todos los disparadores asignados se lanzarán y el marcador del evento se mostrará sombreado.</p>	 <b>Múltiple de envío</b>  <b>Múltiple de recepción</b>
<b>Evento intermedio paralelo múltiple.</b>	<p>Significa que hay múltiples disparadores asignados al evento. A diferencia del evento intermedio múltiple, TODOS los disparadores asignados son requeridos para activar el evento.</p>	 <b>Paralelo</b>

### EVENTOS DE FINALIZACIÓN:

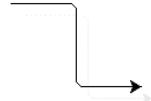
ELEMENTO	DESCRIPCIÓN	NOTACIÓN
<b>Evento de finalización simple</b>	Indica que el flujo finaliza	
<b>Evento de finalización de mensaje</b>	Indica que un mensaje se envía una vez que finaliza el flujo.	
<b>Evento de finalización escalable</b>	Indica que es necesario realizar un escalamiento una vez finaliza el flujo	
<b>Evento de finalización error.</b>	Indica que se debe generar un error. Todas las secuencias activas del proceso son finalizadas. El error será recibido por un evento intermedio de captura de error.	
<b>Evento de finalización cancelación.</b>	Se utiliza dentro de un sub-proceso de transacción e indica que este debe ser cancelado.	

<b>Evento de finalización compensación.</b>	Habilita el manejo de compensaciones. Si una actividad se identifica y fue exitosamente completada, esta será compensada.	
<b>Evento de finalización de Señal.</b>	Indica que una señal es enviada una vez finaliza el flujo.	
<b>Evento de finalización de Múltiple.</b>	Significa que hay múltiples consecuencias de finalizar el flujo. Todas ellas ocurrirán.	
<b>Evento de finalización de Terminal.</b>	Finaliza el proceso y todas sus actividades de forma inmediata.	

#### ARTEFACTOS Y ELEMENTOS DE LEGIBILIDAD:

ELEMENTO	DESCRIPCIÓN	NOTACIÓN
<b>Grupo</b>	Es un artefacto que provee un mecanismo visual para agrupar elementos de un diagrama de manera informal.	
<b>Anotación</b>	Son mecanismos para que un modelador provea información adicional, al lector de un diagrama BPM.	
<b>Imagen</b>	Permite la inserción de imágenes almacenadas en su computador al diagrama.	
<b>Encabezado</b>	Muestra las propiedades del diagrama (autor, versión, descripción) y se actualiza igualmente de manera automáticamente con la información contenida allí. Para editar su información, basta con editar las propiedades del diagrama.	
<b>Artefactos personalizados</b>	Ayuda a definir y utilizar sus propios artefactos. Los artefactos proveen la capacidad de mostrar información adicional acerca del proceso, que no está directamente relacionada al flujo.	

**CONECTORES:**

ELEMENTO	DESCRIPCIÓN	NOTACIÓN
<b>Asociación</b>	Se utiliza para asociar información y artefactos con objetos de flujo. También, se utiliza para mostrar las tareas que compensan una actividad.	
<b>Flujo de secuencia</b>	Un flujo de secuencia es utilizado para mostrar el orden en el que las actividades se ejecutarán dentro del proceso.	
<b>Flujo de mensaje</b>	Se utiliza para mostrar el flujo de mensajes entre dos entidades que están preparadas para enviarlos y recibirlos.	

**1.3.3. Categorías de Procesos: Orquestación, Coreografía y Colaboración****1. ORQUESTACIÓN:**

Dentro de BPMN, los modelos de orquestación tienden a implicar una perspectiva de coordinación única; es decir, representan una vista del proceso específica del negocio o de la organización. Como tal, un proceso de orquestación describe como una única entidad de negocio, fluye entre los elementos.

El término técnico utilizado en BPMN es “Orquestación” para describir procesos en los que la lógica de procesos es explícita y cada instancia del proceso debe seguir alguna ruta en el modelo de proceso.

¿Cómo la lógica del proceso puede ser definida por adelantado si el resultado de una “aprobación” no se puede conocer de antemano?

La forma cómo el ejecutor decide aprobar o rechazar, no es parte de la lógica del proceso. Esto es parte de la lógica de la tarea de “aprobación”. BPMN no describe la lógica de la tarea, solo la lógica del proceso.

La lógica del proceso dice: Si el paso de “aprobación” termina en el estado *aprobado*, entonces sigue esta ruta, si termina en el estado *rechazado* entonces sigue esta otra ruta.

La ruta tomada por cualquier instancia del proceso depende de la información acumulada por la instancia dentro de su flujo.

Esta información incluye: mensajes recibidos, datos producidos en las actividades del proceso y estados finales de las actividades completadas.

BPMN asume que los datos de todas las instancias están disponibles para la lógica del proceso. Con esta información, el proceso sabe cómo fue cada paso completado y a qué paso debe ir.

El modelo del proceso guía de manera inteligente la instancia a través de cada paso. Este guiado a través de los pasos del proceso ayuda a disminuir la variabilidad de los procesos y permite su mejora.

El pool representa una instancia invisible que dirige y controla el flujo de actividades en el proceso. En la figura 10, podemos apreciar como el pool toma el rol de «**dirigente**», por ejemplo, cuando Roberto termina su actividad 1, el dirigente le asigna la actividad 2 a Falko y así sucesivamente.

El concepto se utiliza en analogía a una orquestra de música, en la cual el dirigente guía y controla a cada uno de los músicos, razón por la cual en BPMN también se habla de «**orquestación**».

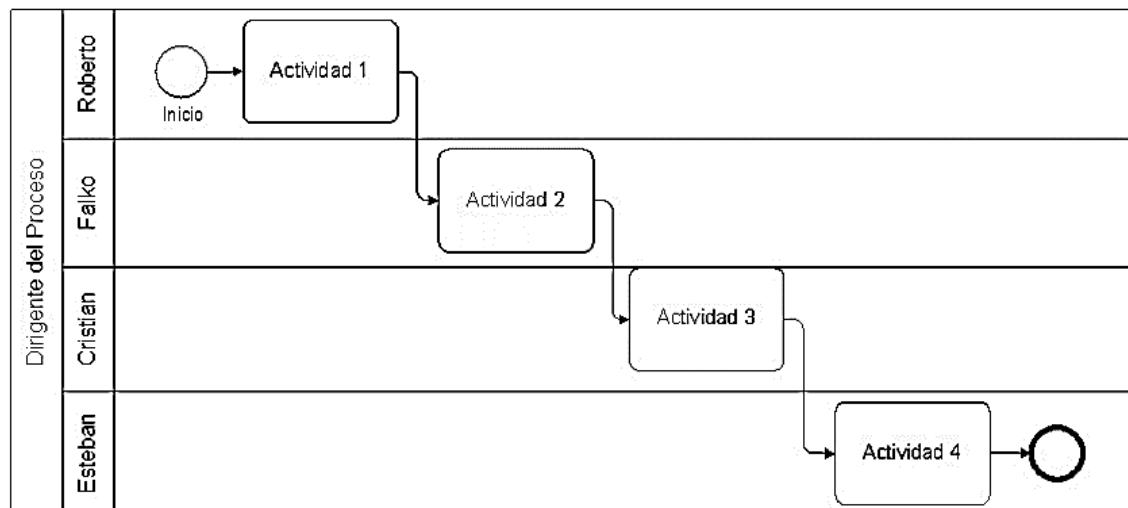


Figura 10: Orquestación en BPMN  
Fuente.- Freund et.al. - BPMN 2.0

Si un pool o dirigente no tiene control sobre un participante, entonces si tiene que obligadamente separarlo y representarlo como un pool propio, por ejemplo **clientes y proveedores**.

A continuación, se muestra un flujo en BPMN (Business Process Management Notation), que expresa la orquestación de servicios para obtener un servicio que realice el cierre de una compra y programe el envío en un sistema de despacho.

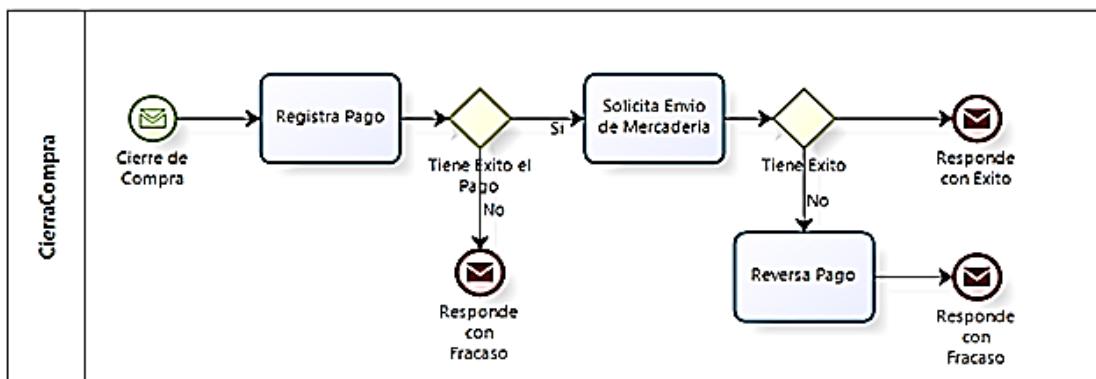


Figura 11: Ejemplo de Orquestación

Elaboración propia

## 2. COREOGRAFÍA:

Una coreografía es una definición del comportamiento esperado entre participantes que interactúan entre sí (una especie de protocolos o contrato procedural). En un formato de diagrama de flujo, define la secuencia de interacciones entre dos o más participantes.

Para realizar la coreografía en diagramas de BPMN, es necesario interactuar entre 2 o más pools (piscinas).

Una coreografía comparte muchas de las características de una orquestación en el sentido que aparece un proceso (es decir, un diagrama de flujo) y que incluye caminos alternativos y paralelos así como también Sub-Procesos.

Un modelo de proceso coreografía es una definición de la conducta que se espera entre los participantes interactúan entre sí.

Describe la interacción de los participantes o la secuencia de interacciones entre dos o más participantes. En BPMN, las interacciones son la comunicación, en la forma de intercambio de mensajes entre dos participantes.

Los diagramas de coreografía son mucho más compactos, porque se concentran solo en el intercambio de información entre los participantes y su debida lógica. El participante que envía información, lleva en el cuadro de coreografía un fondo «blanco» y el participante que recibe y procesa la información con un fondo «gris». No existe una regla en el objeto de coreografía con respecto al orden «arriba o abajo» del participante que envía o recibe; pueden posicionarse como mejor le parezca al modelador. En un diagrama de coreografía, también pueden declararse los objetos como subprocesos.

En este caso el subproceso, contiene su propia lógica de coreografía.

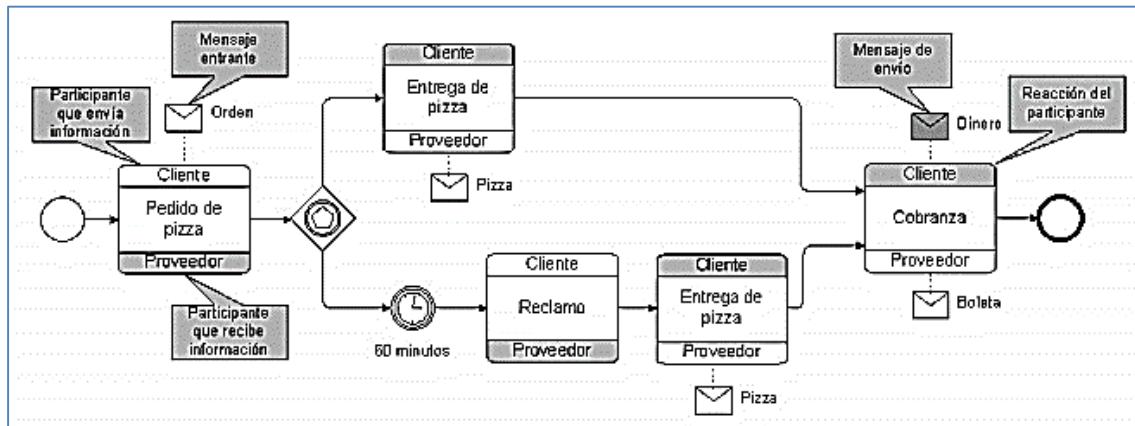


Figura 12: Diagrama de Coreografía  
Fuente.- Freund et.al. - BPMN 2.0

A continuación, se muestra el mismo *Diagrama de Coreografía* “Pedido de Pizza”; pero en su equivalente semántico *Diagrama de Colaboración*:

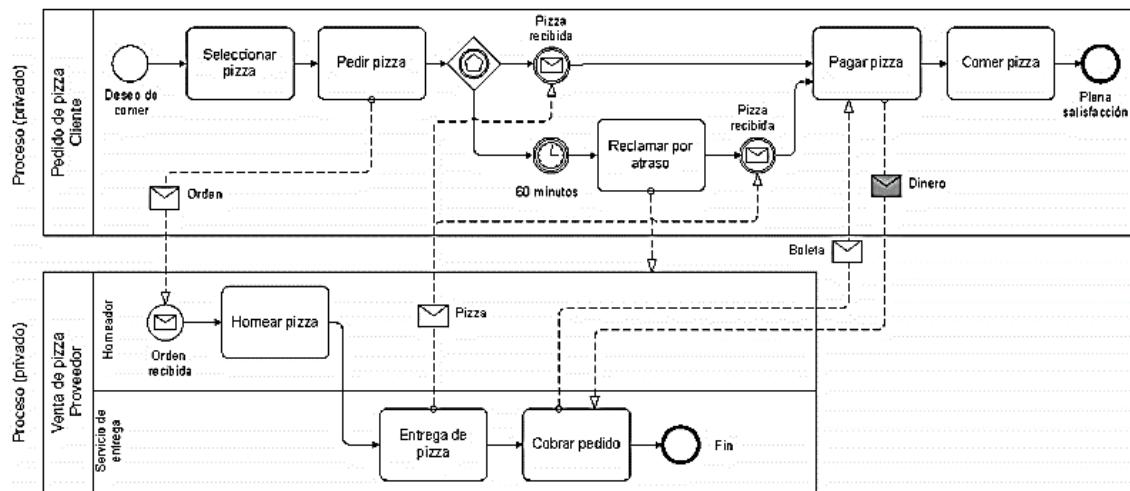


Figura 13: Diagrama de Colaboración  
Fuente.- Freund et.al. - BPMN 2.0

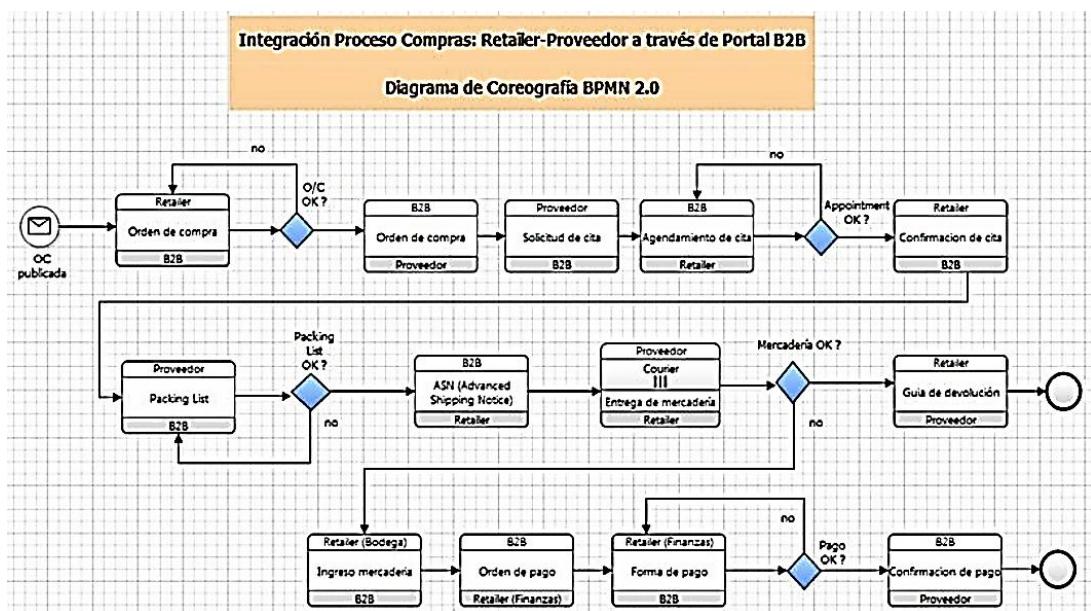


Figura 14: Modelo de Coreografía

Fuente.- Freund et.al. - BPMN 2.0

### 3. COLABORACIÓN:

Un diagrama de colaboración representa la interacción entre dos o más procesos.

Usualmente, contienen dos o más pools que representan los participantes en la colaboración. Tomemos, como ejemplo, los procesos paralelos que llevan a cabo una compañía y sus proveedores cuando se gestiona una compra.

Cada uno lleva procesos independientes, sin embargo, estos interactúan constantemente a través de intercambio de información (llamadas, correos electrónicos, etc.) y ninguno de los procesos finalizará exitosamente sin la información suministrada por el otro.

El siguiente diagrama representa esta situación:

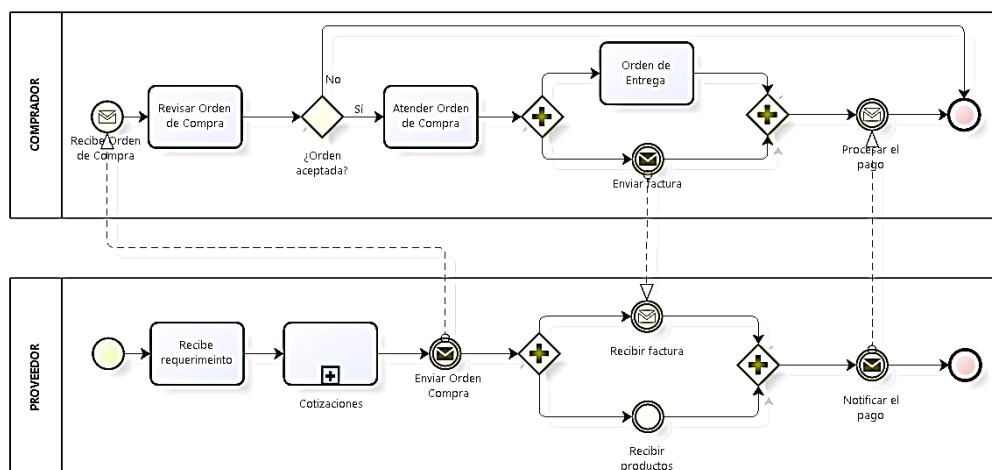


Figura 15: Diagrama de Colaboración

Fuente.- Bizagi BPM Suite: Guía del Usuario

Podemos ver que hay dos participantes involucrados representados por los pools. Uno es la compañía; y el otro, el proveedor.

El proceso es iniciado por la compañía que recibe un requerimiento de compra por parte de algún departamento. Una vez aceptado el requerimiento se inicia un sub-



proceso de *Cotizaciones* . Este sub-proceso gestiona las actividades necesarias para recibir y evaluar cotizaciones del producto solicitado para finalmente seleccionar a un proveedor. Una vez seleccionado el proveedor, se le envía una *orden de compra*, esto se representa en el diagrama mediante un *evento de mensaje*



En los diagramas colaborativos, el flujo de información entre procesos se representa mediante flujo de mensajes. El evento de mensaje activa el mensaje y la línea punteada que se observa salir de él en el diagrama es una línea de flujo de mensaje.

Esta línea conecta dos eventos de mensaje para relacionarlos entre sí. Podemos ver que el evento de mensaje de *Enviar Orden de Compra* está asociado al evento de



inicio de mensaje de *Recibir Orden de Compra*

Este último evento, iniciará una instancia de proceso para el proveedor una vez se reciba la orden de compra.

El proveedor inicia un flujo para procesar la orden del cliente, envía los productos de la orden y la factura de la misma. Esto se representa a través del evento de mensaje de



A su vez, la compañía está a la espera de la factura y la recepción del pedido. El



evento de mensaje de *Recibir Factura* espera a que llegue la factura,



mientras que el evento sin especificar *Recibir Productos*  se habilita para que sea activado manualmente una vez se reciba la orden. Dichos eventos, se habilitan en

paralelo con el uso de una compuerta paralela



Para garantizar que el flujo de proceso de la compañía no continúe hasta que se



reciba la factura y los productos de la orden, se usa compuerta paralela para sincronizar los flujos activos.

Finalmente, a través de una tarea de servicio se procesa el pago al proveedor y se envía una notificación de la realización del pago, de nuevo utilizando *eventos* y *flujos de mensaje*.

Una vez esta es recibida por el proveedor ambos procesos finalizan.

**(Bizagi BPM Suite: Guía del Usuario)**

### ¿Orquestación o Coreografía?

1. La Orquestación es donde un elemento central o maestro controla todos los aspectos del proceso.
2. La Coreografía es donde cada elemento del proceso es autónomo y controla su propia agenda.

Piense en la ejecución de un ballet. La música es Orquestada. El conductor proporciona el control maestro sobre la orquesta. Marca el ritmo y le indica a cada instrumento sobre cuando unírse. El baile es Coreografiado. Cada bailarín sabe su rol individual, y lo ejecuta en respuesta a cada uno de los otros bailarines y en el tiempo asignado por la música. El ballet emerge de la acción general de todos los bailarines, más que de algún control central. También, note que una Empresa dada, también puede usar ambos enfoques, donde cada uno es más adecuado.

La orquestación es el enfoque más comúnmente usado en la composición de servicios y procesos de negocio. Con la orquestación, se define la secuencia de pasos en un proceso, incluyendo condiciones excepcionales, y luego crear un controlador central para implementar la secuencia. En una SOA, los pasos individuales de la secuencia son implementados por operaciones sobre servicios. La secuencia puede ser implementada por una variedad de técnicas diferentes.

Frecuentemente, de forma relativa las composiciones de servicios simples son orquestados en código, tales como Java o C#, que reside dentro del servicio compuesto. Pero para operaciones más complejas, frecuentemente usará una herramienta para crear un modelo visual de la secuencia, y luego para generar el código que ejecute la secuencia, típicamente en un entorno de ejecución dedicado. Esto es el enfoque BPM típico que es bien conocido.

Los estándares para orquestación de hoy, incluyen BPMN (Business Process Modeling Notation) para definir la representación visual de la secuencia, y BPEL (Business Process Execution Language) como el código que ejecuta la secuencia.

Casi todas las infraestructuras de SOA proporcionan algún tipo de motor de ejecución de BPEL, y la mayoría de los productos BPM ya soportan, o están en proceso de soportar estos estándares en su modelado y ejecución. Además, BPEL es amigable con SOA. BPEL es expresado en XML y definido por metadatos de Esquema XML que está estrechamente alineado con los estándares SOA.

BPEL mismo utiliza WSDL en dos niveles. Primero, los servicios web definidos por WSDL son usados para interactuar con las capacidades requeridas por el proceso. Segundo, cada proceso BPEL es por sí mismo un servicio web descrito usando WSDL.

En resumen, la orquestación:

1. Define un solo control maestro de todos los aspectos de un proceso (enfoque top-down). Soporta una vista gráfica de la secuencia.
2. Se mapea fácilmente a SOA.
3. Generalmente, es más simple con el cual iniciar; pero frecuentemente es más difícil de escalar a procesos más complejos.
4. Es manejado por el modelo de secuencia gráfica, por ejemplo, a las funciones le siguen formularios.
5. Representa el estado de la práctica, y es soportado por la mayoría de las herramientas.

De esta forma, la popularidad de este enfoque es comprensible. Pero no soporta todos los tipos de procesos igualmente bien.

La coreografía proporciona un enfoque diferente que está obteniendo aceptación en escenarios que tienen procesos complejos con algunas partes interactivas y sistemas basados en agentes y basados en eventos.

En un enfoque de coreografía, se crean las reglas que determinan el comportamiento de cada participante individual en el proceso.

El comportamiento general del proceso basado en la interacción de las piezas individuales, cada una de forma autónoma siguiendo sus propias reglas.

Si está familiarizado con el Procesamiento de Eventos Complejos (CEP), verá la similitud en el espacio del problema de CEP y un enfoque de coreografía, por ej.: cómo puede administrar la interacción de múltiples eventos independientes (o participantes) para producir un resultado de negocio predecible, general.

Actualmente, hay **dos enfoques** principales para coreografía, basado en mensaje, y basado en componente de trabajo.

El enfoque de mensaje está basado en examinar los mensajes entre participantes en un proceso general. Con este enfoque, define comportamientos exhaustivamente capturando los contratos de mensajes entre las partes colaboradoras.

Este es el mecanismo soportado por el estándar WS-CDL (Lenguaje de Definición de Coreografía de Web Services) y es utilizado frecuentemente para aplicaciones B2B. En aplicaciones B2B, las cuales son por definición cross empresariales, es difícil especificar la implementación de participantes específicos, y no hay autoridad central para el flujo general.

El enfoque de coreografía basado en mensaje es atractivo debido a que solo necesita especificar las definiciones de intercambio de mensajes (sintaxis, semántica y comportamiento).

Otro enfoque prominente se ha hecho con la configuración de componentes de trabajo de proceso. En este enfoque, define el comportamiento de componentes de trabajo individual le permite al comportamiento del proceso emergir como la instancia del proceso específico evolucione.

Por ejemplo, podría implementar comportamiento de enrutamiento en componentes de trabajo individuales con pocas reglas sencillas como qué capacidades de componentes de trabajo necesita para completar; qué roles pueden reunir cada necesidad; lo que causa una necesidad de convertirse en activa, y lo que causa una necesidad para ser considerada completa.

En un sistema sofisticado, estas reglas pueden ser especificadas en los metadatos del componente de trabajo y luego implementadas en un contenedor del componente de trabajo. Los lectores familiarizados con los sistemas basados en agentes deben reconocer la similitud aquí.

En resumen, en la coreografía:

1. El comportamiento del proceso general “emerge” del trabajo de sus partes (bottom up). No se requiere perspectiva global.
2. Los procesos de trabajo complejos son descompuestos en agendas de trabajo donde cada elemento autónomo controla su propia agenda.
3. Mapea fácilmente a sistemas basados en agentes y basados en agentes.
4. Es, generalmente, más difícil de iniciar, pero frecuentemente más fácil de escalar a procesos complejos.
5. Las representaciones gráficas pueden estar derivadas de los procesos, por ej.: los formularios siguen a las funciones.
6. Representa el estado del arte, y está obteniendo soporte con herramientas emergentes.

De esta forma, algunos lineamientos útiles más apropiados para un escenario dado son:

Usar orquestación:

1. Cuando se requieren producto fuera de la plataforma (ya que este es el enfoque implementado por la mayoría de los productos actuales)
2. Para servicios compuestos: donde las semánticas de transacciones pueden ser manejadas por compensación solas.
3. Para definiciones de procesos relativamente estáticos: donde es deseada una definición gráfica del proceso

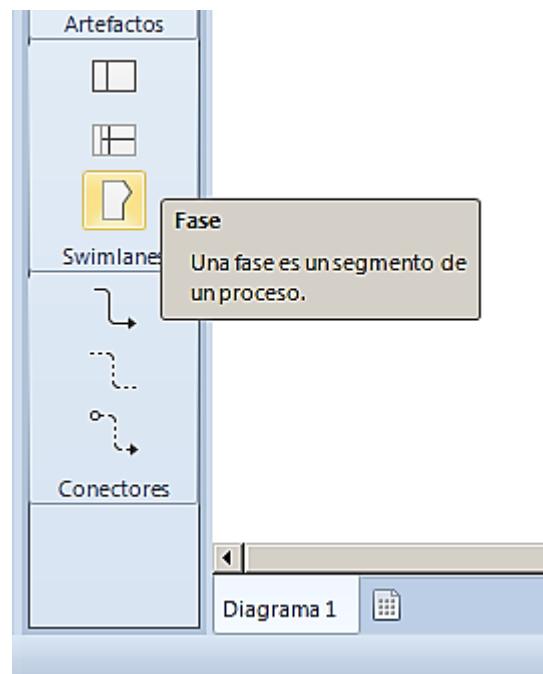
Usar coreografía:

1. Donde los procesos pueden escalar a un alto número de pasos por componente.
2. Donde se desea opacidad de detalles de proceso entre partners de procesos (como B2B).
3. Donde diferentes partners de procesos pueden requerir sus propias personalizaciones de procesos.
4. Donde los procesos son altamente dinámicos o de búsqueda de objetivos.

#### 1.3.4. Uso de Fases en un entorno CASE

Las fases (**Milestones**) son sub divisiones del proceso, las cuales son utilizadas como puntos de referencia. Ellas ayudan a los lectores a entender las diferentes etapas que conforman el proceso.

Los lanes en **Bizagi Process Modeler** se incluyen de manera horizontal; pero si desea una separación vertical, puede optar por usar las **fases**:

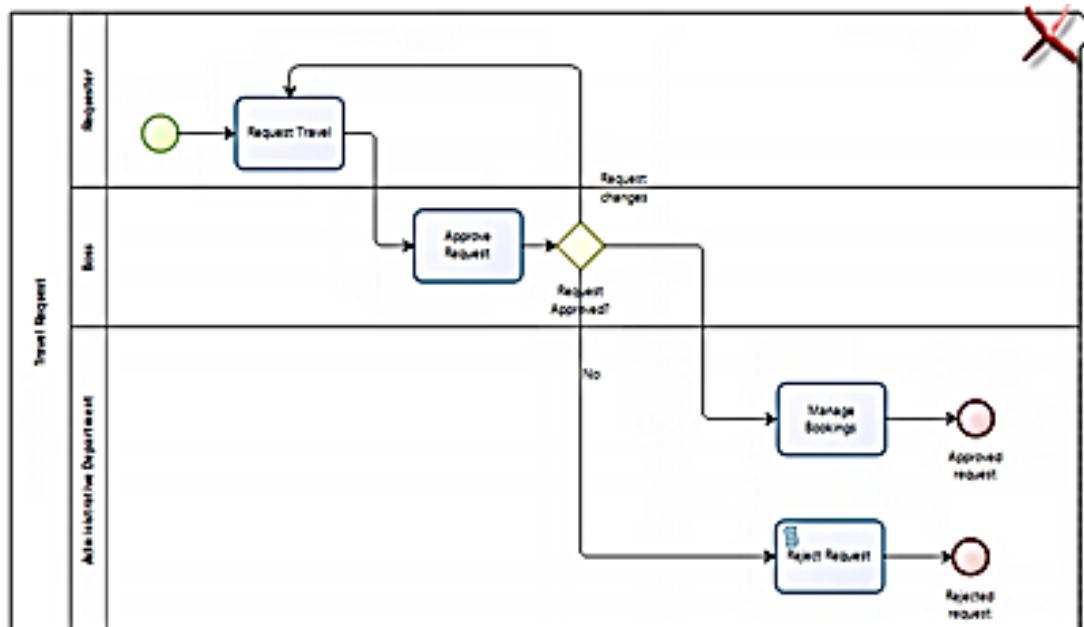


**Figura 16: Diagrama de Colaboración**  
Fuente.- elaboración propia

### Incluir fases (Milestones) es una buena práctica.

Siempre identifique y defina fases; estas representan un periodo de tiempo, objetivo o transición en el proceso.

Así no.



**Figura 17: Diagrama sin fases**  
Fuente.- Bizagi BPM Suite: Guía del Usuario

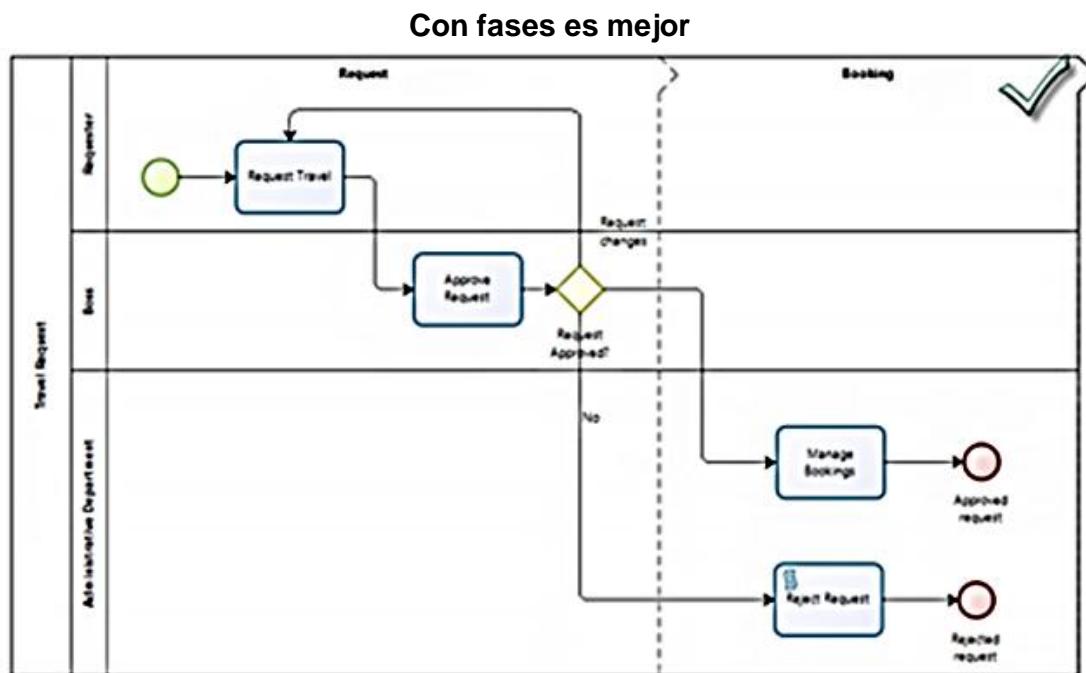


Figura 18: Diagrama con fases

Fuente.- Bizagi BPM Suite: Guía del Usuario

### ¿Cómo colocar las “Fases”?

- a) Para incluir una fase, arrastre el elemento desde la paleta y ubíquelo sobre el diagrama. Arrastre y suelte otras dos fases.

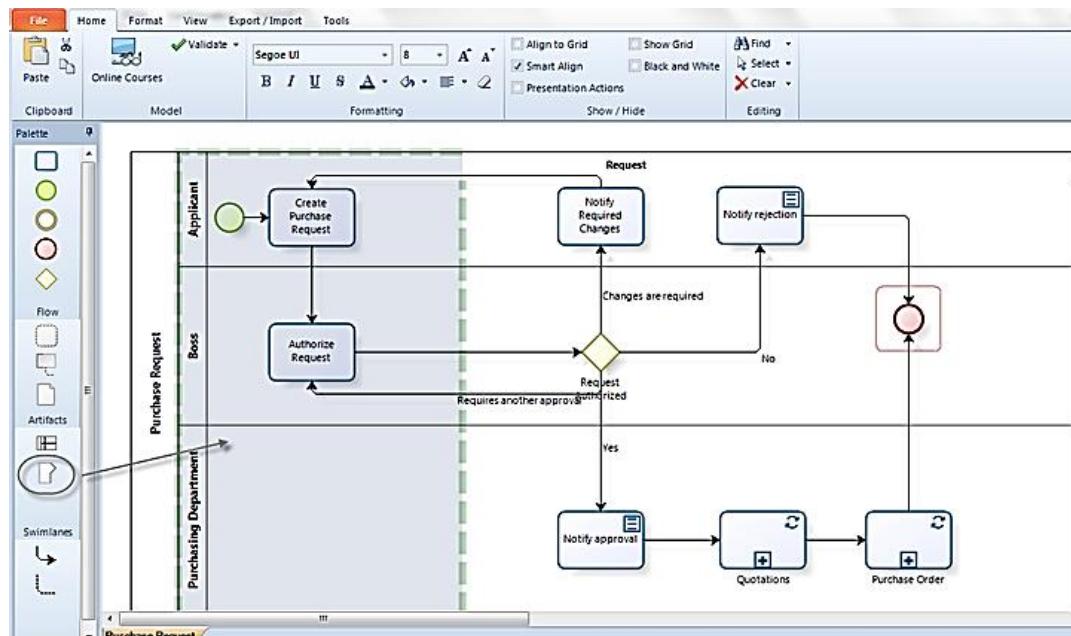


Figura 19: Inserción de fases

Fuente.- Bizagi BPM Suite: Guía del Usuario

b) Arrastra y suelte los elementos que pertenecen a la fase.

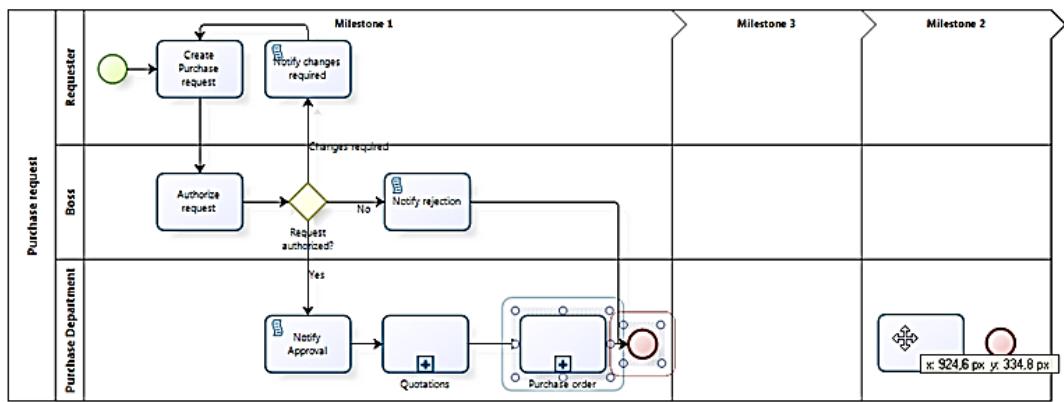


Figura 20: Inserción de fases 2  
Fuente.- Bizagi BPM Suite: Guía del Usuario

c) Para renombrar una fase dé doble clic sobre ella, presione la tecla **F2** o dé **click derecho** sobre ella y seleccione la opción editar texto del menú.

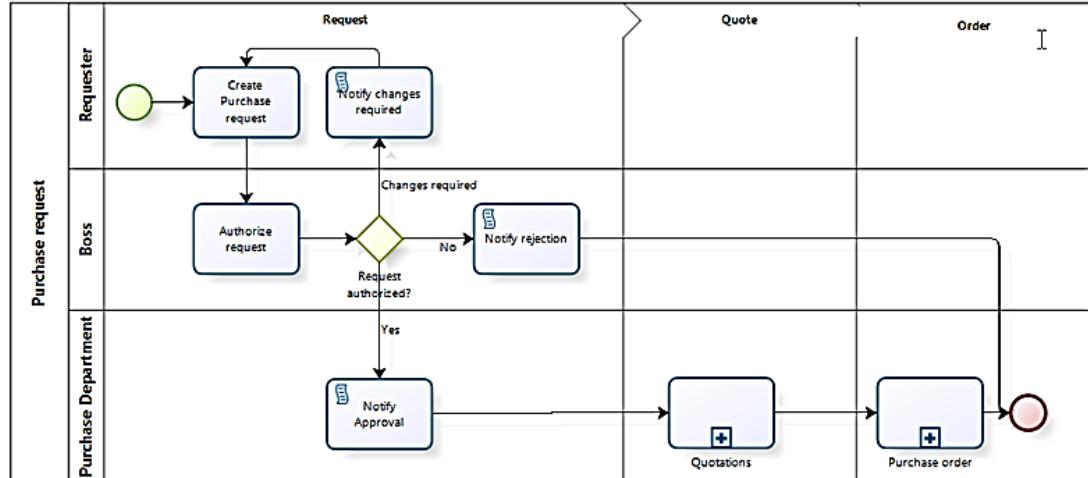


Figura 21: Inserción de fases 3  
Fuente.- Bizagi BPM Suite: Guía del Usuario

### 1.3.5. Elementos de Modelos descriptivos: Compuertas exclusivas y Compuertas inclusivas basadas en datos.

#### Compuertas:

Las compuertas se utilizan para controlar la divergencia y convergencia de flujos de secuencia. Determinan ramificaciones, bifurcaciones, combinaciones y uniones en el proceso. El término “**Compuerta**” implica que hay un mecanismo que permite o limita el paso a través de la misma.

ELEMENTO	DESCRIPCIÓN	NOTACIÓN
<b>Compuerta exclusiva</b>	De <b>divergencia</b> : se utiliza para crear caminos alternativos dentro del proceso, pero solo uno se selecciona. De <b>convergencia</b> : se utiliza para unir caminos alternativos.	

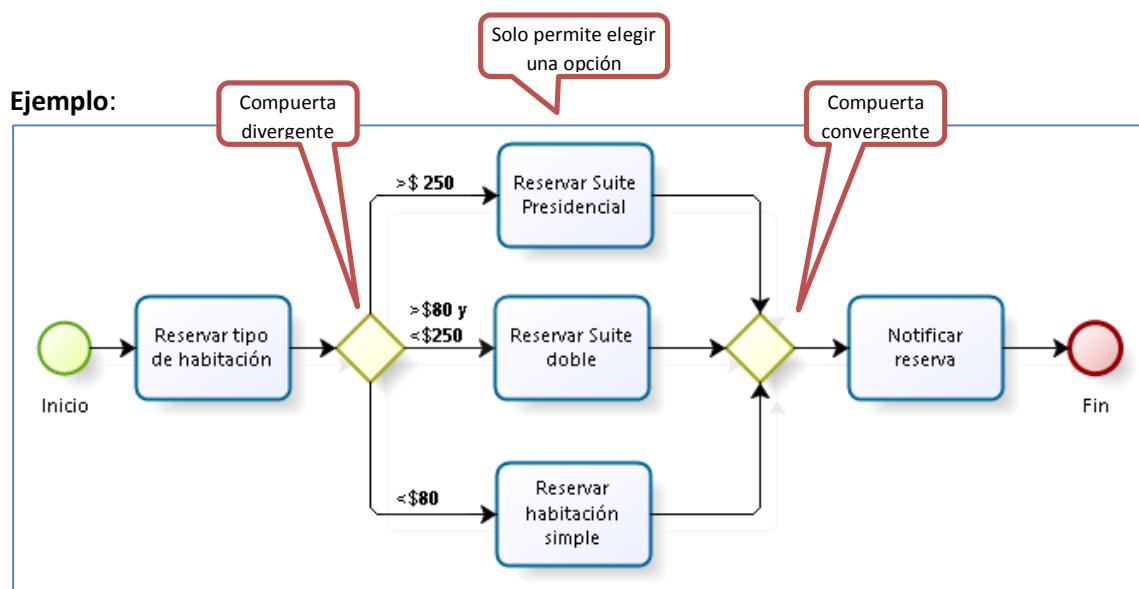


Figura 22: Ejemplo de Compuerta Exclusiva  
Fuente.- elaboración propia

ELEMENTO	DESCRIPCIÓN	NOTACIÓN
Compuerta Inclusiva	<p><b>De divergencia:</b> Representa un punto de ramificación en donde las alternativas se basan en expresiones condicionales.</p> <p>La evaluación VERDADERA de una Condición no excluye la evaluación de las demás condiciones. Todas las evaluaciones VERDADERAS serán atravesadas por un token.</p> <p><b>De convergencia:</b> Se utiliza para unir una combinación de caminos paralelos alternativos.</p>	

**Ejemplo:**

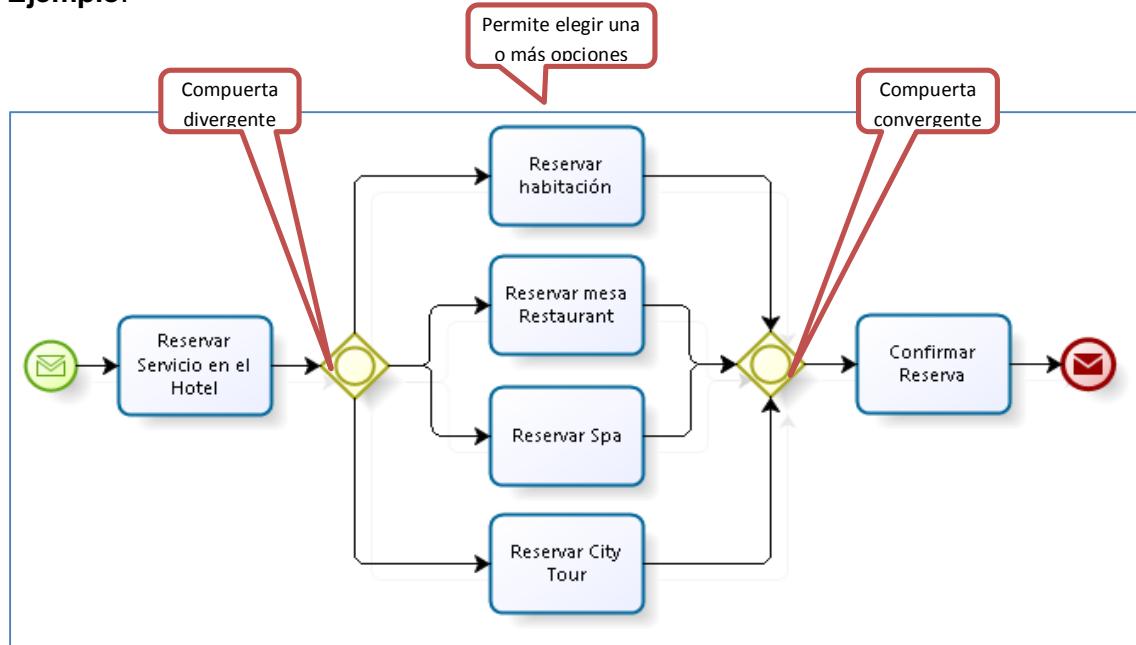


Figura 23: Ejemplo de Compuerta Inclusiva  
Fuente.- elaboración propia

### 1.3.6. Desarrollo de aplicaciones.

#### Caso 1: Requerimiento Organismo de Control

El proceso inicia cuando al Director del área le llega una carta del organismo de Control con un requerimiento de información específico. El director del área revisa el requerimiento y dependiendo de tema solicitado asigna la persona que lo va a responder y le señala la fecha de vencimiento y el tiempo correspondiente que tiene esta persona para dar respuesta.

La persona asignada recibe el requerimiento y verifica el tiempo que tiene disponible para realizar el informe de respuesta, revisa todo el caso, recopila la información necesaria y elabora la respuesta del requerimiento. Una vez realizada la respuesta se lo envía al director para su revisión.

El director recibe el informe de respuesta, lo revisa, cambia lo que considere pertinente y decide si esa es la respuesta definitiva ante el organismo de control, ya que si lo considera necesario, la persona que elaboró la respuesta debe revisar de nuevo el caso y completar y corregir la respuesta, de esta forma indica que la respuesta no es la definitiva, solicita correcciones y le envía de nuevo el requerimiento al empleado para su corrección.

Si el Director, está satisfecho con la respuesta, genera la carta con la respuesta, la imprime y la envía al Organismo de Control.

Realice el Flujo del Proceso utilizando la notación de BPMN.

### Solución:

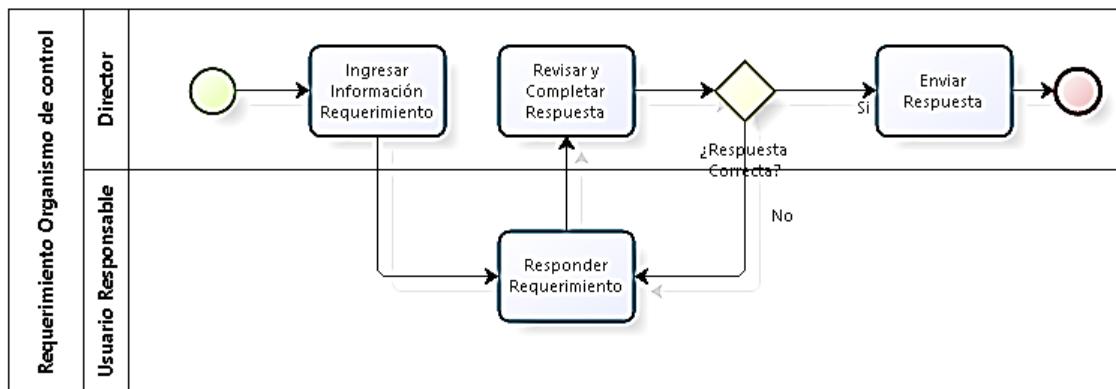


Figura 24: Solución Caso: Requerimiento Organismo Control

Fuente.- Bizagi Elearning – Modelamiento de Procesos

### Caso 2: Proceso de Cotización de Planes Turísticos

Este proceso gestiona las solicitudes de cotización de planes turísticos que los clientes hacen a una agencia de viajes. Cuando un cliente, realiza una solicitud, es necesario que el agente de viajes determine los costos y disponibilidad de cada uno de los servicios que el cliente incluyó en su solicitud (Tiquete, hotel, alquiler de auto).

A continuación, se procede a consolidar un plan turístico que se envía al cliente junto con el valor del mismo. Si el cliente, está interesado en el plan se inicia una gestión de ventas, de lo contrario el proceso finaliza.

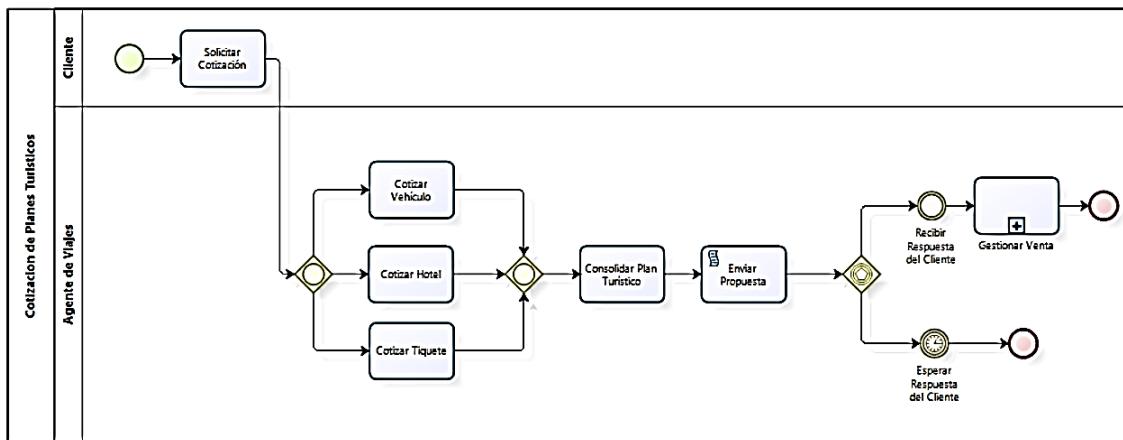


Figura 25: Solución Caso: Requerimiento Organismo Control

Fuente.- Bizagi BPM Suite: Guía del Usuario

El proceso inicia con la actividad **Solicitar Cotización**, donde el cliente manifiesta su interés en un plan turístico y describe sus requerimientos. El cliente puede solicitar planes que incluyan alquiler de vehículo, hotel y tiquetes. De acuerdo a los servicios que escoja, se deberán hacer las gestiones necesarias para determinar disponibilidad y costo de los mismos.



La **compuerta inclusiva** permite que solo se habiliten las actividades de gestión relacionadas a los servicios que el cliente desea, es decir que los caminos no son excluyentes y se puede seguir uno o más de los 3 disponibles.

Para garantizar que el proceso no continúe hasta que las gestiones de los servicios solicitados no hayan sido finalizadas, se utiliza la misma compuerta inclusiva como compuerta de convergencia o sincronización. Esto quiere decir que la compuerta permitirá que el flujo continúe únicamente cuando todos los caminos activos lleguen a ella.

Una vez gestionados los servicios solicitados, el agente de viajes deberá consolidar un plan turístico basado en la disponibilidad y costos de los mismos. Posteriormente, la propuesta se enviará al cliente vía correo electrónico, esto se modela en BPMN



mediante una tarea la **tarea de Script**. Este tipo de tarea permite el ingreso de un script en un lenguaje que el motor de procesos de negocio pueda ejecutar.



A continuación, encontramos una **Compuerta basada en eventos**. Esta compuerta representa un punto de decisión en el proceso, pero dicha decisión no depende de datos del proceso sino de eventos que ocurran.



En este caso, pueden ocurrir dos eventos: el **evento sin especificar** de **Recibir respuesta del cliente**, que se ejecuta de forma manual por el agente de viajes una vez ha recibido una respuesta positiva del cliente, o el **evento de temporización**



**Esperar respuesta**, que espera un tiempo específico por la respuesta del cliente. El primer evento que ocurra habilitará el flujo relacionado a él y el otro será deshabilitado. Esto quiere decir que si el cliente no envía una respuesta a la cotización después de un tiempo determinado, el proceso finalizará al encontrar un **evento de fin**



En caso de que el cliente dé una respuesta positiva a la propuesta enviada, se deberá iniciar una gestión de ventas representada en el diagrama por un **Subprocesos**



Un sub-proceso es una actividad compuesta de un conjunto de tareas.

Es compuesto dado que esta figura incluye a su vez un conjunto de actividades y una secuencia lógica (proceso) que indica que dicha actividad puede ser analizada en más detalle. En este caso, este sub-proceso representa todas las actividades que el área de ventas deberá realizar para entregar el servicio al cliente y facturarlo.



Finalmente, el flujo termina al encontrar el **evento de fin**

# Resumen

1. La OMG establece tres niveles de modelado que se describen a continuación:
  1. Nivel Descriptivo
  2. Nivel Analítico
  3. Nivel Ejecutable
4. El modelado a nivel descriptivo usado para comunicar procesos a través de la organización. Esta descripción de alto nivel, describe los flujos esenciales del proceso de negocio ignorando complejidades e incluso las reglas de Negocio.
5. Los eventos se clasifican en: Eventos de inicio, intermedios y de finalización.
6. Los artefactos y elementos de legibilidad complementan los modelos construidos con la herramienta Bizagi Modeler.
7. Existen tres tipos de conectores: Flujo de mensaje: Flujo de secuencia y Asociación
8. Las tres Categorías de Procesos son: Orquestación, Coreografía y Colaboración.
9. El uso de las fases o segmentos de procesos aclaran la lógica de desarrollo y facilitan el modelamiento.

Pueden revisar los siguientes enlaces para ampliar los conceptos vistos en esta unidad:

- <http://www.omg.org/spec/BPMN/2.0/PDF/>
- <https://books.google.com.pe/books?id=07NJBAAAQBAJ&pg=PA217&lpg=PA217&dq=nivel+descriptivo,+nivel+anal%C3%ADtico+y+nivel+ejecutable&source=bl&ots=E-5IMDkpnk&sig=Xm1Ojj-ANevLWI0LViReLgwggY&hl=es&sa=X&ved=0CDAQ6AEwA2oVChMlo6n0yrfqxgIVUHySCh1a5gxX#v=onepage&q=nivel%20descriptivo%2C%20nivel%20anal%C3%ADtico%20y%20nivel%20ejecutable&f=false>
- <http://www.bpmamericas.org/acerca-de-bpmn-2-0/que-es-el-estandar-bpmn-2-0/>



# MODELADO ANALÍTICO

---

## LOGRO DE LA UNIDAD DE APRENDIZAJE

Al término de la unidad, el alumno usa la notación básica BPMN para construir Modelos a nivel analítico, utilizando la mayor parte de la expresión semántica del BPMN.

## TEMARIO

### 2.1 Tema 4 : Modelado a nivel Analítico

- 2.1.1 : Compuertas paralelas
- 2.1.2 : Compuertas basadas en eventos
- 2.1.3 : Compuertas complejas
- 2.1.4 : Técnicas de simplificación de procesos
- 2.1.5 : Metodología de diseño top-down y modelado BPMN
- 2.1.6 : Identificación de Subprocesos: Reusables y embebidos
- 2.1.7 : Uso de Temporizadores, manejo de errores.
- 2.1.8 : Transacciones: envío y captura de compensaciones

## 2.1 MODELADO A NIVEL ANALÍTICO

### 2.1.1 Compuertas paralelas.

ELEMENTO	DESCRIPCIÓN	NOTACIÓN
Compuerta paralela	De <b>divergencia</b> : Se utiliza para crear caminos alternativos dentro del proceso, pero solo uno se selecciona.  De <b>convergencia</b> : Se utiliza para unir caminos alternativos.	

Ejemplo:

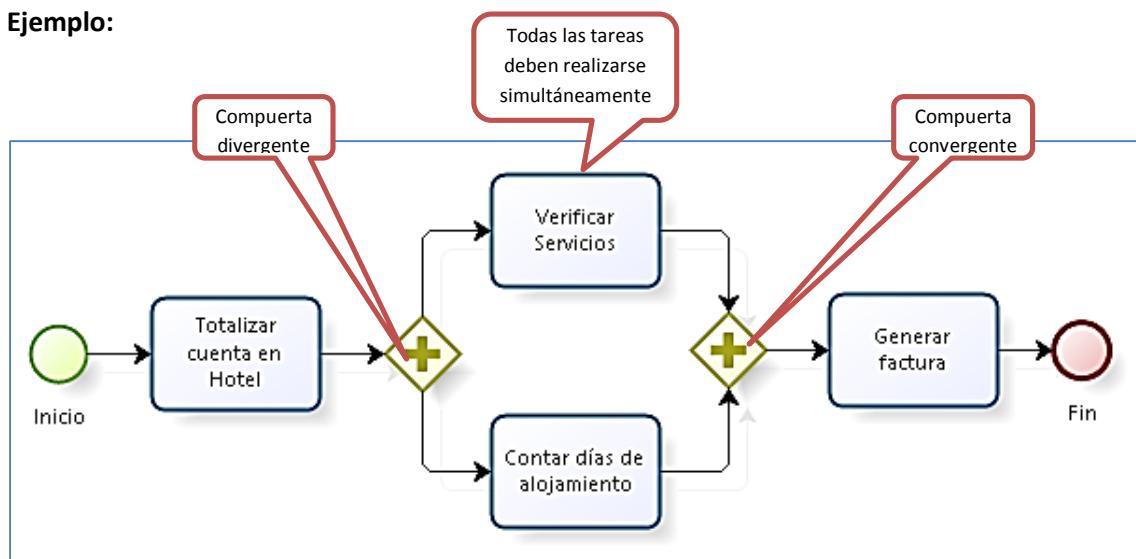


Figura 26: Ejemplo de Compuerta Paralela  
Fuente.- elaboración propia

### 2.1.2 Compuertas basadas en Eventos.

ELEMENTO	DESCRIPCIÓN	NOTACIÓN
Compuerta basada en Eventos	Representa un punto de ramificación en los procesos donde los caminos alternativos que siguen la compuerta están basados en eventos que ocurren.  Cuando el primer evento se dispara, el camino que sigue a ese evento se usará. Los caminos restantes serán deshabilitados.	

<b>Compuerta exclusiva basada en Eventos</b>	<p>Es una variación de la compuerta basada en eventos que se utiliza únicamente para instanciar procesos.</p> <p>Si uno de los eventos de la configuración de la compuerta ocurre, se crea una nueva instancia del proceso.</p> <p>No deben tener flujos de entrada.</p>	
<b>Compuerta paralela basada en Eventos</b>	<p>A diferencia de la Compuerta Exclusiva Basada en Eventos, se crea una instancia del proceso una vez que TODOS los eventos de la configuración de la compuerta ocurren.</p> <p>No deben tener flujos de entrada.</p>	

#### Ejemplo Compuerta basada en Eventos:

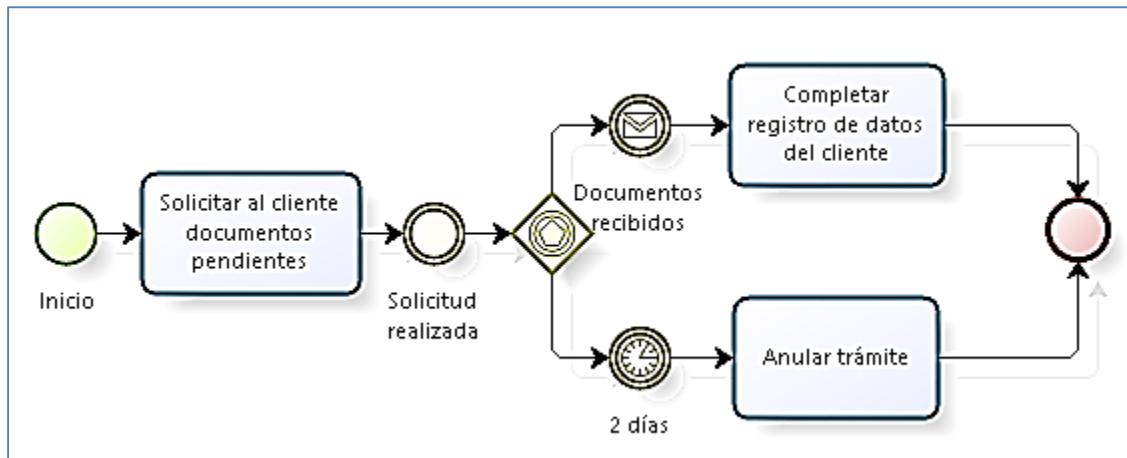


Figura 27: Ejemplo de Compuerta Basada en Eventos

Fuente.- elaboración propia

#### Ejemplo Compuerta exclusiva basada en Eventos:

En la solicitud de crédito, se asume que la entrega por parte del cliente debe hacerse en los próximos días luego de registrar los productos. Si ésta no ocurre dentro de ese tiempo, se deshabilita el "Evento Intermedio" de *receptar documentos* y se procede a *contactar al cliente*. Si los documentos, son entregados por el cliente dentro del tiempo esperado, se revisan los documentos y el tiempo deja de correr.

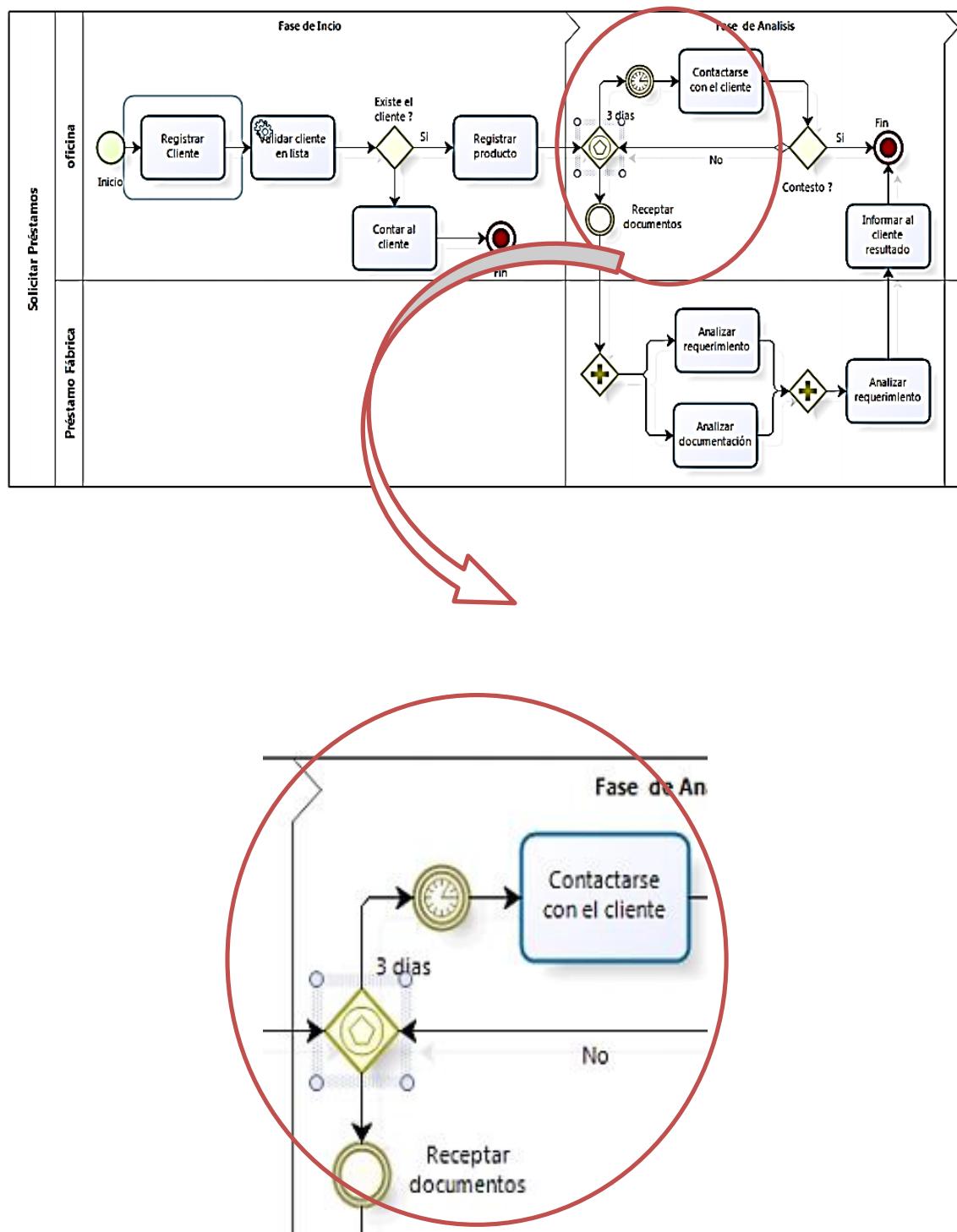


Figura 28: Ejemplo de Compuerta Exclusiva basada en Eventos  
Fuente.- elaboración propia

### Ejemplo Compuerta Paralela basada en Eventos:

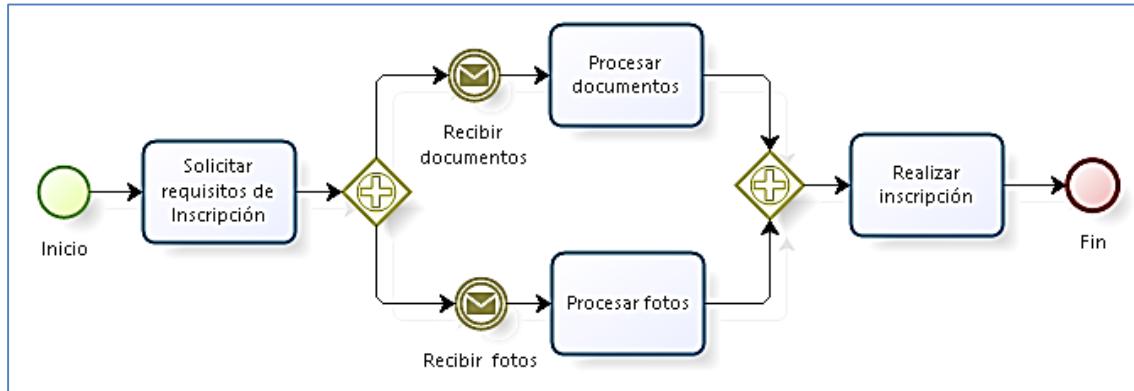


Figura 29: Ejemplo de Compuerta Paralela basada en Eventos

Fuente.- elaboración propia

### 2.1.3 Compuerta compleja.

ELEMENTO	DESCRIPCIÓN	NOTACIÓN
Compuerta compleja	<p><b>De divergencia:</b> Se utiliza para controlar puntos de decisión complejos en los procesos. Crea caminos alternativos dentro del proceso utilizando expresiones.</p> <p><b>De convergencia:</b> Permite continuar al siguiente punto del proceso cuando una condición de negocio se cumple.</p>	

### Ejemplo Compuerta Compleja:

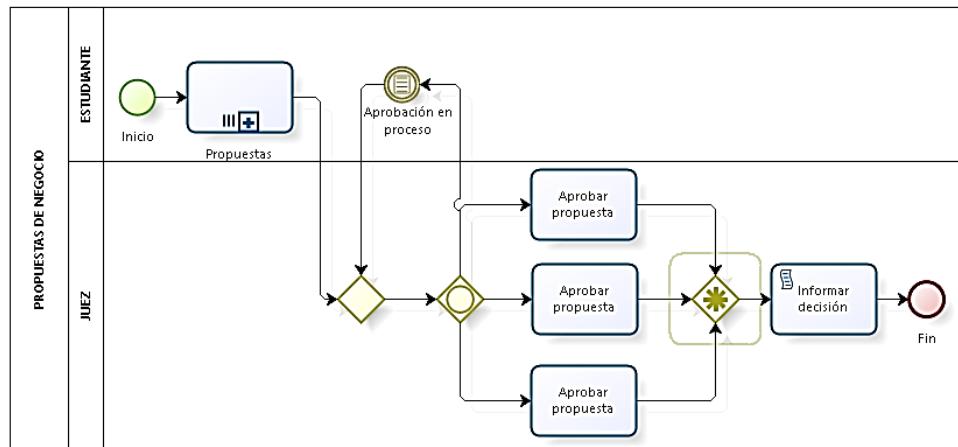


Figura 30: Ejemplo de Compuerta Compleja

Fuente.- Bizagi Process Modeler: Patrones de Modelado de Procesos

En el ejemplo anterior, una compañía decide invertir en propuestas de negocios. Estas propuestas son enviadas por estudiantes de varias universidades, cada una de ellas es evaluada de forma individual por 3 jueces. Si 2 de los 3 jueces dan su aprobación, el aspirante será informado acerca de la decisión.

Cuando el último de los jueces decida sobre la propuesta, será posible empezar a estudiar una nueva.

El patrón utiliza una compuerta Paralela o una compuerta Inclusiva para dividir la rama entrante. Una compuerta compleja es utilizada para validar el número de aprobaciones (N) necesarias para informar sobre la decisión.

Para bloquear las ramas entrantes hasta que la unión sea reiniciada, se utiliza un Evento Condicional.

El evento evalúa si hay alguna aprobación en progreso, y bloqueará la entrada de nuevas propuestas hasta que todos los jueces hayan dado su veredicto de la propuesta en curso.

#### 2.1.4 Técnicas de simplificación de procesos.

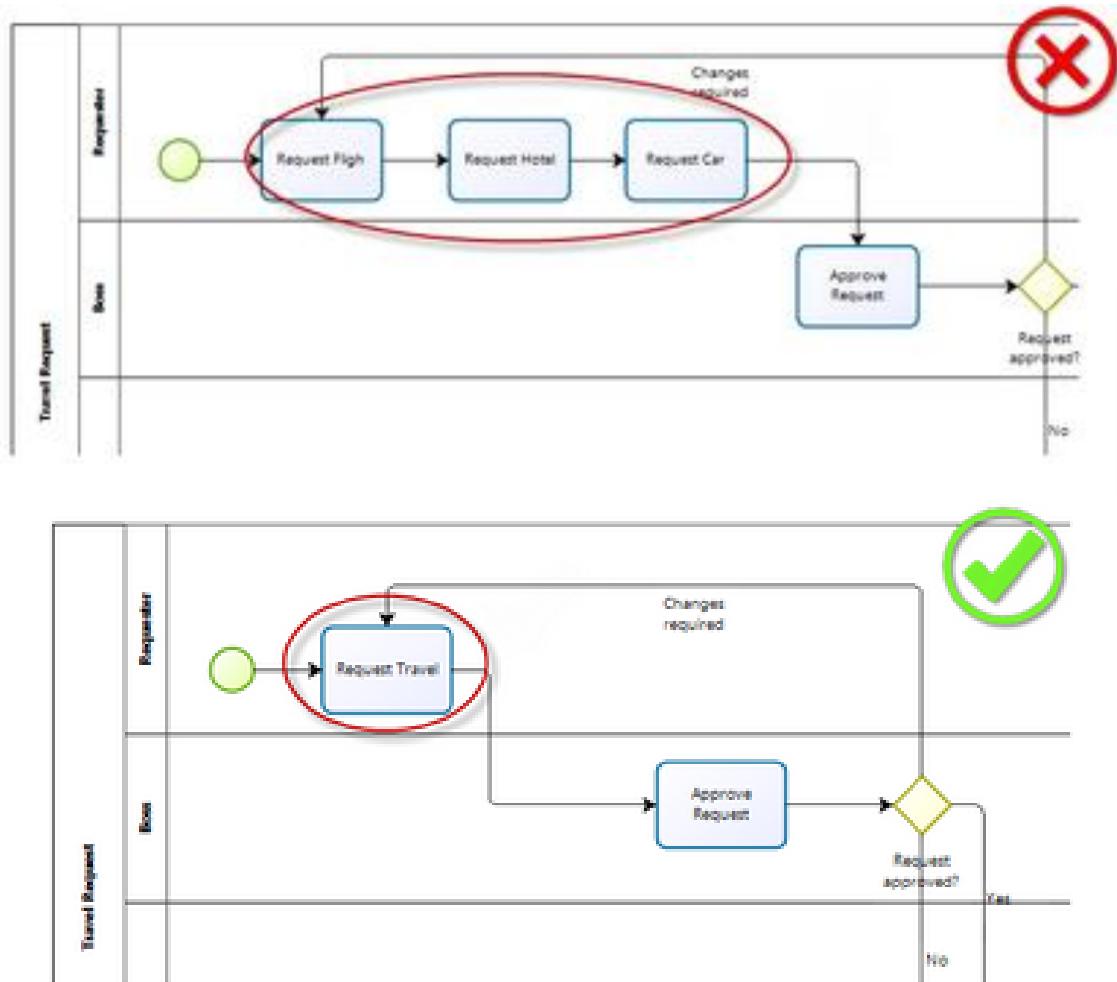
Los diagramas grandes no permiten dar una perspectiva de extremo a extremo para los lectores. Son difíciles de leer y el propósito del proceso es difícil de comunicar.

Es fundamental definir el alcance correcto de las tareas y el nivel de detalle de los procesos para reducir el exceso de información. Los siguientes consejos le ayudarán:

a) *Reduzca el número de tareas redundantes*

El nivel de detalle en un proceso a veces es un verdadero reto. En muchos casos, usted puede enfrentar dificultades para definir el alcance de una sola tarea. Tome en cuenta que:

1. Cuando diagrame, es útil imaginar que usted es el usuario final. Si un conjunto de actividades consecutivas las puede realizar la misma persona, al mismo tiempo, entonces estas actividades podrían integrarse en una sola actividad.
2. Un conjunto de actividades consecutivas en el mismo Lane puede indicar falta de un participante, demasiado detalle, o un desajuste el alcance de las tareas. Revise estos patrones para identificar oportunidades de la integración de la actividad.



**Figura 31: Simplificación de tareas**  
Fuente.- Bizagi BPM Suite: Guía del Usuario

### b) Agrupe las actividades

Utilice subprocesos para agrupar actividades con el mismo propósito. Después, puede ampliar los subprocesos para exponer los detalles de los niveles inferiores de la jerarquía. Un proceso contendrá varias páginas, pero internamente, se mantiene la integridad de un modelo único.

Utilice *subprocesos embebidos* cuando:

1. Un conjunto de actividades consecutivas tiene un propietario diferente del propietario del proceso principal (por ejemplo, un proceso de *Solicitud de Compra* se realiza el área de *Compras* y el proceso de *Cuentas por Pagar* se realiza el área *Financiera*).

2. Un conjunto de actividades consecutivas tiene un objetivo diferente al del proceso principal (por ejemplo, Solicitud de Crédito se centra en la gestión de todas las actividades para aprobar una solicitud de crédito y Verificar la información del solicitante se centra en comprobar si el solicitante está en lista negra, así como la información presentada).

Utilice *subprocesos reutilizables* cuando:

- El subproceso debe ser invocado desde diferentes procesos (por ejemplo, el subproceso *Verificar la información del solicitante* se puede invocar desde el proceso de *Solicitud de Crédito* o desde *Solicitud de Seguros*).

c) Aplique patrones de procesos

No reinvente la rueda. Los expertos de BPMN han trabajado en la definición de patrones de modelado para diferentes situaciones de negocios. Úselos para modelar las condiciones de negocio requeridas al tiempo que simplifica sus diagramas.

Para más información sobre los patrones de modelado, por favor, revise la Unidad 3: Patrones de Modelado.

d) No utilice antipatrones

Mientras simplificar sus modelos tenga en cuenta no utilizar Antipatrones. Estos son patrones de modelado que se usan generalmente en la automatización, pero son malas prácticas que no son recomendados por Bizagi

e) Documente los detalles menores

Deje los detalles a la documentación. No incluya toda la información en diagramas. La información adicional debe ser documentada como propiedades de forma, no como objetos o texto en el diagrama.

### 2.1.5 Metodología de diseño top-down y modelado BPMN.

Esta propuesta metodológica pretende simplificar, a través de una serie de pasos, plasmar un **proceso de negocio en la notación BPMN 2.0**.

Los pasos son:

**Paso 1:** Definir el alcance del proceso: Se debe entender lo siguiente:

El modelado jerárquico inicia con un acuerdo del alcance del proceso, donde comienza y donde termina. Este acuerdo debe darse antes de iniciar el modelado.

Esto no es algo simple de obtener y puede tomar algún tiempo y algunas discusiones, pero es mejor tener estas discusiones antes de entrar en los detalles del proceso.

Algunas preguntas que nos ayudan son:

- a. ¿Cómo empieza el proceso?**
- b. ¿Qué es lo que representa cada instancia del proceso, quienes son los participantes del proceso?**
- c. ¿Qué significa el final del proceso?**
- d. ¿Ha diferentes formas en que el proceso pueda terminar?**

**Paso 2:** Definir el diagrama BPMN de nivel más alto para el camino ideal: Normalmente los procesos siguen un flujo que va variando conforme las decisiones o los eventos que van ocurriendo, por lo que se piensa inicialmente en dibujar el curso de eventos ideal para el proceso. Se realiza mediante los siguientes pasos:

- a. Adicionando los “Pools”:** Los pools encierran las sucesiones de eventos y actividades que ocurren en un proceso. Cuando hay procesos externos de los que no se tiene conocimiento pleno, pero que actúan de alguna forma en el proceso a modelar, se pueden dibujar en el diagrama como un pool solitario. También, se dibujan en pools solitarios los participantes del proceso no pertenecientes a la organización.
- b. Adicionando los “lanes” a los “Pools”:** Los lanes encierran las actividades y eventos cubiertos por un rol de negocio, el cual es asumido por una persona.
- c. Adicionar el evento de inicio y de final del proceso:** El evento de inicio puede describirse como la llegada de un mensaje (puede ser para descubrir una solicitud, la llegada de un documento) o la llegada de un evento de tiempo (como por ejemplo, cuando un proceso debe ejecutarse regularmente cada cierto tiempo). Si lo inicia un participante directamente, se define un evento de inicio básico.
- d. Se identifican los pasos principales (o actividades globales) del proceso y se dibujan.** Se aclara que estos pasos principales van a contener una serie de subpasos, por los se dibujan con la notación de las actividades de tipo subprocesso.
- e. Se conectan los pasos a través de flujos de secuencia.** Normalmente son flechas que indican el paso de una actividad a otra. Hay flujos en los que se toman decisiones para ir a una actividad un otra, o que se inician actividades en paralelo, o que esperan a que un conjunto de actividades finalice para avanzar a la siguiente. Para definir los dos últimos flujos, se utilizan las compuertas lógicas (o gateways).

**Paso 3.** Se adicionan los caminos de excepción del nivel superior. Para esto, se pueden seguir estos pasos:

- a. Identificar los estados de finalización no exitosa del proceso.
- b. Ingresar eventos de finalización para cada estado final no exitoso del proceso.
- c. Se insertan los gateways, que son compuertas para especificar condiciones que van a dirigir las actividades tanto en caso de éxito como en caso que el proceso tenga que finalizar de manera no exitosa.

**Paso 4.** Para cada actividad de subproceso definida, se tiene que detallar toda la secuencia de pasos que deben ocurrir para que se cumpla, y describir los pasos a los eventos de finalización no exitosa de la actividad de tipo subproceso.

**Paso 5.** Si se tienen Pools externos, se trazan los flujos de mensajes que van desde y hacia estos Pools por parte del proceso encerrado en un Pool.

**Paso 6.** Si se quiere describir flujo de datos, se pueden utilizar las figuras mensaje (tiene apariencia de carta), objeto de datos (para describir variables que cambian en el proceso, tiene apariencia de una hoja de papel) o un almacén de datos (para descubrir fuentes externas de datos que se usan de cierta forma en el proceso).

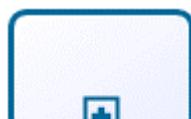
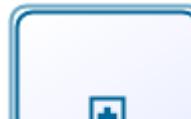
### 2.1.6 Identificación de Subprocesos: Reusables y embebidos

**Subproceso:** Un sub proceso es una actividad compuesta que se incluye dentro de un proceso. Compuesta significa que puede ser desglosada en niveles más bajos, esto es, que incluye figuras y elementos dentro de él.

Por defecto, los sub procesos son creados como **embebidos**. En BPMN, este tipo de sub-procesos no tienen pools, ni lanes. Si usted necesita incluir pools y lanes, usted puede utilizar sub-procesos **reusables** que, de acuerdo al estándar BPMN, pueden incluir pools y lanes. Estos tipos de sub-procesos son actividades predefinidas. Esto es, son independientes y por lo tanto son creados individualmente, luego, pueden ser relacionados al elemento del sub-proceso.

En BPMN este tipo de sub-proceso es conocido como **Actividad de Llamada**, en Bizagi Modeler lo denominamos **Reusable**.

**Tipos de Subprocesos:**

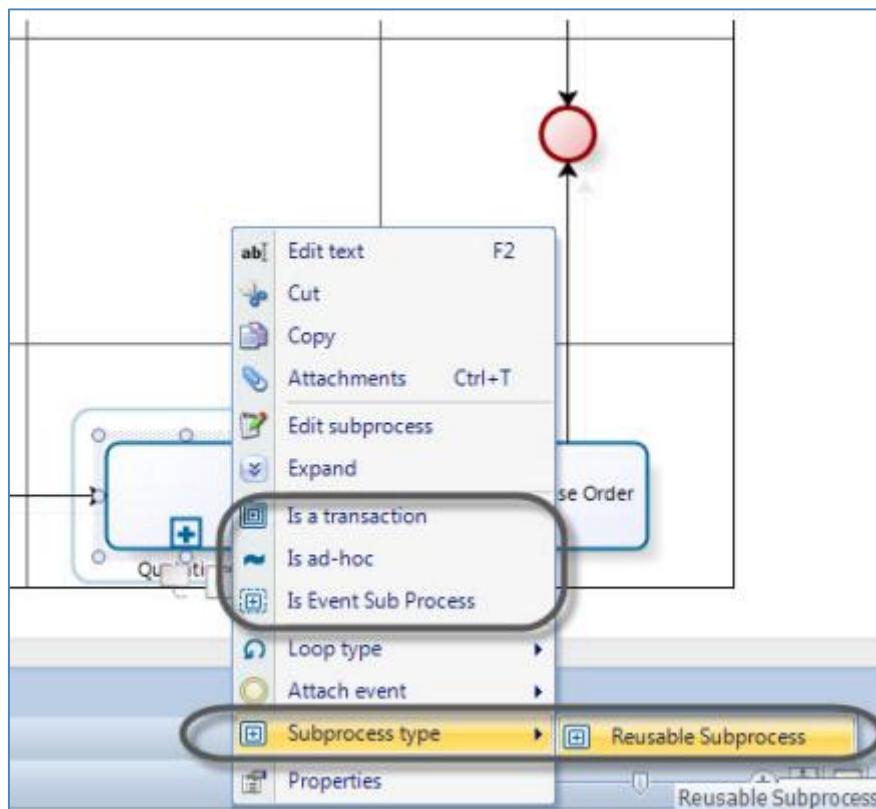
ELEMENTO	DESCRIPCIÓN	NOTACIÓN
<b>Sub-proceso</b>	Es una actividad cuyos detalles internos han sido modelados utilizando actividades, compuertas, eventos y flujos de secuencia.	 Subproceso
<b>Sub-proceso Reusable</b>	Identifica un punto en el flujo donde se invoca un proceso pre-definido. Los procesos reusables se conocen como Actividades de Llamada en BPMN.	 Subproceso reusable
<b>Sub-proceso de Transacción</b>	Es un sub proceso cuyo comportamiento es controlado a través de un protocolo de transacción.  Este incluye los tres resultados básicos de una transacción: Terminación exitosa, terminación fallida y evento intermedio de cancelación.	 Transacción
<b>Sub-proceso Ad-Hoc.</b>	Es un grupo de actividades que no requieren relaciones de secuencia. Se puede definir un conjunto de actividades, pero su secuencia y número de ejecuciones es determinada por sus ejecutantes.	 Ad-Hoc
<b>Ciclo estándar.</b>	Los sub procesos pueden repetirse secuencialmente comportándose como un ciclo.  Esta característica define un comportamiento de ciclo basado en una condición booleana. La actividad se ejecutará siempre y cuando la condición booleana sea verdadera.	 Ciclo estándar
<b>Ciclo Multiinstancia</b>	Los sub procesos pueden repetirse secuencialmente comportándose como un ciclo. El ciclo multi-instancias permite la creación de un número deseado de instancias de actividad que pueden ser ejecutadas de forma paralela o secuencial.	 Ciclo Multiinstancia

### Cambiar el tipo de sub-proceso:

BPMN define varios tipos de sub proceso que responden a necesidades de negocio particulares. Por defecto los sub-procesos, se crean como embebidos y usted puede cambiar el tipo de sub proceso en cualquier momento.

Una vez que usted ha creado un elemento de sub-proceso en el diagrama, dé clic derecho sobre él y seleccione una de las siguientes opciones:

1. Es transacción
2. Es ad-hoc
3. Es sub proceso de evento
4. Tipo de sub proceso -> Sub proceso reusable



**Figura 32: Simplificación de tareas**  
Fuente.- Bizagi Modeler: Guía del Usuario

Adicionalmente usted puede definir el Tipo de Ciclo de sus procesos, el cual puede ser:

1. Ninguno.
2. Multi-instanciación: El atributo multi-instanciación de un sub-proceso permite la creación de un número deseado de instancias de la actividad.
3. Estándar: Esta característica define un comportamiento cíclico basado en una condición booleana. Este sub proceso se ejecutará siempre y cuando la condición booleana sea verdadera.

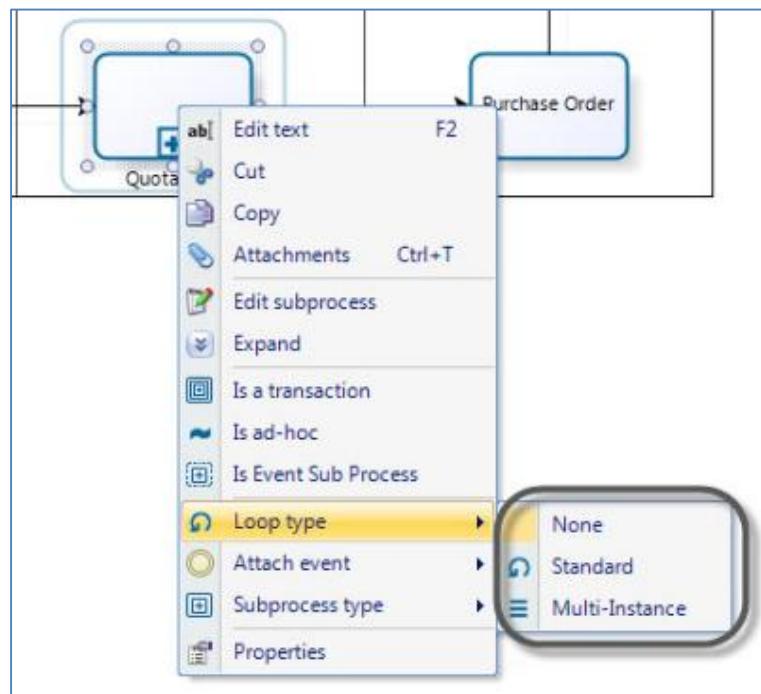


Figura 33: Simplificación de tareas  
Fuente.- Bizagi Modeler: Guía del Usuario

#### Convertir a sub-proceso reusable:

Usted puede convertir un sub proceso por defecto a un sub proceso reusable o Actividad de Llamada 1. Dé clic derecho sobre el sub proceso que desea convertir, seleccione la opción Tipo de Sub proceso y luego dé clic en Proceso Reusable.

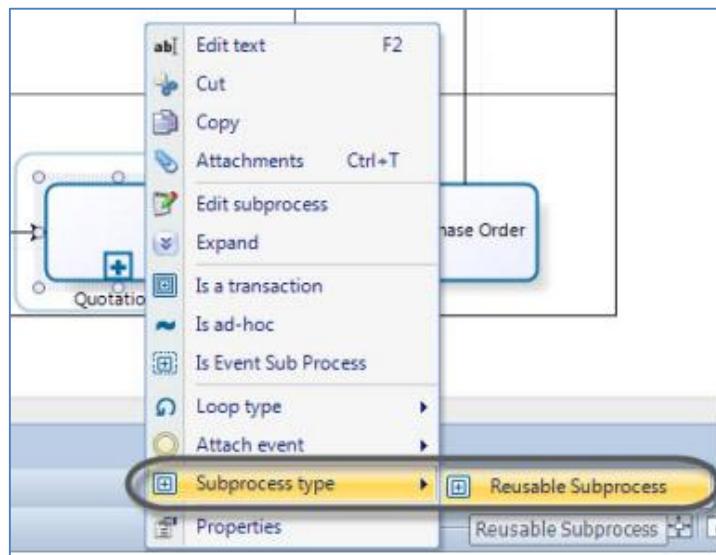


Figura 34: Simplificación de tareas

Fuente.- Bizagi Modeler: Guía del Usuario

Los bordes de la figura cambiarán indicando que ahora es un sub proceso reusable.

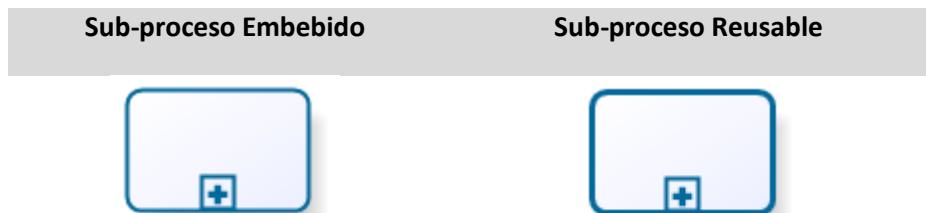


Figura 35: Diferencias entre Subproceso embebido y reusable

Fuente.- Elaboración propia

### 2.1.7 Uso de Temporizadores, manejo de errores.

#### Temporizador:

El evento de tiempo (también llamado temporizador) se utiliza cuando una condición de tiempo ocurre. Gráficamente, se reconoce como un reloj como marcador dentro del evento.

Como **evento de inicio** se puede utilizar para:

1. iniciar cada ciertos intervalos un proceso,
2. iniciar un proceso regularmente en una fecha y hora indicada,
3. iniciar un proceso en una relación temporal con otro evento e iniciar un proceso por única vez en una fecha y hora determinada.

Como **evento intermedio** el temporizador puede detener el proceso, hasta que:

1. un tiempo definido se haya alcanzado,
2. un período de tiempo haya transcurrido,
3. se haya alcanzado un tiempo, que se encuentre en relación a otro evento.

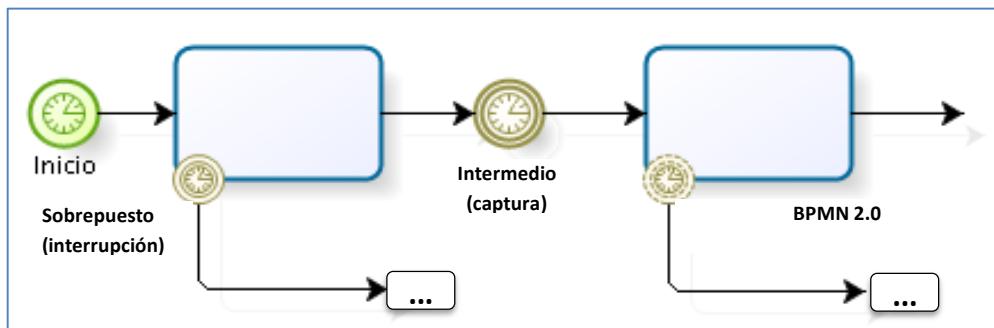


Figura 36: Tipos de Eventos de tiempo

Fuente.- Freund et.al. - BPMN 2.0

A continuación, se aprecian algunos ejemplos de uso del temporizador. Un evento de tiempo no puede ser impulsado por un proceso, porque sobre el tiempo no tenemos influencia, razón por la cual este evento existe solo en forma de «**evento de captura**».

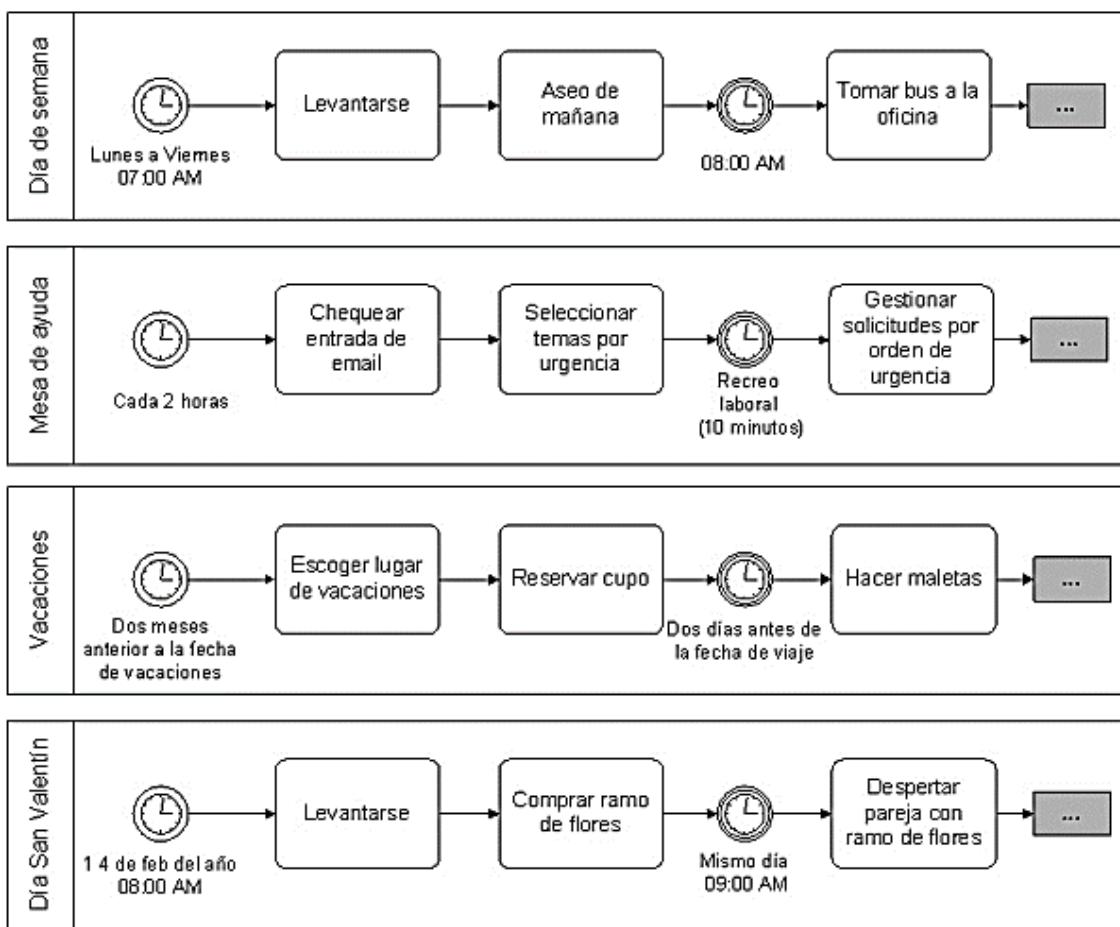


Figura 37: Utilización de Eventos de tiempo

Fuente.- Freund et.al. - BPMN 2.0

Muy a menudo, se utiliza el temporizador **sobrepuerto** como «**timeout**», tiempo máximo permitido para la ejecución de una actividad. En la figura 37, mostramos un ejemplo donde solo nos damos un tiempo de **30 minutos** para escoger una pizza.

Si no nos ponemos de acuerdo dentro este lapso, cancelamos la búsqueda y cocinamos pasta, pero en ambos casos vamos a comer lo que llegue a la mesa.

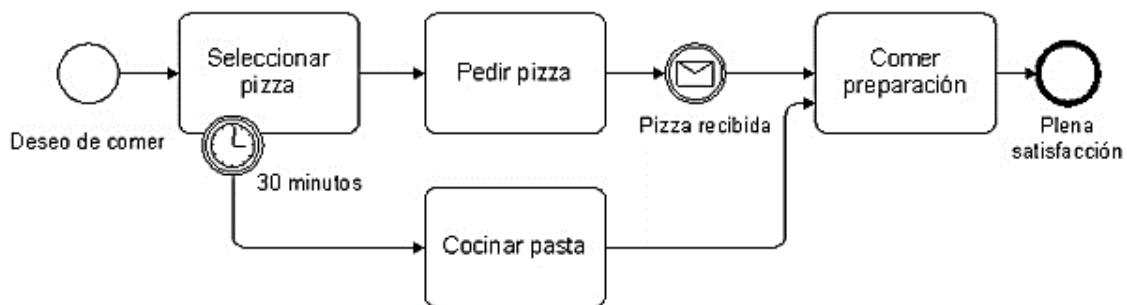


Figura 38: el “timeout” para la actividad “seleccionar pizza” es de 30 minutos.

Fuente.- Freund et.al. - BPMN 2.0

A partir de la versión **2.0** en BPMN también se pueden utilizar **temporizadores sobrepuertos que no interrumpen la actividad**. Un ejemplo para un evento del tipo temporizador sin interrupción lo vemos en la figura 38.

Antes de poder cenar juntos hay que preparar la comida y poner la mesa, que lo haremos solo **10 minutos** antes que esté lista la comida.

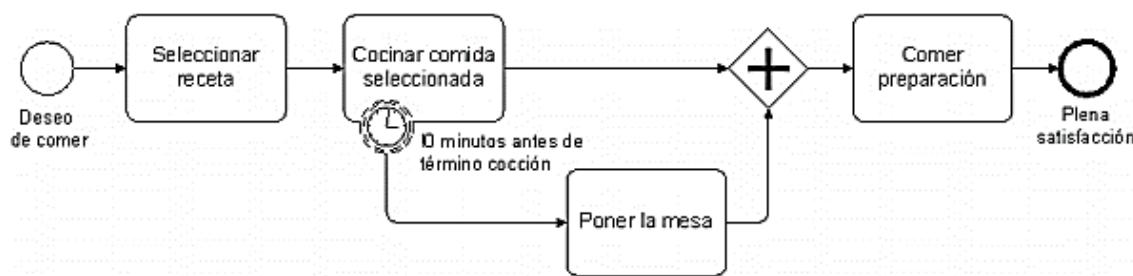


Figura 39: Utilización de evento de tiempo, intermedio y sobrepuerto del tipo "no interrupción".

Fuente.- Freund et.al. - BPMN 2.0

### Manejo de errores:

En pocas ocasiones, se puede asumir que durante la ejecución de los procesos no ocurren errores. Si usted, identifica los puntos donde pueden ocurrir errores, los puede interceptar utilizando este tipo de eventos. Gráficamente, se representan con un signo de rayo dentro del círculo del evento como lo muestra la figura 39.

La notación de BPMN no especifica tipos de errores. Sin embargo, en BPMN se considera un error como un evento excepcional, razón por la cual sólo se puede modelar como evento intermedio sobrepuerto y que además requiere de un tratamiento excepcional. Como tipo disparador solo se debe usar como evento final, indicando que el proceso ha sido cancelado por error, o bien el evento es capturado por un subproceso superior que lo lleva a un tratamiento especial.

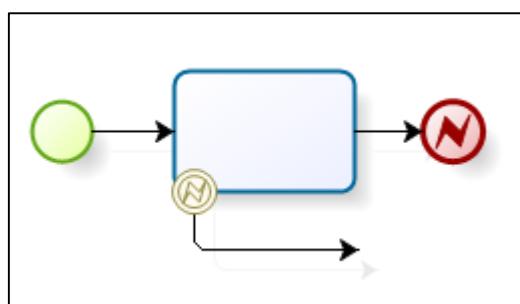


Figura 40: Tipos de Eventos de Error.

Fuente.- Freund et.al. - BPMN 2.0

Debido a lo anterior, vamos a darles en la sección 4.5.1 algunas recomendaciones prácticas al respecto.

La colaboración entre subprocesos (superior vs inferior) se explicará en la sección 2.8. Allí, encontrará ejemplos para el tratamiento de eventos por error.

## 2.1.8 Transacciones: envío y captura de compensaciones

### Evento de Compensación:

Compensar en BPMN significa volver al estado inicial de una actividad. En la práctica, utilizamos el evento de compensación solo en el contexto de transacciones que tienen que ser reversadas. BPMN también permite «compensar» sin la contemplación de transacciones, pero en la práctica no hemos visto casos en que no estén involucradas las transacciones, razón por la cual las tratamos en conjunto.

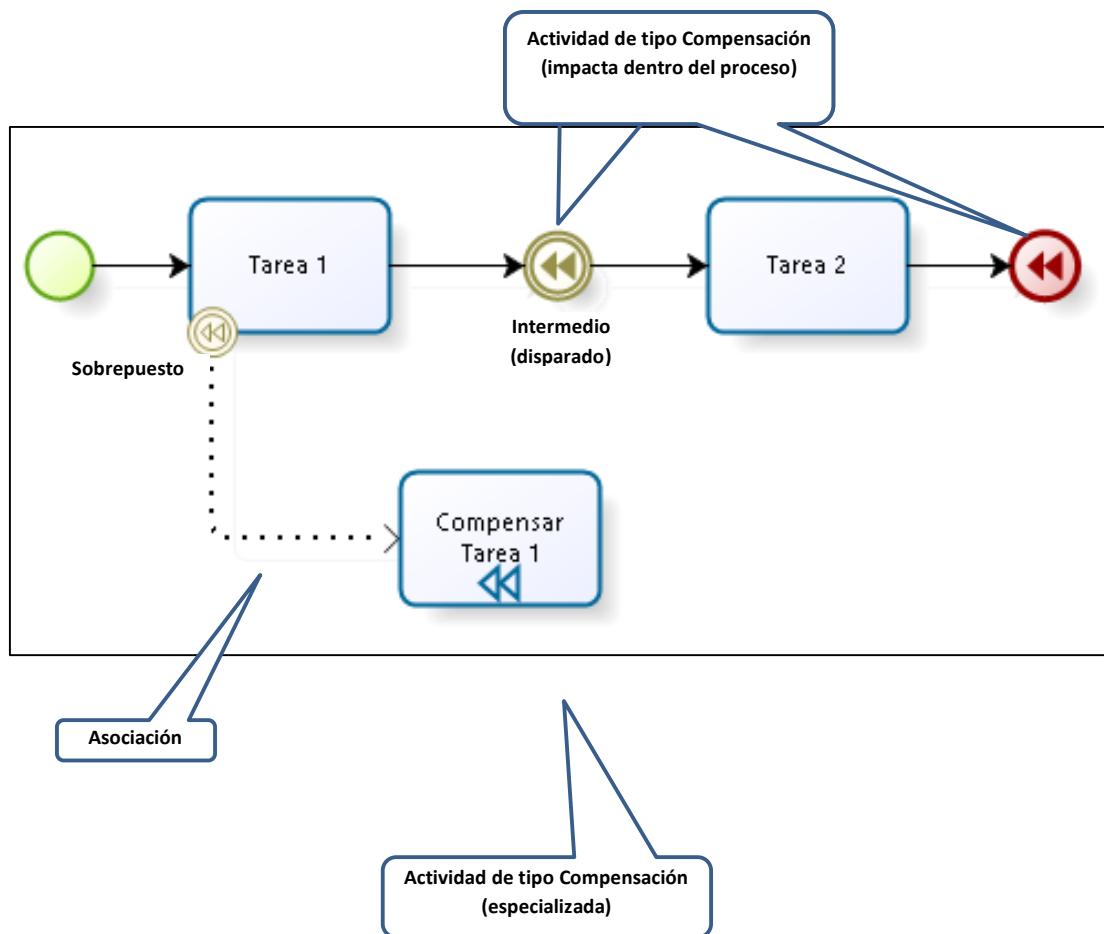


Figura 41: Tipos de Eventos de Compensación.

Fuente.- Freund et.al. - BPMN 2.0

Típicos ejemplos de compensación podrían ser:

1. la reserva de un pasaje
2. la reserva de un hotel
3. el pago con tarjeta de crédito
4. una transferencia electrónica

### **Transacción:**

Muchos procesos funcionan solo bajo el principio «todo o nada», es decir o todos los pasos están correctos o no se debe hacer nada. En el tema anterior, conocimos el evento de compensación, el cual impulsa una acción de reversar todo lo hecho anteriormente.

En BPMN, entendemos bajo el elemento de **Transacción** un subproceso especializado que apoya las funcionalidades de procesos transaccionales. La figura 41 muestra en forma ejemplar como funciona un proceso transaccional:

Vamos a suponer que queremos visitar en nuestras vacaciones a nuestros familiares que viven en otro país.

Una vez tomada esta decisión vamos a planificar y preparar nuestro viaje. Primero, vamos a coordinar con ellos una fecha y la duración de la estadía. Luego, vamos a reservar pasajes de vuelo en una línea aérea económica y reservar una habitación en un hotel cercano a nuestros familiares, a pesar que ellos insisten que nos alojemos en su casa.

En realidad, pensamos en nuestra independencia, además de no comprometernos o abusar de ellos. Por último, tenemos que solicitar la fecha de vacaciones a nuestro jefe.

Si todo resulta bien, podemos emprender nuestro viaje, pero ¿qué hacemos si el hotel está copado y no encontramos otro alternativo debido a la sobre demanda de la época, o si nuestro jefe no nos aprueba la fecha de vacaciones solicitada?

En este caso, tenemos que abortar nuestra preparación del viaje. En BPMN, existe un símbolo que solo se puede utilizar para cancelar transacciones (cruz blanca al borde del subproceso).

Si una transacción es cancelada, se disparan automáticamente todos los eventos de compensación que son necesarios. Entonces, le avisamos a nuestros familiares que no podemos viajar, el ticket de vuelo lo perdemos, porque las condiciones tarifarias no permiten devolución de dinero (compensación no posible) y cancelamos la reserva de hotel.

Luego de compensar todas las actividades posibles, el evento de transacción cancelada nos lleva nuevamente a la planificación de viaje y buscamos otra fecha alternativa.

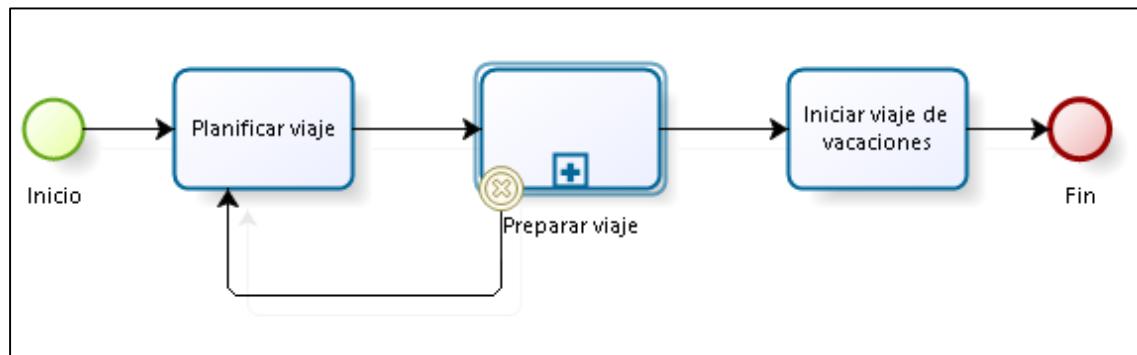


Figura 42: Caso con uso de subprocesso colapsado del tipo Transacción.

Fuente.- Freund et.al. - BPMN 2.0

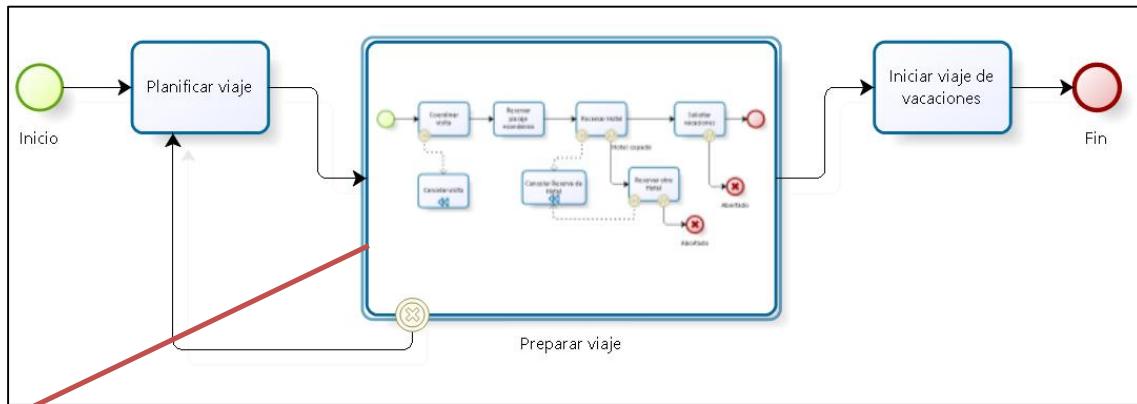


Figura 43: Caso con uso de subprocesso expandido del tipo Transacción.

Fuente.- Freund et.al. - BPMN 2.0

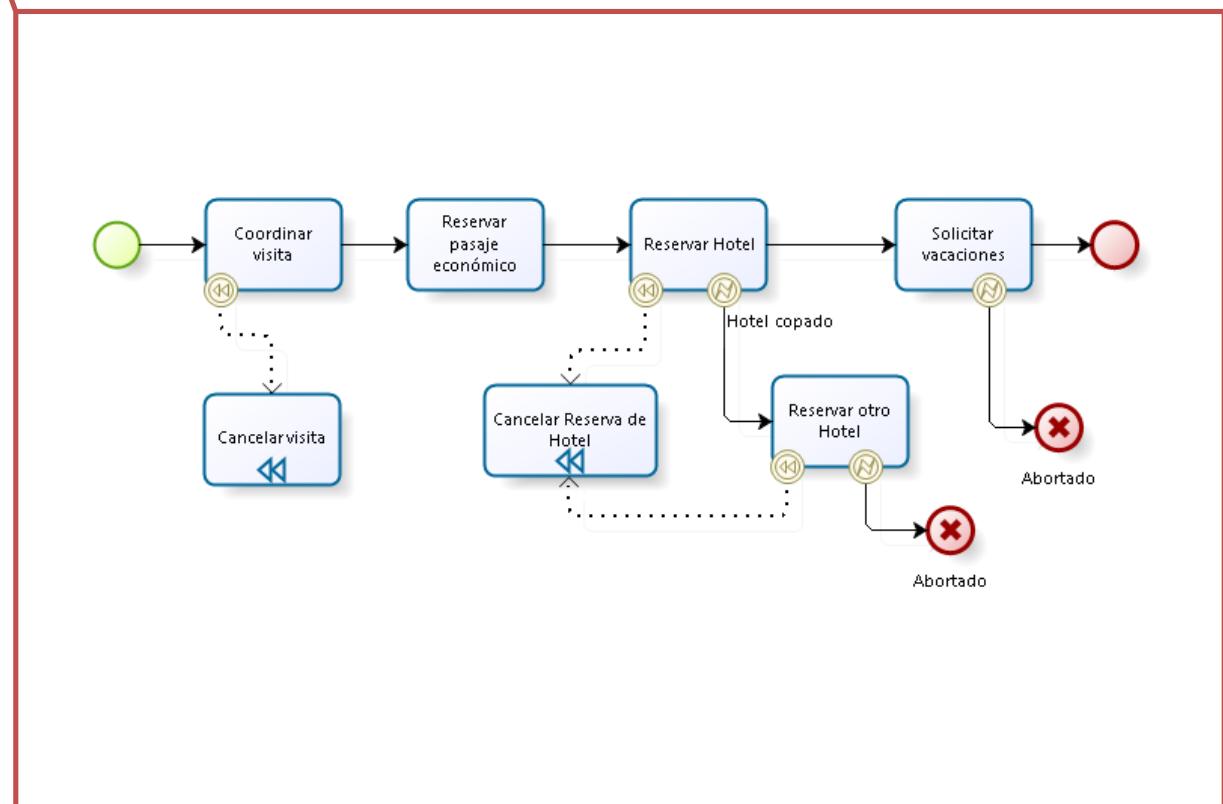


Figura 44: Subproceso transaccional magnificado.

Fuente.- Freund et.al. - BPMN 2.0

# Resumen

1. El modelado a nivel Analítico describe todas las actividades, decisiones y excepciones necesarias para analizar un proceso completamente y crear requerimientos detallados. Expresa toda la lógica de negocio y las reglas de negocio.
2. Las compuertas paralelas permiten la ejecución de actividades simultáneas.
3. Las compuertas basadas en eventos representan puntos donde lo que sigue son ocurrencias de eventos.
4. Las compuertas exclusivas basadas en eventos se utilizan para instanciar procesos.
5. Las compuertas paralelas basadas en eventos también crean instancias; pero de todos los procesos del ámbito de la compuerta.
6. Es conveniente aplicar las técnicas de simplificación de procesos para reducir el exceso de información y optimizar la ejecución de los procesos.
7. El uso del diseño top-down para modelar procesos facilita la construcción del modelo y proporciona un orden lógico de desarrollo.
8. Un Sub proceso es una actividad compuesta que se incluye dentro de un proceso.
9. Un Sub proceso Embebido es aquel que usa los datos del proceso padre y sólo puede ser usado por el proceso principal. Es el tipo de sub proceso por defecto.
10. Un Sub proceso Reusable es aquel que es independiente del proceso padre y puede usarse en otros procesos.
11. Un temporizador puede usarse como evento de inicio o intermedio.
12. Los subprocesos de transacción son útiles para controlar los errores; pero las compensaciones permiten manejar el error.

Pueden revisar los siguientes enlaces para ampliar los conceptos vistos en esta unidad:

1. <http://www.bpmvision.com.br/diferenca-entre-gateways-de-evento-3/>
2. [https://books.google.com.pe/books?id=B2WyaSJD-P8C&pg=PA80&lpg=PA80&dq=top-down+y+bpmn&source=bl&ots=sXgXw9XRHe&sig=W5z1\\_htATJ2WEcJKynHjxw2bZWs&hl=es&sa=X&ved=0CEIQ6AEwBWoVChMI9eL6q8L1xgIVSKACH0gRA6c#v=onepage&q=top-down%20y%20bpmn&f=false](https://books.google.com.pe/books?id=B2WyaSJD-P8C&pg=PA80&lpg=PA80&dq=top-down+y+bpmn&source=bl&ots=sXgXw9XRHe&sig=W5z1_htATJ2WEcJKynHjxw2bZWs&hl=es&sa=X&ved=0CEIQ6AEwBWoVChMI9eL6q8L1xgIVSKACH0gRA6c#v=onepage&q=top-down%20y%20bpmn&f=false)
3. <http://blog.crossnet.ws/>



# PATRONES DE DISEÑO

## LOGRO DE LA UNIDAD DE APRENDIZAJE

Al término de la unidad, el alumno aplica patrones de modelado estandarizados, para describir procesos de negocios reutilizables, mediante estereotipos comunes.

## TEMARIO

### 3.1 Tema 5 : Patrones de Control básico de Flujo

- 3.1.1 : Secuencia – División paralela
- 3.1.2 : Sincronización
- 3.1.3 : Decisión exclusiva – Unión simple
- 3.1.4 : Elección múltiple – Unión sincronizada
- 3.1.5 : Unión múltiple – Discriminador estructurado

### 3.2 Tema 6 : Patrones basados en Eventos y de Iteración

- 3.2.1 : Estructura y bloqueo parcial de unión
- 3.2.2 : Unión y división de caminos – otros.
- 3.2.3 : Decisión implícita – ejecución paralela intercalada
- 3.2.4 : Milestone – Sección crítica
- 3.2.5 : Ciclos arbitrarios – Bucle estructurado.

## 3.1 PATRONES DE CONTROL BÁSICO DE FLUJO

### 3.1.1 Secuencia – División paralela

**Secuencia:** Este patrón es utilizado para modelar dependencia entre tareas, es decir, una tarea no puede empezar hasta que otra no haya terminado (ejecución en secuencia)

#### Ejemplo

Cuando una compañía de seguros recibe un reclamo en alguna póliza, es necesario realizar varias actividades con un orden definido. Primero, el cliente debe reportar el reclamo, luego, éste debe ser evaluado para poder autorizar el pago y finalmente se desembolsa la cantidad reclamada. No tendría sentido que una reclamación sea evaluada antes de ser reportada, o pagar el dinero de la reclamación antes de evaluarla.

#### Implementación

Para modelar este patrón, es necesario conectar las actividades (en el orden de ejecución definido) utilizando conectores de flujo de secuencia como se muestra en el diagrama.

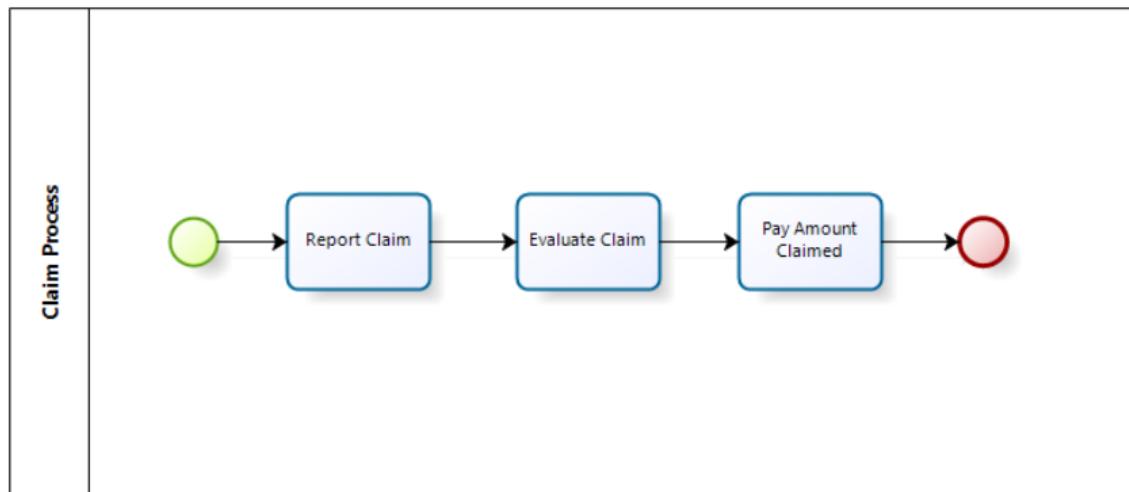


Figura 45: Ejemplo de patrón de secuencia.  
Fuente.Bizagi Process Modeler – Patrones de Modelado

**División paralela:** Una división paralela es un punto del proceso donde un camino es dividido en dos o más ramas paralelas las cuales son ejecutadas al mismo tiempo.

### Ejemplo

Cuando un nuevo empleado llega a la compañía, es necesario realizar varias actividades, por ejemplo, darle acceso a la información de su cargo, firmar algunos documentos legales y alistar su puesto de trabajo.

### Implementación

Para implementar la División Paralela, es necesario utilizar la Compuerta Paralela. Esta compuerta activa caminos alternativos sin verificar condiciones.

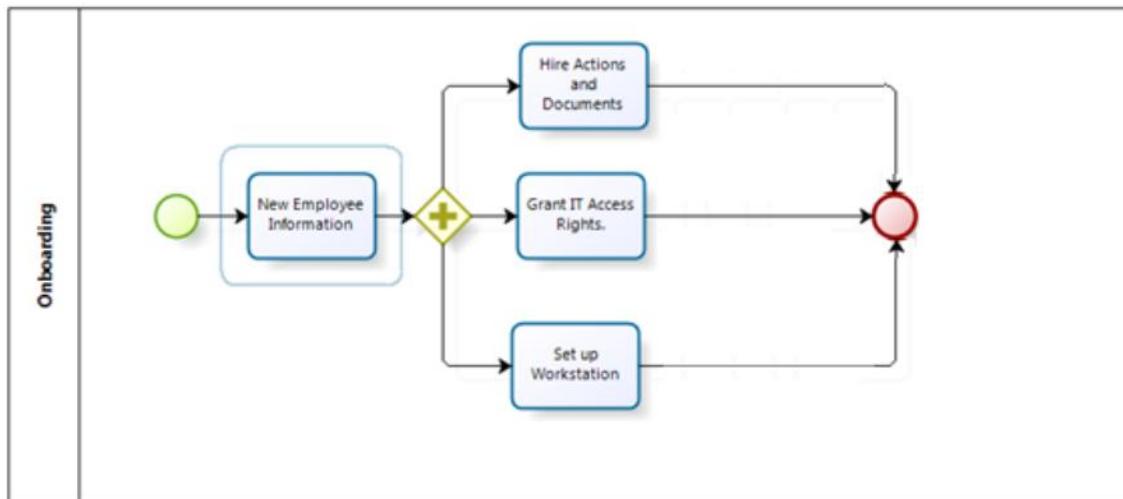


Figura 46: Ejemplo de patrón de División paralela.  
Fuente.Bizagi Process Modeler – Patrones de Modelado

### 3.1.2 Sincronización:

La sincronización es un punto en el proceso donde dos o más ramas del proceso se unen en una sola. Se llama sincronización porque se espera a que todas las ramas entrantes se completen antes de continuar con la siguiente actividad [2].

### Ejemplo

Cuando un nuevo empleado llega a la compañía es necesario realizar varias actividades, por ejemplo, darle acceso a la información de su cargo, firmar algunos documentos legales y alistar su puesto de trabajo.

El empleado no puede empezar a trabajar hasta que todas las actividades se hayan completado.

### Implementación

Para este ejemplo, vamos a utilizar una Compuerta Paralela como elemento convergente para lograr la unión de todas las ramas.

El patrón de sincronización también puede ser modelado utilizando compuertas inclusivas o exclusivas dependiendo de los requerimientos del negocio.

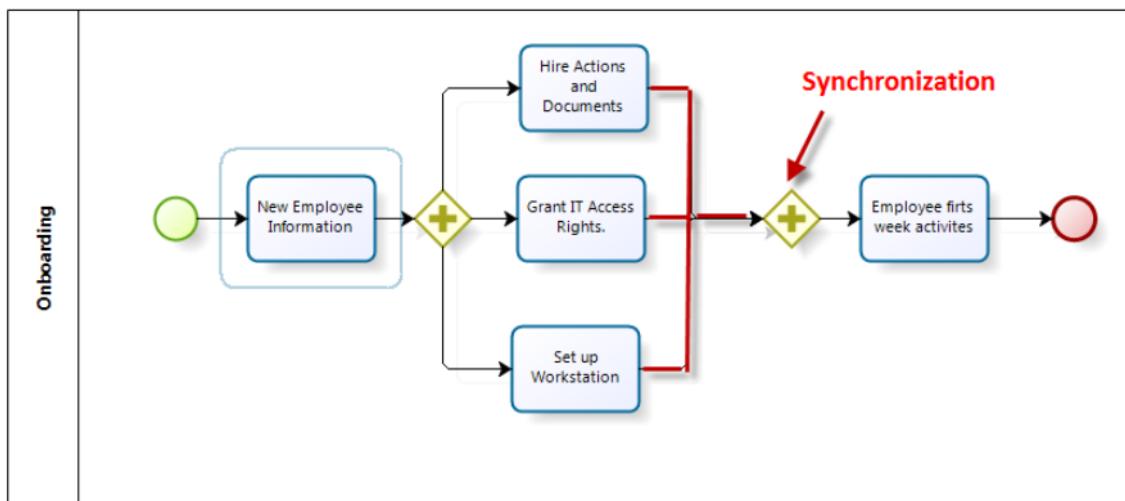


Figura 47: Ejemplo del patrón de Sincronización.  
Fuente.Bizagi Process Modeler – Patrones de Modelado

### 3.1.3 Decisión exclusiva – Unión simple

**Decisión exclusiva:** La decisión exclusiva representa un punto en el proceso donde se debe escoger un solo camino de varios disponibles dependiendo de una decisión o de datos del proceso.

### Ejemplo

El Departamento de Contabilidad recibe y paga las facturas de la compañía. El pago de una factura se puede realizar a través de transferencia bancaria, cheque o tarjeta de crédito. Solo es posible realizar el pago de la factura con un medio de pago.

## Implementación

El patrón de decisión exclusiva puede ser modelado a través de una Compuerta Exclusiva. Para este caso, la compuerta tiene tres secuencias de flujo pero el proceso solo debe continuar por uno de ellos de acuerdo al cumplimiento de la condición definida.

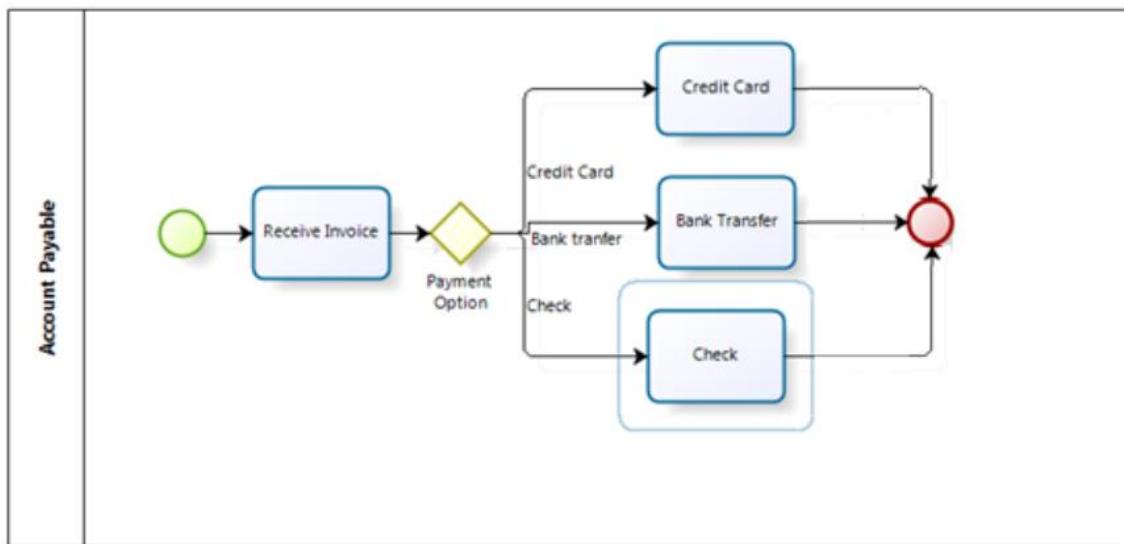


Figura 48: Ejemplo del patrón de Decisión exclusiva.

Fuente.Bizagi Process Modeler – Patrones de Modelado

**Unión simple:** La unión simple es un punto del proceso donde dos o más caminos alternativos convergen en uno solo. Es un supuesto de este patrón que los caminos alternativos no son ejecutados en paralelo.

## Ejemplo

Tomemos el ejemplo que utilizamos en el patrón WCP 4. Suponga que luego de realizar el pago con alguna de las opciones disponibles el ERP financiero es actualizado. Note en el diagrama que la actividad *Actualizar ERP Financiero* solo será ejecutada una vez, esto debido a que solo una de las ramas entrantes es activada.

## Implementación

Para realizar la implementación de este patrón, se utilizan conectores de flujo de secuencia para conectar las actividades de las ramas entrantes con la siguiente actividad.

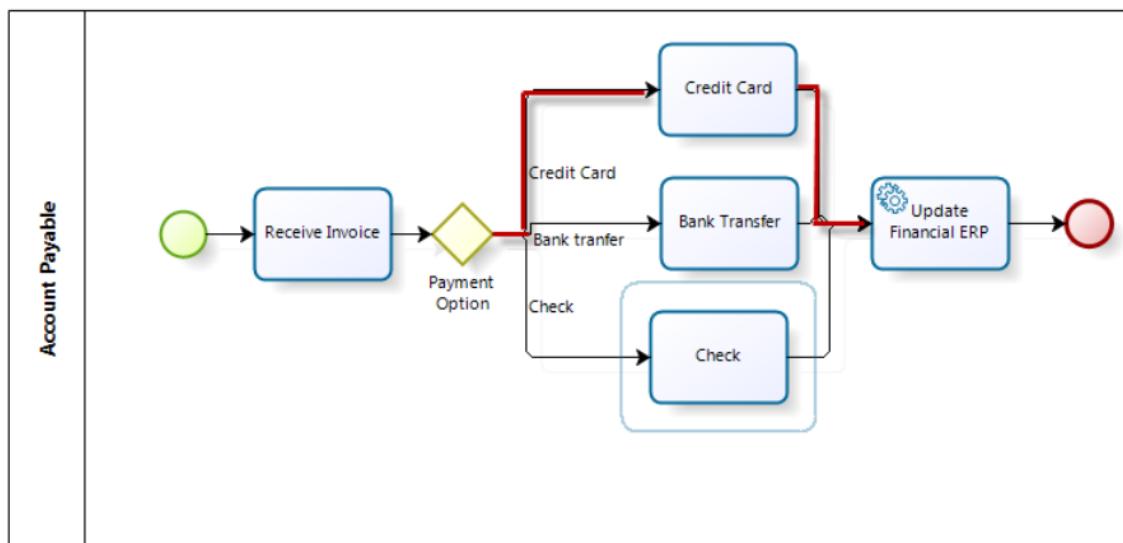


Figura 49: Ejemplo del patrón de Unión simple.

Fuente.Bizagi Process Modeler – Patrones de Modelado

### 3.1.4 Elección múltiple – Unión sincronizada

**Elección múltiple:** El patrón de elección múltiple es utilizado para modelar puntos en el flujo de proceso donde varios caminos son escogidos dependiendo de datos del proceso y/o puntos de decisión.

#### Ejemplo

Durante un proceso de auditoría, es muy común encontrar no conformidades dentro del proceso auditado. Las no conformidades deben ser evaluadas y corregidas por el dueño del proceso.

La corrección de una no conformidad puede ser realizada de diferentes formas, por ejemplo, con una acción correctiva, una acción preventiva, una acción inmediata o una combinación entre las opciones anteriores.

Para implementar el patrón de elección múltiple, se utiliza una **Compuerta Inclusiva**. Esta compuerta es utilizada para activar uno o más caminos dependiendo de datos del proceso.

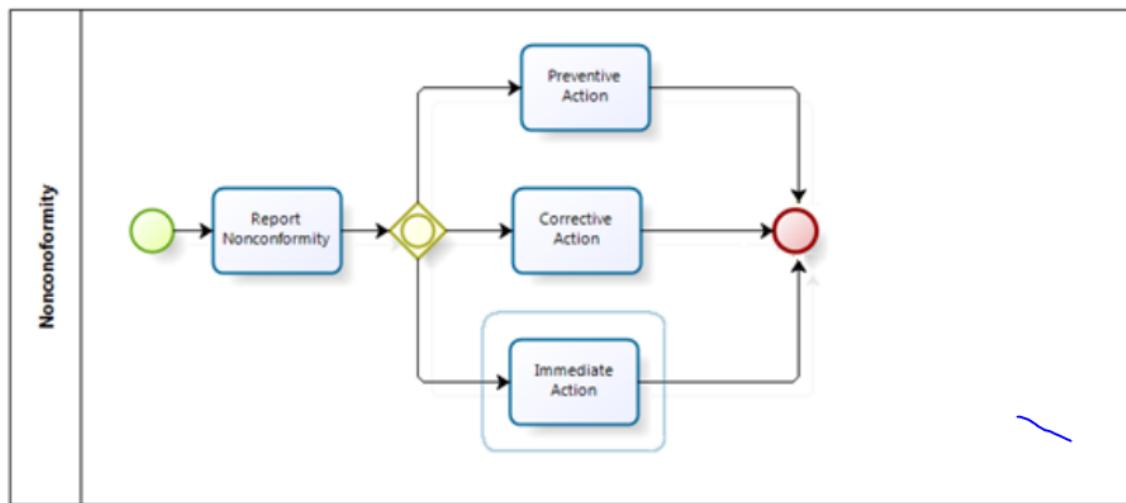


Figura 50: Ejemplo del patrón de elección múltiple.

Fuente.Bizagi Process Modeler – Patrones de Modelado

**Unión sincronizada:** Es un punto en el proceso donde múltiples caminos que fueron activados antes en el proceso convergen en una sola rama, el proceso con continua hasta que todas las actividades de las ramas entrantes hayan terminado [2].

### Ejemplo

Continuando con el ejemplo utilizado en WCP 4, suponga ahora que la no conformidad no puede ser cerrada hasta que las actividades que fueron seleccionadas hayan terminado.

### Implementación

Para implementar el patrón, se deben utilizar dos compuertas inclusivas, una como elemento divergente (activar algunas de las ramas salientes) y otro como elemento de sincronización o convergente (para esperar que todas las ramas activadas lleguen a un determinado punto para continuar con el proceso).

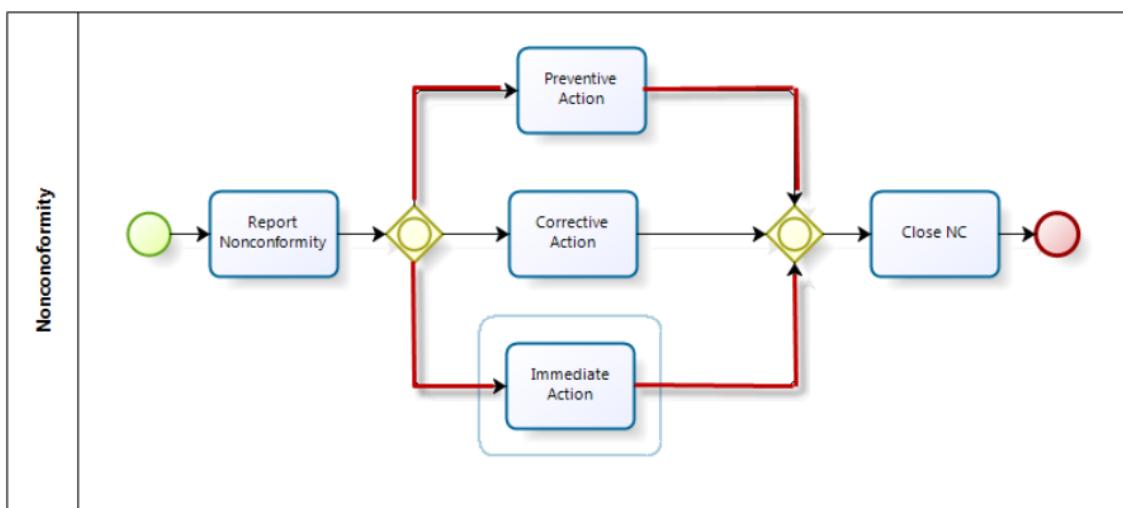


Figura 51: Ejemplo del patrón de Estructura de unión sincronizada.

Fuente.Bizagi Process Modeler – Patrones de Modelado

### 3.1.5 Unión múltiple – Discriminador estructurado

**Unión múltiple:** El patrón de unión múltiple es usado para converger dos o más ramas en un solo camino. Cada activación de una de las ramas entrantes resulta en la activación de la siguiente actividad en el proceso.

#### Ejemplo

Durante el Proceso de Selección de personal es necesario verificar las referencias del nuevo empleado. Es muy importante verificar las referencias personales y laborales suministradas. Cada vez que una referencia es verificada, el Jefe de Recursos Humanos debe ser notificado.

#### Implementación

El patrón utiliza una compuerta paralela para habilitar los dos caminos. Cada vez que la actividad de uno de los caminos es terminada, la siguiente actividad, en este caso Informar sobre Referencias, es ejecutada.

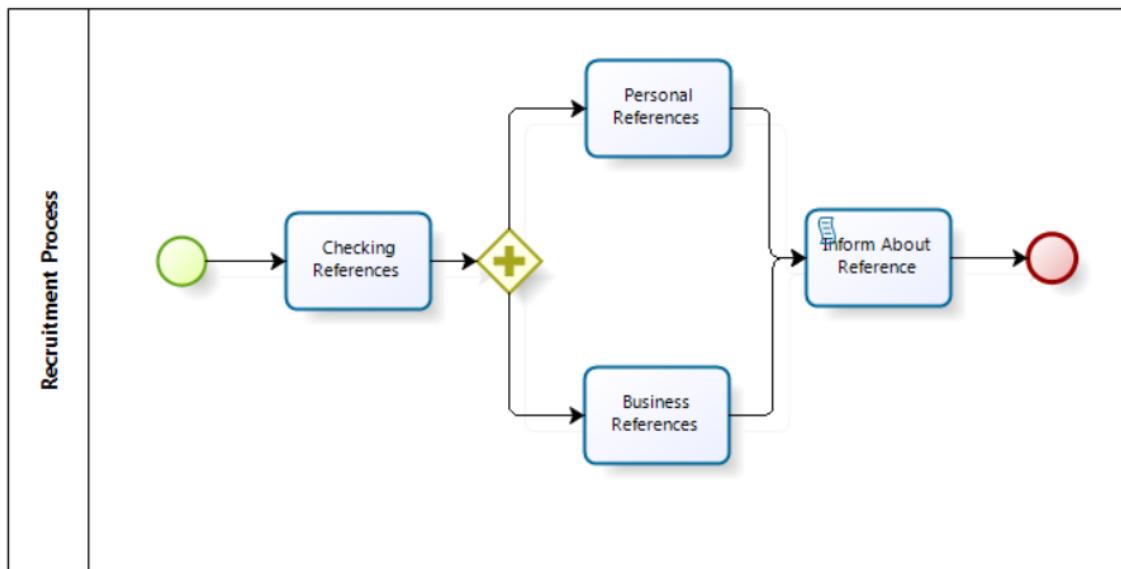


Figura 52: Ejemplo del patrón de Unión múltiple.

Fuente.Bizagi Process Modeler – Patrones de Modelado

**Discriminador estructurado:** Este patrón describe un punto en el proceso donde se espera a que una de las ramas entrantes haya sido completada antes de continuar con la siguiente actividad, las otras ramas son omitidas después de ser completadas. Una vez todas las ramas entrantes han sido completadas, el discriminador se reinicia.

### Ejemplo

Un empleado solicita un préstamo a su compañía. Para otorgar el crédito, es necesaria que la solicitud sea aprobada por su jefe o por el jefe del área financiera. Cuando alguno de los dos dé su aprobación, el dinero puede ser desembolsado al empleado.

### Implementación

Este patrón puede utilizar compuertas **Paralelas** o **Inclusivas** para dividir los caminos.

Para modelar el discriminador, es necesario utilizar una **Compuerta Compleja**, la cual espera hasta que una de las dos ramas entrantes sea completada.

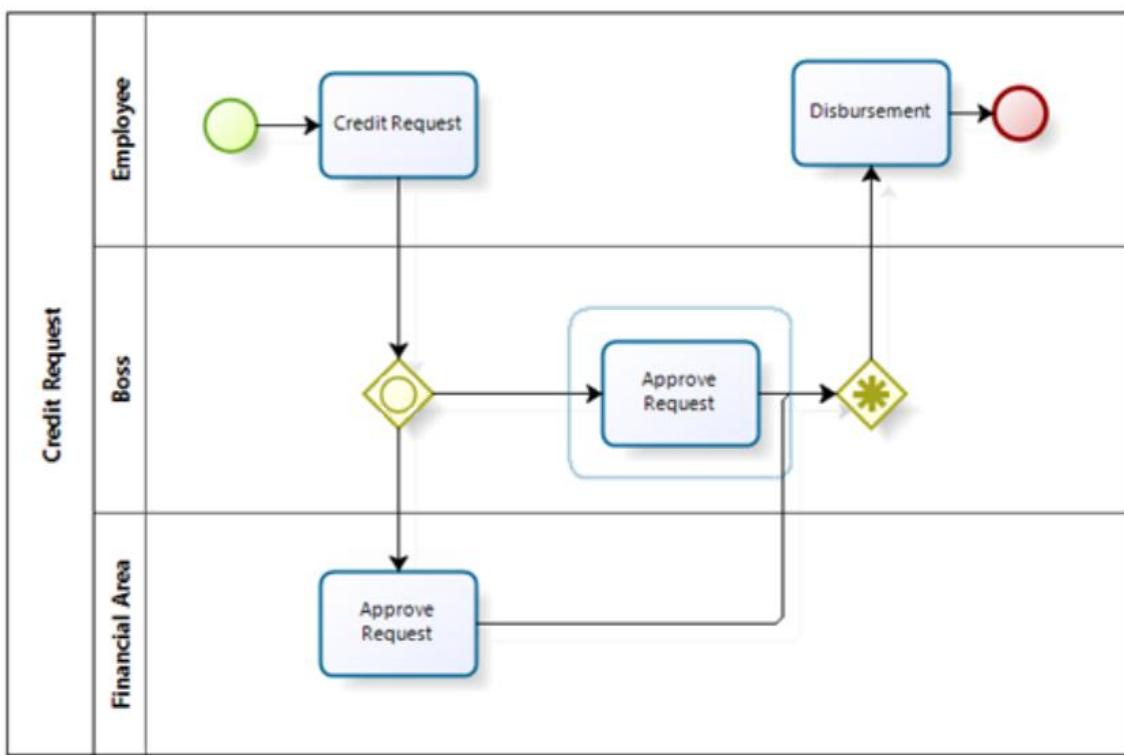


Figura 53: Ejemplo del patrón Discriminador estructurado.

Fuente.Bizagi Process Modeler – Patrones de Modelado

# Resumen

1. Los patrones de modelado describen situaciones que comúnmente se encuentran en los procesos de negocio.
2. Los patrones de modelado de procesos propuestos por el profesor Van Der Aalst utilizando el Modelador de Procesos de Bizagi, tienen el auspicio de la OMG.
3. Los patrones de control de flujo básico muestran los aspectos básicos para el control del flujo de procesos.

Pueden revisar los siguientes enlaces para ampliar los conceptos vistos en esta unidad:

- [https://www.bizagi.com/docs/Workflow\\_Patterns\\_using\\_BizAgi\\_Process\\_Modeler\\_Esp.pdf](https://www.bizagi.com/docs/Workflow_Patterns_using_BizAgi_Process_Modeler_Esp.pdf)
- <https://grupocinco.files.wordpress.com/2007/11/bpmn.pdf>
- <http://help.bizagi.com/processmodeler/es/>

## 3.2 PATRONES BASADOS EN EVENTOS Y DE ITERACIÓN

### 3.2.1 Estructura y bloqueo parcial de unión.

**Estructura parcial de Unión:** El patrón modela la convergencia de M ramas que han sido habilitadas en un punto anterior del proceso en un solo camino. El proceso continúa cuando N de las M ramas han sido completadas.

#### Ejemplo

Un empleado solicita un préstamo a la compañía donde labora. Para otorgar el crédito, es necesaria que la solicitud sea aprobada por su jefe y por el jefe del área financiera.

#### Implementación

Esta compuerta puede utilizar una **Compuerta Paralela** o **Inclusiva** para dividir los caminos. Para modelar la estructura parcial de unión se utiliza una **Compuerta Compleja**, la cual espera hasta que las dos ramas entrantes sean completadas.

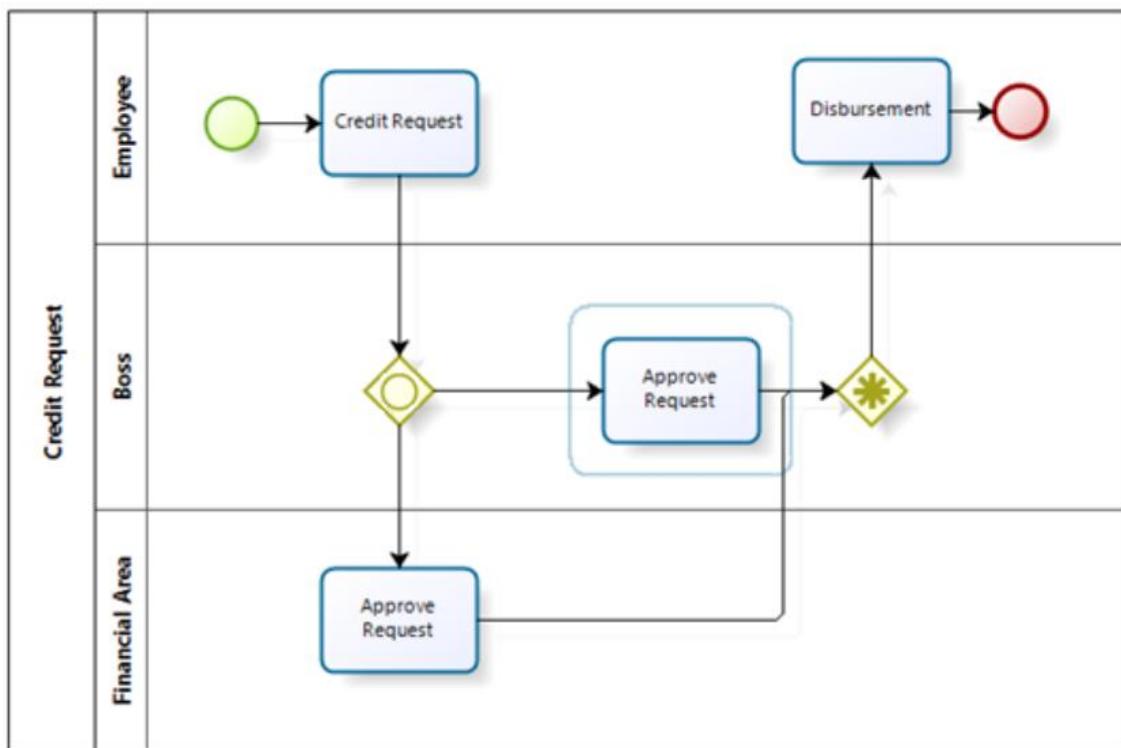


Figura 54: Ejemplo del patrón de Estructura parcial de unión.

Fuente.Bizagi Process Modeler – Patrones de Modelado

**Bloqueo parcial de Unión:** El patrón representa la convergencia de dos o más ramas, que han sido activadas antes en algún punto del proceso, en un solo camino. El proceso continúa cuando N de las M ramas habilitadas anteriormente han terminado.

La unión es reiniciada cuando todas las ramas entrantes han sido ejecutadas una vez en la misma instancia del proceso. La ejecución de las ramas entrantes se bloquea hasta que el elemento de unión ha sido reiniciado.

### Ejemplo

Una compañía decide invertir en propuestas de negocios. Estas propuestas son enviadas por estudiantes de varias universidades, cada una de ellas es evaluada de forma individual por 3 jueces.

Si 2 de los 3 jueces dan su aprobación, el aspirante será informado acerca de la decisión.

Cuando el último de los jueces decida sobre la propuesta, será posible empezar a estudiar una nueva.

### Implementación

El patrón utiliza una compuerta Paralela o una compuerta Inclusiva para dividir la rama entrante.

Una compuerta compleja es utilizada para validar el número de aprobaciones (N) necesarias para informar sobre la decisión.

Para bloquear las ramas entrantes hasta que la unión sea reiniciada, se utiliza un Evento

Condicional. El evento evalúa si hay alguna aprobación en progreso, y bloqueará la entrada de nuevas propuestas hasta que todos los jueces hayan dado su veredicto de la propuesta en curso.

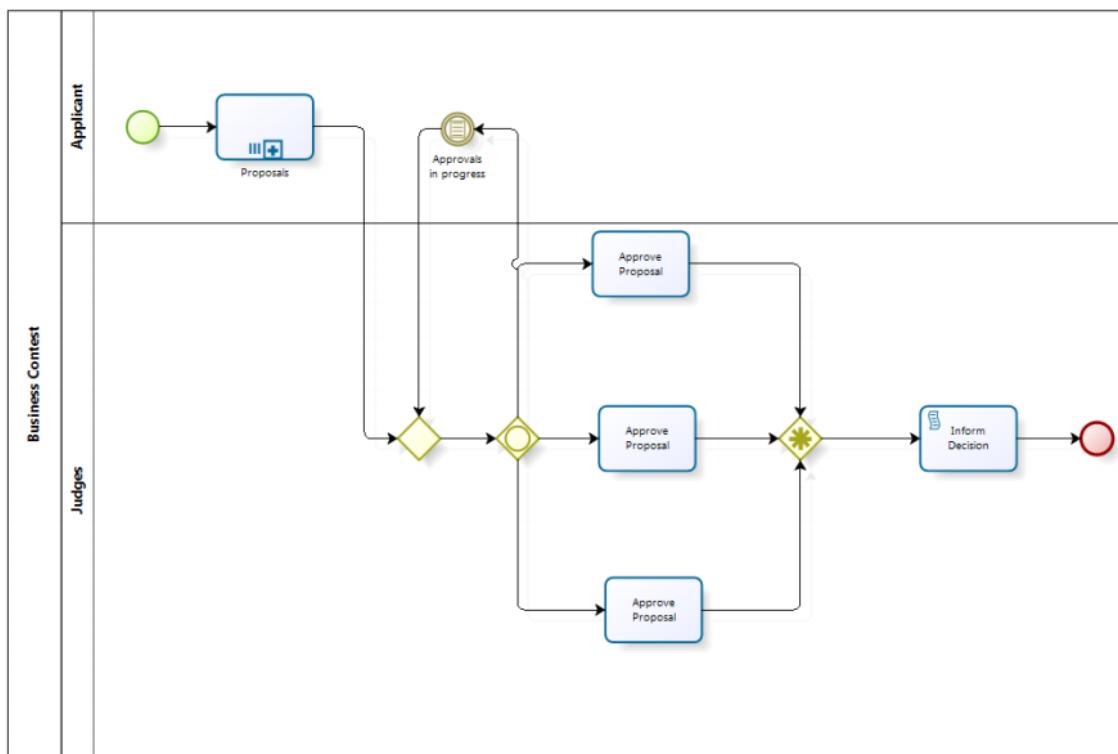


Figura 55: Ejemplo del patrón de Bloqueo parcial de unión.

Fuente.Bizagi Process Modeler – Patrones de Modelado

### 3.2.2 Unión y división de caminos – otros.

**Unión de caminos:** El patrón describe un punto en el proceso donde un número de ejecuciones, de un mismo camino en una misma instancia del proceso, convergen en un solo camino de ejecución.

#### Ejemplo

Una compañía decide evaluar la percepción del ambiente de trabajo entre los empleados. Una encuesta es enviada a 100 empleados. Cuando todos los empleados hayan enviado sus resultados, el analista puede evaluar la información.

#### Implementación

Para implementar este patrón, es necesario utilizar un sub proceso para crear las 100 instancias de la actividad Completar encuesta. En este caso, el sub proceso Enviar Encuesta es instanciado 100 veces.

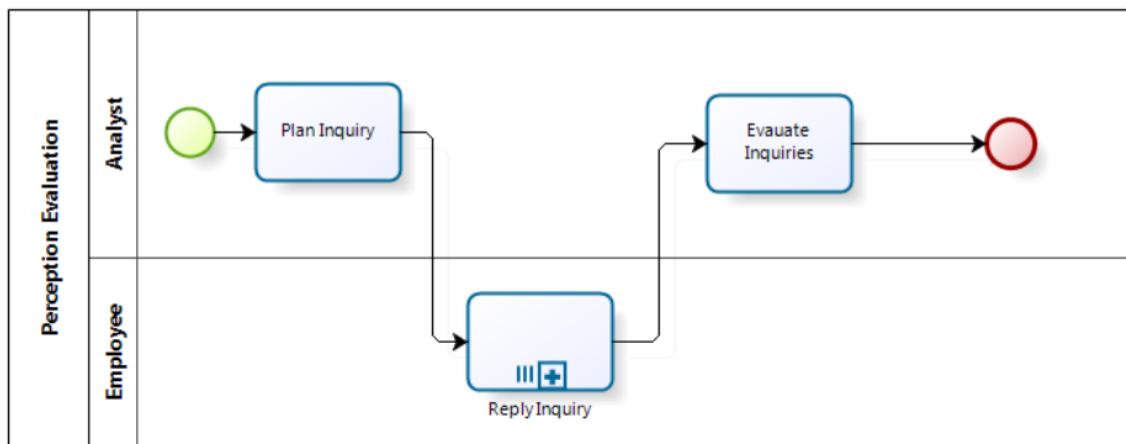
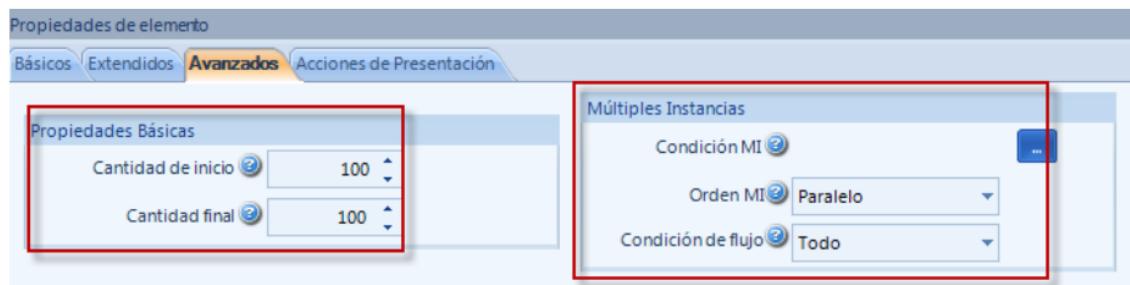


Figura 56: Ejemplo del patrón de Unión de caminos.

Fuente.Bizagi Process Modeler – Patrones de Modelado

Para configurar el comportamiento del sub proceso para la condición que se desea, es necesario definir las siguientes propiedades:



**Cantidad de Inicio = 100:** Esto significa que el proceso debe ser instanciado 100 veces.

**Cantidad Final= 100:** El sub proceso se considera como terminado cuando se hayan completado todas (en este caso) las 100 instancias.

**Condición MI= Ninguna.** Esto quiere decir que no hay condición para el número de instancias para el sub proceso.

**Orden MI = Paralelo:** Las instancias del sub proceso serán ejecutadas en paralelo.

**Condición de Flujo = Todo:** El proceso continúa cuando todas las instancias del subproceso han terminado.

**División del camino:** El patrón describe un punto del proceso donde un determinado número de ejecuciones de un mismo camino deben ser instanciados.

### Ejemplo

Una compañía decide evaluar la percepción del ambiente de trabajo entre los empleados. Una encuesta es enviada a 100 empleados. Cada vez que un empleado completa la encuesta, esta debe ser evaluada por el analista.

### Implementación.

Para implementar este patrón, es necesario utilizar un sub proceso para crear las 100 instancias de la actividad *Completar encuesta*. En este caso, el sub proceso *Enviar Encuesta* es instanciado 100 veces.

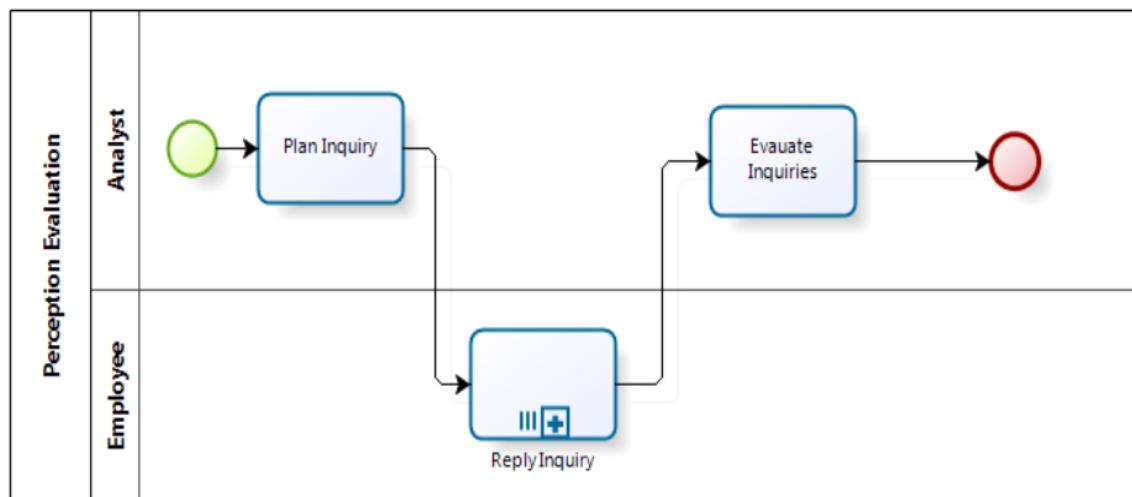
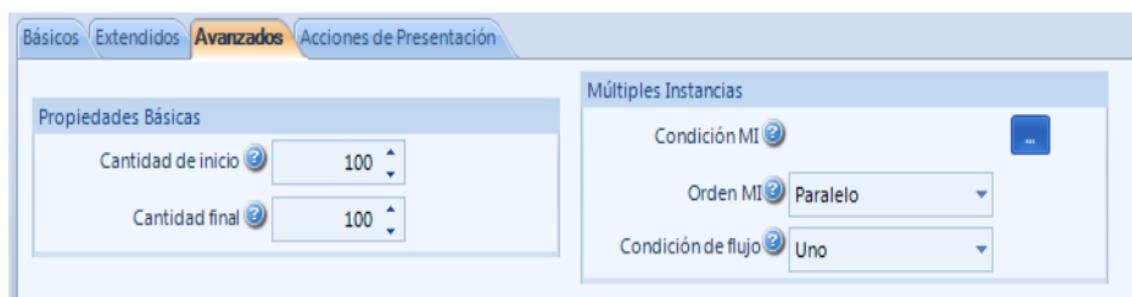


Figura 57: Ejemplo del patrón de División del camino.

Fuente: Bizagi Process Modeler – Patrones de Modelado

Para configurar el comportamiento del sub proceso para la condición que se desea, es necesario definir las siguientes propiedades:



**Cantidad de Inicio = 100:** Esto significa que el proceso debe ser instanciado 100 veces.

**Cantidad Final= 100:** El sub proceso se considera como terminado cuando se hayan completado todas (en este caso) las 100 instancias.

**Condición MI= Ninguna.** Esto quiere decir que no hay condición para el número de instancias para el sub proceso.

**Orden MI = Paralelo:** Las instancias del sub proceso serán ejecutadas en paralelo.

**Condición de Flujo = Uno.** El proceso continua cada vez que una instancia del sub proceso es terminada.

### 3.2.3 Decisión implícita – ejecución paralela intercalada

**Decisión implícita:** El patrón describe un punto en el proceso donde una rama es escogida de varias disponibles. La decisión es tomada de acuerdo a los datos del proceso. Cuando una rama es escogida, las demás se deben deshabilitar.

#### Ejemplo

Cuando un cliente realiza una solicitud de crédito, es necesario solicitarle diferentes documentos; Si el cliente no trae los documentos antes de 5 días se le debe contactar y preguntar si continúa o no en el proceso. Si el cliente no continúa en el proceso, no es necesario esperar por los documentos y el proceso debe terminar. Si el cliente, trae los documentos, no es necesario contactarlo.

#### Implementación

Para implementar el patrón es necesario utilizar una compuerta exclusiva basada en eventos. Esta compuerta representa un punto en el proceso donde solo una de las ramas disponibles de debe ejecutar. Las ramas restantes deben ser deshabilitadas.

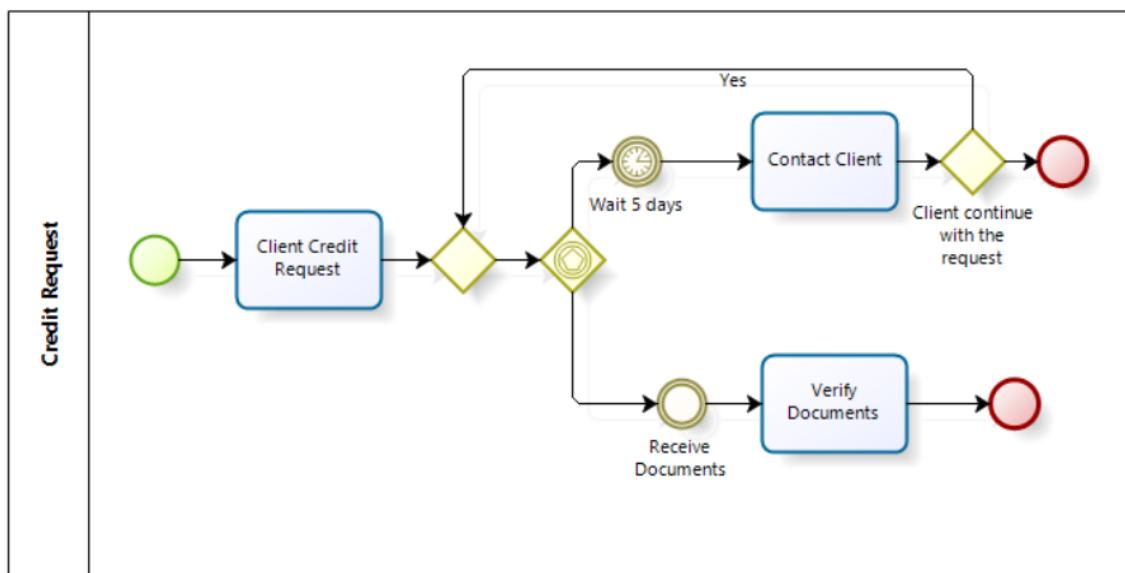


Figura 58: Ejemplo del patrón de Decisión implícita.

Fuente.Bizagi Process Modeler – Patrones de Modelado

**Ejecución paralela intercalada:** Un conjunto de actividades tienen un orden de ejecución definido. Cada actividad, puede ser ejecutada una vez y se puede completar en cualquier momento de acuerdo al orden definido. Sin embargo, como requisito adicional, dos actividades no pueden ser ejecutadas al mismo tiempo.

### Ejemplo

El Jefe de Recursos Humanos estable que durante los procesos de selección los aspirantes deben presentar tres pruebas de aptitud: una prueba psicológica, un test de inteligencia y un test de conocimientos específicos.

El orden en que se realizan las pruebas es importante. El test de inteligencia debe ser realizado antes de la prueba psicológica. La prueba de conocimientos debe ser realizada en cualquier momento. No es posible realizar dos pruebas al mismo tiempo.

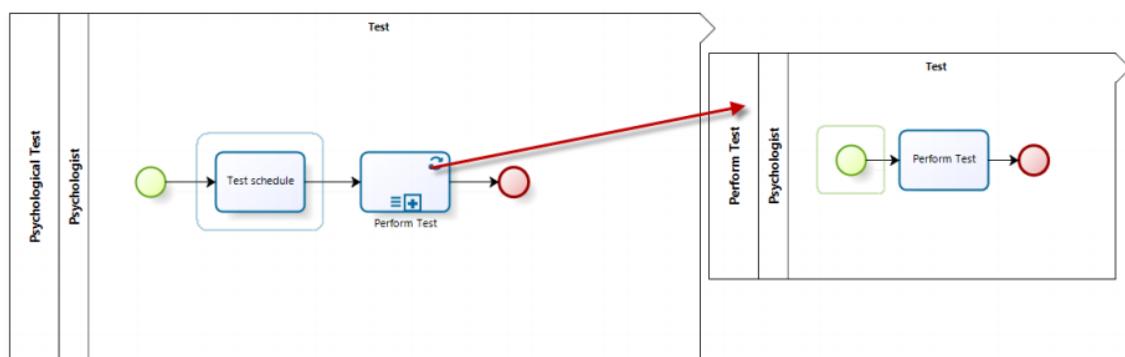


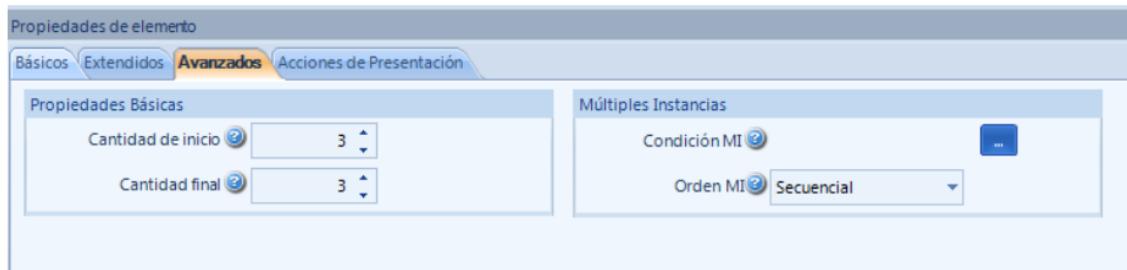
Figura 59: Ejemplo del patrón de Ejecución paralela intercalada.

Fuente.Bizagi Process Modeler – Patrones de Modelado

## Implementación

Para modelar el patrón, se utiliza un sub proceso múltiple, el sub proceso solo incluye una actividad.

Para definir el orden en el cual se realizan las pruebas, es necesario establecer como propiedades del sub proceso los siguientes datos.



La secuencia en que las pruebas se realizan es definida en la primera actividad del proceso.

### 3.2.4 Milestone – Sección crítica

**Milestone:** El patrón define que una actividad es habilitada solamente cuando el proceso alcance un estado específico (Hito). Si el proceso, ha avanzado más allá de ese estado, la tarea no puede volver a ser habilitada.

## Ejemplo

En una agencia de viajes, es posible cambiar las fechas de las reservas de vuelos, carros y hoteles mientras las facturas no hayan sido impresas.

## Implementación

El patrón utiliza una *compuerta exclusiva basada en eventos* para controlar la disponibilidad de la actividad *Cambiar reservas*. Una vez la actividad *Imprimir Factura* se ha terminado, no es posible cambiar las reservas.

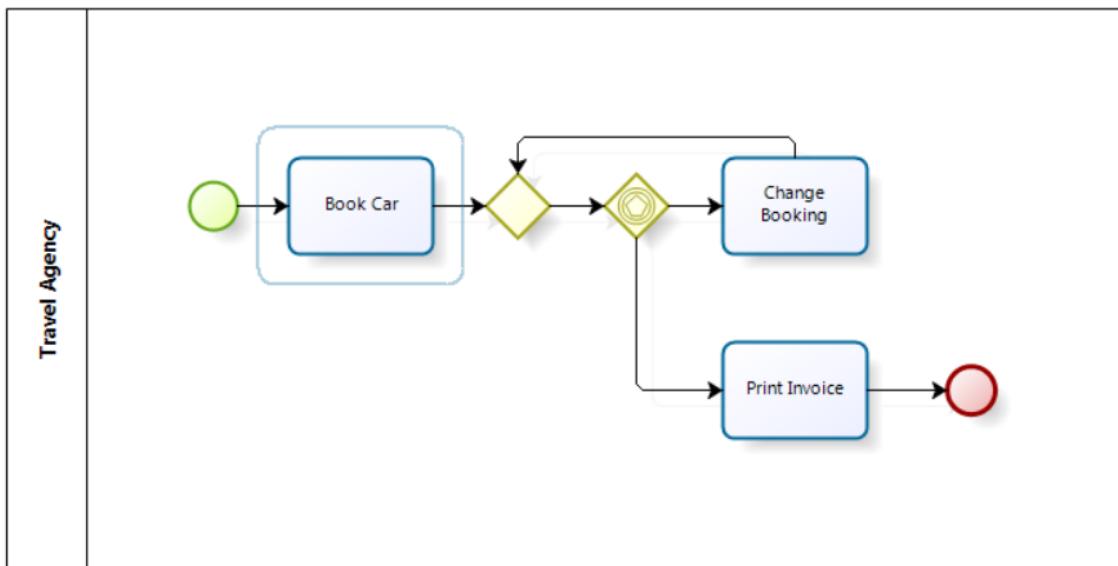


Figura 60: Ejemplo del patrón de Hito.

Fuente.Bizagi Process Modeler – Patrones de Modelado

**Sección crítica:** El patrón de Sección Critica describe la identificación de dos o más sub procesos o actividades como secciones críticas. Cuando una de estas secciones críticas, es activada, es decir la actividad dentro de la sección es habilitada, las otras secciones críticas no pueden ser activadas.

El proceso espera hasta que la sección crítica sea completada para permitir la ejecución de otra sección crítica.

### Ejemplo

Dos administradores tienen acceso al servidor para realizar cambios en la configuración.

Mientras que uno de ellos, se encuentre trabajando en el servidor no es posible que el otro administrador realice cambios. Él o ella deberán esperar hasta que el primer administrador termine su trabajo.

### Implementación

El patrón es modelado con la ayuda de una **Compuerta Basada en Eventos** para controlar que los dos administradores no trabajen de forma simultánea en la ejecución de tareas críticas. En este caso, la tarea crítica es la actualización del servidor y solo puede ser realizada por un administrador.

Una vez que el servidor ha sido actualizado, una compuerta exclusiva evalúa si se necesitan más cambios para volver a habilitar las tareas críticas o si el proceso puede terminar.

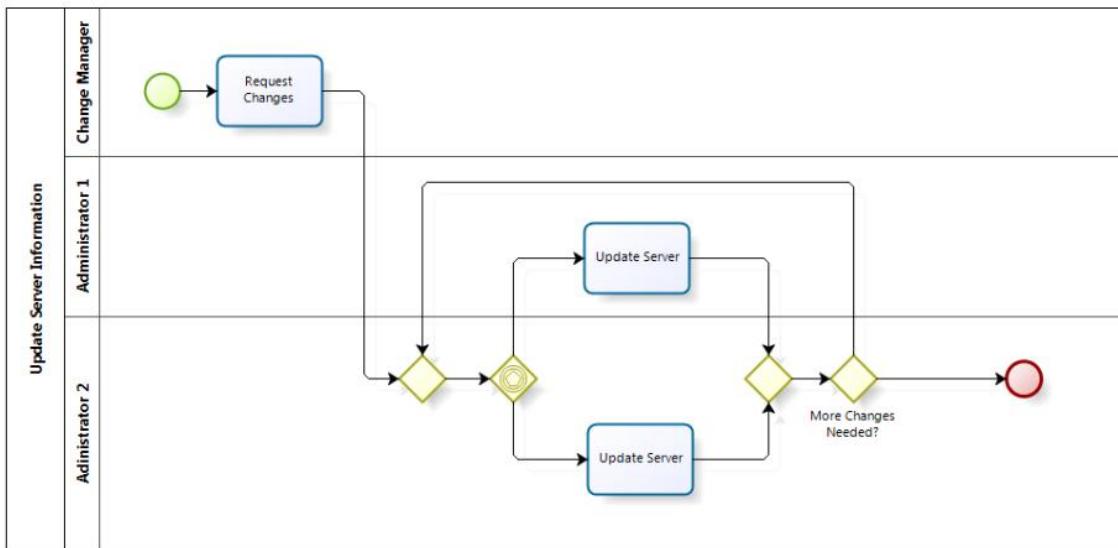


Figura 61: Ejemplo del patrón de Sección crítica.  
Fuente.Bizagi Process Modeler – Patrones de Modelado

### 3.2.5 Ciclos arbitrarios – Bucle estructurado

**Ciclos arbitrarios:** El patrón modela un punto en el flujo del proceso donde una o más actividades son realizadas varias veces.

#### Ejemplo

Los entregables de un proyecto necesitan la aprobación de varias personas. El número de aprobaciones es definido por el Gerente del Proyecto.

#### Implementación

El patrón utiliza una compuerta exclusiva que controla el número de veces que se ha aprobado el entregable, cuando se cumpla la condición (número de aprobaciones requeridas) el proceso continúa por el flujo normal.

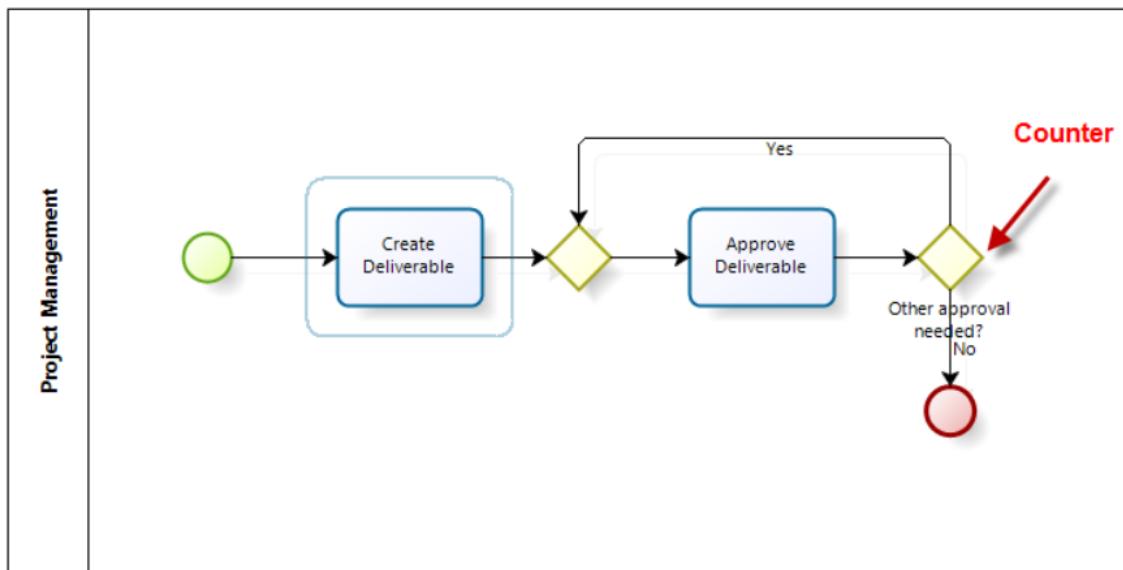


Figura 62: Ejemplo del patrón de Ciclo arbitrario.  
Fuente.Bizagi Process Modeler – Patrones de Modelado

**Bucle estructurado:** El patrón describe la posibilidad de ejecutar una actividad o un sub proceso varias veces. El ciclo o bucle tiene una validación de condición asociado a él. La condición es verificada al inicio o al final del ciclo para determinar si se continua. La estructura del ciclo tiene una sola entrada y un solo punto de salida.

### Ejemplo

En el proceso de *Cuentas por Pagar*, se reciben facturas. Si la factura no es aprobada, se devuelve al proveedor y se espera su envío de nuevo. Este proceso es realizado hasta que la factura sea aprobada.

### Implementación

Para implementar el patrón, es necesario utilizar una **compuerta exclusiva**. La compuerta valida una condición, en este caso, si la condición se cumple, el proceso continúa por el flujo normal, sino, se devuelve a la actividad *Recibir y Aprobar Factura*.

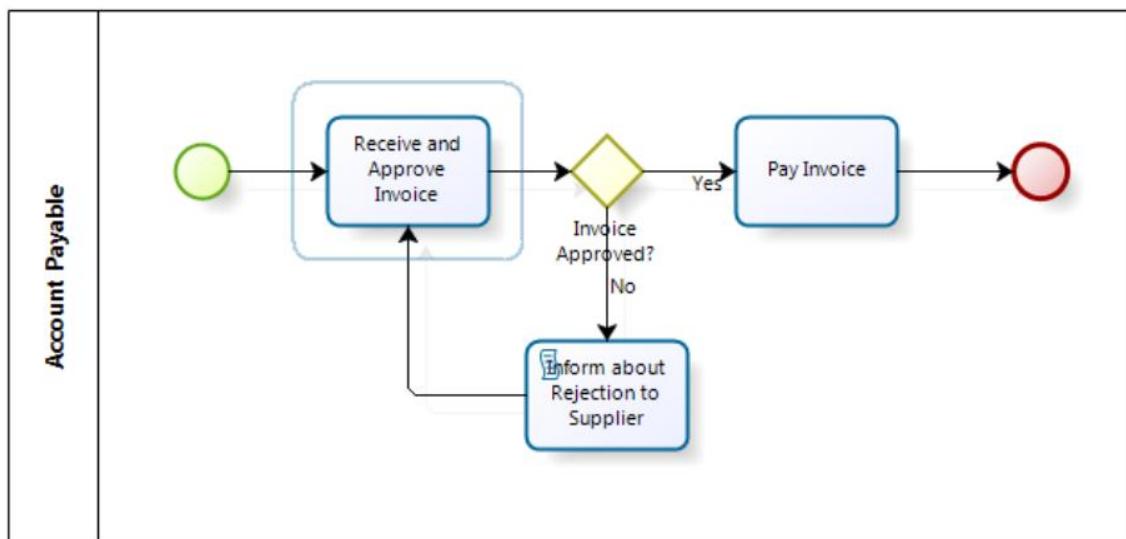


Figura 63: Ejemplo del patrón de Bucle estructurado.

Fuente. [Bizagi Process Modeler – Patrones de Modelado](#)

# Resumen

1. Los patrones basados en eventos, aplican los estereotipos más frecuentes, donde intervienen conceptos de eventos.
2. Los patrones de Iteración permiten reconocer modelos de procesos repetitivos o cílicos.
3. BPMN es más expresivo que otras herramientas, dando soporte (total o parcial) a 48 patrones de work-flow.

Pueden revisar los siguientes enlaces para ampliar los conceptos vistos en esta unidad:

- [https://www.bizagi.com/docs/Workflow\\_Patterns\\_using\\_BizAgi\\_Process\\_Modeler\\_Esp.pdf](https://www.bizagi.com/docs/Workflow_Patterns_using_BizAgi_Process_Modeler_Esp.pdf)
- <https://grupocinco.files.wordpress.com/2007/11/bpmn.pdf>
- <http://help.bizagi.com/processmodeler/es/>



# FUNDAMENTOS DE UML

---

## LOGRO DE LA UNIDAD DE APRENDIZAJE

Al término de la unidad, el alumno plantea escenarios de negocios, diseñando modelos de Casos de uso de negocio y de actividades, con la notación UML.

## TEMARIO

### 4.1 Tema 7 : UML – Modelado de Negocios

- 4.1.1 : Nociones generales de UML
- 4.1.2 : Definiciones básicas y diagramas principales
- 4.1.3 : Introducción al Modelo de Casos de Uso de Negocio
- 4.1.4 : Elementos del Modelo de Negocio: Simbología

### 4.2 Tema 8 : UML – El Modelo de las Actividades

- 4.2.1 : Actividades y Decisiones
- 4.2.2 : Rutas concurrentes e indicaciones.
- 4.2.3 : Marcos de responsabilidad - Ejemplos

## 4.1 UML – Modelado de Negocios

### 4.1.1 Nociones generales de UML

**UML** es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

Lenguaje Unificado de Modelado (**UML**, por sus siglas en inglés, Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el **OMG** (Object Management Group).

Desde el año 2005, UML es un estándar aprobado por la **ISO** como **ISO/IEC 19501:2005** Information technology — Open Distributed Processing — Unified Modeling Language (UML) Versión 1.4.2.

UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y compuestos reciclados.

Es importante remarcar que UML es un "lenguaje de modelado" para especificar o para describir métodos o procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo.

Se puede aplicar en el desarrollo de software gran variedad de formas para dar soporte a una metodología de desarrollo de software (tal como el Proceso Unificado Racional o RUP), pero no especifica en sí mismo qué metodología o proceso usar.



Figura 64: Logo de UML.

Fuente. G. Booch et.al. "El Lenguaje Unificado de Modelado"

## Un poco de Historia

El lenguaje UML comenzó a gestarse en octubre de 1994, cuando Rumbaugh se unió a la compañía Rational fundada por Booch (dos destacados investigadores en el área de metodología del software).

El objetivo de ambos era unificar dos métodos que habían desarrollado: el método Booch y el OMT (Object Modelling Tool). El primer borrador apareció en octubre de 1995. En esa misma época otro conocido investigador, Jacobson, creador de OOSE (*Object Oriented Software Engineer*) se unió a Rational y se incluyeron ideas suyas.

Estas tres personas son conocidas como los “tres amigos”.

Además, este lenguaje se abrió a la colaboración de otras empresas para que aportaran sus ideas. Todas estas colaboraciones condujeron a la definición de la primera versión de UML.

	<p><b>Grady Booch</b>          Es un diseñador de software, un metodologista de software y entusiasta de diseño de patrones. Es director científico de Rational Software y editor de una serie de Benjamin/Cummings. (<a href="http://es.wikipedia.org/wiki/Grady_Booch">http://es.wikipedia.org/wiki/Grady_Booch</a>)</p> <p><b>Fecha de nacimiento:</b> 27 de febrero de 1955 (edad 58), Texas, Estados Unidos</p> <p><b>Educación:</b> Universidad de California en Santa Bárbara, Academia de la</p>
	<p><b>Ivar Jacobson</b>          Ingeniero sueco en Ciencias de la computación. Inventó el diagrama de secuencia y desarrolló los diagramas de colaboración. (<a href="http://es.wikipedia.org/wiki/Ivar_Jacobson">http://es.wikipedia.org/wiki/Ivar_Jacobson</a>)</p> <p><b>Fecha de nacimiento:</b> 2 de septiembre de 1939 (edad 73), Ystad, Suecia</p> <p><b>Educación:</b> Real Instituto de Tecnología, Chalmers University of Technology.</p>
	<p><b>James Rumbaugh</b>          Científico de la computación y un metodologista de objeto. Es mejor conocido por su trabajo en la creación de la Técnica de Modelado de Objetos y el Lenguaje Unificado de Modelado. <a href="http://es.wikipedia.org/wiki/James_Rumbaugh">http://es.wikipedia.org/wiki/James_Rumbaugh</a></p> <p><b>Fecha de nacimiento:</b> 24 de septiembre de 1947 (edad 65), Bethlehem, Pensilvania, Estados Unidos</p> <p><b>Educación:</b> Instituto Tecnológico de Massachusetts, Instituto de Tecnología de California</p>

**Figura 65: “Los tres amigos”**  
 Fuente. Booch et al. "El Lenguaje Unificado de Modelado"

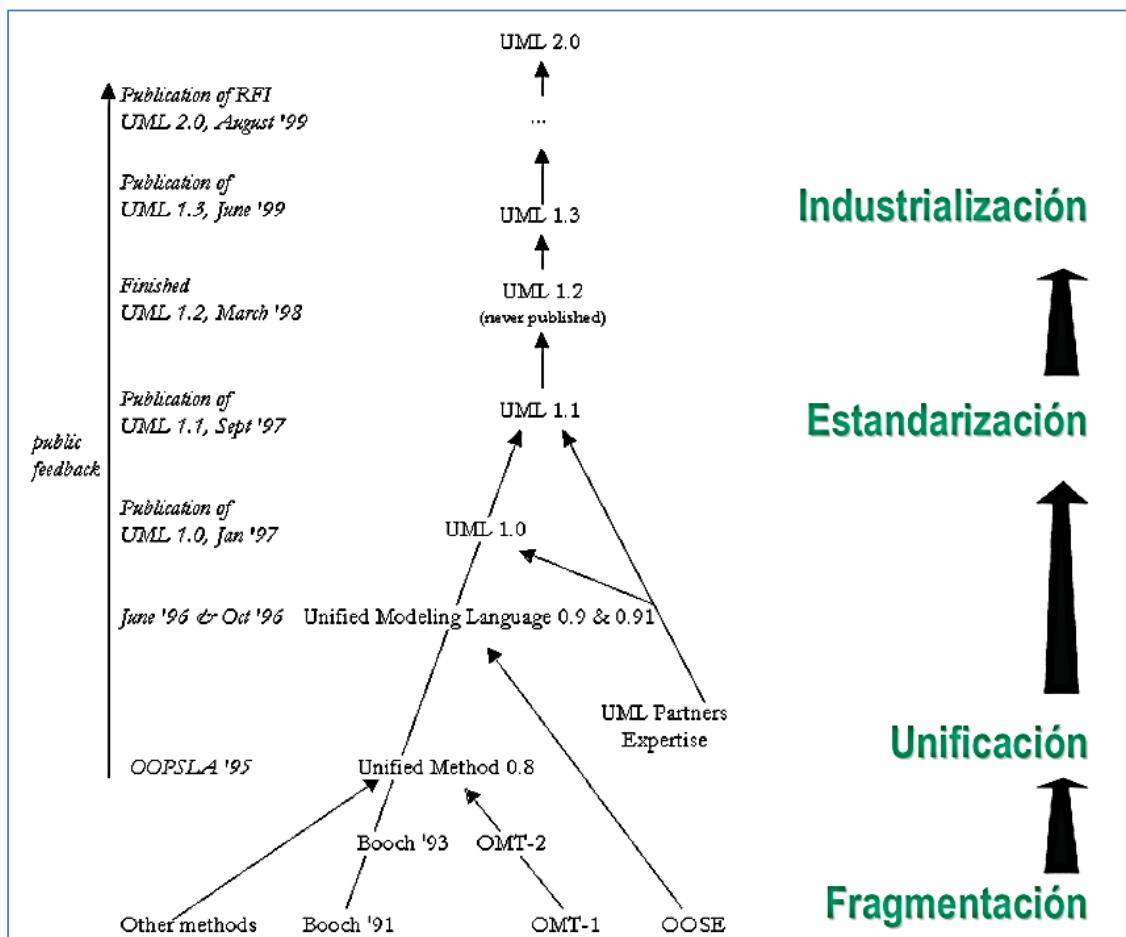


Figura 66: Evolución de UML  
Fuente. Booch et al. "El Lenguaje Unificado de Modelado"

#### 4.1.2 Definiciones básicas y diagramas principales

##### DIAGRAMAS UML

En la **versión 2** de UML, se agregaron 3 tipos de diagramas (el de estructura compuesta, el de interacción y el de tiempo) a los 11 ya existentes en la versión 1. Así que tenemos 14 tipos de diagramas diferentes (13 si no tenemos en cuenta el Diagrama de colaboración que no forma parte de UML 2.x).

Los diferentes tipos de diagramas se dividen en dos grupos, los que describen la estructura del sistema y los que describen su comportamiento. Dentro de los de comportamiento, hay a su vez un subgrupo con los de interacción.

Vamos a repasarlos.

## I) DIAGRAMAS DE ESTRUCTURA

**Diagrama de clases:** Describe los diferentes tipos de objetos en un sistema y las relaciones existentes entre ellos. Dentro de las clases, muestra las propiedades y operaciones, así como las restricciones de las conexiones entre objetos.

**Diagrama de objetos:** (También llamado Diagrama de instancias) Foto de los objetos en un sistema en un momento del tiempo.

**Diagrama de paquetes:** Muestra la estructura y dependencia entre paquetes, los cuales permiten agrupar elementos (no solamente clases) para la descripción de grandes sistemas.

**Diagrama de despliegue:** Muestra la relación entre componentes o subsistemas software y el hardware donde se despliega o instala.

**Diagrama de estructura compuesta:** Descompone jerárquicamente una clase mostrando su estructura interna.

**Diagrama de componentes:** Muestra la jerarquía y relaciones entre componentes de un sistema software.

## II) DIAGRAMAS DE COMPORTAMIENTO

**Diagrama de casos de uso:** Permite capturar los requerimientos funcionales de un sistema.

**Diagrama de estado:** Permite mostrar el comportamiento de un objeto a lo largo de su vida.

**Diagrama de actividad:** Describe la lógica de un procedimiento, un proceso de negocio o workflow.

**Diagramas de interacción (Subgrupo dentro de los diagramas de comportamiento):**

Describen cómo los grupos de objetos colaboran para producir un comportamiento. Son los siguientes:

**Diagrama de secuencia:** Muestra los mensajes que son pasados entre objetos en un escenario.

**Diagrama de comunicación:** Muestra las interacciones entre los participantes haciendo énfasis en la secuencia de mensajes.

**Diagrama de colaboración:** (Solamente en UML 1.X) Muestra las interacciones organizadas alrededor de los roles.

**Diagrama de (visión de conjunto o resumen de) interacción:** Se trata de mostrar de forma conjunta diagramas de actividad y diagramas de secuencia.

**Diagrama de tiempo:** Pone el foco en las restricciones temporales de un objeto o un conjunto de objetos.

## ESTRUCTURA DEL UML.

UML es una notación estándar para modelar sistemas, desde diferentes vistas o niveles de abstracción, que pueden ir desde la definición del problema (casos de uso), la vista lógica (clases, objetos), la vista de procesos (comportamiento), implementación y hasta distribución.

Es tan amplia que incluso ayuda a entender procesos de negocio complejos, lo cual lo convierte en una buena herramienta de comunicación entre las diferentes capas, participantes y clientes de un proyecto

UML se compone de muchos elementos de esquematización que representan las diferentes partes de un sistema de software.

Los elementos UML se utilizan para crear diagramas, que representan alguna parte o punto de vista del sistema. En UML existen cinco vistas que permiten, visualizar, especificar, construir y documentar la arquitectura del software.

UML permite representar cada vista mediante un conjunto de diagramas, tal y como se describe a continuación:



Figura 67: Estructura del UML 2.0

Fuente: <http://www.que-informatica.com/.../arquitectura-uml.jpg>

1. **Vista de Casos de uso:** muestra la funcionalidad del sistema desde el punto de vista de un actor externo que interactúa con él. Esta vista es útil para clientes, diseñadores y desarrolladores. Uno de sus diagramas es:

### 1. Diagrama de Casos de Uso:

1. Usados Para Comunicarse con el usuario final y el Experto de dominio:
  1. Proporciona credibilidad en una etapa inicial del desarrollo del sistema.
  2. Asegura una comprensión mutua de los requisitos.
2. Usados para Identificar:
  1. Quién interactuará con el sistema
  2. y qué deberá hacer el sistema
  3. Qué interfaz deberá tener el sistema
3. Usados Para Verificar:
  1. Que se hayan capturado todos los requerimientos
  2. Que los desarrolladores hayan entendido los requerimientos

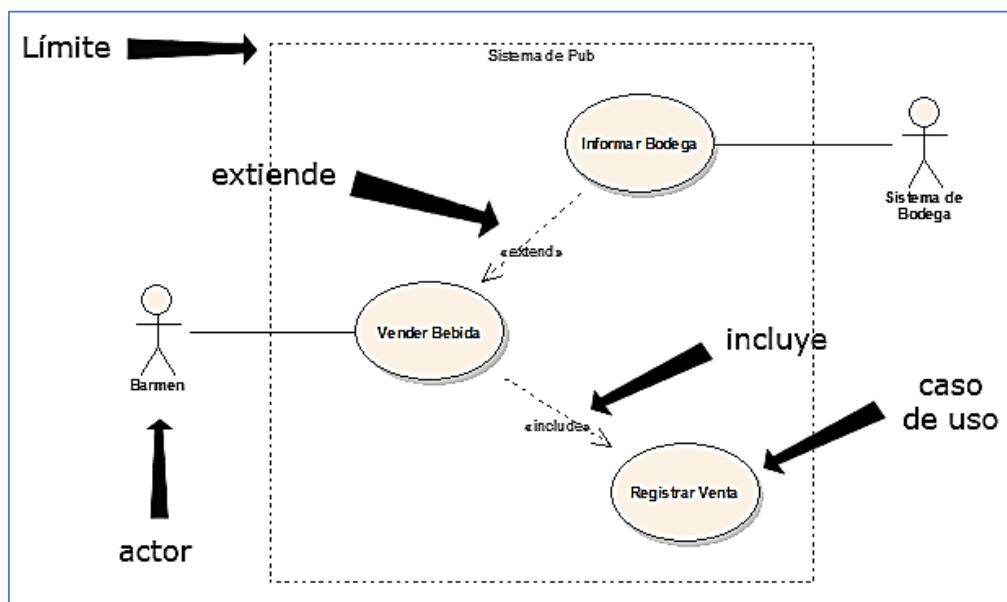


Figura 68: Diagrama de Casos de Uso

Fuente. Elaboración propia

2. **Vista de Diseño:** muestra la funcionalidad del diseño dentro del sistema en términos de la estructura estática y comportamiento dinámico del sistema. Esta vista es útil para diseñadores y desarrolladores. Uno de sus Diagramas es:

### 1. Diagrama de Clases:

1. Usados para mostrar la Estructura Estática de un sistema computacional o una parte relevante del mundo real.
2. Son los diagramas más frecuentemente usados y se les puede considerar con tres perspectivas posibles:
  1. Conceptual – muestra las entidades del mundo real con sus relaciones.
  2. Especificación – muestra la estructura del sistema o sus partes, destacando las interfaces
  3. Implementación – el “blueprint” del código fuente.

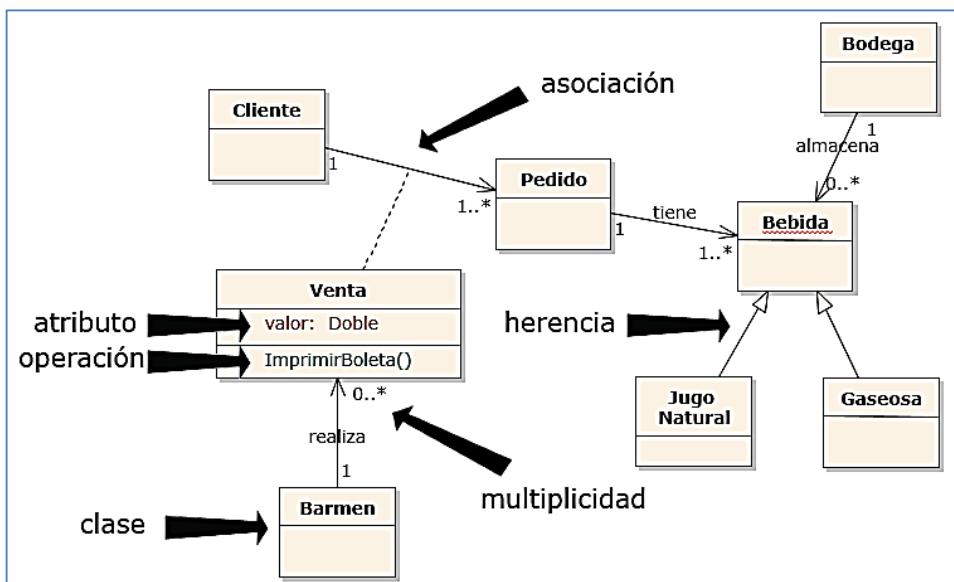


Figura 69: Diagrama de Clases (de diseño)

Fuente. Elaboración propia

3. **Vista de Interacción (procesos):** muestra la concurrencia del sistema, comunicación y sincronización. Útil para desarrolladores e integradores. Uno de sus Diagramas es:

1. **Diagrama de Secuencias:**

1. Usados para representar el comportamiento del sistema.
2. Muestran la colaboración a través de mensajes entre los objetos del sistema.
3. Destacan por:
  1. Mensajes enviados entre los objetos
  2. Orden secuencial entre los mensajes
  3. Un escenario concreto, sin condiciones
4. Útiles tanto en análisis (identificación de clases), como en diseño (especificación de componentes)

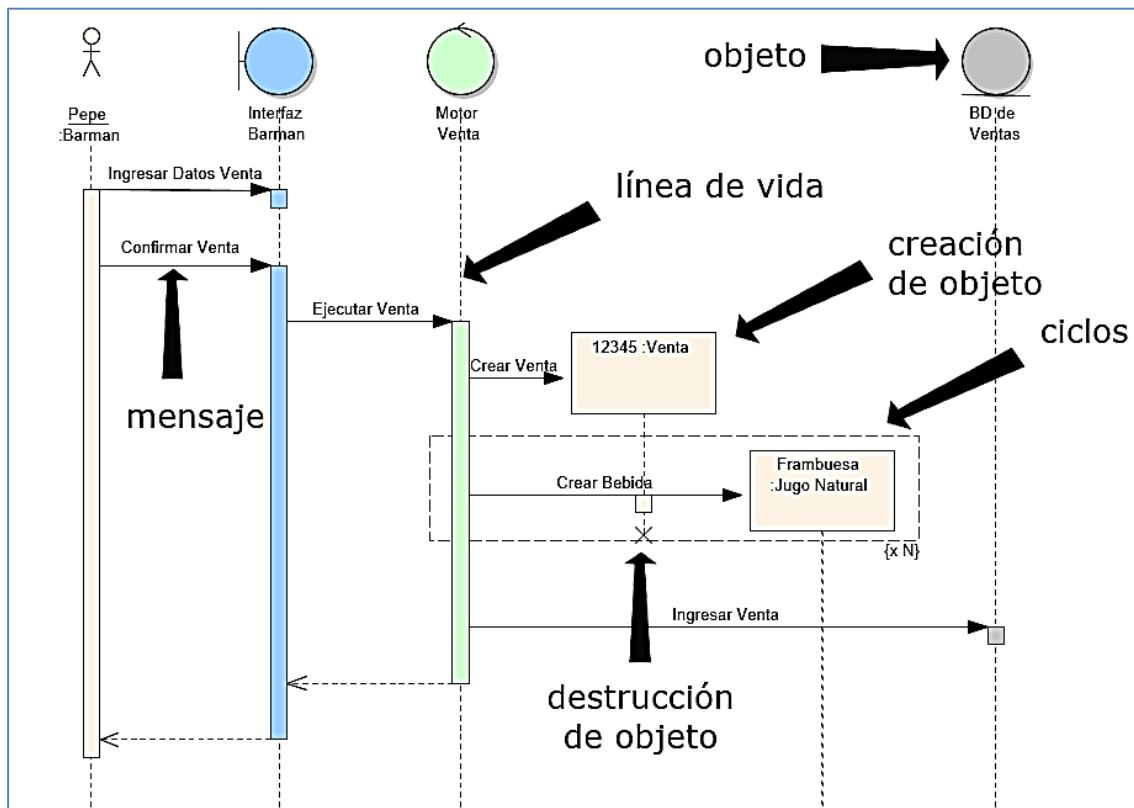


Figura 70: Diagrama de Secuencia  
Fuente. Elaboración propia

4. **Vista de Implementación:** muestra la organización de los componentes de código. Útil para los desarrolladores. Uno de sus Diagramas es:

1. **Diagrama de Componentes:**

1. Usados para mostrar los **Módulos Físicos** de software:
  1. Los ejecutables y librerías dinámicas
  2. Las páginas WEB y los scripts
  3. Los módulos o funciones, etc.
2. Sin embargo se usan más bien para capturar la **Organización de los Componentes** de Software (EXE, DLL, EJB, etc)
3. Destacan **Dependencias entre los Componentes**.

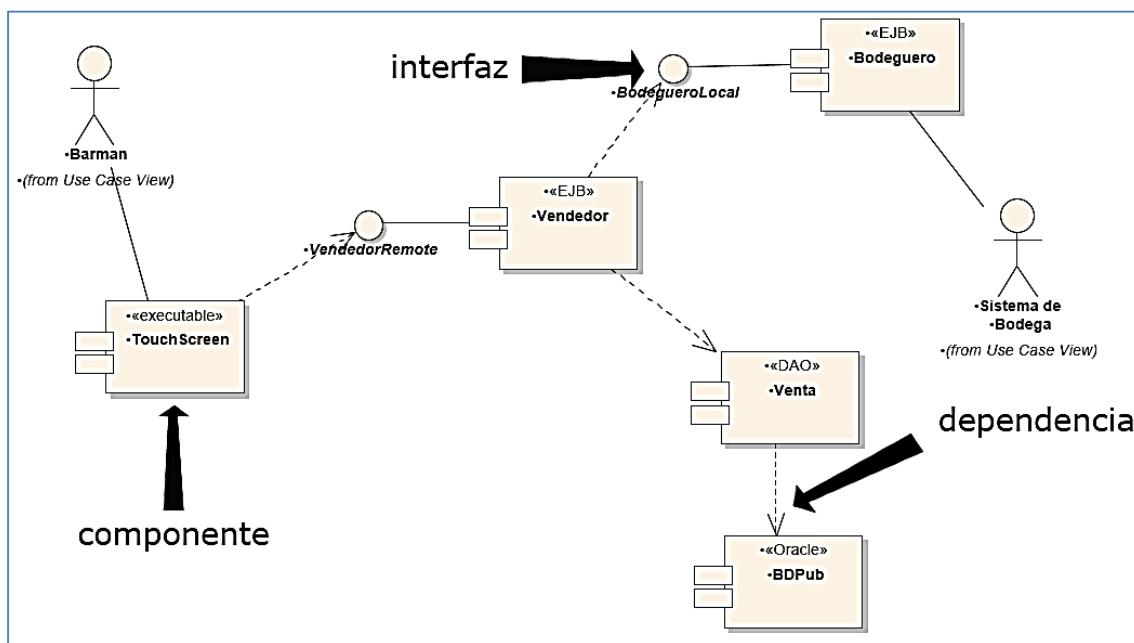


Figura 71: Diagrama de Componentes

Fuente. Elaboración propia

5. **Vista de Despliegue o implantación:** muestra la implantación del sistema en la arquitectura física. Útil para desarrolladores, integradores y verificadores. Uno de sus Diagramas es:

**1. Diagrama de Despliegue:**

1. Usados para modelar las relaciones entre el software y el hardware.
2. Mapeo de los componentes de software a los nodos de Hardware.
3. Típicamente contienen elementos tales como:
  1. Servidores
  2. Procesadores
  3. Impresoras
  4. Redes computacionales
  5. Etc.

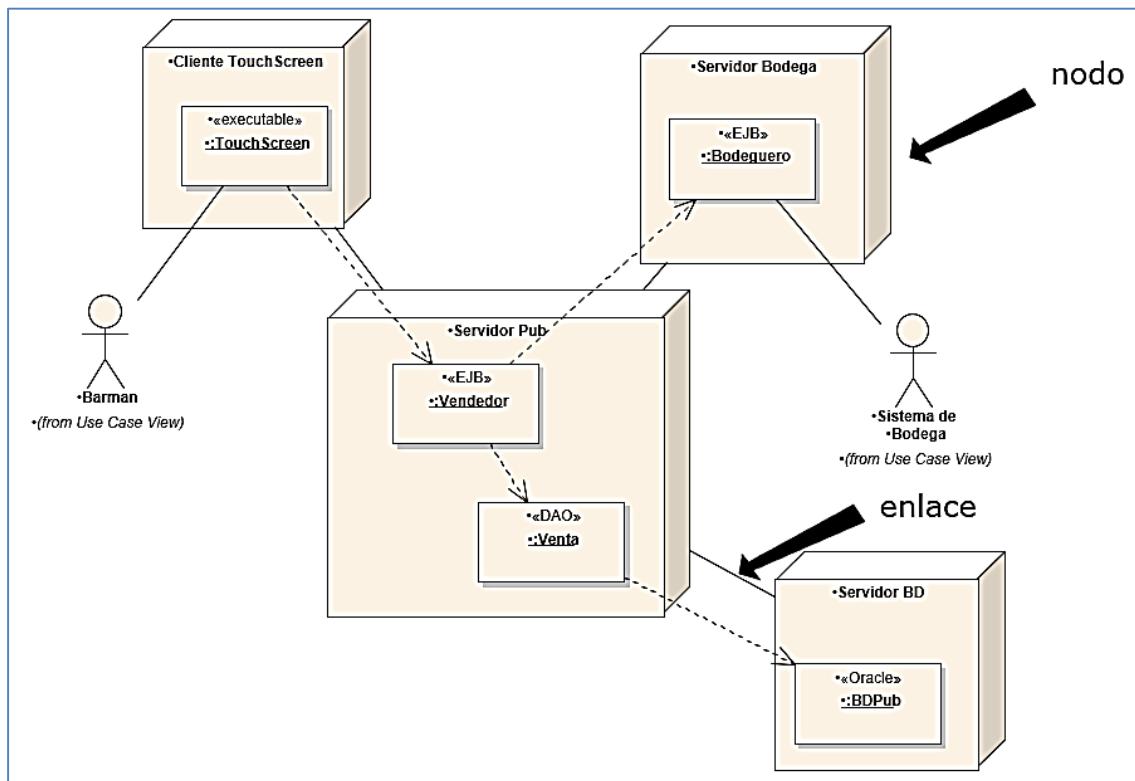


Figura 72: Diagrama de Despliegue

Fuente. Elaboración propia

#### 4.1.3 Introducción al Modelo de Casos de Uso de Negocio

##### 1. MODELADO DE NEGOCIO

La disciplina del Modelado del negocio describe la organización actual y desarrolla la visión de una nueva. Los creadores de **RUP** señalan que el modelo de negocio está soportado por dos artefactos principales:

1. Modelo de casos de uso del negocio.
2. Modelo de análisis del negocio.

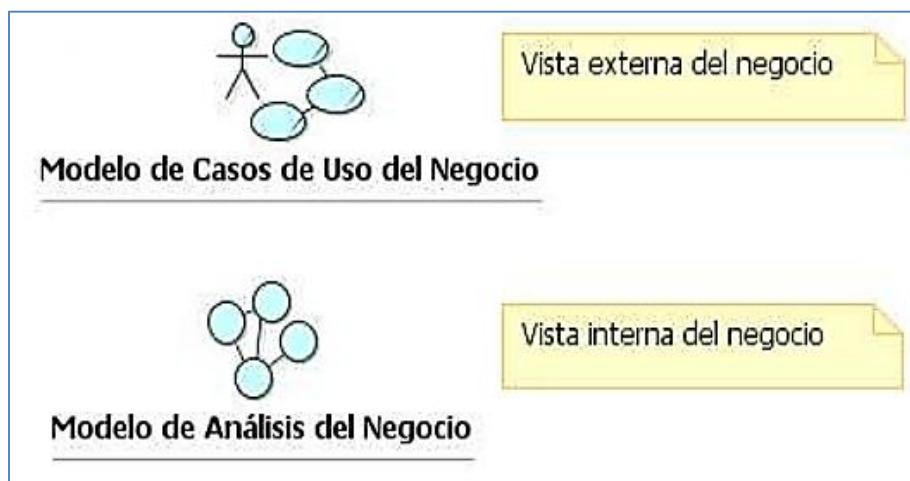


Figura 73: Vistas del Negocio

Fuente. Curso ADSI - Cibertec

### I) Modelo de casos de uso del negocio (Vista externa)

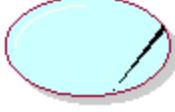
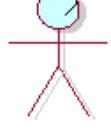
El modelo de casos de uso del negocio describe los procesos de negocio de una empresa en términos de casos de uso del negocio y actores del negocio que se corresponden con los procesos del negocio y los clientes, respectivamente.

### II) Modelo de análisis del negocio (Vista interna)

El modelo de análisis del negocio es un modelo interno a un negocio, que describe cómo cada caso de uso de negocio es llevado a cabo por un grupo de trabajadores que utilizan entidades del negocio.

#### 4.1.4 Elementos del Modelo de Negocio: Simbología

##### Notación para el Modelo de casos de uso del negocio: (Vista externa)

ARTEFACTO	DESCRIPCIÓN
 <b>Caso de Uso de Negocio</b>	<p>Describe un proceso de negocio desde un punto de vista externo que percibe algún tipo de valor.</p> <p>Es la descripción de la secuencia de acciones necesarias para entregar un producto o servicio, con valor tangible, a un consumidor (cliente).</p>
 <b>Actor de Negocio</b>	<p>Representa un rol que algo o alguien externo desempeña en relación con el negocio.</p> <p>Puede iniciar el proceso o participar en él debido a que recibirá algún resultado de valor del proceso.</p>
 <b>Objetivo de Negocio</b>	<p>Describe el valor deseado de una medida en particular a futuro, y se utiliza para planear y administrar las actividades del negocio.</p> <p>El objetivo debe ser claro, measurable, alcanzable, realista y sensible al tiempo.</p>

**Notación para el Modelo de análisis del negocio: (Vista interna)**

ARTEFACTO	DESCRIPCIÓN
 <b>Trabajador de Negocio</b>	<p>Representa un rol interno al negocio.</p> <p>Colabora con trabajadores de otro sector, es notificado de acontecimientos del negocio y manipula entidades de negocio para realizar sus responsabilidades.</p>
 <b>Entidad de Negocio</b>	<p>Ente manipulado por actores del negocio y trabajadores del negocio.</p>
 <b>Realización de Caso de Uso de Negocio</b>	<p>Colección de diagramas que muestra cómo los trabajadores del negocio y entidades del negocio llevan a cabo el caso de uso del negocio.</p> <p>Por ejemplo: diagramas de clases y diagramas de actividades para realizar el detalle de cada proceso de negocio.</p>

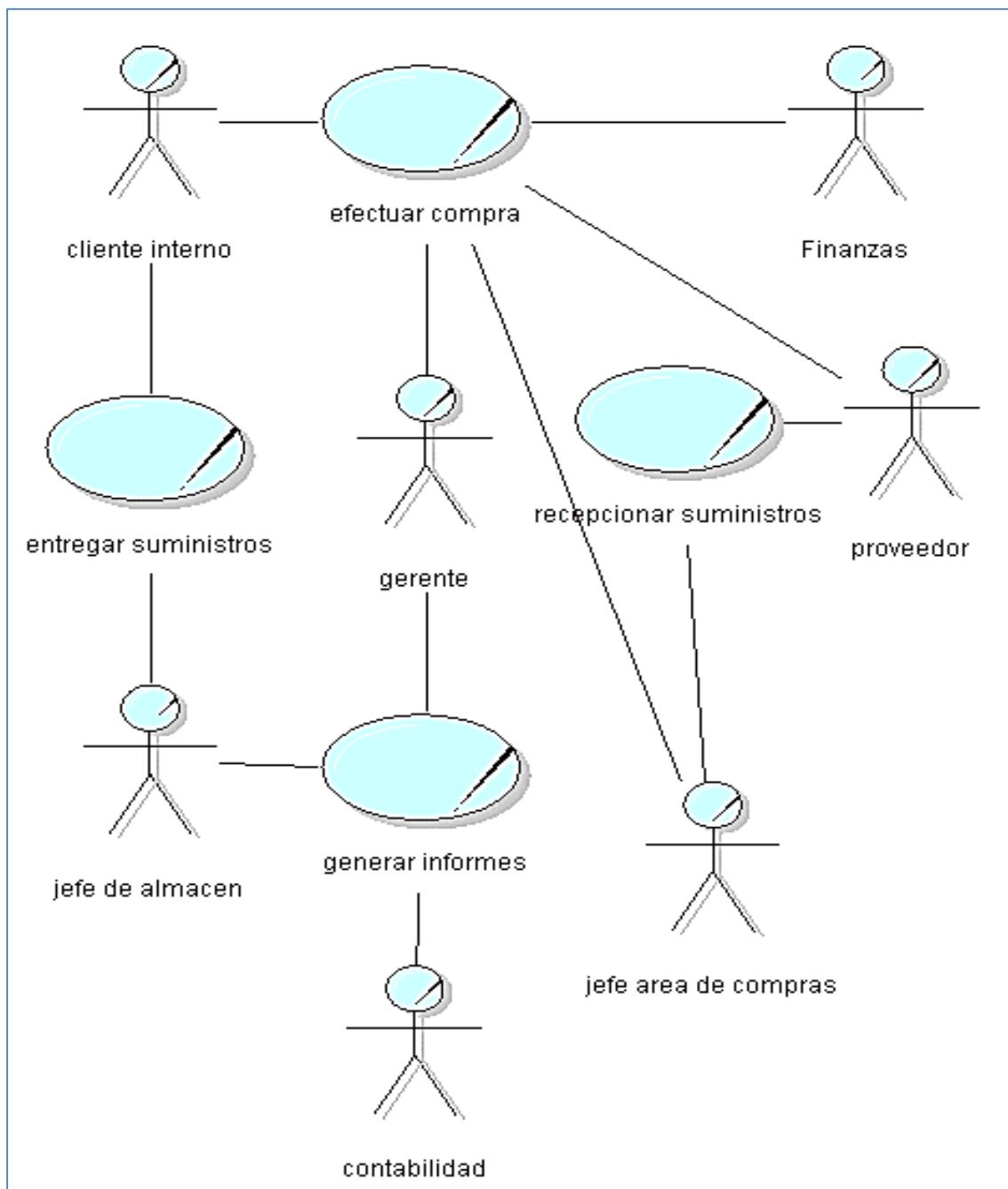


Figura 74: Ejemplo de Diagrama de Casos de Uso de Negocio

Fuente. Sabana M. – UML con RSA

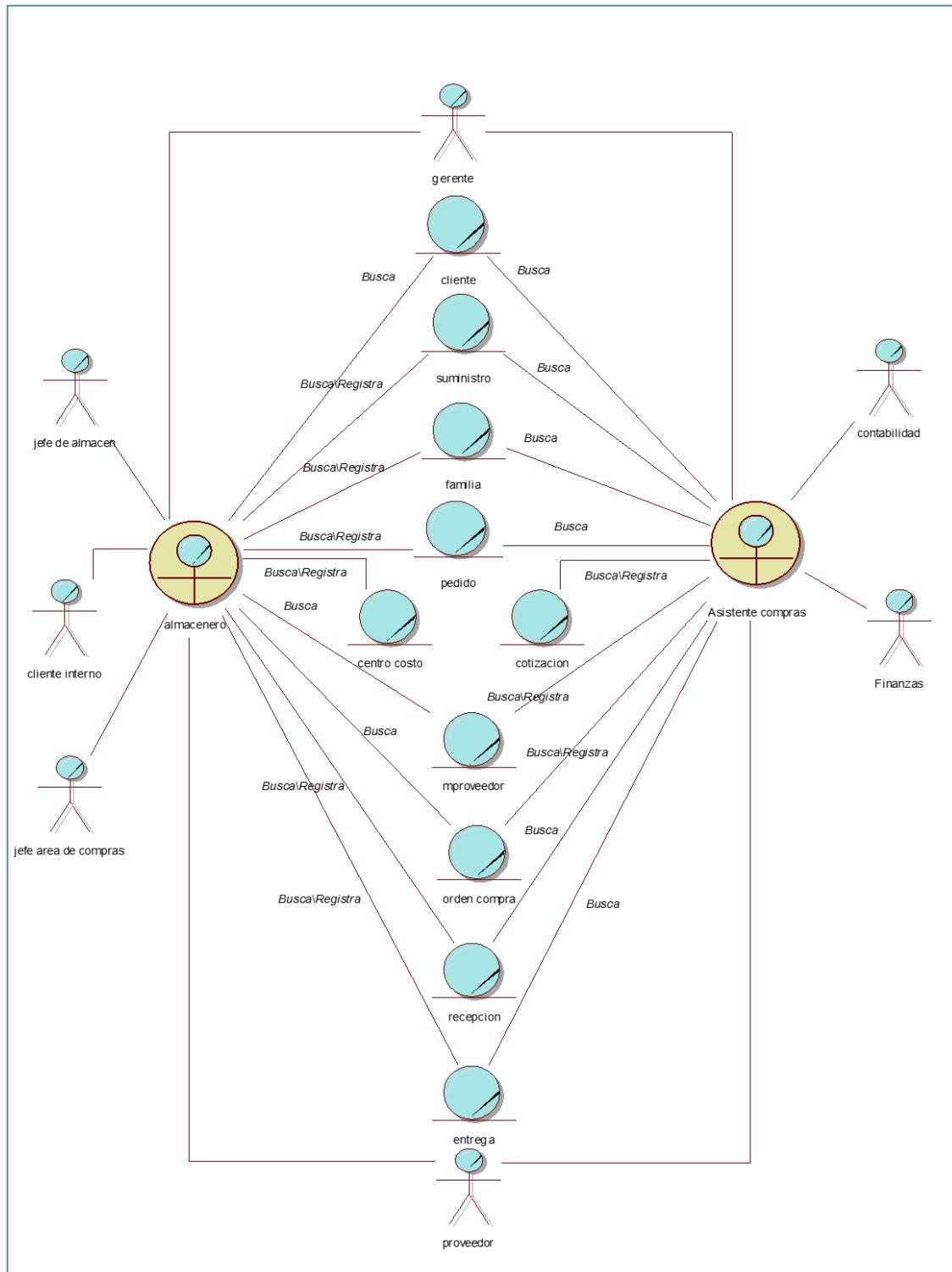


Figura 75: Ejemplo de Diagrama de Modelo de Negocio  
Fuente. Sabana M. – UML con RSA

# Resumen

1. UML significa Lenguaje Unificado de Modelado por sus siglas en inglés, Unified Modeling Language.
2. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.
3. UML 2.0 ofrece 14 tipos de Diagramas diferentes clasificados en dos categorías: Diagramas de Estructura y Diagramas de Comportamiento.
4. La arquitectura de un Sistema (UML 2.0) se puede especificar a través de 5 Vistas:
  1. Vista de Casos de Uso
  2. Vista de Diseño
  3. Vista de Interacción
  4. Vista de Implementación
  5. Vista de Despliegue
6. El modelo de negocio está soportado por dos artefactos principales:
  1. Modelo de casos de uso del negocio.
  2. Modelo de análisis del negocio.

Pueden revisar los siguientes enlaces para ampliar los conceptos vistos en esta unidad:

1. [http://www.epidataconsulting.com/tikiwiki/tiki-read\\_article.php?articleId=15](http://www.epidataconsulting.com/tikiwiki/tiki-read_article.php?articleId=15)
2. [http://www.epidataconsulting.com/tikiwiki/tiki-read\\_article.php?articleId=31](http://www.epidataconsulting.com/tikiwiki/tiki-read_article.php?articleId=31)
3. [https://books.google.com.pe/books?id=rXU-WS4UatYC&pg=PA340&lpg=PA340&dq=modelo+de+casos+de+uso+de+negocio&source=bl&ots=vvsLzd0n\\_V&sig=4L7BylTN-3kEE-oO2cMeVOXVi5E&hl=es&sa=X&ved=0CEEQ6AEwBjgKahUKEwiA88bF4YbHAhWGIA0KhbDNAcY#v=onepage&q=modelo%20de%20casos%20de%20uso%20de%20negocio&f=false](https://books.google.com.pe/books?id=rXU-WS4UatYC&pg=PA340&lpg=PA340&dq=modelo+de+casos+de+uso+de+negocio&source=bl&ots=vvsLzd0n_V&sig=4L7BylTN-3kEE-oO2cMeVOXVi5E&hl=es&sa=X&ved=0CEEQ6AEwBjgKahUKEwiA88bF4YbHAhWGIA0KhbDNAcY#v=onepage&q=modelo%20de%20casos%20de%20uso%20de%20negocio&f=false)

## 4.2 UML – El Modelo de las Actividades

El diagrama de actividades ha sido diseñado para mostrar una visión simplificada de lo que ocurre durante una operación o proceso. Es una extensión de un diagrama de estados. El diagrama de estados muestra los estados de un objeto y representa las actividades como flechas que conectan a los estados. El diagrama de actividades resalta, precisamente a las actividades.

Es una notación que forma parte de UML y que se utiliza principalmente para modelar procesos de negocio, especificando:

- La secuencia de actividades que componen los procesos de negocio.
- Los actores que realizan las actividades (opcional).
- La información que fluye de unas actividades a otras (opcional).

Dentro del proceso de ingeniería de requisitos, se utilizarán para modelar los procesos de negocio, tanto actuales como a implantar, de la organización para la que se va a desarrollar el sistema software.

A partir del modelo del negocio al que el sistema software debe dar soporte, se plantean los objetivos y requisitos del sistema a desarrollar.

A cada actividad se le representa por un rectángulo con las esquinas redondeadas (más angosto y ovalado que la representación del estado).

### 4.2.1 Actividades y decisiones.

Una actividad representa un paso dentro de proceso de negocio.

- Su nombre, que debe ser siempre una forma verbal, debe ser representativo y coherente dentro del proceso de negocio.

**Actividad**

- Si una actividad es compleja, puede ser necesario mostrar su descomposición en actividades más simples en otro diagrama.

**Actividad  
compleja**

- En cada diagrama de actividades, las actividades deben tener un nivel de abstracción similar.

#### Actividades iniciales y finales:



La actividad inicial, que debe ser única, indica dónde comienza el proceso de negocio.

Una actividad final, de las que puede haber varias o ninguna (proceso sin fin), indica dónde puede terminar el proceso de negocio.



#### Transiciones:

Indican la secuencia de actividades que componen el proceso de negocio.



Cuando una actividad termina de realizarse se produce la transición hacia la siguiente actividad.

#### Decisiones:

Indican que la siguiente actividad a realizar depende de cierta condición.

Como mínimo y como máximo, solo puede haber una opción válida al evaluar la condición.

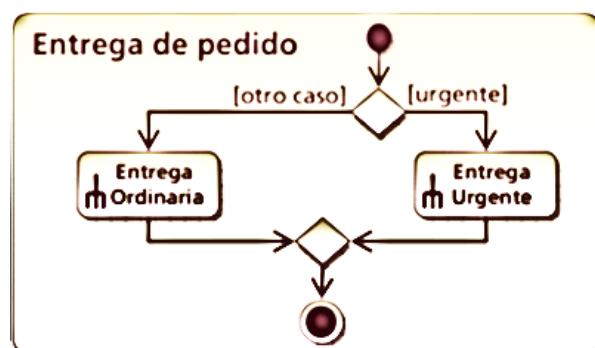


Figura 76: Ejemplo de Decisiones  
Fuente. C.Larman – Applying UML and Patterns

El símbolo de condición se puede usar también para unir varios caminos condicionales (opcional).

Casi siempre una secuencia de actividades, llegará a un punto donde se realizará alguna decisión. Ciertas condiciones le llevarán por un camino y otras por otro (pero ambas son mutuamente exclusivas).

Podrá representar un punto de decisión de una de dos formas: la primera es mostrar las rutas posibles que parten directamente de una actividad y la segunda es llevar la transición hacia un rombo - reminiscencias del símbolo de decisión en un diagrama de flujo- y que de allí salgan las rutas de decisión (como usuario de los antiguos diagramas de flujo, prefiero la segunda opción). De cualquier forma, indicará la condición con una instrucción entre corchetes junto a la ruta correspondiente.

#### 4.2.2 Rutas concurrentes e indicaciones (Paralelismo)

A veces, algunos pasos de un proceso de negocio se realizan simultáneamente (en paralelo) o sin un orden definido.

Para indicar que comienzan varias actividades a la vez se usa un símbolo de comienzo de paralelismo (fork), al que llega una transición y del que salen varias (al menos dos).

Para indicar que todas las actividades que se hacían en paralelo han terminado se usa un símbolo de fin de paralelismo (join), al que llegan varias transiciones (al menos dos) y del que sale una sola transición.

La transición de salida del join solo se realiza cuando han terminado todas las actividades que se realizaban en paralelo.

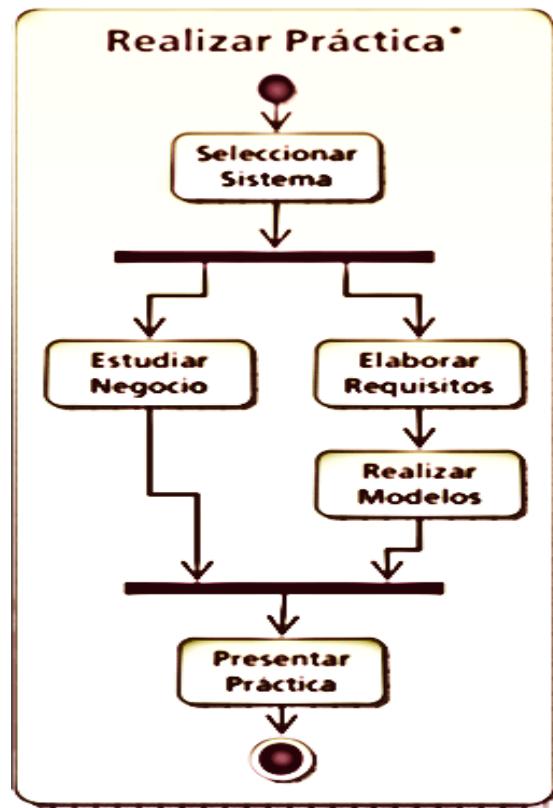


Figura 77: Ejemplo de Paralelismo  
 Fuente. C.Larman – Aplying UML and Patterns

Conforme modele actividades tendrá la oportunidad de separar una transición en dos rutas que se ejecuten al mismo tiempo (es decir, de forma concurrente) y luego se reúnan. Para representar esta división, utilizará una línea gruesa perpendicular a la transición y las rutas partirán de ella. Para representar la reincorporación, ambas rutas apuntarán a otra línea gruesa

#### Indicaciones:

Durante una secuencia de actividades, es posible enviar una indicación. Cuando se reciba, la indicación provocará que se ejecute una actividad. El símbolo para enviar una indicación es un pentágono convexo, y el que la recibe es un pentágono cóncavo.

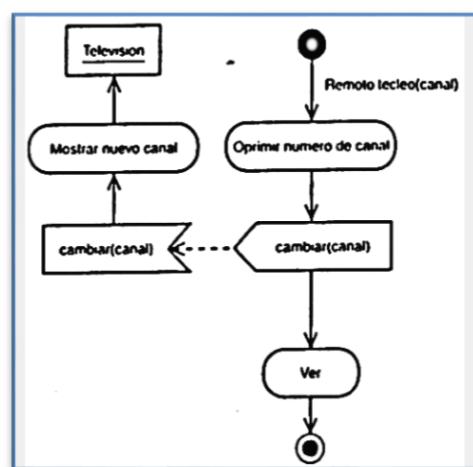


Figura 78: Ejemplo de Indicaciones  
 Fuente. C.Larman – Aplying UML and Patterns

#### 4.2.3 Marcos de Responsabilidad (Swim lines)

Uno de los aspectos más importantes del Diagrama de Actividades es su facultad para expandirse y mostrar quien tiene las responsabilidades en el proceso. Para ello separará el Diagrama en segmentos paralelos donde se muestra el nombre de un responsable en la parte superior, y presenta las actividades de cada uno.

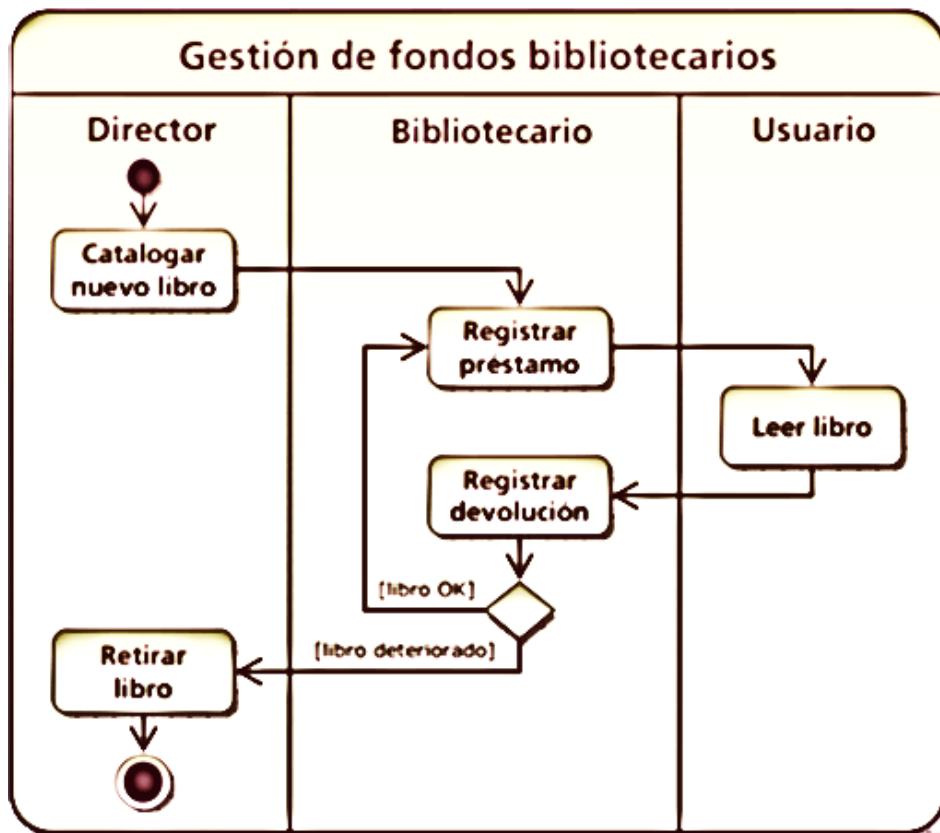
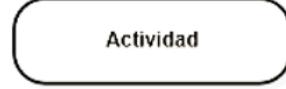


Figura 79: Ejemplo de Marcos de Responsabilidad

Fuente. C.Larman – Applying UML and Patterns

**Simbología:**

Símbolo	Significado
 <b>Inicio</b>	Punto en donde se inician las actividades
 <b>Actividad</b>	Acción o conjunto de acciones a ejecutarse
 <b>Barra de sincronización</b>	Indica el comienzo (fork) o sincronización (join) de actividades concurrentes.
 <b>Transición</b>	Denota traspaso del control desde una actividad a otra.
 <b>Decisión</b>	Flujos de control alternativos
 <b>Fin</b>	Denota término de las actividades.

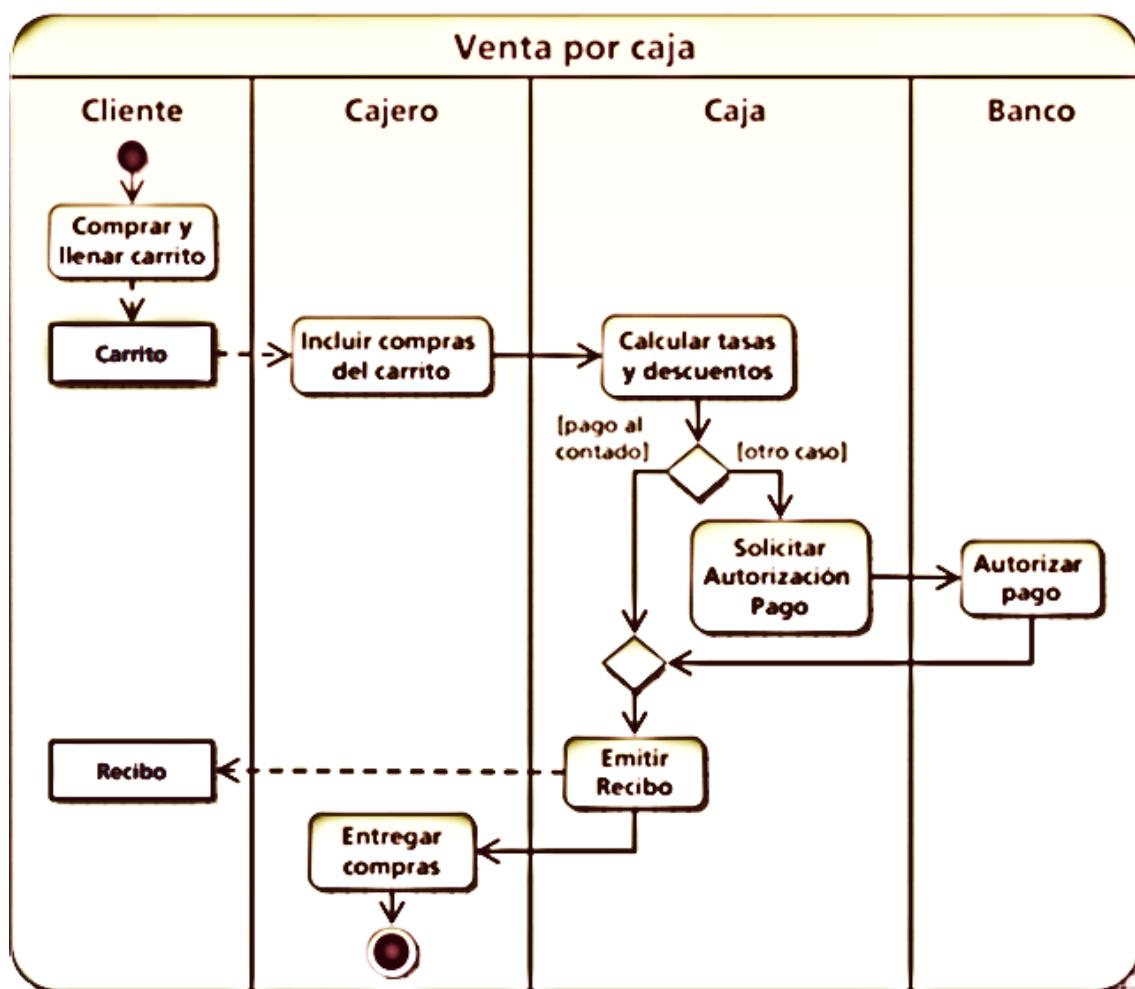


Figura 80: Ejemplo de Diagrama de Actividades

Fuente. C.Larman – Applying UML and Patterns

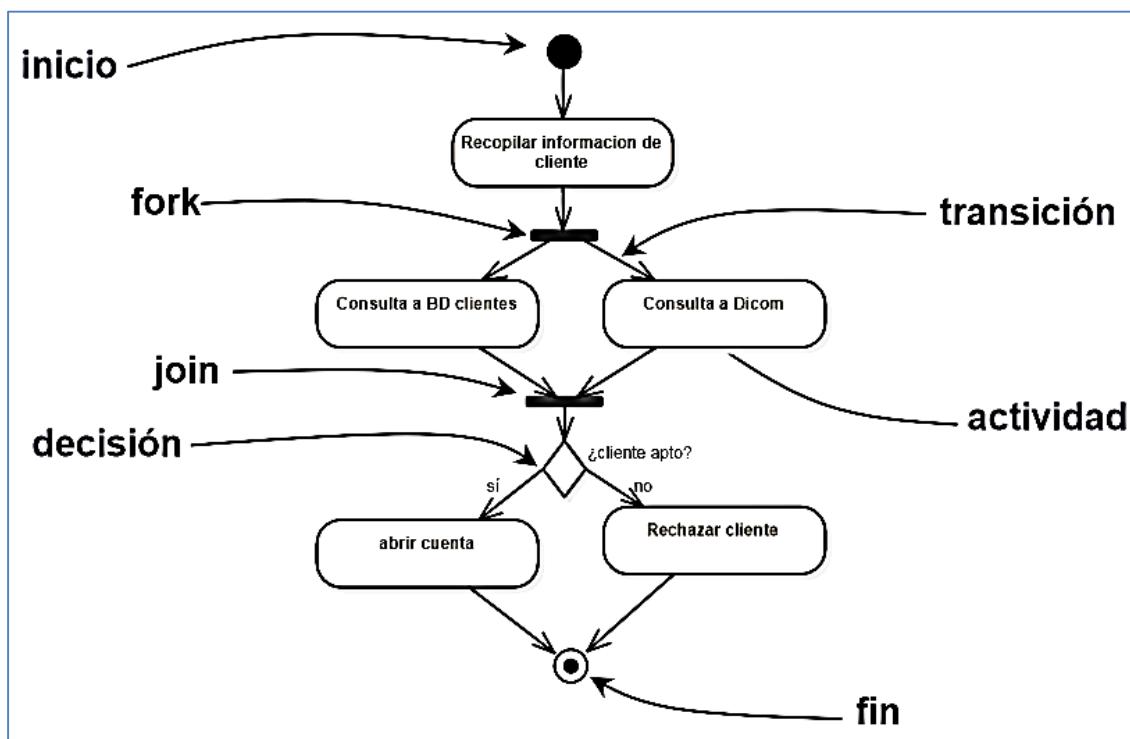


Figura 81: Elementos de un Diagrama de Actividades

Fuente. C.Larman – Applying UML and Patterns

# Resumen

1. El Diagrama de Actividades es una notación que forma parte de UML y que se utiliza principalmente para modelar procesos de negocio.
2. Se utiliza para modelar los procesos de negocio, tanto actuales como a implantar, de la organización para la que se va a desarrollar el sistema software.
3. Los elementos principales de un Diagrama de Actividades son:
  1. Actividades iniciales y finales
  2. Transiciones
  3. Decisiones lógicas.
  4. Rutas concurrentes
  5. Indicaciones.
  6. Marcos de responsabilidad.
7. Es fácil predecir que UML será el lenguaje de modelado de software de uso universal. Las principales razones para ello son:
  1. En el desarrollo, han participado investigadores de reconocido prestigio.
  2. Ha sido apoyado por prácticamente todas las empresas importantes de informática.
  3. Se ha aceptado como un estándar por la OMG.
  4. Prácticamente todas las herramientas CASE y de desarrollo la han adaptado como lenguaje de modelado.
  5. En resumen, UML resuelve de forma bastante satisfactoria un viejo problema del desarrollo de software como es su modelado gráfico. Además, se ha llegado a una solución unificada basada en lo mejor que había hasta el momento, lo cual lo hace todavía más excepcional.

Pueden revisar los siguientes enlaces para ampliar los conceptos vistos en esta unidad:

1. <http://iearobotics.com/alberto/lib/exe/fetch.php?media=teaching:uml.pdf>
2. <https://drive.google.com/file/d/0BwWwTdM7msEKMXVSOHpOc05wNTg/view?pli=1>
3. <http://elvex.ugr.es/decsai/java/pdf/3E-UML.pdf>
4. [http://ocw.unizar.es/ciencias-experimentales/modelos-matematicos-en-bases-de-datos/uml/02UML\\_DiagramaActividades.pdf](http://ocw.unizar.es/ciencias-experimentales/modelos-matematicos-en-bases-de-datos/uml/02UML_DiagramaActividades.pdf)

# Anexos

MEJORES PRÁCTICAS

EN EL MODELADO DE PROCESOS

## MEJORES PRÁCTICAS EN EL MODELADO DE PROCESOS

El estándar de Modelo y Notación de Procesos de Negocio (BPMN; por sus siglas en inglés) proporciona a las organizaciones la capacidad de comprensión de sus procesos internos de negocio en una notación gráfica y la capacidad de comunicar sus procedimientos de manera estándar.

Sin embargo, el uso del estándar no garantiza que los procesos se modelen de forma clara y eficaz; la forma en que los modeladores interpretan las condiciones de negocio y cómo definen su estructura, es crucial para asegurar que se entienden correctamente.

Esta sección proporciona modeladores de procesos algunas pautas para construir modelos claros y eficaces compatibles con el estándar BPMN.

### Principios de modelado BPMN

Cuando se definen los diagramas de proceso que debe tomar en cuenta los siguientes principios básicos:

1. Mantenga una secuencia lógica y clara
2. Utilice el estándar BPMN
3. Utilice un etiquetado estricto
4. Simplifique los diagramas

A continuación, encontrará consejos útiles para seguir estos principios y ayudar a la definición correcta procesos y su comunicación.

#### 1. Mantenga una secuencia lógica y clara

Esto pareciera ser obvio, pero es uno de los errores más comunes en el modelado de procesos. Los diagramas pueden se pueden difíciles de leer y muy confusos cuando la lógica de proceso no es explícita y clara.

Las siguientes técnicas le ayudarán a mantener una secuencia lógica y clara en sus modelos.

Definir un comienzo y un final claro

En BPMN, comienzan y los eventos extremos son opcionales. Sin embargo, los procesos con los eventos de inicio y fin implícitos son indeseables y podrían dar lugar a malas interpretaciones.

Utilice eventos de inicio y final de cada proceso y subproceso para representar su comienzo y finalización.

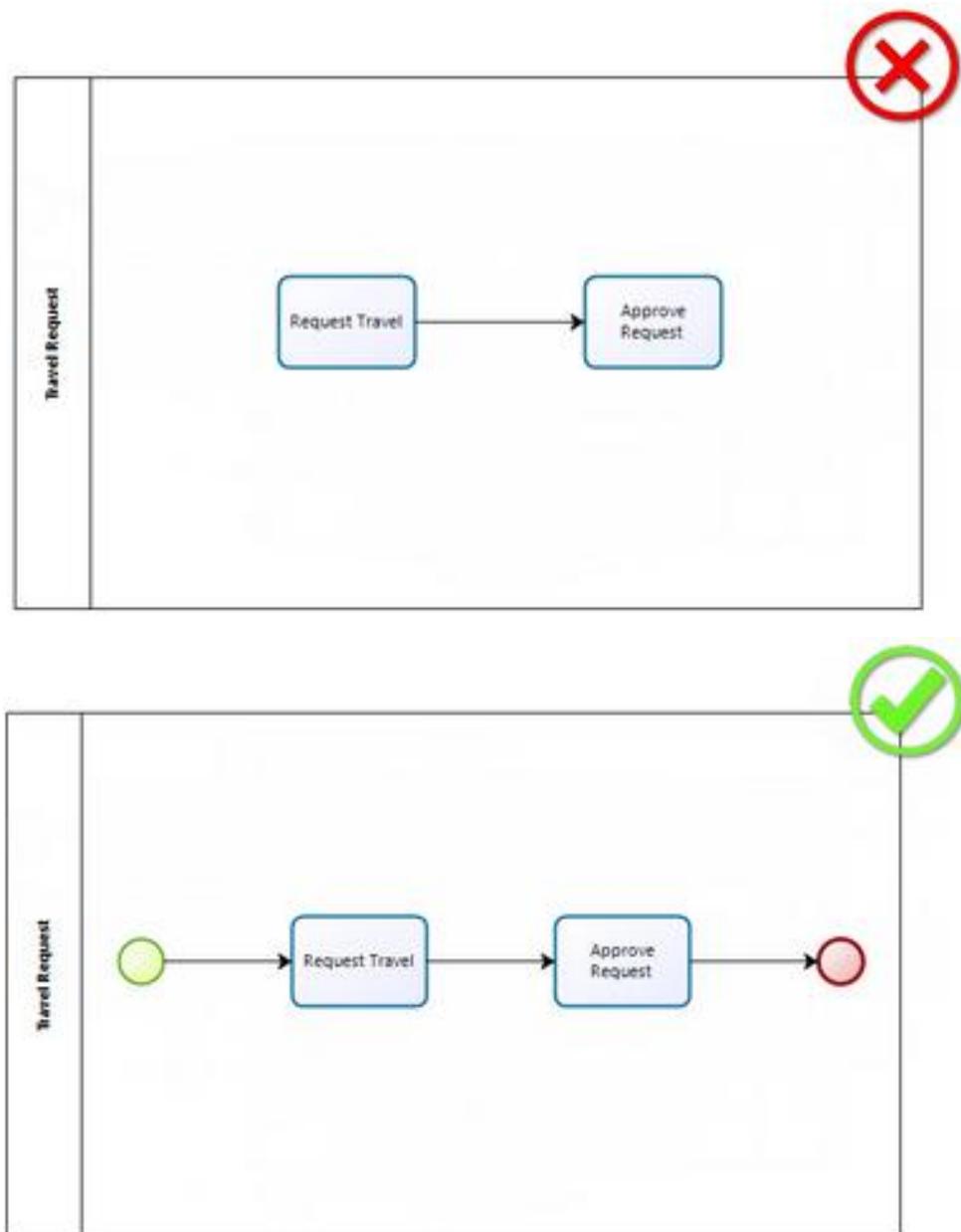


Figura 82: Comienzo y final claros  
Fuente.- Bizagi BPM Suite: Guía del Usuario

Siga una dirección consistente del flujo

Haga visible la lógica del proceso en el diagrama. Evite las líneas cruzadas (conectores), mantenga una secuencia de tiempo y una dirección de flujo constante. La lectura diagrama será más fácil y su comunicación eficiente.

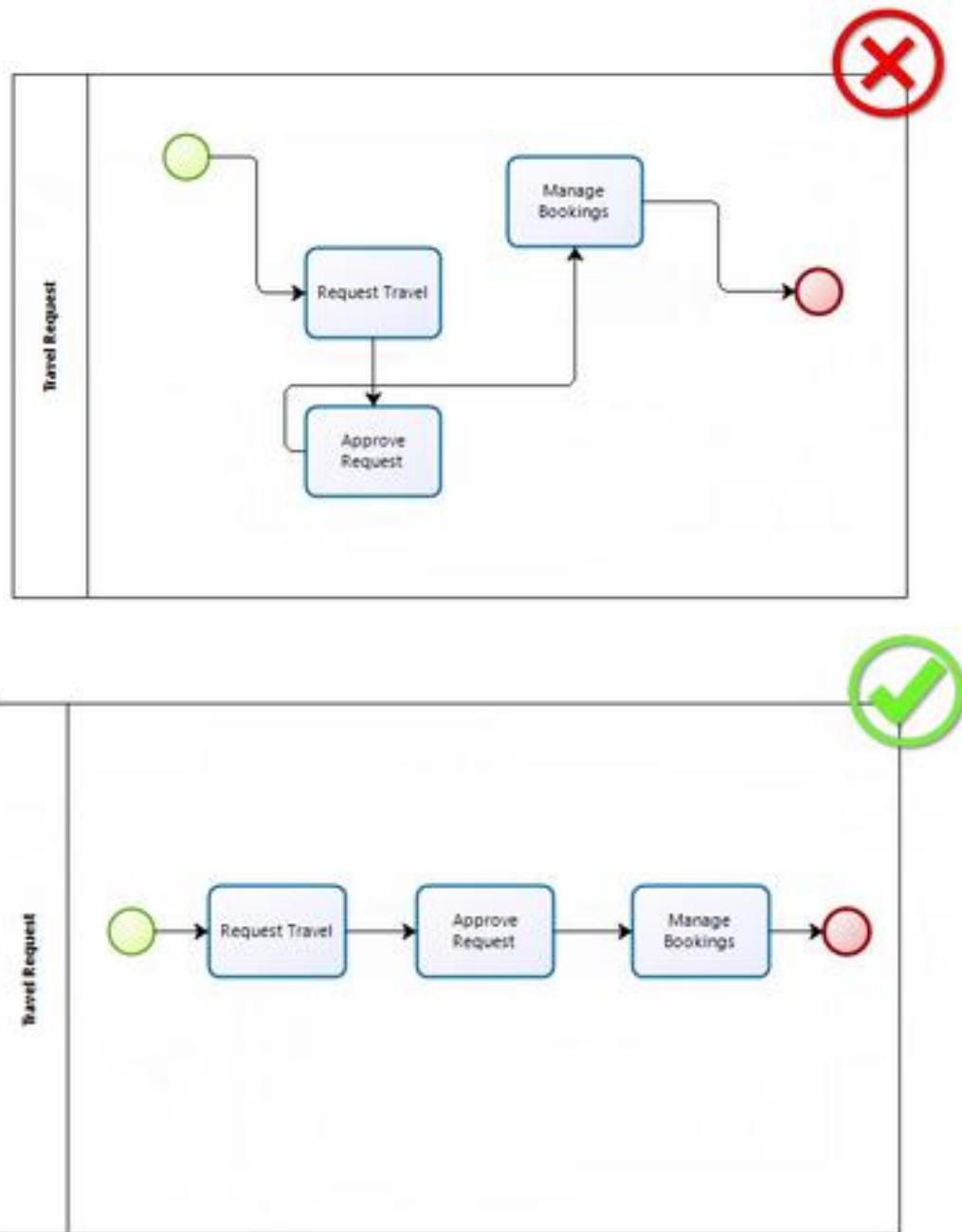


Figura 83: Dirección consistente de flujo  
Fuente.- Bizagi BPM Suite: Guía del Usuario

Mantenga claro el escenario principal

El escenario principal debe ser fácilmente identificado al leer el diagrama. Diagrama el escenario principal primero y luego los flujos alternativos.

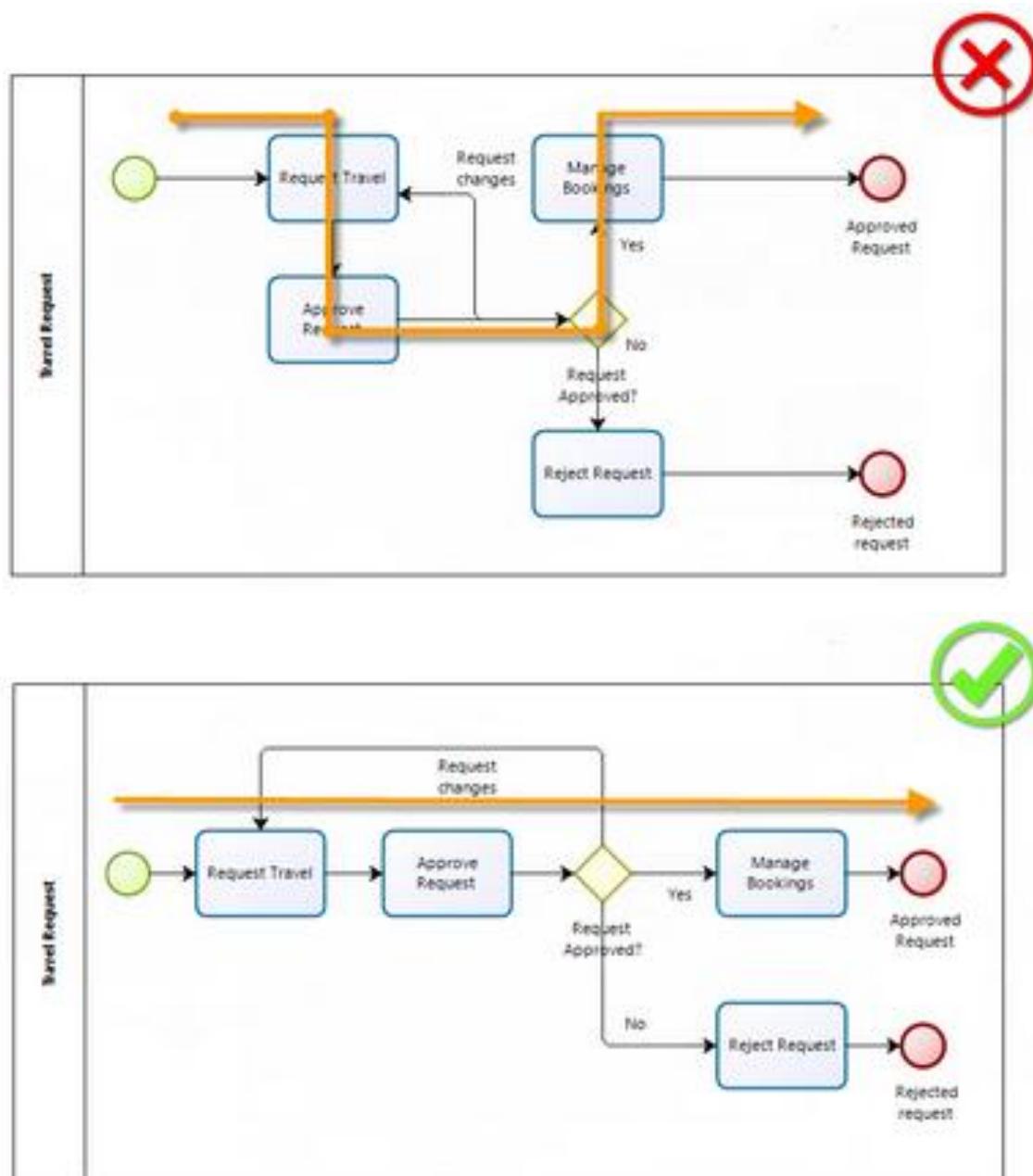


Figura 84: escenario principal claro  
Fuente.- Bizagi BPM Suite: Guía del Usuario

Mantenga claros los escenarios alternativos

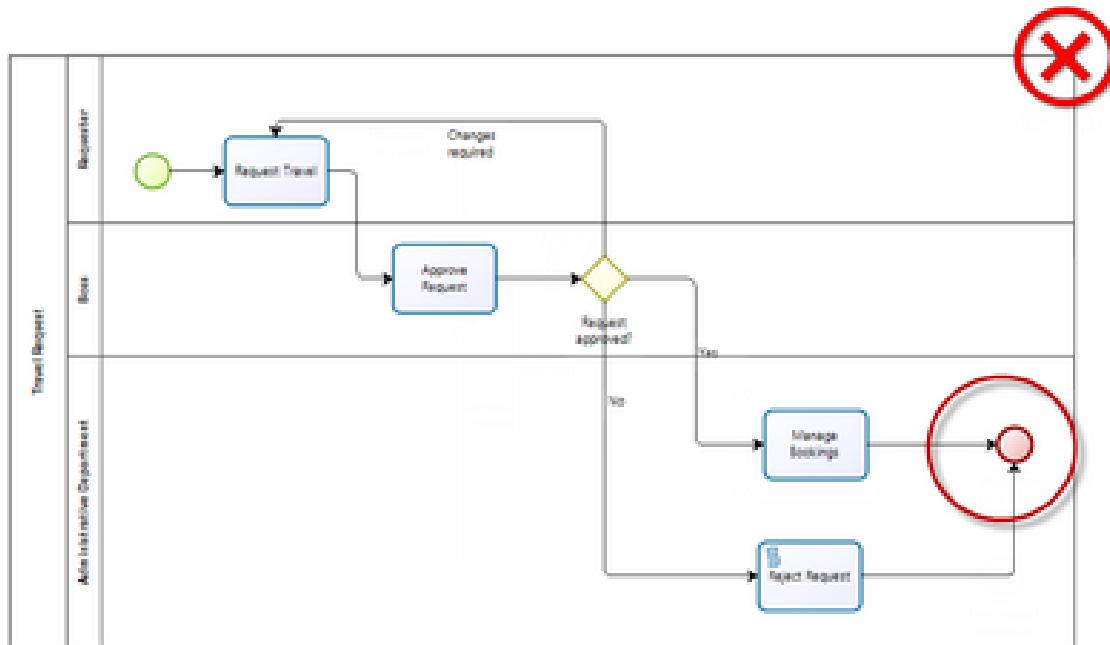
BPMN ofrece las herramientas necesarias para representar en el diagrama la lógica del manejo de excepciones de forma explícita. Una vez que el escenario principal es diagramado, haga uso de los siguientes elementos para modelar los flujos alternativos según sea necesario:

#### Utilice procesos transaccionales

Los procesos transaccionales permiten escenarios de negocios con transacciones. Un conjunto de actividades debe realizarse con éxito, de lo contrario compensación o cancelación flujos son seguidos. Para más información ver subprocesos.

#### Distinga los estados finales exitosos y no exitosos

Utilice eventos finales separados para identificar cuando un proceso terminó con éxito y cuando no, para propósitos de documentación y revisión.



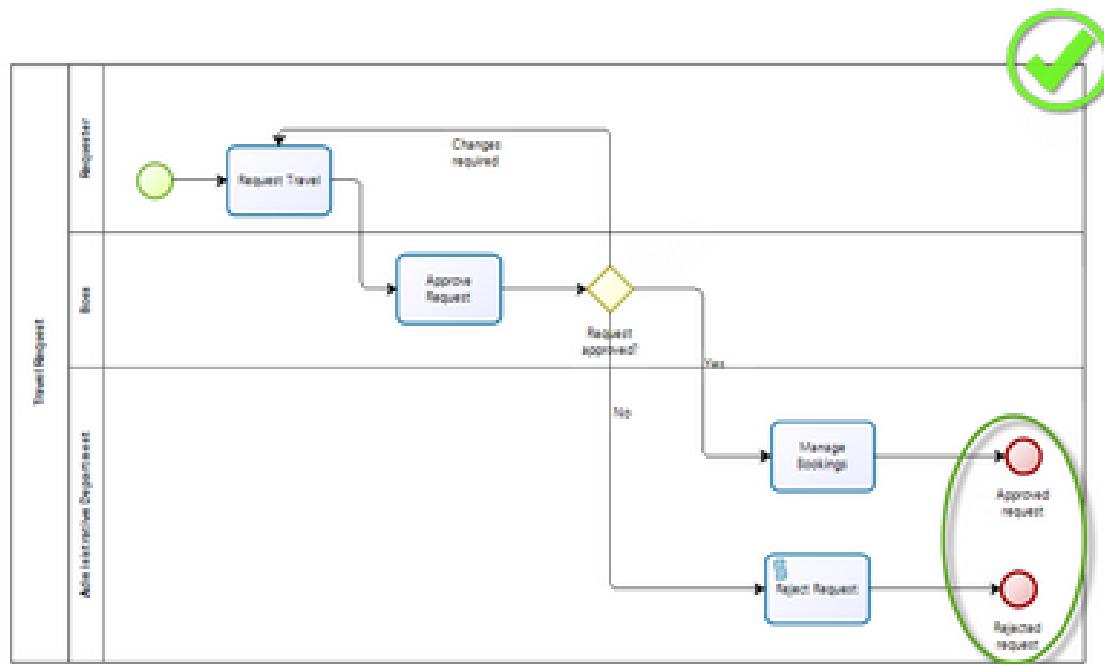
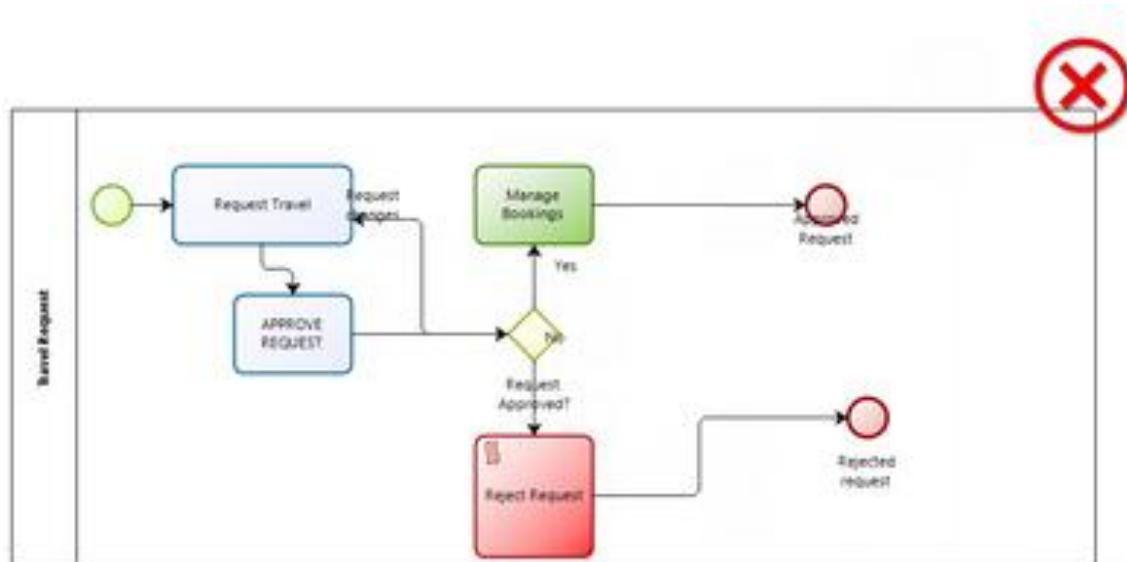


Figura 85: Escenarios alternativos claros

Fuente.- Bizagi BPM Suite: Guía del Usuario

Mantenga un formato estándar

Mantenga un formato único a lo largo de sus diagramas y enfóquese en una apariencia limpia y agradable. El uso de diferentes tamaños de fuente, colores, dimensiones de cajas o etiquetas superpuestas podrían hacer que la lectura de los diagramas sea un desafío.



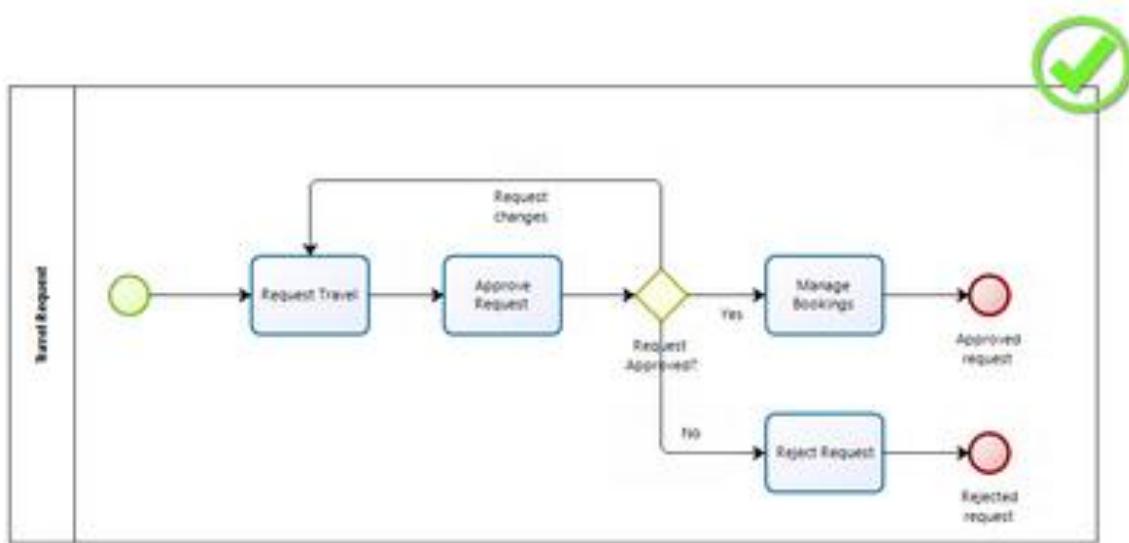


Figura 86: Mantener un formato estándar

Fuente.- Bizagi BPM Suite: Guía del Usuario

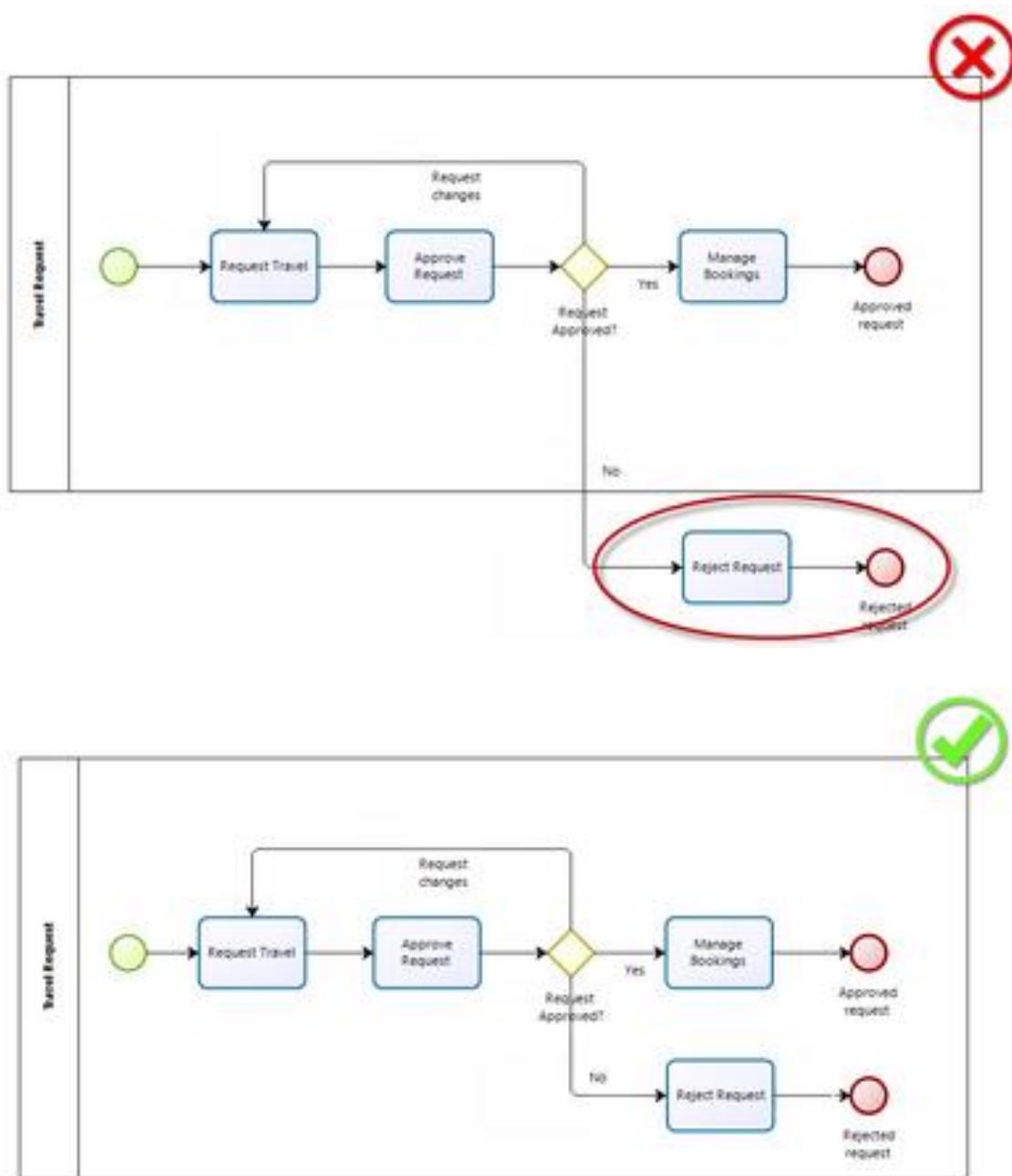
## 2. Utilice el estándar BPMN

El estándar BPMN define los lineamientos utilizados para diagramar los procesos de negocio. Sin embargo, seguir las directrices de BPMN está completamente en sus manos. Asegúrese de que sus modelos cumplen con la norma para asegurar su correcta comprensión.

Una vez se ha definido la lógica del proceso, valide sus diagramas asegurándose de utilizar correctamente los diferentes elementos de BPMN. El siguiente aspecto debe ser revisado para cada elemento BPMN:

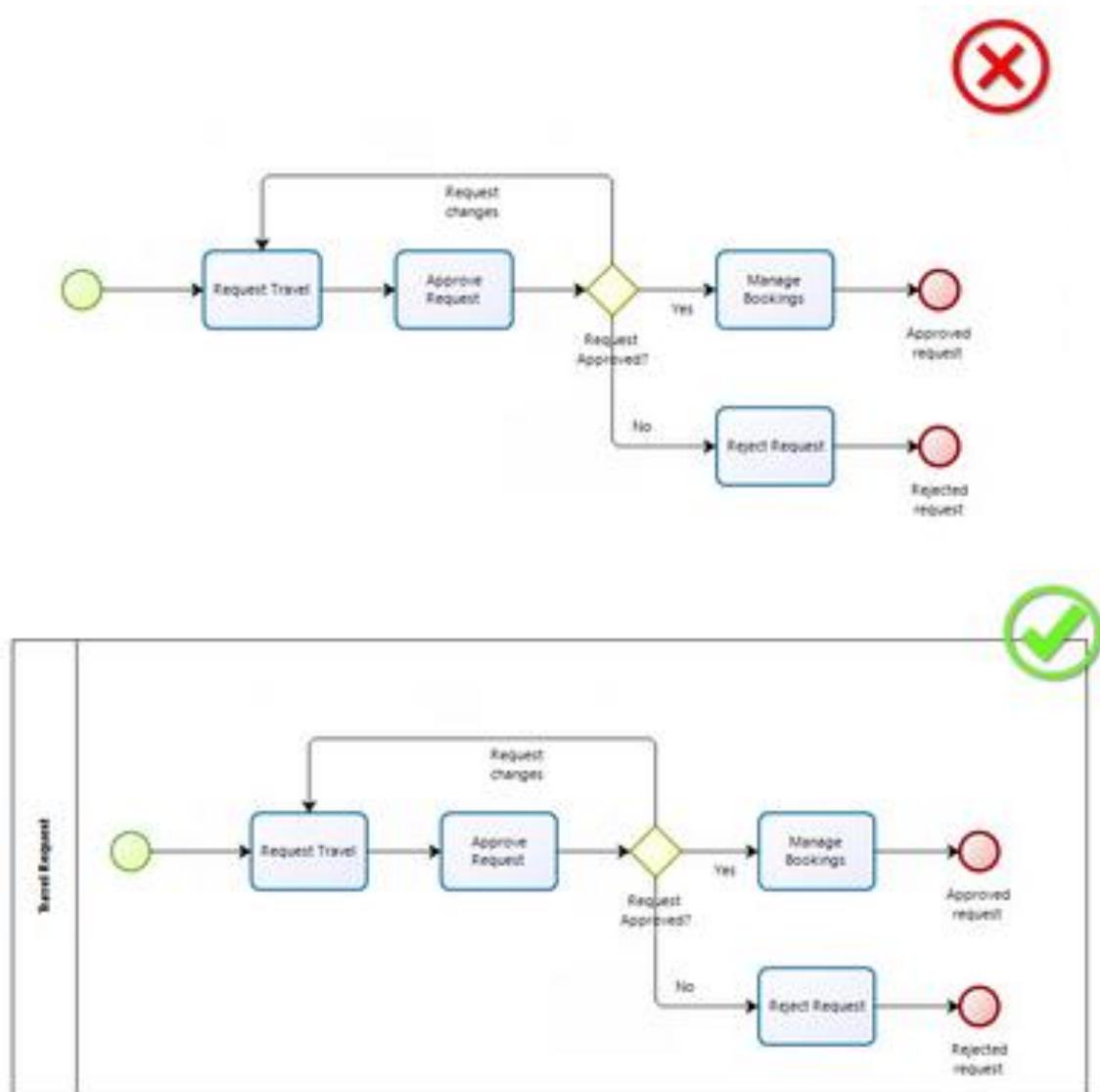
### Lo que hay que revisar en Pools

Diagrama los procesos completamente dentro de un Pool. Nunca diagrama flujos fuera de los límites de un Pool.



**Figura 87: Todo dentro del Pool**  
 Fuente.- Bizagi BPM Suite: Guía del Usuario

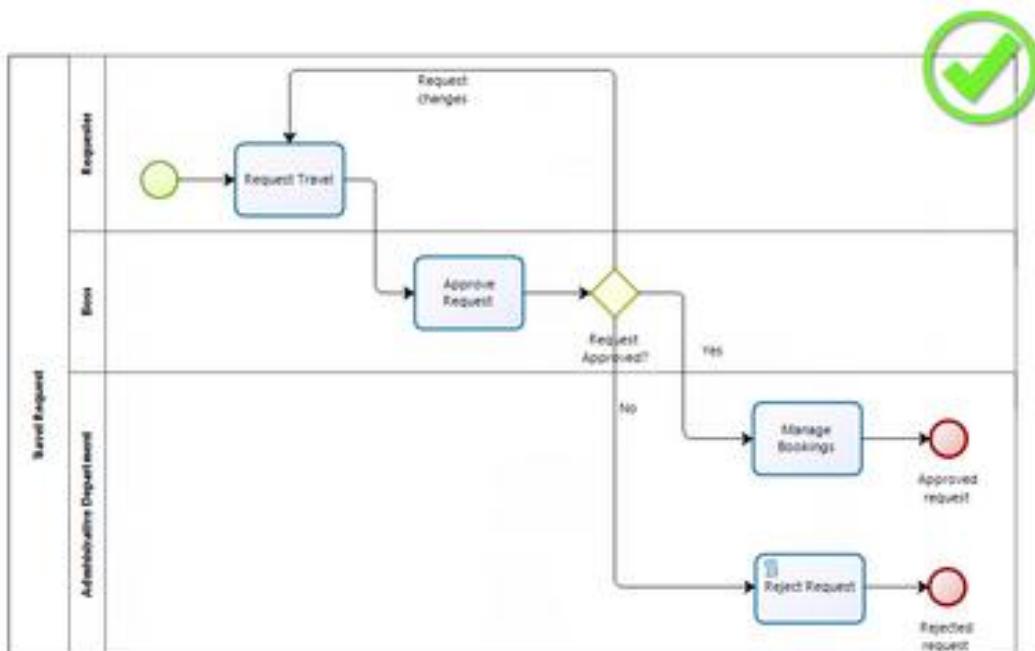
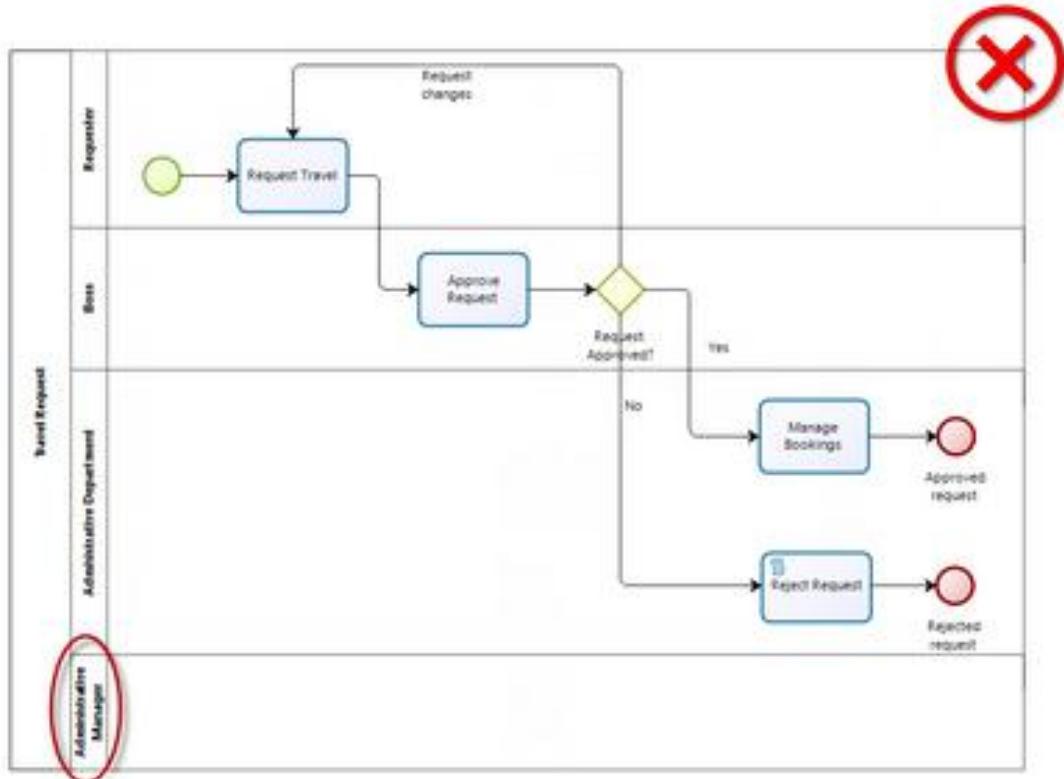
Defina tantos Pools como procesos. Debe haber siempre al menos un Pool.



**Figura 88: Al menos un Pool**  
Fuente.- Bizagi BPM Suite: Guía del Usuario

## ¿Qué verificar en Lanes?

Cree un Lane solo si se ejecuta al menos una tarea o un evento intermedio en él.



**Figura 89: No crear Lanes vacíos**  
Fuente.- Bizagi BPM Suite: Guía del Usuario

No cree Lanes para representar un área o una entidad que lleva a cabo una tarea automática o una compuerta.

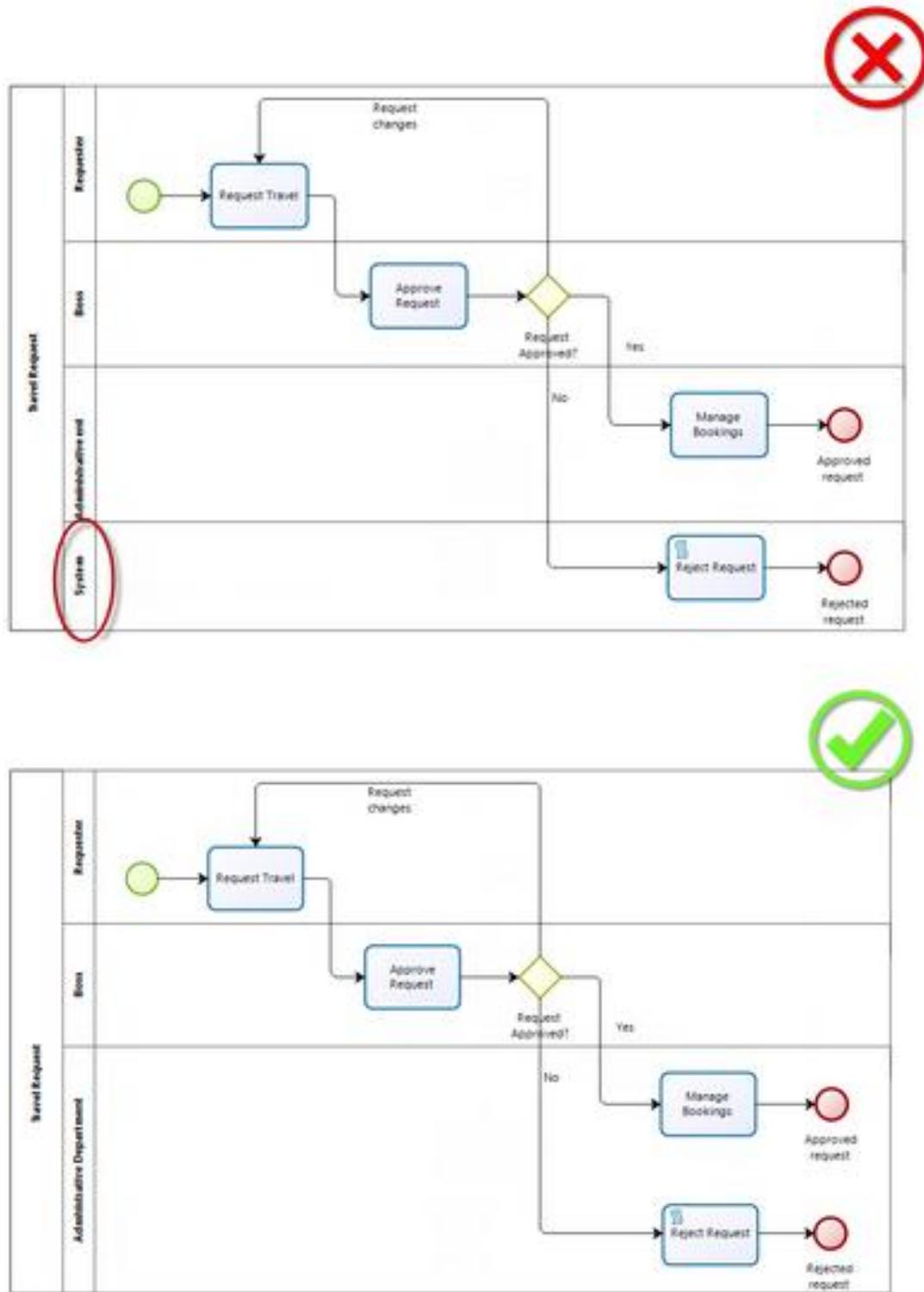


Figura 90: No crear Lanes para tareas automáticas

Fuente.- Bizagi BPM Suite: Guía del Usuario

No diagrama tareas, compuertas o eventos en medio de dos Lanes.

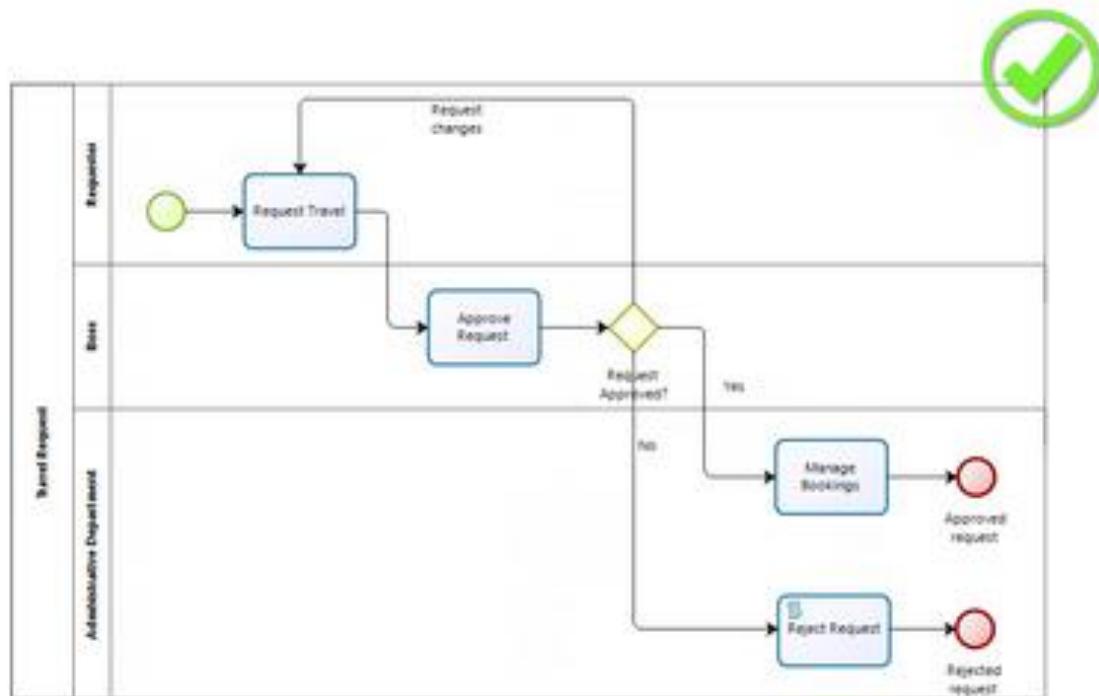
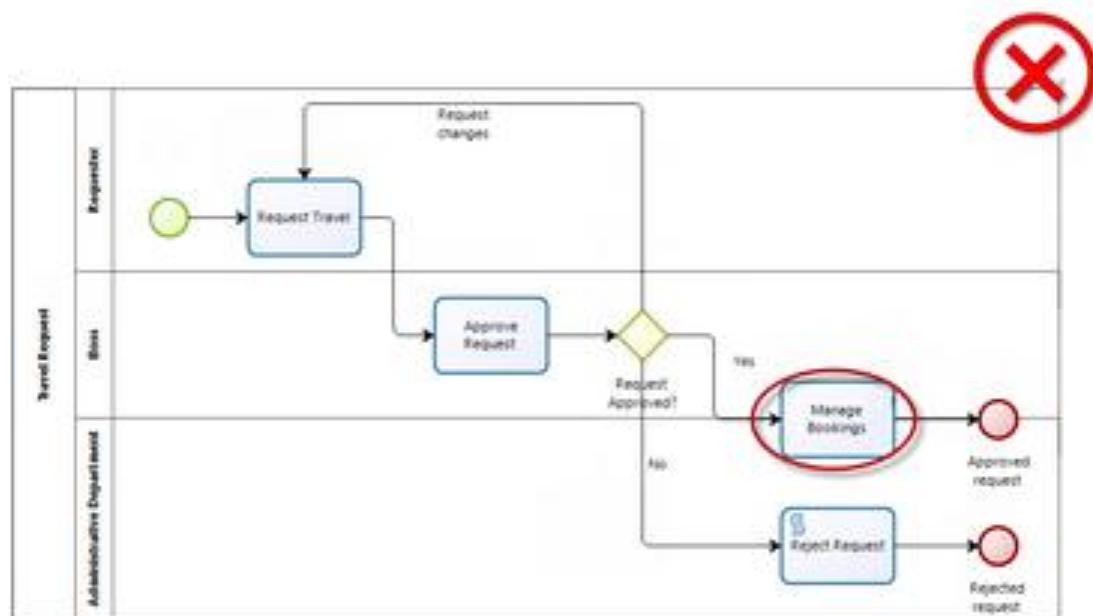


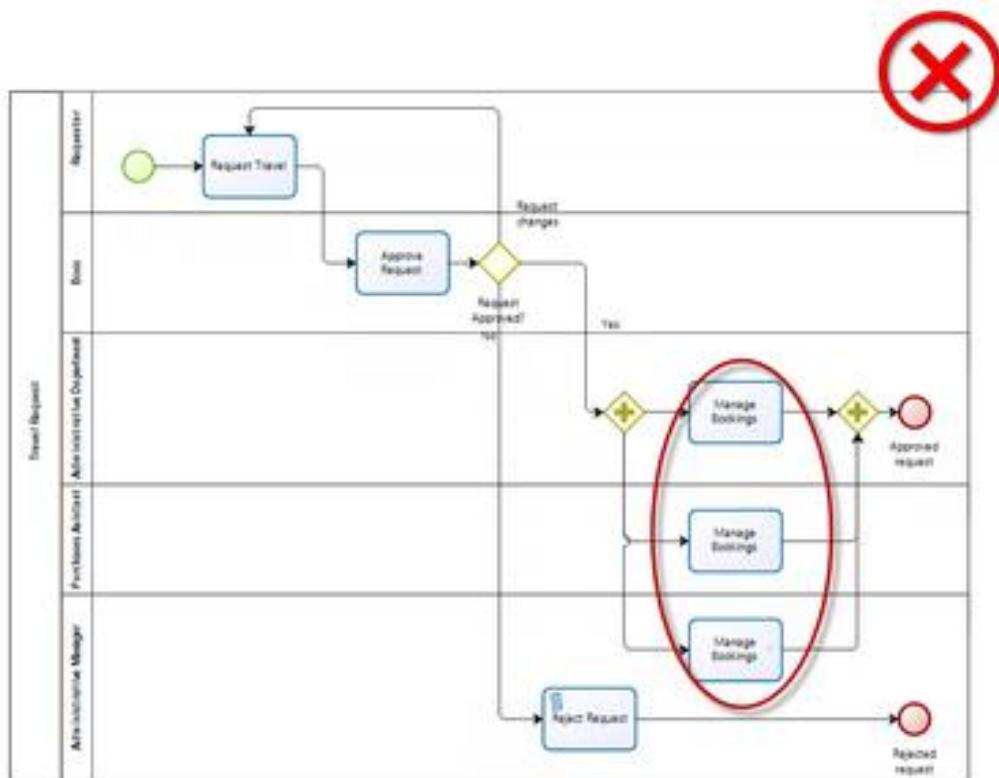
Figura 91: No crear tareas en medio de dos Lanes

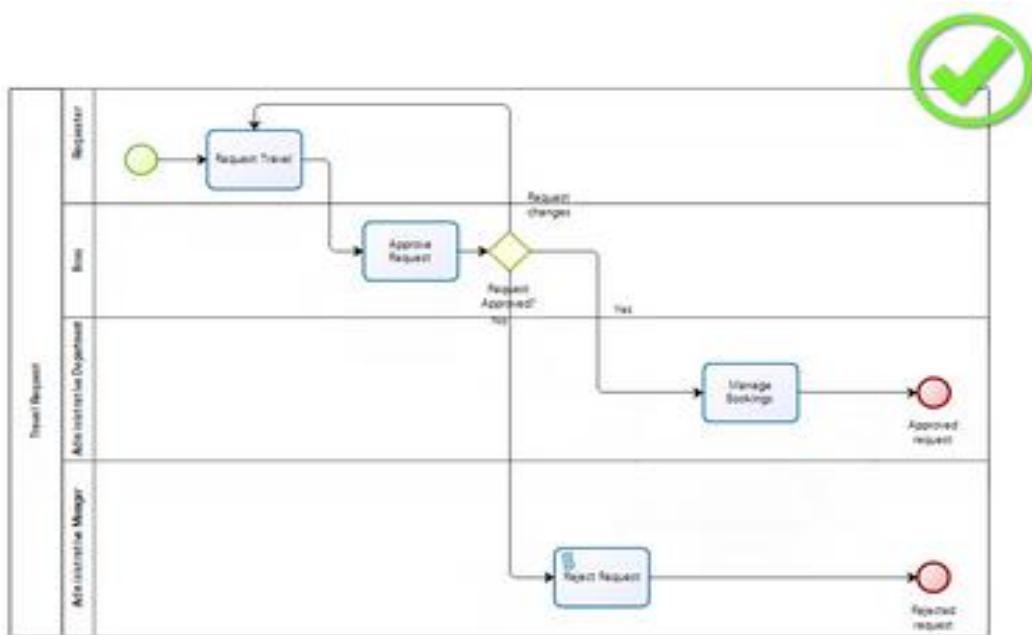
Fuente.- Bizagi BPM Suite: Guía del Usuario

### ¿Qué verificar en Actividades?

No diagrame varias instancias de la misma tarea para representar a varios participantes.

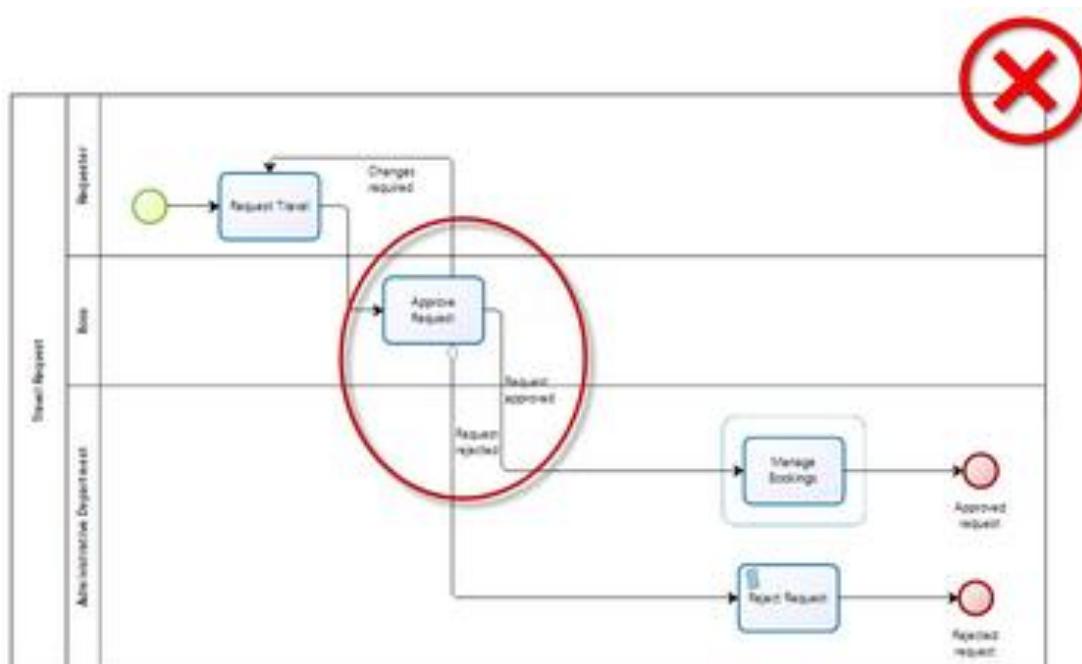
Solo diagrame una tarea en un área. Defina los participantes como condiciones de asignación en la documentación y en reglas de asignación.





**Figura 92: Sólo un tarea en un área**  
 Fuente.- Bizagi BPM Suite: Guía del Usuario

No ramifique los flujos usando tareas. Siempre use las compuertas.



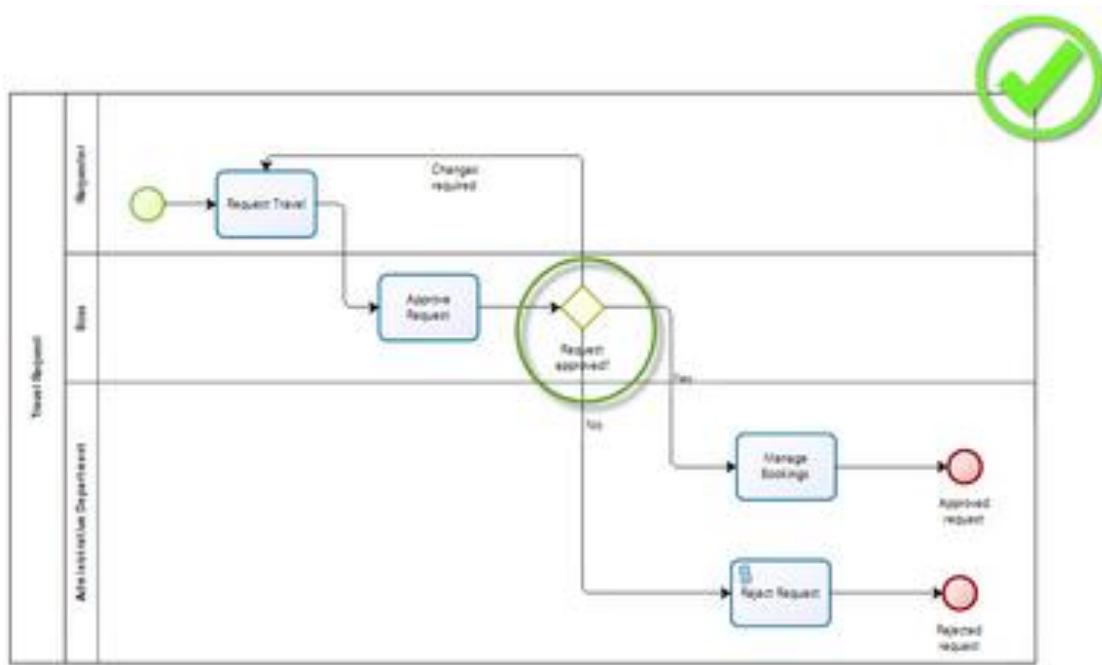
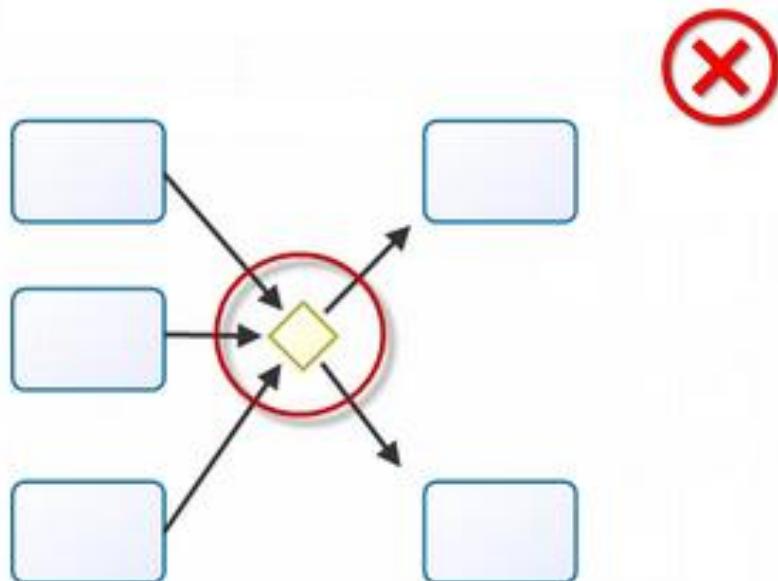


Figura 93: Usar compuertas en vez de ramificar

Fuente.- Bizagi BPM Suite: Guía del Usuario

### ¿Qué verificar en compuertas?

No use compuertas para juntar y separar al mismo tiempo. Esto producirá un error en tiempo de ejecución.



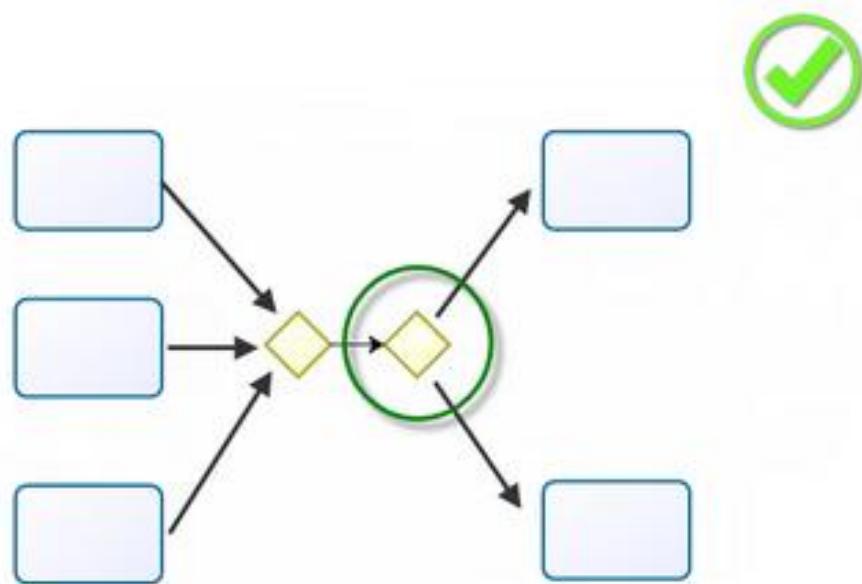
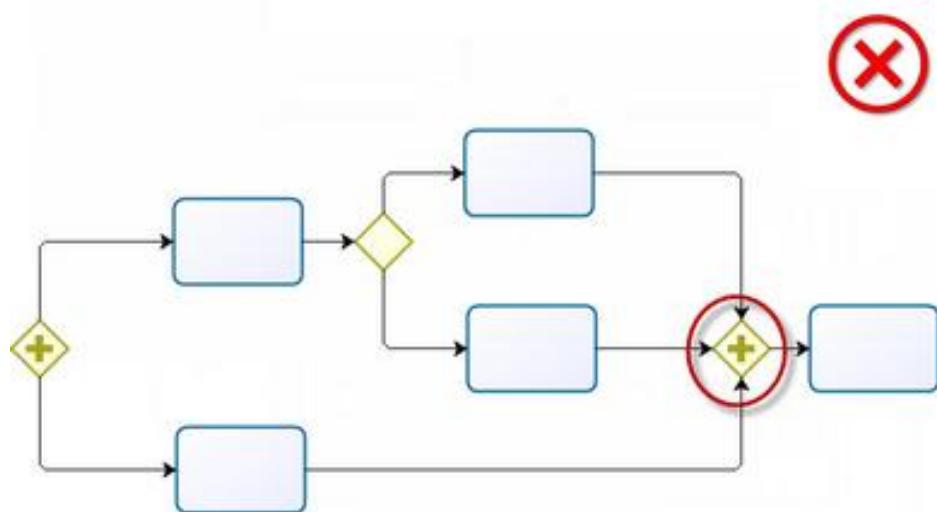


Figura 94: No usar compuertas para juntar y separar al mismo tiempo  
Fuente.- Bizagi BPM Suite: Guía del Usuario

Balancee las compuertas. Las divisiones deben unirse de manera equivalente.



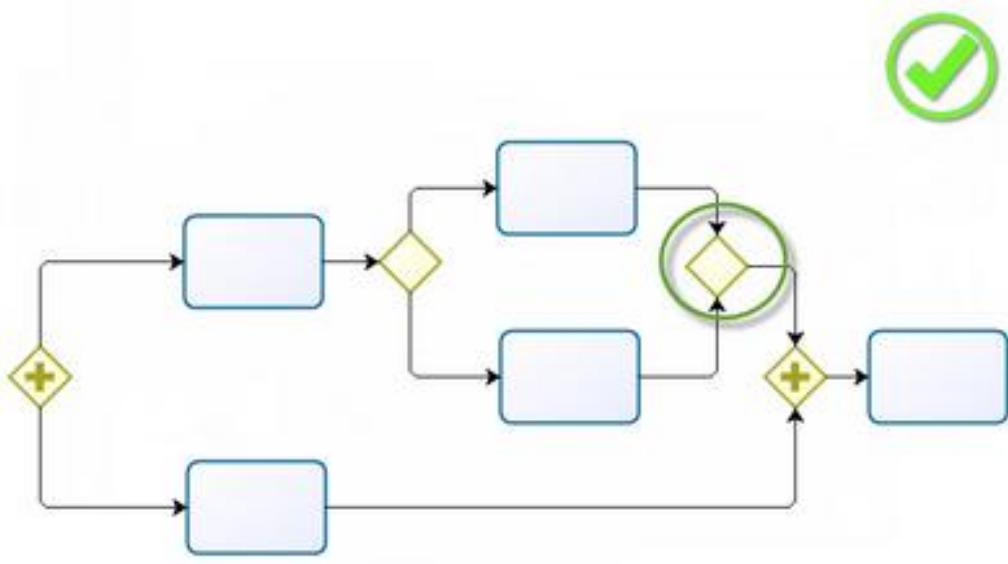
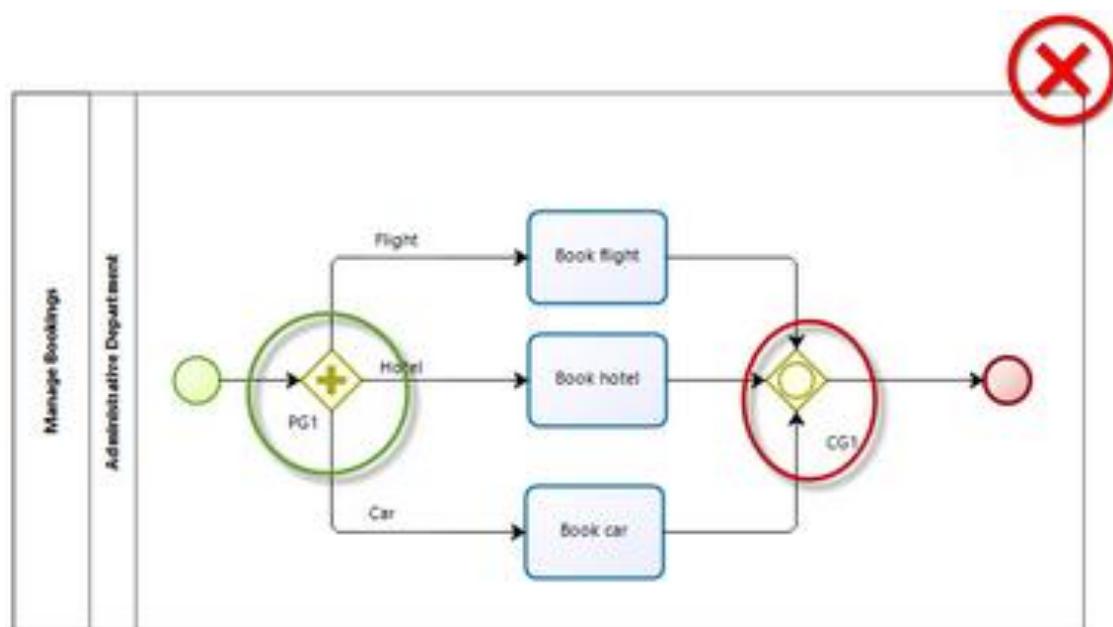


Figura 95: No usar compuertas para juntar y separar al mismo tiempo

Fuente.- Bizagi BPM Suite: Guía del Usuario

Siempre use el mismo tipo de compuerta para juntar los flujos que fue usado para dividirlos.



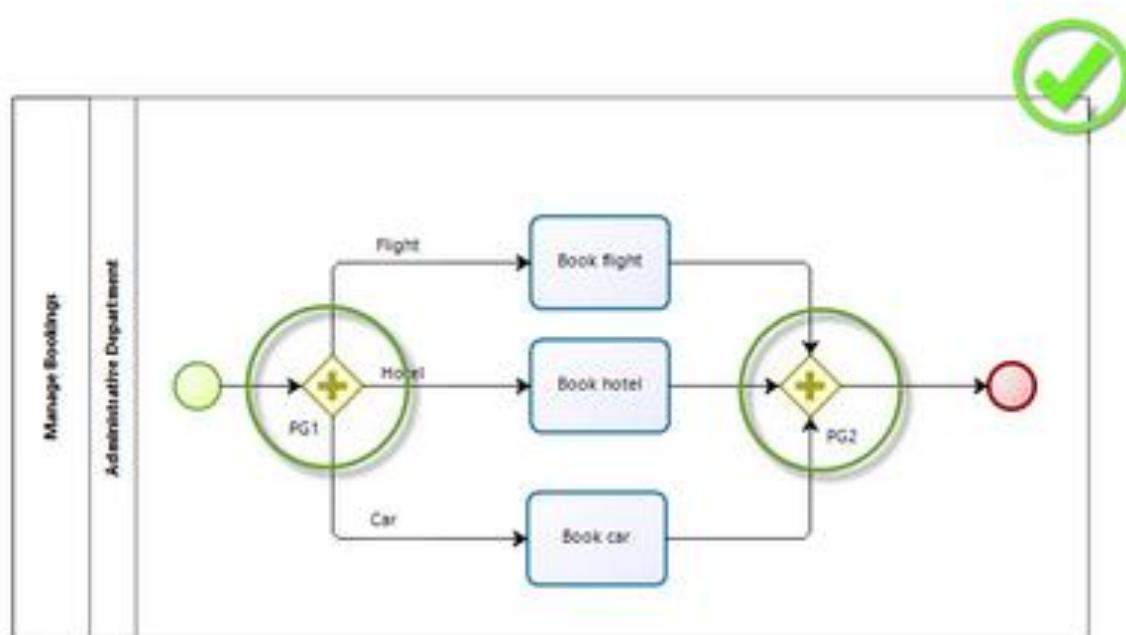
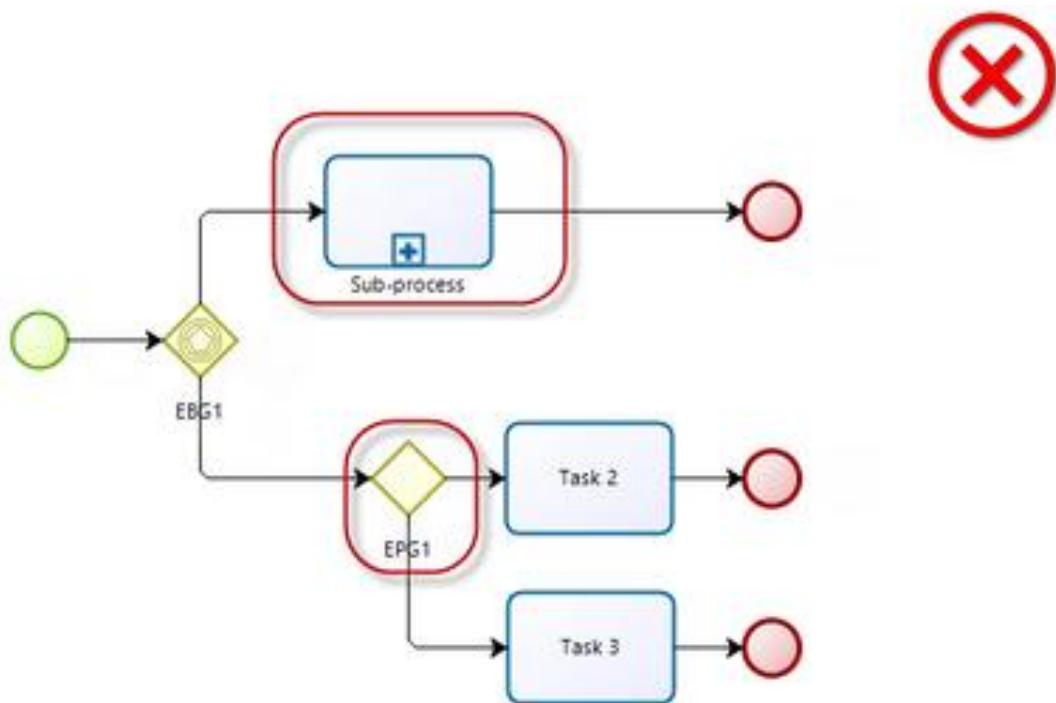


Figura 96: La misma compuerta para dividir y juntar los flujos.

Fuente.- Bizagi BPM Suite: Guía del Usuario

Use solo Eventos y/o Tareas después de una compuerta basada en eventos.



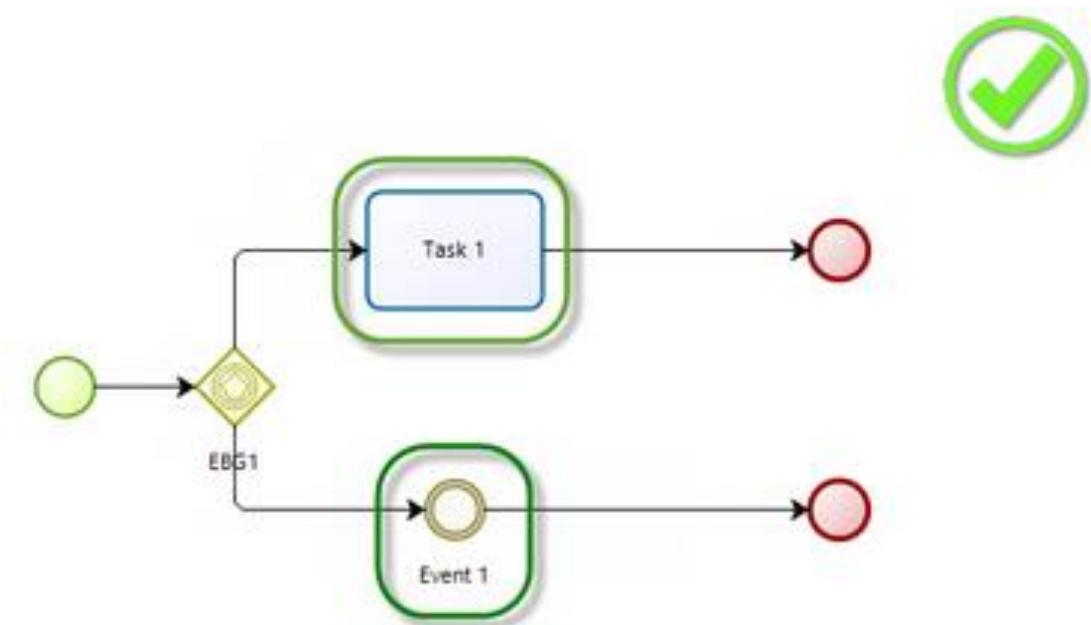
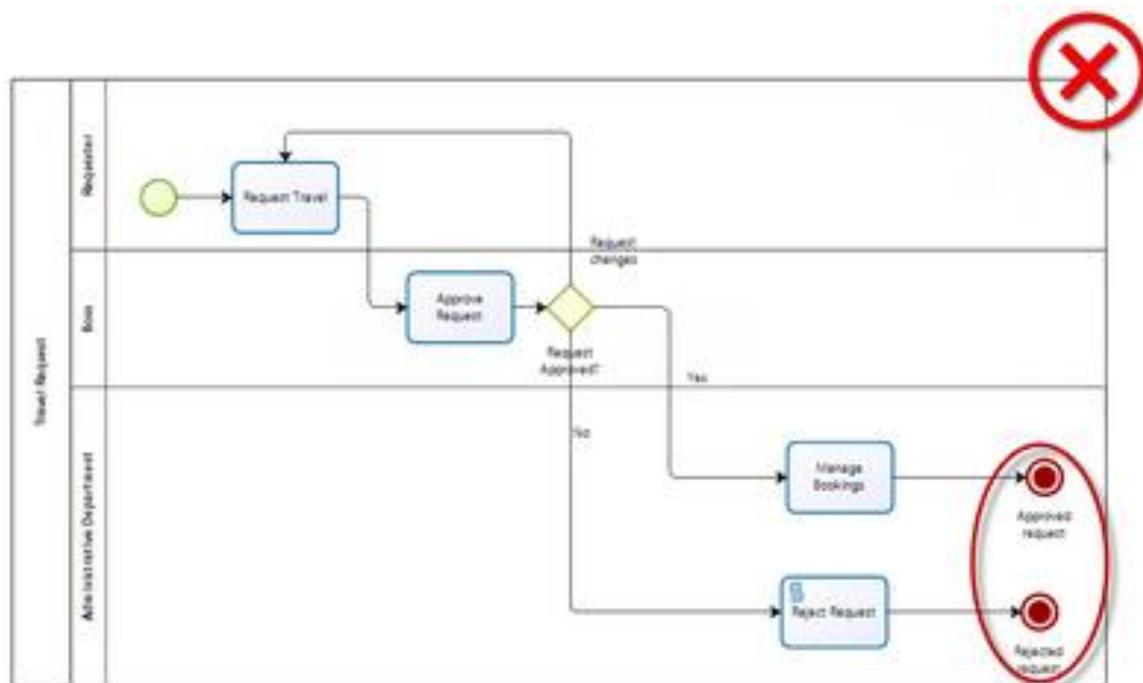


Figura 97: Sólo Eventos y/o Tareas después de una compuerta basada en eventos.  
Fuente.- Bizagi BPM Suite: Guía del Usuario

### ¿Qué verificar en Eventos?

Utilice eventos de terminación solo cuando sea estrictamente necesario. Estos se utilizan para modelar situaciones donde se habilitan varios caminos alternativos y todo el proceso tiene que ser terminado cuando uno de ellos se ha completado.

*Esto tiene una excepción descrita en el siguiente ítem.*



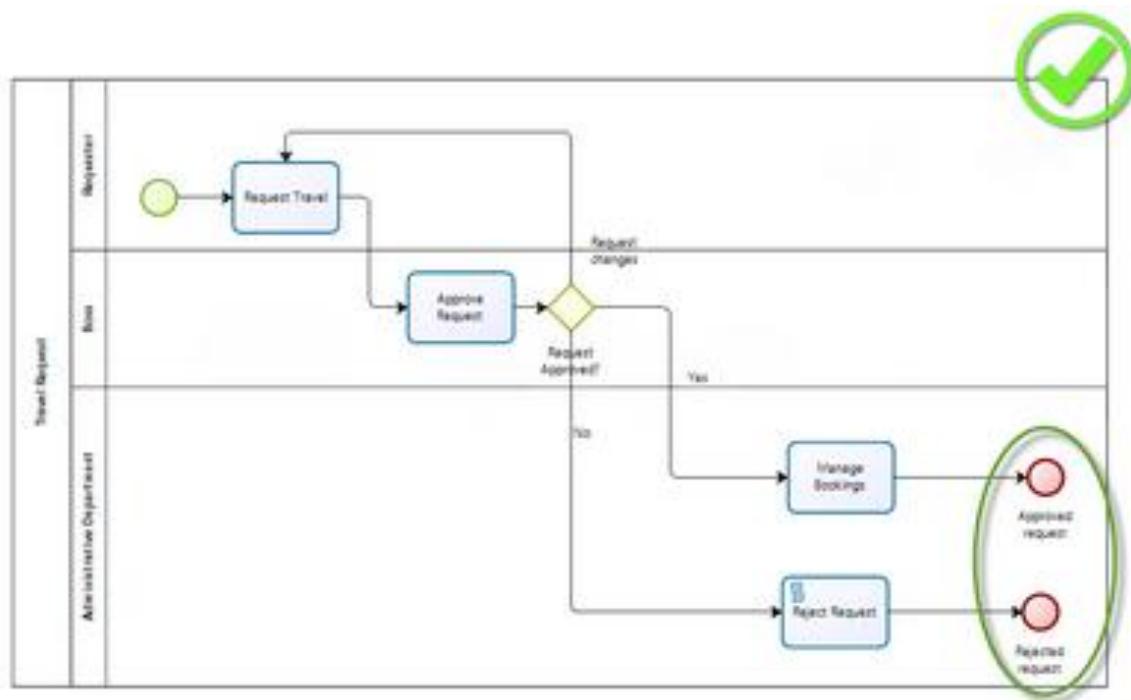
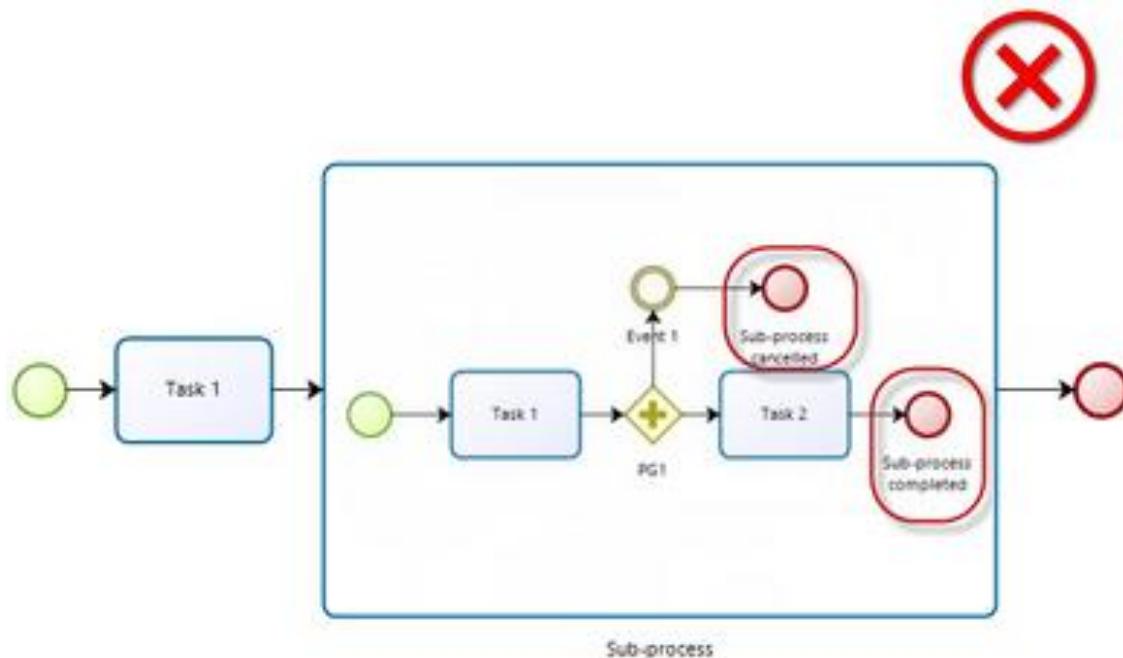


Figura 98: Usar eventos de terminación sólo cuando sea necesario.

Fuente.- Bizagi BPM Suite: Guía del Usuario

Use los Eventos de finalización terminal en vez de eventos de terminación en subprocesos embebidos.



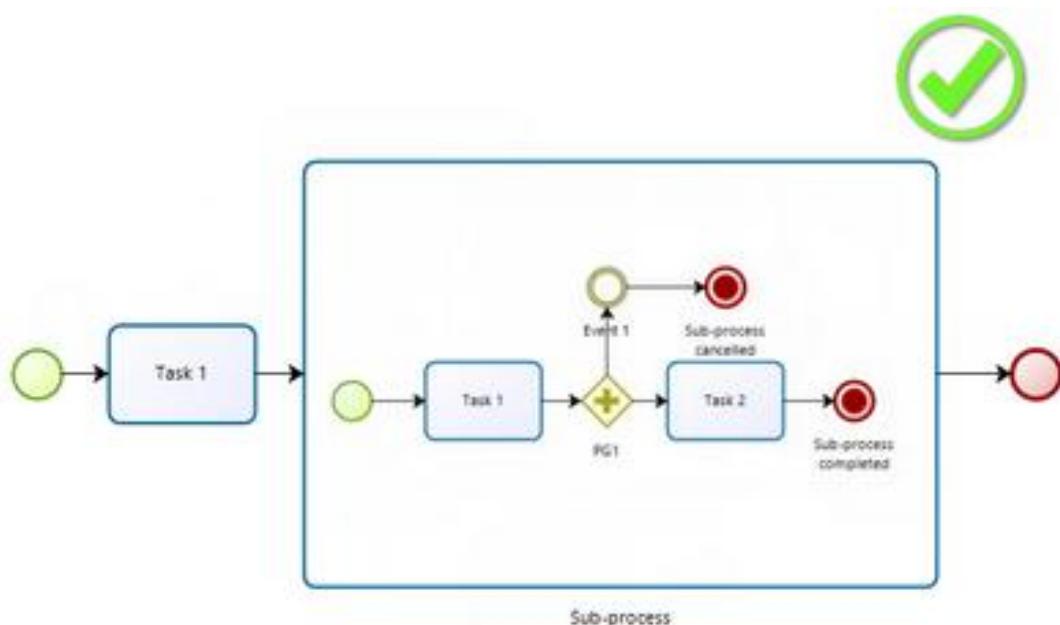


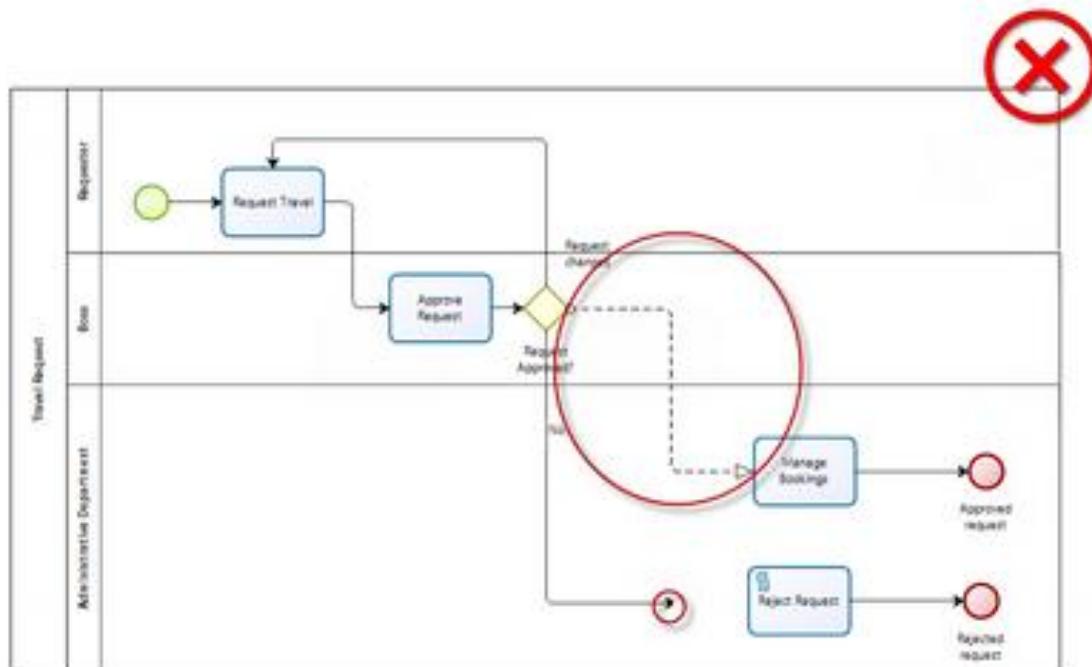
Figura 99 Usar Eventos de finalización terminal en subprocesos embebidos.

Fuente.- Bizagi BPM Suite: Guía del Usuario

### ¿Qué verificar en Conectores?

Use flujos de secuencia para conectar todas las actividades, eventos y compuertas.

Nunca use el flujo de mensajes para conectar las actividades del mismo Pool o deje formas sin conectar.



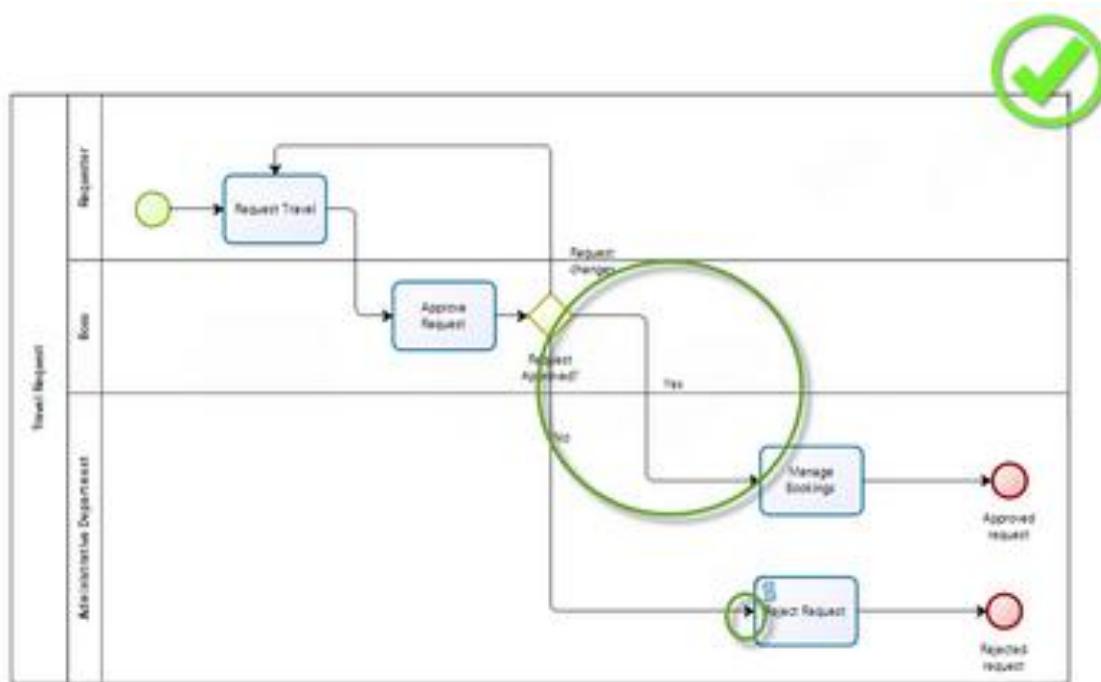
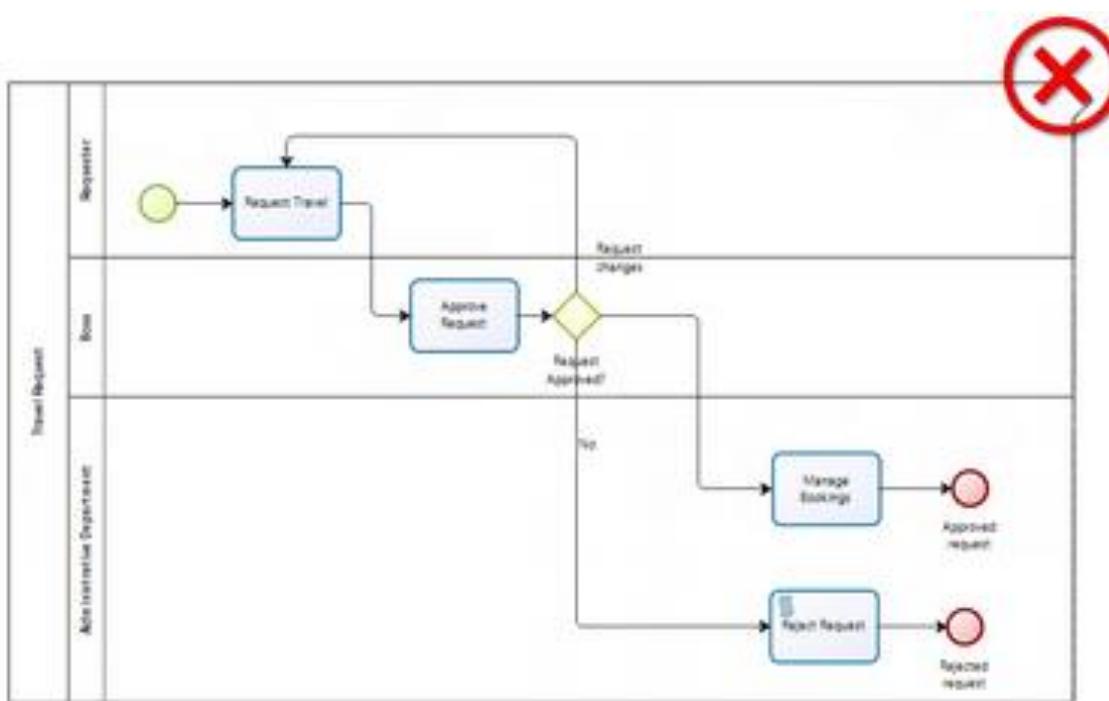


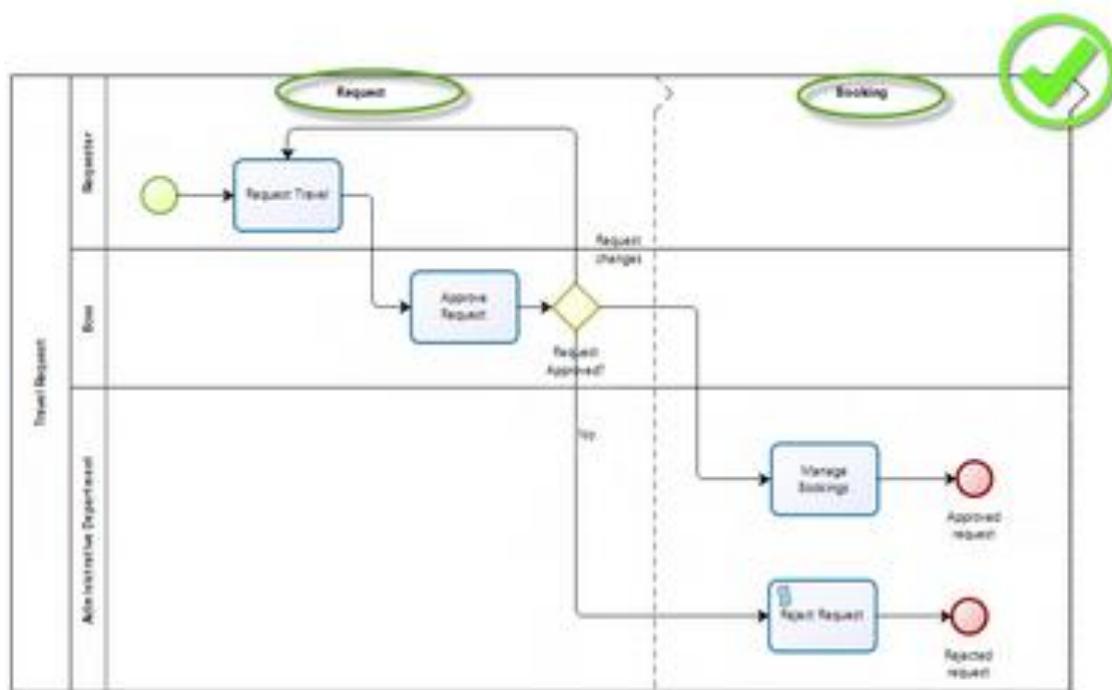
Figura 100: Usar flujos de secuencia para conectar actividades, eventos y compuertas, dentro de un mismo Pool.

Fuente.- Bizagi BPM Suite: Guía del Usuario

### ¿Qué verificar en Milestones?

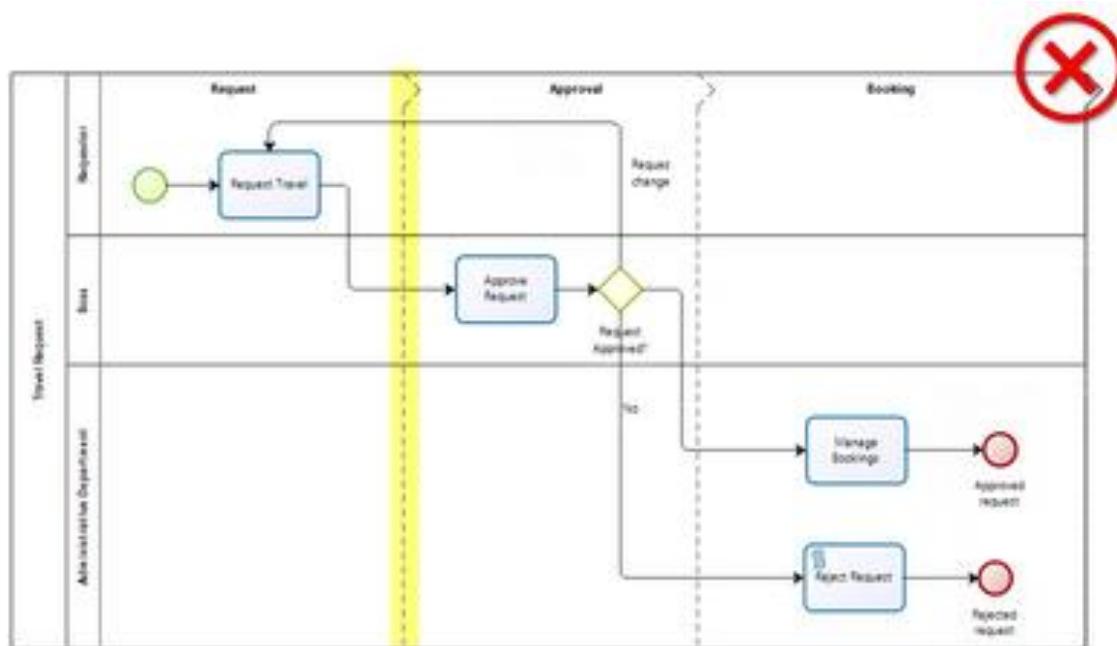
Siempre identifique y defina fases; estas representan un periodo de tiempo objetivo o una transición en el proceso.





**Figura 101: Siempre usar fases.**  
 Fuente.- Bizagi BPM Suite: Guía del Usuario

En lo posible, evite regresar entre Milestones.



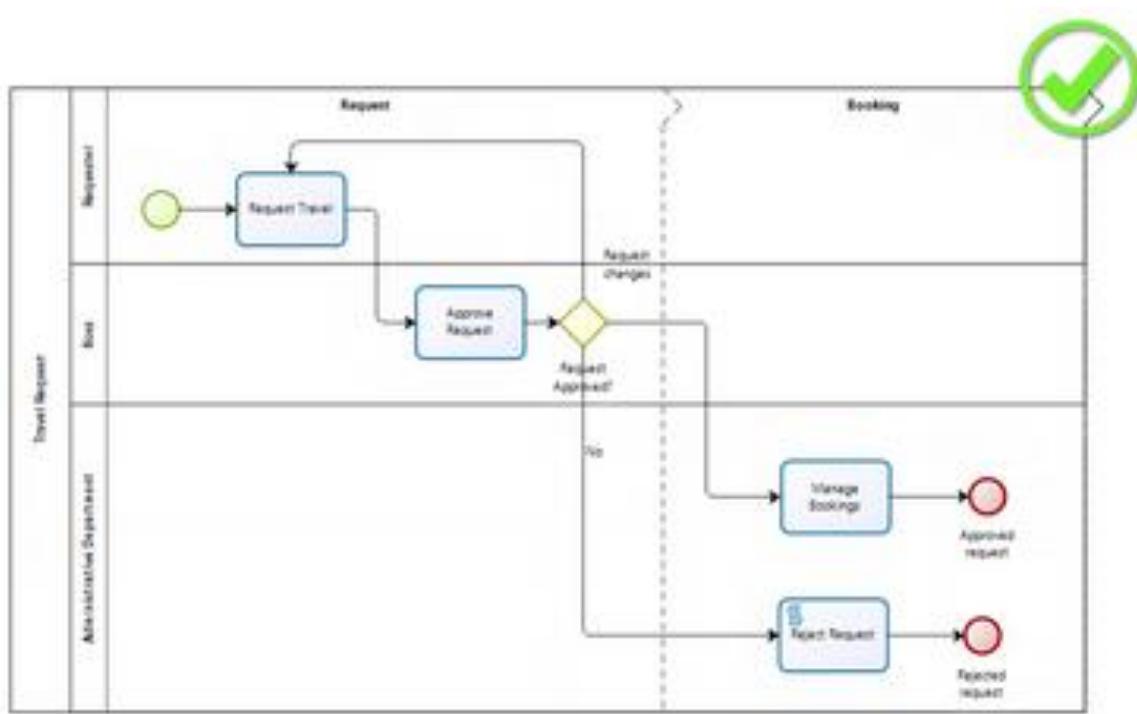


Figura 102: Evitar regresar entre Milestones.

Fuente.- Bizagi BPM Suite: Guía del Usuario

### 3. Utilice un etiquetado estricto

El nombramiento correcto de los diferentes elementos de los diagramas es fundamental para una comprensión fácil y correcta de los procesos.

Al revisar los logs, es muy útil saber cómo se ejecutó el proceso. Cuando no se nombra alguna forma en el proceso, los Logs, se muestran en blanco, lo que hace que sea difícil de entender. Estas son algunas recomendaciones que le ayudarán a hacerlo:

#### Etiquetas de los procesos

Los nombres de los procesos deben describir claramente su propósito principal.

Asegúrese de no utilizar nombres cortos o abreviaturas.

**Prefijo:** utilizando el nombre de proceso, cree un prefijo que se utilizará en todos los componentes que pertenecen al proceso.

Por ejemplo, si el nombre del proceso es: Solicitud del cliente, puede usar el prefijo SC\_.

Los siguientes prefijos están restringidos para otros fines: "P\_", "M\_".

#### Etiquetas de las actividades

Dé a las actividades un nombre compuesto por **un verbo** y **un objeto**.

De esta manera, los lectores puedan entender con claridad el objetivo de una tarea.

Además, asegúrese de que usted no utiliza nombres cortos o abreviaturas.

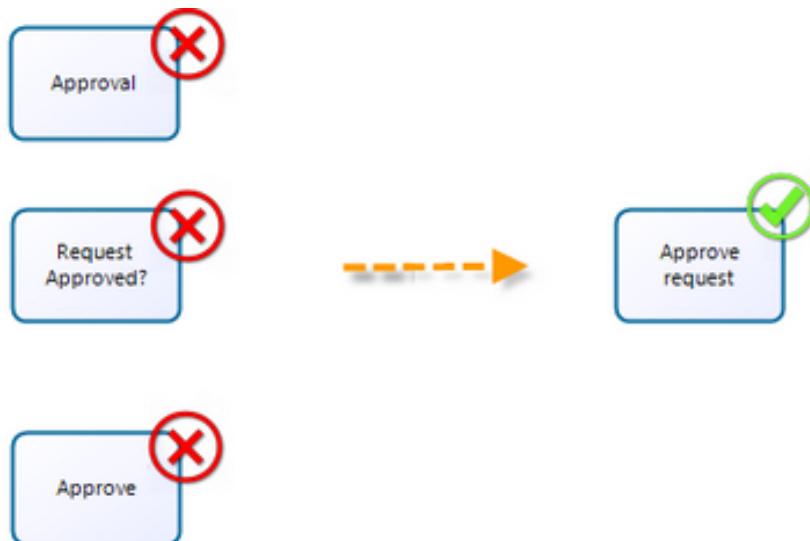


Figura 103: Usar un verbo y un objeto para el nombre de las actividades.

Fuente.- Bizagi BPM Suite: Guía del Usuario

#### Etiquetas de los Eventos

Utilice el etiquetado cuando se utilizan múltiples eventos de inicio y fin. Nómbrelos para que el diagrama pueda explicarse por sí mismo y permitir que los usuarios sepan cómo termina el proceso.

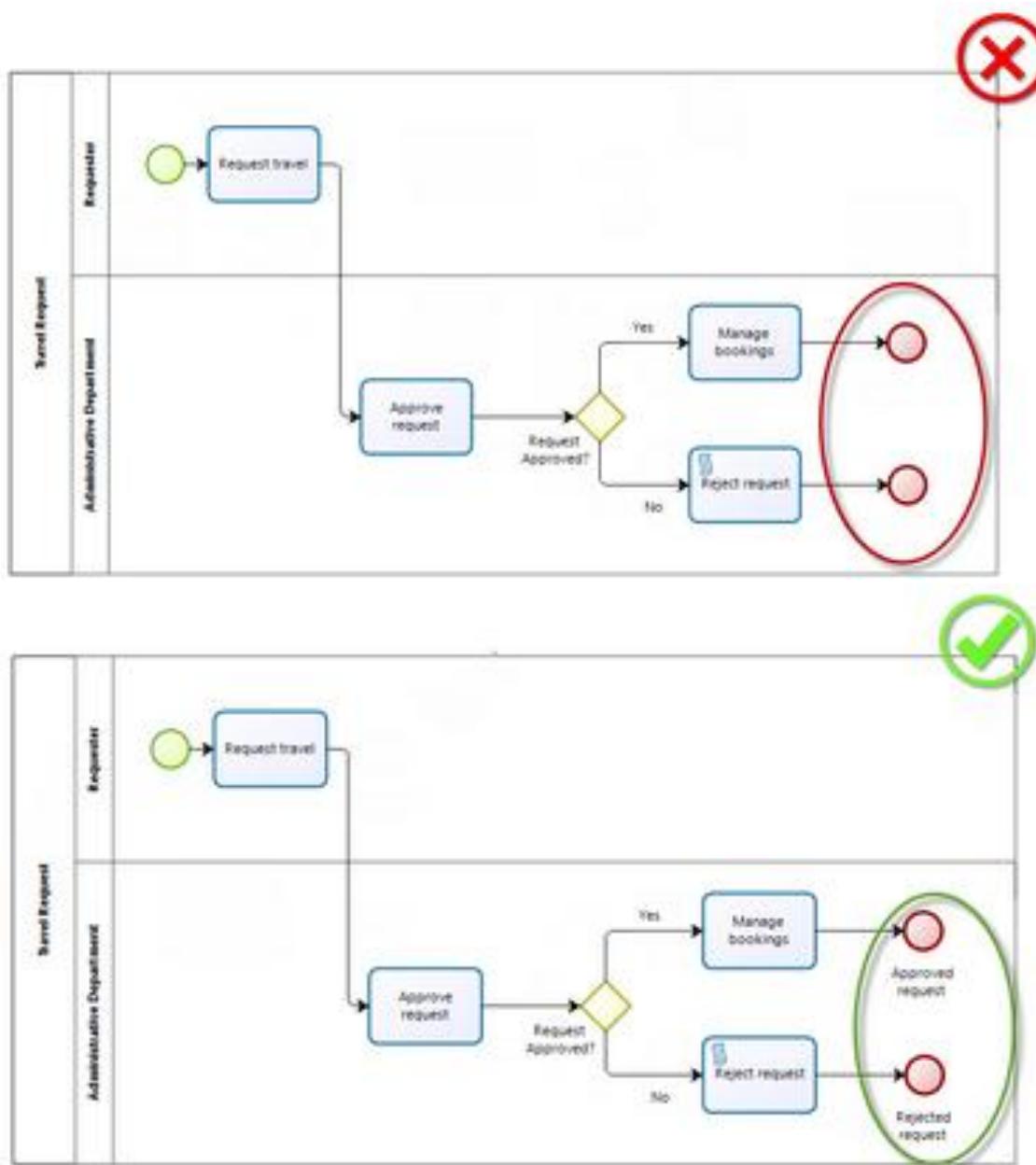


Figura 104: Etiquetar cuando se utilicen múltiples eventos de inicio y fin.

Fuente.- Bizagi BPM Suite: Guía del Usuario

## Etiquetas de los Milestones

Los Milestones deberían ser nombrados con un sustantivo que haga referencia a un periodo de tiempo (verano, madurez) o a lo que sucede en un periodo de tiempo (creación, aprobación, entrega).

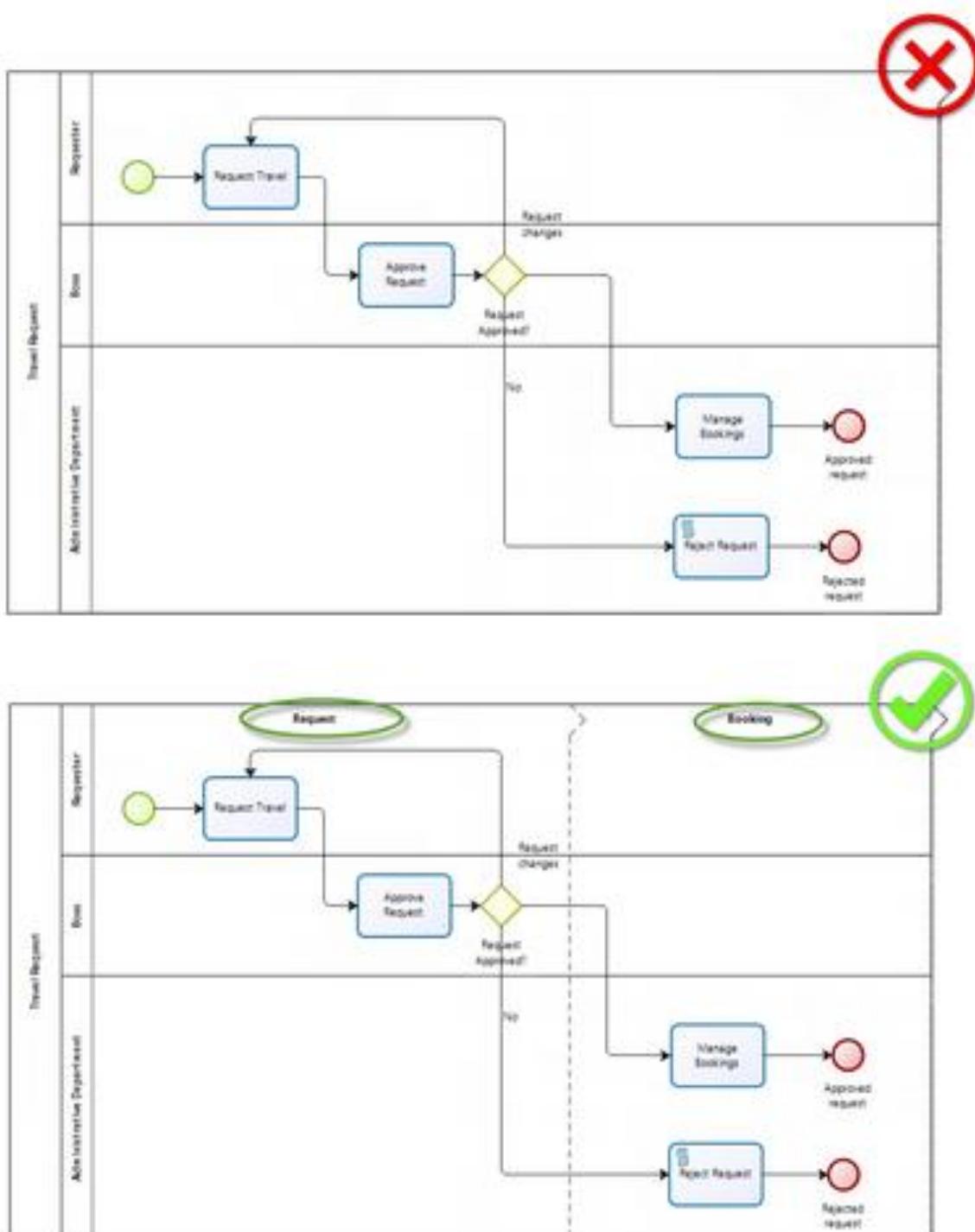


Figura 105: Nombrar las Fases con un sustantivo que haga referencia a un período.

Fuente.- Bizagi BPM Suite: Guía del Usuario

### Etiquetas de las Compuertas

Las compuertas de divergencia deben tener un nombre que indique claramente la decisión o condición evaluada cuando aplique.

Utilice un nombre compuesto por un verbo, un objeto, y un signo de interrogación para identificar lo que se está evaluando. Usted puede incluso utilizar preguntas para aclarar la decisión en cuestión.



Figura 106: Nombrar claramente las compuertas de divergencia.

Fuente.- Bizagi BPM Suite: Guía del Usuario

Si no aplica tener un nombre para la compuerta, use abreviaciones o números para diferenciarlas.

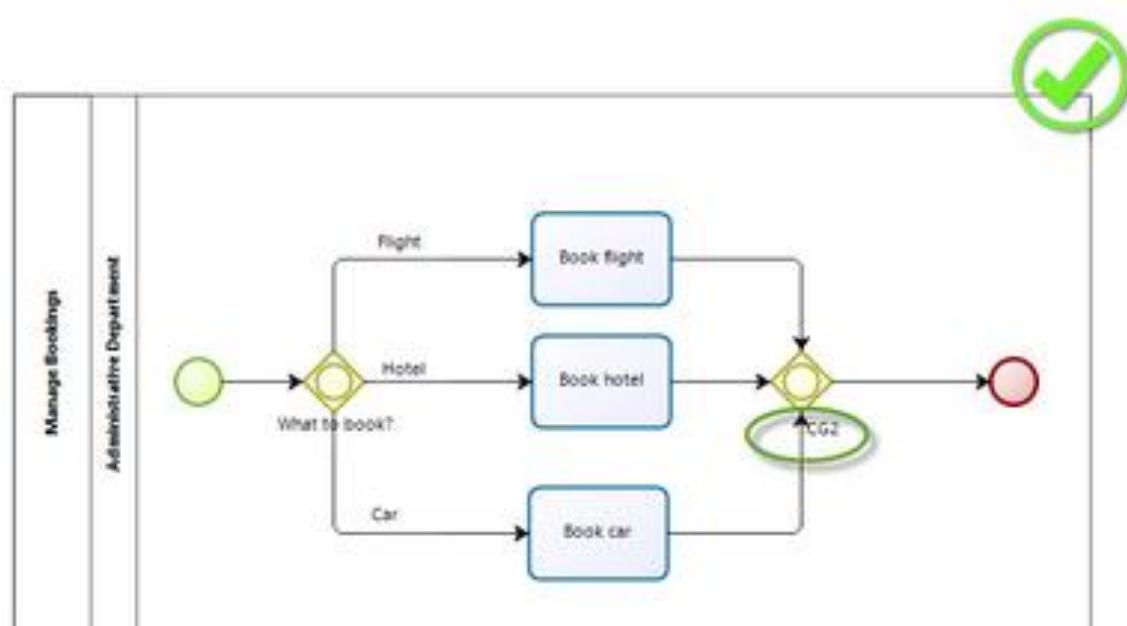
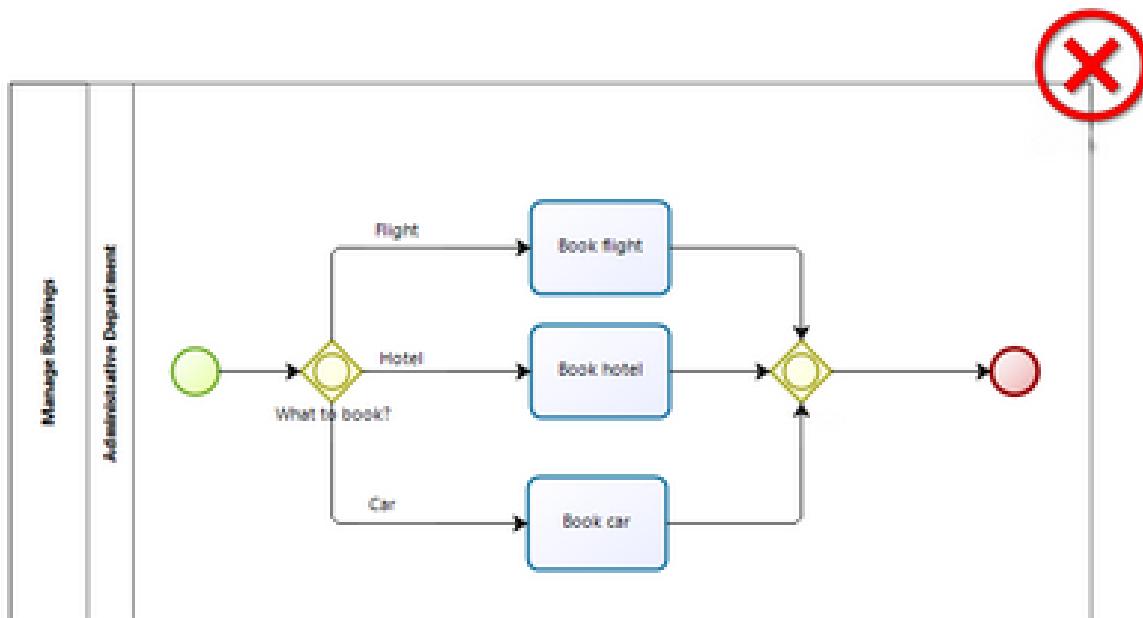


Figura 107: Usar abreviaturas o nombres para diferenciar compuertas.

Fuente.- Bizagi BPM Suite: Guía del Usuario

Nombre las transiciones indicando la condición relacionada.

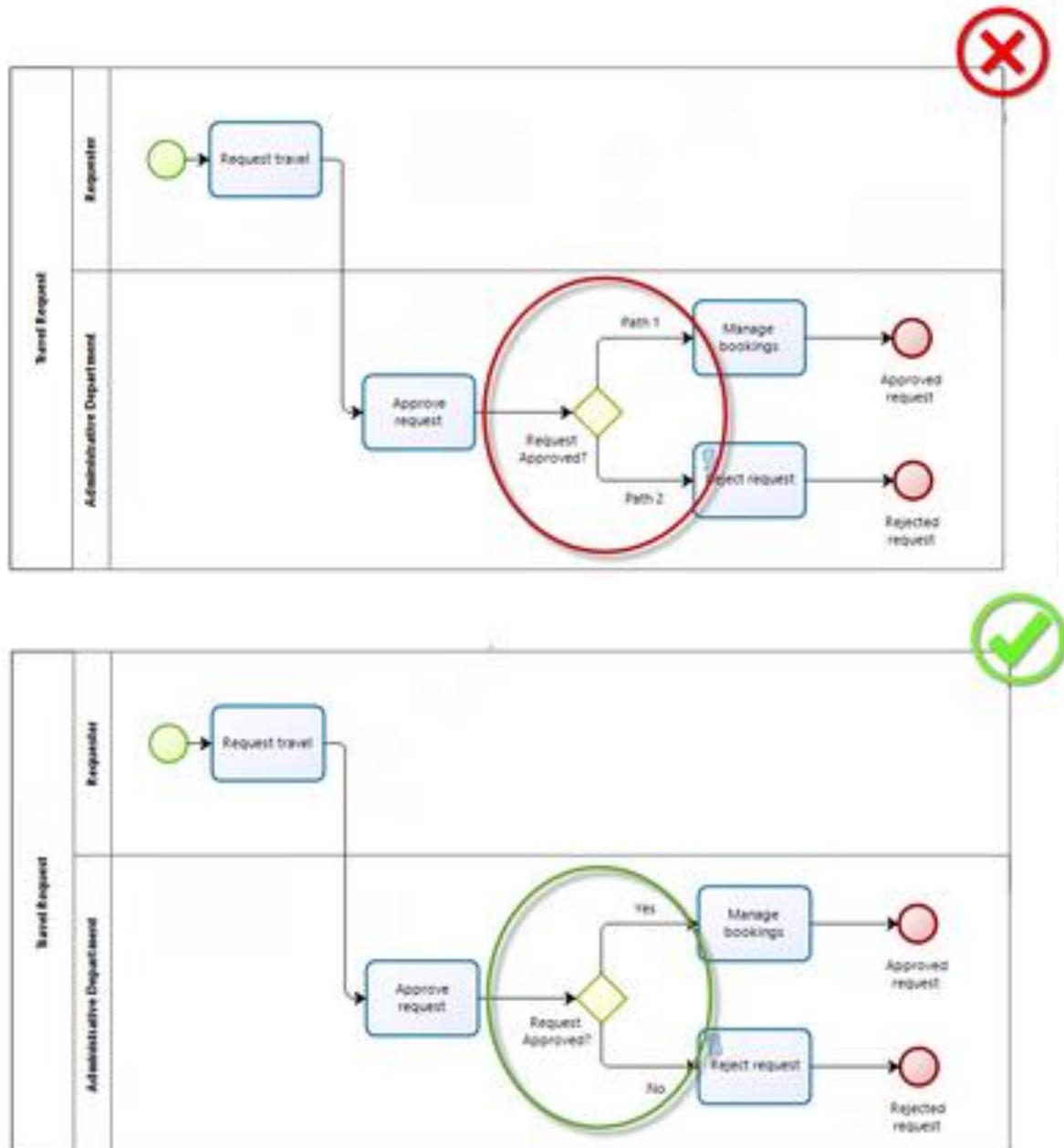


Figura 108: Nombrar transiciones de acuerdo a la condición relacionada.

Fuente.- Bizagi BPM Suite: Guía del Usuario