



Revista EIA

ISSN: 1794-1237

revista@eia.edu.co

Escuela de Ingeniería de Antioquia

Colombia

Tabares, Marta Silvia; Pineda, Juan Diego; Barrera, Andrés Felipe
UN PATRÓN DE INTERACCIÓN ENTRE DIAGRAMAS DE ACTIVIDADES UML Y SISTEMAS
WORKFLOW

Revista EIA, núm. 10, diciembre, 2008, pp. 105-120

Escuela de Ingeniería de Antioquia

Envigado, Colombia

Disponible en: <http://www.redalyc.org/articulo.oa?id=149212844008>

- Cómo citar el artículo
- Número completo
- Más información del artículo
- Página de la revista en redalyc.org

redalyc.org

Sistema de Información Científica

Red de Revistas Científicas de América Latina, el Caribe, España y Portugal

Proyecto académico sin fines de lucro, desarrollado bajo la iniciativa de acceso abierto

UN PATRÓN DE INTERACCIÓN ENTRE DIAGRAMAS DE ACTIVIDADES UML Y SISTEMAS WORKFLOW

MARTA SILVIA TABARES*
JUAN DIEGO PINEDA**
ANDRÉS FELIPE BARRERA***

RESUMEN

Actualmente en los ambientes de desarrollo de software hay un gran interés en buscar y desarrollar técnicas que puedan integrar los sistemas transaccionales con los flujos de trabajo que soportan los procesos del negocio de las organizaciones. Sin embargo, en la industria del software no es común encontrar técnicas o prácticas que faciliten el desarrollo de los modelos del sistema en función de los procesos del negocio. En este artículo se define un patrón de desarrollo que estandariza la interacción entre diagramas de actividades de UML 2.0, que representan las operaciones de un sistema, y procesos del negocio automatizados bajo tecnologías *workflow*. La trazabilidad de dicha interacción se mantiene por medio de modelos de trazabilidad que controlan la evolución de las operaciones del negocio y del sistema. Para mostrar la aplicación del patrón se desarrolla un caso de estudio.

PALABRAS CLAVE: modelado de procesos del negocio; flujo de trabajo; transformación de modelos; trazabilidad; diagrama de actividades; proceso de desarrollo de software; UML 2.0.

AN INTERACTION PATTERN BETWEEN UML ACTIVITY DIAGRAMS AND WORKFLOW SYSTEMS

ABSTRACT

In software development environments there is a big interest to look and develop techniques that could integrate transactional systems with Workflow systems in order to support the business processes in organizations.

* Ph. D (c) en Ingeniería de Sistemas, Universidad Nacional de Colombia. Docente del Área de Ingeniería de Software y Bases de Datos, Escuela de Ingeniería de Antioquia. pfmstabare@eia.edu.co

** Ingeniero Informático, Escuela de Ingeniería de Antioquia. Analista de Negocio, Electronic Data Systems Corporation (EDS). juandiego61@gmail.com.

*** Ingeniero Informático, Escuela de Ingeniería de Antioquia. Analista de Investigación y Desarrollo, Choucair Testing S. A. anbarrera@gmail.com

Nevertheless, in the software industry it is not common to find techniques or practices that facilitate the development of system models according to the business processes. In this article we define a development pattern to standardize the interaction between UML 2.0 activity diagrams, which represent the operations of a system, and the business processes automated by means of Workflow technologies. The traceability of the above mentioned interaction is supported by means of traceability models that control the evolution of both operations of the business and of the system. To show the application of the pattern a case study is developed.

KEY WORDS: business process modeling; workflow systems; model transformation; traceability; activity diagram; software development process; UML 2.0.

1. INTRODUCCIÓN

Las empresas de desarrollo de software buscan constantemente técnicas y tecnologías que puedan adoptar con facilidad y que agilicen su proceso en la producción de artefactos de software. En la actualidad existe un gran interés en desarrollar o usar técnicas que puedan integrar los procesos del negocio con el proceso de desarrollo, sus modelos de desarrollo y los productos de software que se construyen.

En los últimos tiempos, herramientas tales como BPM (*Business Process Modeling*), SOA (*Service Oriented Architecture*) y *workflow* están tomando mayor fuerza, tanto en las organizaciones como en la industria del software. Esto ha llevado a que las organizaciones piensen en el desarrollo y la adquisición de aplicaciones de software adoptando el concepto de arquitecturas empresariales. Estas arquitecturas conducen al desarrollo de los procesos del negocio desde su concepción organizacional hasta su aplicación en soluciones, tales como SOA o flujos de trabajo (*workflows*) soportados por sistemas transaccionales.

Para lograr esta transición, los procesos del negocio se pueden modelar por medio de diagramas de actividades UML y automatizar por medio de las tecnologías asociadas a la administración de flujos de trabajo. Mientras los diagramas de actividades proveen los elementos de modelo que identifican las acciones que se realizan durante la ejecución de un proceso o actividad [1], los *workflows* pro-

veen la representación de un proceso del negocio facilitando la comunicación y colaboración entre los integrantes del grupo de trabajo o coordinando la secuencia de sus actividades, dependiendo de las reglas de negocio que definan el proceso de la organización [2].

Aunque esta tecnología se viene adoptando discretamente en algunas empresas de servicio, para las empresas de desarrollo de software no es tan evidente hacerla parte de los desarrollos de aplicaciones transaccionales, tales como ERP (*Enterprise Resource Planning*), financieras, recurso humano, entre otras.

Desarrollar software transaccional integrado a los flujos de trabajo de los procesos del negocio es una estrategia competitiva que puede traer beneficios para las empresas de desarrollo y sus clientes. Por esta razón, en este artículo se define un patrón de interacción entre diagramas de actividades de UML 2.0 [1], que representan las operaciones de un sistema, y los procesos del negocio modelados y automatizados bajo tecnologías *workflow* [3, 4]. El control de su evolución se realiza por medio de modelos de trazabilidad que facilitan el refinamiento o la correlación con otros elementos en diferentes niveles de abstracción. Para mostrar la aplicación del patrón se desarrolla un caso de estudio.

Este artículo está organizado de la siguiente forma. La sección 2 presenta definiciones de términos usados en el desarrollo del enfoque. La sección 3 define el patrón de interacción. La sección 4 presenta



el modelo de trazabilidad que soporta la evolución de la interacción. La sección 5 expone un caso de estudio que demuestra la aplicación del enfoque propuesto. La sección 6 presenta algunos trabajos relacionados. Finalmente, la sección 7 contiene las conclusiones y trabajo futuro.

2. CONCEPTOS BÁSICOS

2.1 Procesos del negocio

Los procesos del negocio determinan la forma como un conjunto de actividades pueden lograr los objetivos específicos de una organización describiendo su forma de operar, tomar decisiones y establecer el flujo de la información necesario entre los participantes del proceso. Cada proceso es motivado por un evento interno o externo a la organización; se procesa la información de entrada, se manipulan los objetos necesarios, se toman las decisiones requeridas y se generan la información y los eventos de salida. Los procesos están restringidos por un conjunto de reglas de negocio, que determinan las políticas y la estructura de la información del negocio [2].

Un proceso puede estar compuesto de subprocesos o actividades, reglas del negocio y de flujos de control (figura 1).

Los procesos del negocio son la base de todo buen desarrollo de software y normalmente se identifican en la primera fase del proceso. Facilitan la abstracción del problema para construir la solución informática, y su análisis está orientado a la identificación de los objetivos generales, los requisitos del software, la selección del tipo de sistema de información y la arquitectura sobre la cual se debe construir el sistema.

2.2 Tipos de sistemas de información

Algunos de los tipos de sistemas de información más representativos son:

- *Transaccionales*. Son sistemas que automatizan las tareas y los procesos operativos del negocio, por medio de interfaces de usuario-máquina, funciones de control y operaciones sobre las bases de datos.
- *Apoyo a la toma de decisiones*. Son sistemas de almacenamiento y consultas especializadas de información para soportar la toma de decisiones en los mandos medios y en la alta gerencia de las organizaciones.
- *Flujos de trabajo (workflows)*. Automatizan el flujo de la información que soporta el proceso administrativo del negocio, por medio de roles, mensajes, actividades y notificaciones, entre otros.

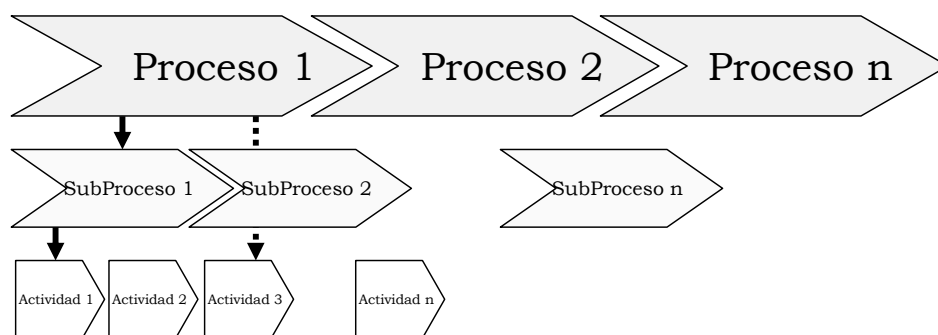


Figura 1. Un esquema de representación de los procesos del negocio, subprocesos y actividades

La selección del tipo de sistema de información por lo general es excluyente, es decir, la aplicación es del tipo transaccional o una de las otras dos. Sin embargo, un sistema transaccional puede ser parte de la automatización de un flujo de un proceso de negocio y su implementación ser complementaria durante el mismo proceso de desarrollo (por lo general son independientes).

Normalmente, en el análisis de una aplicación transaccional se identifican las operaciones o actividades del negocio, p. ej. “registrar la idea de un negocio” (referente al caso de estudio de la sección 5, véase la figura 9) y, a su vez, esta operación puede generar tanto información para la administración del proceso del cual forma parte como actividades dentro del flujo v. gr. “notificar registro”. Estos requisitos son parte del mismo proceso del negocio, pero la diferencia está en su implementación.

2.3 Flujos de trabajo (*workflows*)

Un *workflow* (WF) es un sistema para gestionar los procesos del negocio con la integración de

subprocesos y actividades que facilitan su operación, así como la automatización y colaboración basada en procesos. Permite el modelado, la automatización y la mejora continua del proceso del negocio, encaminando información de cualquier tipo, según el usuario haya definido las reglas del negocio. La integración con sistemas transaccionales o de otro tipo debe ser realizada manual o automáticamente, dependiendo de las herramientas o aplicaciones que pueda componer. En otras palabras, describe la ruta controlada de la información, ya que está diseñado para conseguir los objetivos de procesamiento de alguna clase, como transformación física, provisión de servicio o proceso de información. Stohr *et al.* en la figura 2 presentan un mapa conceptual de cómo un proceso del negocio puede ser automatizado desde su definición y su uso en un sistema (motor) administrador de *workflow* [4].

Tecnologías tales como Oracle Workflow [5], Microsoft Workflow [6] y Skelta [7], entre otras, soportan el diseño y la construcción de los flujos de trabajo de los procesos del negocio. Un *workflow* está compuesto por los siguientes elementos:

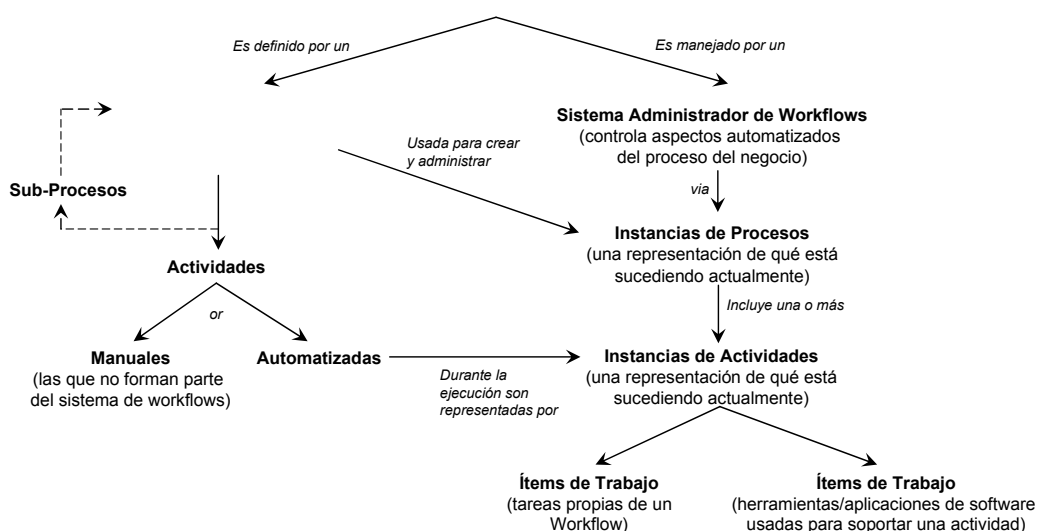


Figura 2. Un proceso de negocio desde dos puntos de vista, su definición y los sistemas de *workflow* (adaptación de [4])



- **Ítem.** Es una clasificación de un componente que hace parte de un proceso. En otros términos, es todo objeto/documento que requiera tramitarse por medio del *workflow*.
 - **Atributos.** Son las propiedades que describen cada ítem identificado en el *workflow*. Por ejemplo, si un ítem es “Registrar Idea de Negocio”, entonces algunos atributos de ese ítem pueden ser la descripción de la idea y el sector de negocio.
 - **Proceso.** Está compuesto de actividades (íconos) y transiciones (líneas de conexión o flechas).
 - **Actividad.** Es una unidad de trabajo que contribuye a la ejecución de un proceso. Puede ser una notificación, una función, un evento, un proceso o subproceso.
 - **Notificación.** Es el resultado de cada actividad dentro del flujo de trabajo, es decir, es la forma de anunciar los cambios de estado de los objetos del sistema que interesan a los actores del proyecto. Así, en la figura 3 hay varias notificaciones, entre ellas: “Aprueba idea de negocio”.
 - **Función.** Actividad usada para ejecutar pasos totalmente automatizados en el proceso. Los pasos automatizados de ordinario los realiza una aplicación transaccional.
 - **Evento.** Representa un acontecimiento del negocio dentro de un proceso. Un evento puede recibir, atender o enviar un acontecimiento del negocio.
- Las actividades de función y de evento tienen un costo asociado. El costo es un valor que representa el tiempo (número de segundos) que la actividad tarda en recibir una función, un evento o una notificación para darle continuidad al flujo. El tiempo es calculado por el motor del *workflow*.
- Mensajes.** Se refiere a una actividad de notificación asociada a un ítem que puede enviar a un usuario o rol.

En la figura 3 se ilustra un ejemplo de un *workflow* construido con la tecnología Oracle Workflow para soportar el proceso de *Tramitar Idea de Negocio* (ver caso de estudio en la sección 5).

2.4 Diagramas de actividades

Los diagramas de actividades (DA) son parte de los diagramas de comportamiento UML, que describen la funcionalidad del software en un nivel alto de abstracción. En la actualidad, los procesos

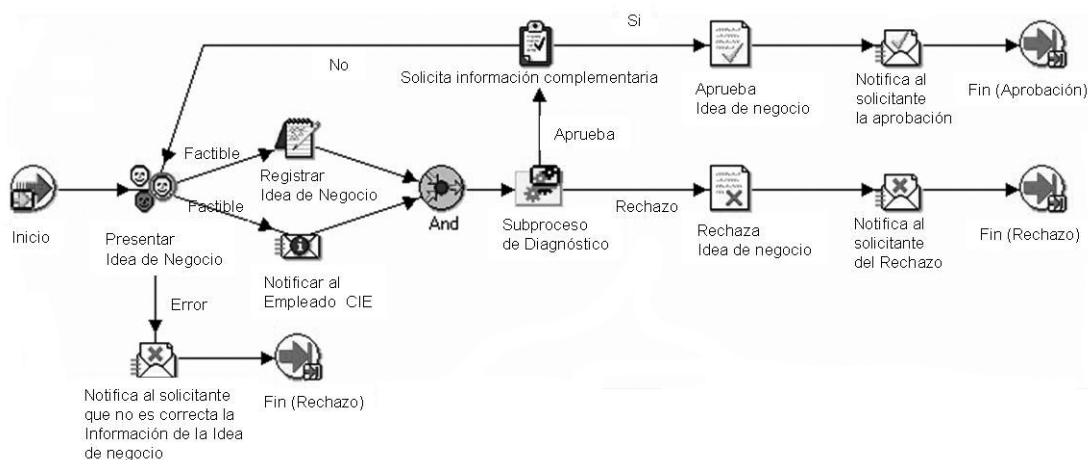


Figura 3. Ejemplo de un *workflow* [5] que automatiza el proceso de *Tramitar Idea de Negocio* [9]

del negocio se pueden representar en diagramas de actividades de UML 2.0 [1, 8] (es decir, procesos, subprocesos y actividades), conforme con las técnicas de modelado de procesos de negocio definidas por la BPMI¹, la cual es parte del OMG² y se formaliza en el documento *Business Process Modeling Notation* –BPMN–³.

Los diagramas de actividades describen los flujos de control que son creados, desde los modelos de procesos del negocio hasta los modelos de operación del sistema descritos por elementos tales como: modelos de casos de uso (flujos básicos, subflujos y flujos alternos), clases, operaciones, interfaces, componentes y colaboraciones. Un diagrama de actividades está compuesto por elementos de modelo identificados como nodos de acción (actividad/acción, llamada a actividad externa o subproceso), nodos de control, nodos objeto, flujos de control y flujos de objeto [8] (figura 4).

3. PATRÓN DE INTERACCIÓN

En este enfoque se define un patrón⁴ que determina la forma como debe realizarse la interacción entre diagramas de actividades de UML 2.0 y los *workflows* que automatizan los procesos del negocio. Con este patrón se busca definir los elementos y la forma como deben correlacionarse los modelos dinámicos de comportamiento que representen las operaciones del sistema transaccional con los procesos del negocio, sus subprocesos y actividades.

La figura 5 ilustra el patrón por medio de un diagrama de actividades que describe la interacción que debe realizarse entre los nodos de un diagrama

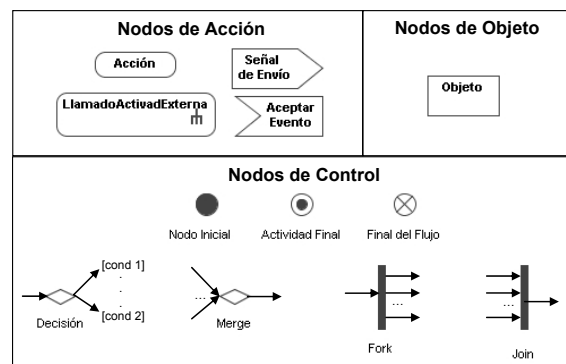


Figura 4. Elementos de los diagramas de actividades en UML 2.0

de actividades con las actividades de un sistema *workflow*.

El procedimiento para usar el patrón de interacción es el siguiente.

Paso 1. *Adaptar interacción del nodo de acción:* en cada diagrama de actividades se identifican los nodos de acción que permiten operacionalizar el sistema transaccional, se evalúa la información que procesa y se establece la relación con actividades de función del *workflow* por medio de acciones de señal de envío.

Los diagramas de actividades y sus nodos de acción son adecuados para interactuar con los *workflows* que automatizan el proceso, realizando tres tareas.

- Identificar los siguientes datos: nombre de la acción, tipo, actores (de entrada, ejecución y salida) y operaciones transaccionales.

¹ Business Process Management Initiative: <http://www.bpmi.org>.

² The Object Management Group: www.omg.org.

³ Business Process Management Notation v1.0 (2006): <http://www.bpmn.org/Documents/>.

⁴ Un patrón describe un problema que ocurre una y otra vez en nuestro ambiente, de tal forma que describe una solución para el problema que se puede usar n veces, sin tener que hacer lo mismo dos veces [16].

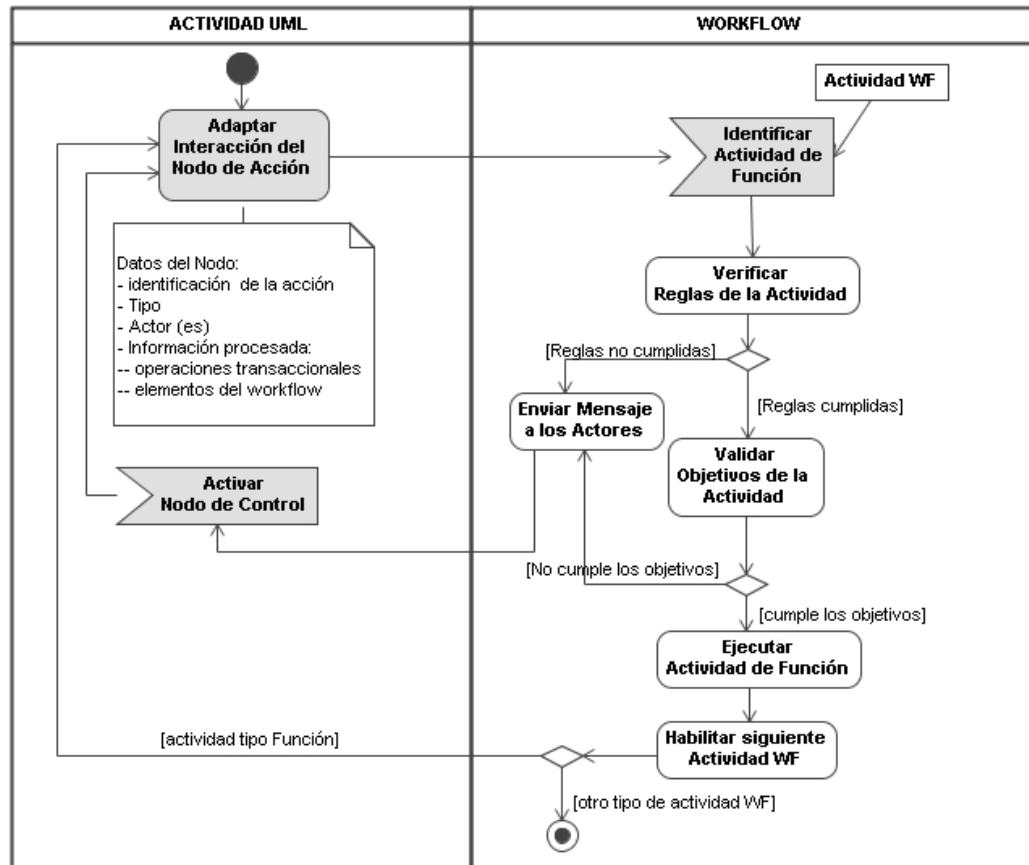


Figura 5. Patrón de interacción entre un diagrama de actividades y un sistema *workflow*

- Adicionar un nuevo actor (participante) del flujo con una banda funcional (*swimlane*) que se identifique como *workflow* (figura 6, parte (a)).
- Complementar las acciones del flujo de actividades con nodos de acción de señal de envío que indiquen interacción con el *workflow*. Cada uno de estos se encargará de establecer la interacción entre el evento del nodo de acción descrito y las acciones del *workflow* (figura 6, parte (b)).
- Este nodo se encarga de gestionar la información de control del flujo con los siguientes parámetros:
- **FlujoEntrada.** Se refiere a los valores o información que retorna la operación del sistema transaccional para la actividad del *workflow* (representada por un nodo de acción).
- **ActividadRecibe.** Se refiere a la actividad del *workflow* que se correlaciona con la operación del flujo de entrada. Se gestiona la siguiente información: tipo (p. ej. Función), nombre y medio (forma que se utiliza para realizar la actividad en el *workflow*; puede ser un formulario u otro como el servicio de correo electrónico).
- **ActividadGenerada.** Se refiere a una actividad del *workflow* que se correlaciona con la acción

de señal. Generalmente son notificaciones o eventos. Se gestiona la siguiente información: tipo (p. ej. notificación, evento), mensaje y medio.

- **FlujoSalida.** Se refiere a los valores o información que conecta o activa una nueva actividad en el *workflow*. Suelen ser valores de verdad con una etiqueta de identificación del flujo.

Para identificar la necesidad de la interacción se sugiere verificar la siguiente información:

- ¿Las operaciones de un proceso o subproceso generan información para la administración del negocio, como una notificación, un evento, una función ejecutable?
- ¿Hay transacciones que se realizan entre varios actores que deban soportarse por medio de un flujo de información controlada para la toma de decisiones?
- Crear un nodo de actividad estructurada llamado *Gestionar Acciones del Workflow* para el nuevo actor *workflow*, de tal forma que reciba y despache la información necesaria para interactuar con otras acciones del diagrama de actividades (figura 6, parte (c)).

El nodo *Gestionar Acciones del Workflow* recibe la información de los nodos de señal y la valida por medio de las acciones indicadas en la figura 5 para el actor *workflow* (descritas en los siguientes pasos). Si la información está incompleta o no cumple con

los objetivos establecidos en el *workflow*, activa un nodo de control en el diagrama de actividades que se encarga de reportar al nodo de acción los problemas encontrados. Esto permite controlar la interacción entre los modelos y administrar los cambios en cualquiera de los dos. Además, la interacción se debe soportar en modelos de trazabilidad que mantengan los vínculos de trazado entre los nodos de control y otros elementos de modelo tales como clases y operaciones, entre otros.

Paso 2. Identificar la actividad de Función: acción del *workflow* que está a la espera de que ocurra la acción en el diagrama de actividades para identificar una actividad de función preestablecida en el *workflow*. Esta acción recibe la información necesaria para identificar la actividad en el *workflow*.

Paso 3. Verificar Reglas de Actividad: acción que comprueba que la información recibida permita evaluar las reglas definidas dentro de la actividad (tipo función) del *workflow*. Las reglas de una actividad por lo regular establecen la realización de ciclos, evaluación de decisiones, seguimiento de rutas o ramales dentro del flujo y otras reglas propias del motor del *workflow*. Si no se puede cumplir alguna de las reglas, se envía un mensaje a los actores (participantes) del proceso y se retorna información a una acción de control del diagrama de actividades.

Paso 4. Validar los objetivos de la actividad: el analista de desarrollo al crear las actividades del *workflow* define los objetivos que debe cumplir cada una de ellas. Los objetivos pueden ser ingresar información complementaria, establecer información para la notificación de aceptación o rechazo, identificar el medio de envío de los mensajes de notificación o evento, entre otros. Estos objetivos facilitan a los participantes tomar decisiones para continuar o terminar el proceso del negocio. Si esta información es incorrecta o incompleta, se envía un mensaje a los actores (participantes) del proceso, y se retorna información a una acción de control del diagrama de actividades.

DIAGRAMA DE ACTIVIDADES

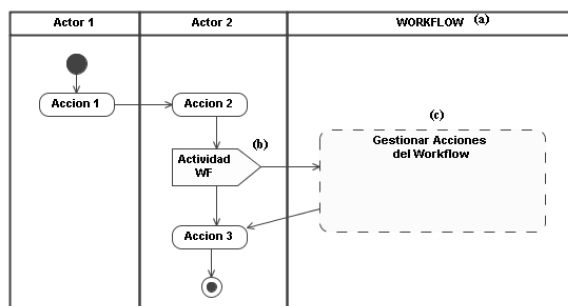


Figura 6. Adaptación de un diagrama de actividades para interactuar con un *workflow*



Paso 5. Ejecutar Actividad de Función: el *workflow* ejecuta la actividad de acuerdo con las reglas verificadas y los objetivos definidos.

Paso 6. Habilita siguiente actividad Workflow: al terminar una actividad, el *workflow* automáticamente habilita otra actividad que continúa o termina el proceso del negocio.

4. TRAZABILIDAD

La trazabilidad (de la palabra *traceability* en inglés) o rastreabilidad de la interacción se puede extender a otros elementos de modelo que se construyan durante el proceso de desarrollo, por ejemplo, casos de uso y clases. Para lograr esto se define un modelo de trazado que está compuesto por elementos trazables (elementos UML o definidos con otro metamodelo) y vínculos de trazado por medio de los cuales se hará el control de la evolución y transformación de los procesos del negocio o los requisitos del sistema durante el proceso de desarrollo. Los desarrolladores crean el modelo de trazado como un perfil de desarrollo en el nivel del metamodelo; así es posible tener diferentes modelos para diferentes tipos de proyectos [10].

Los elementos trazables se identifican por medio de un rol que les da responsabilidades de trazado durante el proceso de desarrollo. Los roles definidos son:

- Eje de trazado (`<<axisTracing>>`). Los elementos identificados por este estereotipo tienen asociadas tareas de control de la transformación para generar los predecesores y sucesores y la gestión del cambio (análisis de impacto y propagación).
- Predecesores (`<<predecessor>>`). Los elementos identificados por este estereotipo preceden a los ejes del trazado y permiten mantener la traza hacia atrás (cuando sea necesaria una ingeniería inversa).
- Sucesores (`<<successor>>`): Los elementos identificados por este estereotipo suceden a los ejes del trazado y permiten mantener la traza hacia delante.

El modelo de trazado ilustrado en la figura 7, en particular, se orienta al soporte de procesos de desarrollo centrados en casos de uso, como lo es el Proceso Unificado [8]. Para soportar la transformación de los procesos del negocio se definen

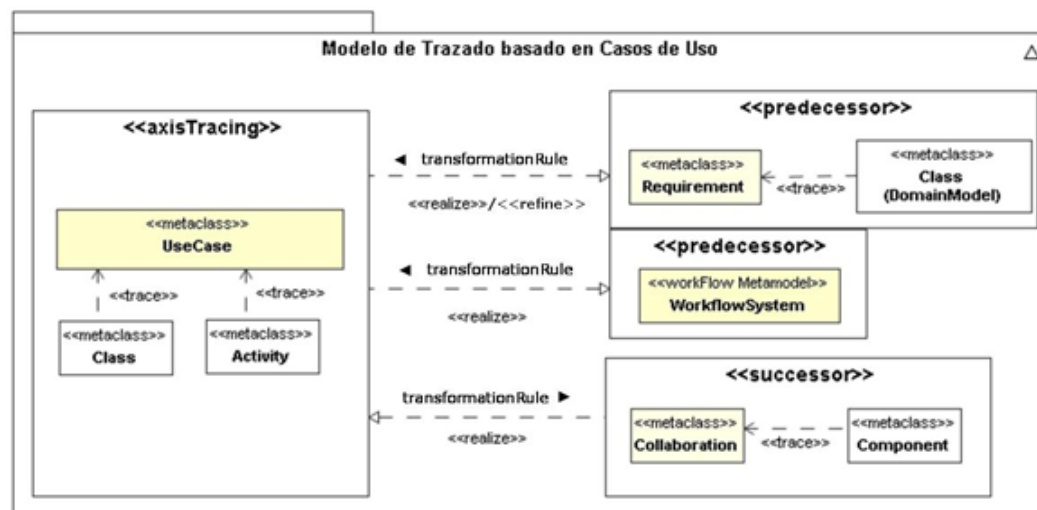


Figura 7. Modelo de trazado basado en casos de uso (definición con base en [10])

como ejes del trazado los metaelementos UML: *UseCase*, *Activity* y *Class* (interrelacionan por medio de una relación de abstracción <<trace>>). Estos se encargarán de controlar el patrón de interacción y otras instancias de los elementos definidos como predecesores y sucesores.

Los predecesores se definen en función de los metaelementos *BusinessObject*, *Requirement* (extensiones del metamodelo de UML) y *WorkflowSystem* (del *WorkFlow* metamodel) cuyas instancias definen los procesos del negocio y los requisitos en el espacio del problema. Los sucesores se definen en función de los metaelementos UML *Collaboration* y *Component*

(interrelacionan por medio de una relación de abstracción <<trace>>) que pueden representar los procesos del negocio en el nivel del diseño.

El caso de uso como elemento principal de los ejes del trazado se relaciona con predecesores y sucesores por medio de las relaciones de trazado <<refine>> y <<realize>>, lo que permite obtener transitividad, entre otros elementos, de ejes, predecesores y sucesores. Por ejemplo, la correlación de *Activity* (*Diagrama de Actividad*) y el *Workflow* (*Flujo de Trabajo*) a través del caso de uso: *UseCase* traza a *Activity* y *UseCase* realiza al *WorkflowSystem*; así, por transitividad *Activity* realiza al *WorkflowSystem* (ver expresión 1):

$$\text{UseCase}_{\langle\langle\text{trace}\rangle\rangle\text{Activity}} \wedge \text{UseCase}_{\langle\langle\text{realize}\rangle\rangle\text{WorkflowSystem}} \rightarrow \text{Activity}_{\langle\langle\text{realize}\rangle\rangle\text{WorkflowSystem}} \quad (1)$$

Los vínculos de trazado son relaciones de abstracción y realización UML que usan los estereotipos <<trace>>, <<refine>> y <<realize>>. Las reglas de transformación están asociadas a los vínculos de trazado entre los ejes del trazado y sus predecesores y sucesores. La tabla 1 presenta algunas relaciones de trazado y sus reglas de transformación (p. ej., R_a, \dots, R_z) que pueden definirse en un lenguaje de transformación como ATL [15].

5. CASO DE ESTUDIO

El problema que se toma como caso de estudio corresponde a un sistema de información para un centro de innovación y emprendimiento (CIE) cuyo objeto principal es gestionar los proyectos de los emprendedores, desde que se registran, y evalúa una idea de negocio hasta que se desarrollan una serie de actividades del proyecto para lograr los

Tabla 1. Definición de los vínculos de trazado y sus reglas de transformación

Nombre	Vínculo fuente del trazado	Vínculo fuente del trazado	Regla de transformación
<<trace>>	<<axisTracing>>Activity	<<axisTracing>>UseCase	$\{R_a, R_b\}$
<<trace>>	<<axisTracing>>Class	<<axisTracing>>UseCase	$\{R_c, R_d\}$
<<realize>>	<<axisTracing>> UseCase	<<axisTracing>> Requirement	$\{R_n\}$
<<realize>>	<<axisTracing>> UseCase	<<axisTracing>> BusinessObject	$\{R_m\}$
<<realize>>	<<axisTracing>> WorkflowSystem	<<axisTracing>> UseCase	$\{R_i, R_z\}$
...



objetivos de los planes estratégicos definidos para el negocio propuesto [9].

5.1 Identificación de los procesos del negocio

La gestión del centro de innovación está compuesta por tres procesos del negocio: Gestionar Emprendedor, Tramitar Idea de Negocio y Administrar Proyecto. Para mostrar la aplicación del patrón de interacción, sólo se trabaja con el proceso *Tramitar Idea de Negocio*.

La figura 8 ilustra el diagrama de actividades del proceso *Trámite de Ideas de Negocio* que describe el flujo de actividades que existe entre los actores: Emprendedor y Empleado CIE. El proceso empieza cuando el emprendedor presenta una idea de negocio, luego el Empleado CIE evalúa que la información presentada esté correcta para que el emprendedor proceda a registrar la idea propuesta. De esta forma, el Empleado CIE hace un diagnóstico y determina si la idea queda en trámite, enviando sugerencias de complementación, si lo considera necesario, o la aprueba o la rechaza. Si la idea es aprobada, se notifica al emprendedor y genera la

información necesaria para activar el proceso de Formalizar Proyecto. El proceso se mecaniza por medio de un *workflow* para automatizar el flujo de información y facilitar la toma de decisiones entre los diferentes participantes del sistema.

5.2 Interacción de los nodos de acción

En la figura 9(a) se muestra el diagrama de actividades que soporta la operación de *Tramitar Ideas de Negocio* en el sistema transaccional (este diagrama soporta la especificación del caso de uso del mismo nombre). Los nodos de acción identificados, numerados del 1 al 4, se correlacionan con las actividades de función del *workflow* *Tramitar Idea de Negocio* (figura 9(b)). La correlación por nombre se soporta en la tabla 2.

Es importante anotar que los nombres de los nodos de actividades y las actividades WF deben ser lo más similares que se pueda. Esto ayudará al seguimiento de la traza por nombre del elemento de modelo y la propagación de los cambios que afecten el *workflow* durante el proceso de desarrollo, ante todo cuando los nodos de acción se implementen

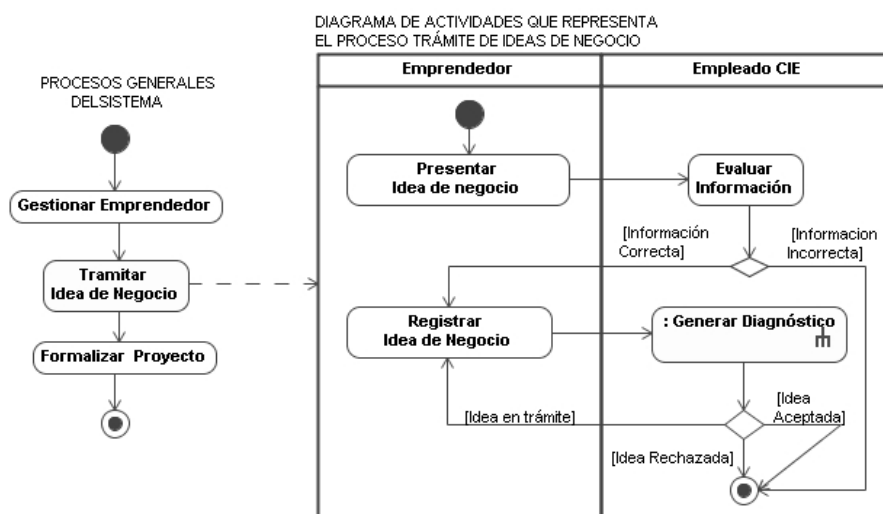


Figura 8. Procesos generales del centro de innovación y emprendimiento

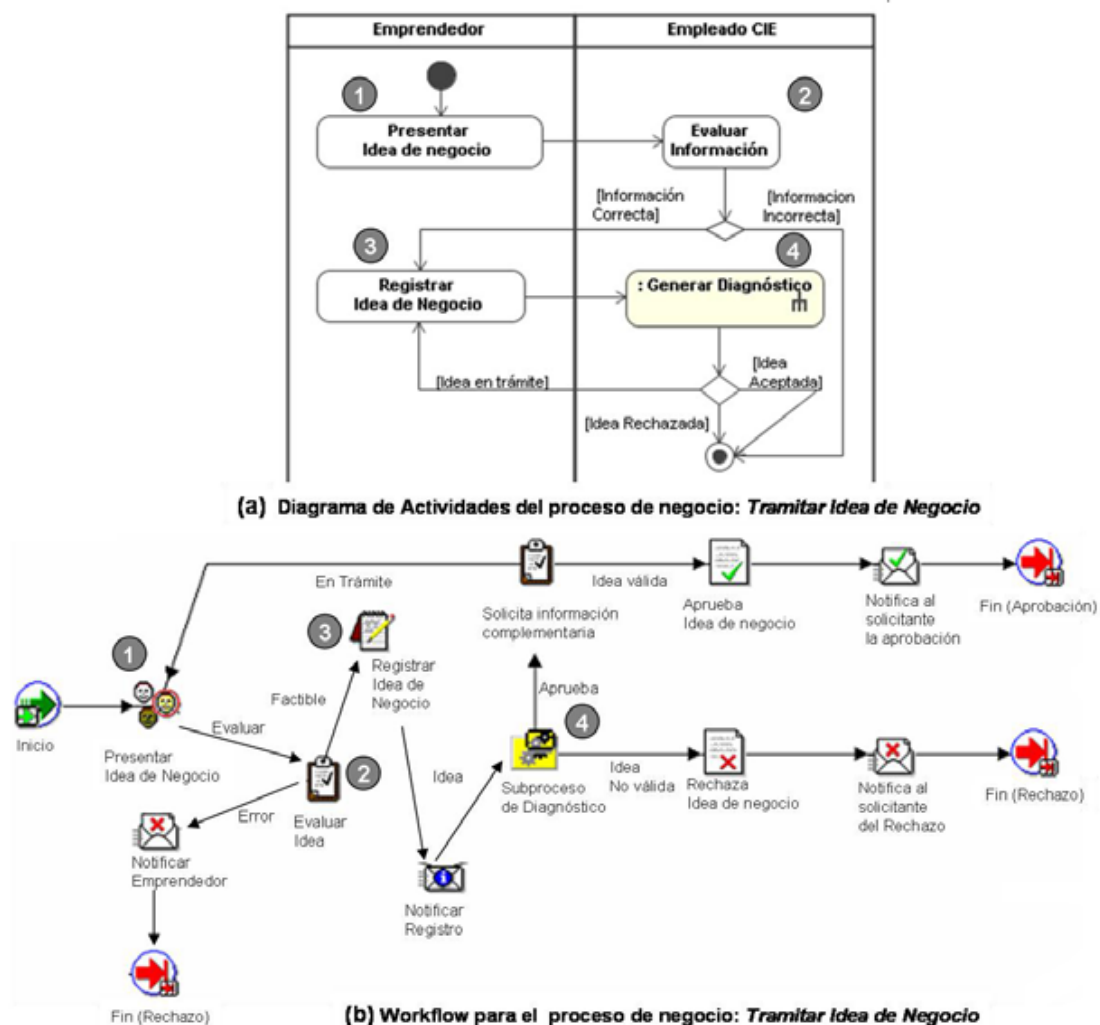


Figura 9. Correlación de nodos de acción con actividades de función del *workflow*

Tabla 2. Elementos de modelo que participan en la correlación

Diagrama de actividades (DA)	Workflow (WF)
Presentar idea de negocio	Presentar idea de negocio
Evaluar información	Evaluar idea
Registrar ideas de negocio	Registrar idea de negocio
Generar diagnóstico	Empleado CIE valida factibilidad

en otros elementos de modelo como clases, operaciones, componentes o conectores, y, finalmente, en el código.

Desde cada nodo de acción se verifica qué notificaciones, eventos u otras actividades pueden generar información para la toma de decisiones dentro del flujo de trabajo. Por ejemplo, a partir de la acción *Evaluar Información*, el Empleado CIE genera una de dos notificaciones, dependiendo del grado de asertividad de la idea de negocio presentada. Si hay error, se notifica al Emprendedor del error y rechazo



de la idea. Si la idea inicial es factible, el Emprendedor la registra completamente y se le notifica al Empleado CIE para su evaluación.

Por lo tanto, en el diagrama de actividades del proceso *Tramitar Idea de Negocio* se crean cuatro acciones de señales de envío que indican la interacción entre un nodo de acción y una actividad del *workflow* (en este caso, de *notificación*) que refleja el flujo de las decisiones que intervienen en las acciones del proceso (figura 10).

Cada acción de señal de envío representa una acción en el *workflow* y gestiona los datos del nodo de acción que la genera (para este ejemplo cada acción genera datos de envío, pero esto no quiere decir que todo nodo de acción tenga que interactuar con una actividad *workflow*).

5.3 Trazabilidad

La trazabilidad entre los diagramas de actividades y los *workflows* se consigue inicialmente por

medio de la información que se genera durante la interacción. Cualquier cambio que ocurra en uno de los dos modelos durante el proceso de desarrollo se gestiona con la información registrada en la traza.

La tabla 3 presenta dos de las actividades trazadas durante la interacción del proceso *Trámite de Idea de Negocio*.

Una instancia del modelo de trazado (presentado en la sección 4) se ilustra en el modelo de trazabilidad Sistema CIE v. 1.0.0 de la figura 11. Este se genera a partir de la creación del caso de uso *Tramitar Idea de Negocio* o la definición del proceso *Tramitar Idea de Negocio* y su interacción entre el diagrama de actividades con el *workflow*. Las reglas de transformación asociadas a estos elementos son ejecutadas para generar sus predecesores y sucesores. Por la relación de traza del caso de uso con *Actividad* y el *Workflow*, dichas reglas de transformación utilizan la información de la interacción para complementar directamente elementos tales como el proceso del negocio *Tramitar Idea de Negocio* y la clase *Idea de Negocio*.

DIAGRAMA DE ACTIVIDADES QUE REPRESENTA LA INTERACCIÓN DEL PROCESO TRÁMITE DE IDEAS DE NEGOCIO CON SU WORKFLOW

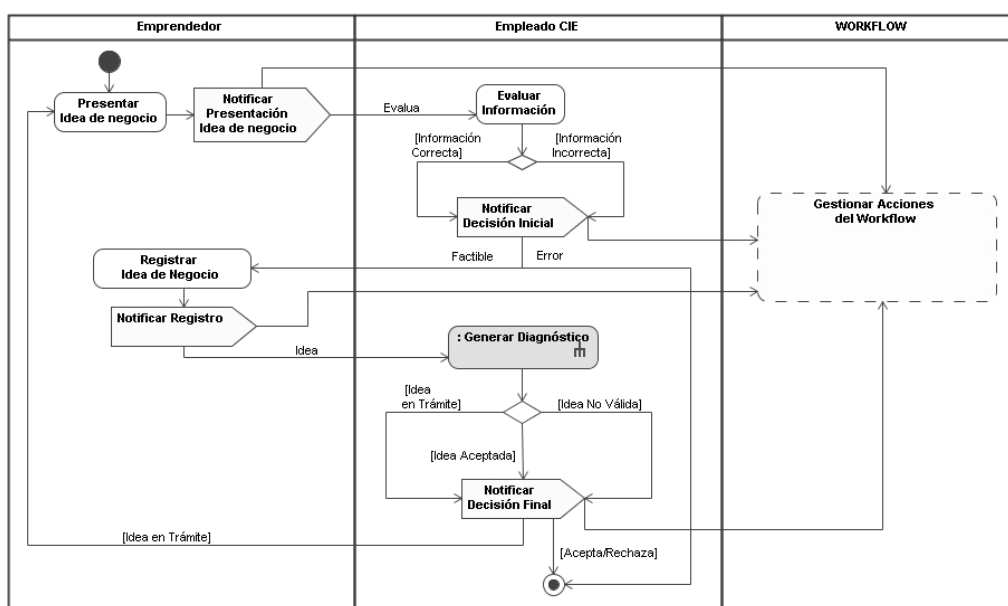


Figura 10. Definición de la interacción del diagrama de actividades “Trámite de Ideas de Negocio” y el workflow “Trámite de Idea de Negocio”

Tabla 3. Algunas trazas de correlación entre el diagrama de actividades del proceso de negocio y el sistema *workflow* que soporta el mismo proceso

Diagrama de actividades	Sistema workflow
Subproceso/Actividad/Acción: Emprendedor::PresentarIdeaNegocio[Nodo Acción]	Flujo entrada: CIESystem::Emprendedor::PresentarIdeaNegocio(NombreEmprendedor, Idea)
Operación: IngresoInicial_IdeaNegocio	ActividadRecibe: Tipo: Función Nombre: Presentación Idea de Negocio
AcciónSeñalEnvío: Notificación: Idea de Negocio	ActividadGenerada: Tipo: Notification (IdeaNegocio) Mensaje: NombreEmprendedor, Idea (descripción) Media: servicio e-mail. FlujoSalida: Enviado/Recibido (booleano).
Subproceso/Actividad/Acción: EmpleadoCIE::EvaluarInformacion[NodoAccion]	Flujo entrada: CIESystem::EmpleadoCIE::EvaluarInformacion(NombreEmpleadoCIE, Idea)
Operación: IngresoInicial_IdeaNegocio	ActividadRecibe: Tipo: Función Nombre: EvaluarInformacion Medio: Formulario Evaluación(estados: correcta, incorrecta)
AcciónSeñalEnvío: Notificación: Decisión Evaluación	ActividadGenerada: Tipo: Notification (Decisión) Mensaje: NombreEmpleadoCIE, ResultadoEvaluacion(descripción) Media: servicio e-mail. FlujoSalida: Estado: Error/Factible.

Si se crean instancias de elementos, el proceso del negocio o los requisitos antes del caso de uso o la interacción, las reglas de transformación de dichos elementos se ejecutarán de acuerdo con la relación de trazado establecida entre los elementos de modelo y el eje del trazado.

Una vez generada la primera versión del modelo de trazabilidad para el Sistema CIE v.1.0.0, cualquier cambio en el proceso del negocio se controla por medio de la relación de traza entre el casos de uso Tramitar Idea de Negocio, la clase *IdeaNegocio*, y la interacción entre el diagrama de actividades y el *workflow* Tramitar Idea de Negocio que se puede lograr por transitividad de estos elementos trazables.

6. TRABAJOS RELACIONADOS

García *et al.* presentan una forma sistemática para obtener el modelo de casos de uso y el modelo conceptual, desde los modelos del negocio basado en diagramas de actividades UML [11]. El enfoque es conceptual y descriptivo, no formaliza la interacción y la trazabilidad entre los modelos que se generan desde la identificación de los modelos de procesos.

Rusell *et al.* proporcionan una evaluación de las capacidades de los diagramas de actividades de UML 2.0 (fortalezas y debilidades) cuando se usan para el modelado de procesos del negocio. La evaluación se realiza usando los patrones del flujo de trabajo [12], pero no provee una conexión directa a posibles interacciones con sistemas *workflow*.

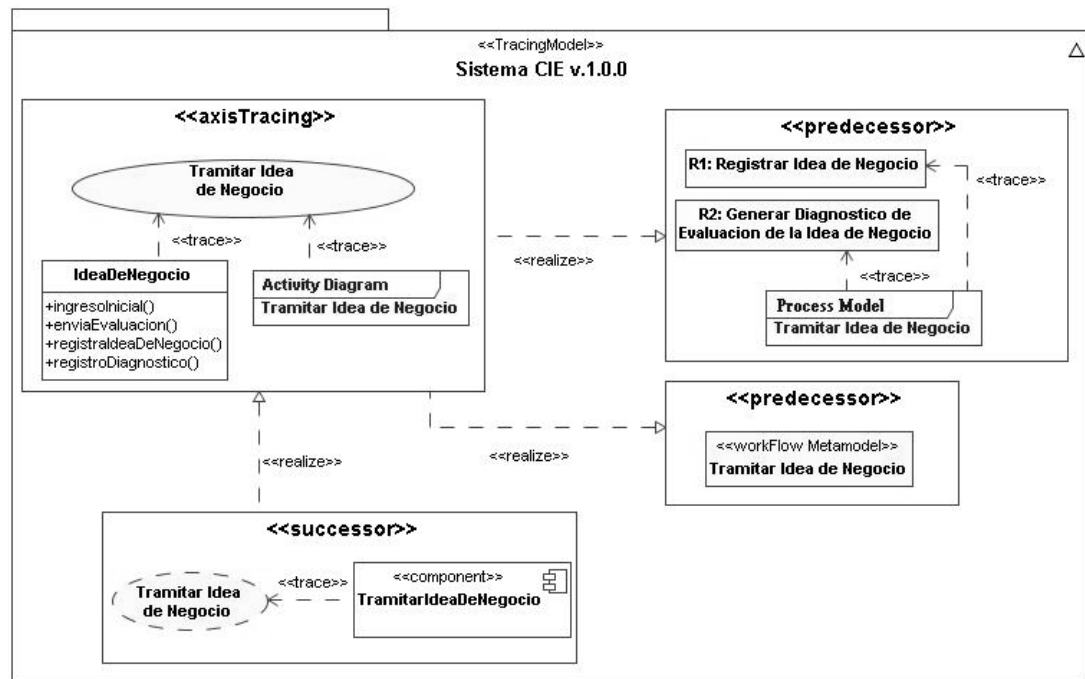


Figura 11. Modelo de trazabilidad para el proceso para el caso de uso Tramitar Idea de Negocio

7. CONCLUSIONES Y TRABAJO FUTURO

En este artículo se define un patrón para gestionar la interacción entre un diagrama de actividades UML 2.0 y sistemas *workflow*. Este enfoque proporciona varios elementos de soporte al proceso de desarrollo de software y su interacción con los procesos del negocio tales como:

- La formalización de la gestión de los procesos del negocio por medio de los sistemas *workflow* y su modelado en diagramas de actividades UML 2.0, para soportar su correlación con artefactos de desarrollo de los sistemas transaccionales u otro tipo de sistemas.
- La automatización de los procesos del negocio por medio de sistemas *workflow* integrada a los sistemas transaccionales que participan o proveen información al proceso del negocio. Los

workflows ayudan a controlar la gestión de la información que fluye en un proceso de negocio y la interrelación de sus actividades con la información necesaria para tomar la decisión correcta de forma eficiente.

- El soporte de un modelo de trazado para controlar la interacción de los procesos del negocio y los modelos de desarrollo que lo soportan.

Aunque el modelado de los procesos del negocio es un punto vulnerable en toda organización, su automatización en sistemas *workflow* facilita el control y mantenimiento del flujo de información y su correlación directa con los sistemas de información que se desarrollen o se implanten en las áreas del negocio.

En la actualidad, se desarrolla un modelo de transformación bajo arquitecturas orientadas a modelos (Model-Driven Architecture) [13], que toma el

patrón de interacción como base para la definición de los modelos independientes de la computación (*Computation Independent Models –CIM–*), donde su transformación se define y controla por medio de patrones de trazabilidad como el que se presenta en este artículo. Además, se trabaja en la evaluación y valoración del impacto del cambio desde los procesos del negocio y los diferentes modelos de desarrollo del sistema.

En un trabajo futuro, este patrón se utilizará como elemento de soporte para facilitar la implementación de los procesos en motores Workflow o BPEL (<http://www.ibm.com/developerworks/library/specification/ws-bpel>), de tal forma que pueda usarse para la transición hacia las arquitecturas empresariales y los sistemas SOA [14]. Además, la interacción que define el patrón también puede ayudar a definir catálogos de procesos establecidos por áreas del negocio, que se puedan automatizar y formen parte del análisis de arquitecturas empresariales.

REFERENCIAS

1. UML-OMG, Unified Modeling Language: superstructure v. 2.0. (2005).
2. Zhu, L. Osterweil, L. J.; Staples, M. and Kannengiesser, U. (2008). Challenges observed in the definition of reference business processes. LNCS Springer, Berlin/Heidelberg, p. 95-107.
3. White S. A. (2004). Process modeling notations and workflow patterns. BPTrends (March).
4. Stohr, E. A.; zur Muehlen, M. and Zhao, J. L. (2002). Workflow and process automation in the age of e-business: technical, organizational and educational aspects. In: HICSS-35 tutorials, Advanced Seminars, and Workshops. Waikoloa, Hawaii, United States.
5. Oracle workflow guide release 2.6.2. Oracle Corporation. http://download-uk.oracle.com/docs/cd/B10501_01/Workflow.920/a95265/toc.htm. [citado: 7 octubre 2008].
6. Chappell, D. Introducing Microsoft Windows workflow foundation: An early look. Ago 2005. <http://msdn.microsoft.com/en-us/library/aa480215.aspx>.
7. Skelta BPM.Net, <http://www.skelta.com/>. [citado: 7 octubre 2008].
8. Arlow, J. and Neustad, I. (2005). UML 2 and the Unified Process: Practical object-oriented analysis and design (2nd ed.). Addison-Wesley Object Technology Series.
9. Barrera, A. F. y Pineda, J. D. (2008). Análisis y diseño del sistema de información del Centro de Innovación y Emprendimiento de la Escuela de Ingeniería de Antioquia. Trabajo de grado. Escuela de Ingeniería de Antioquia.
10. Tabares, M. S., Anaya, R., Moreira, A., Araújo, J. and Arango, F. (2008). Traceability models to control an aspectual model-driven development. Proceedings of the Twentieth International Conference of Engineering & Knowledge Engineering. ISBN 1-891706-22-5.
11. García, J., Ortín-Ibáñez, M.-J., Moros, B. and Nicolás, J. (2002). Transforming the OOram three-model architecture into a UML-based process. Journal of Object Technology 1(4): 119-135.
12. Russell, N.; van der Aalst, W.M. P.; ter Hofstede, A. H. M. and Wohed, P. (2006). On the suitability of UML 2.0 activity diagrams for business process modeling. APCCM ,06: Proceedings of the 3rd Asia-Pacific Conference on Conceptual Modelling. Vol. 53, p. 95-104.
13. MDA-Guide (2003). OMG Document v1.0.1. www.omg.org.
14. Krafzig, D.; Banke, K. and Slama, D. (2004). Enterprise SOA: service-oriented architecture best practices. Prentice Hall PTR. ISBN 9780131465756.
15. Jouault, F. and Kurtev, I. (2005). Transforming models with ATL. In: Proceedings of the Model Transformations in Practice Workshop at MoDELS 2005, Montego Bay, Jamaica.
16. Alexander, C.; Ishikawa, S.; Silverstein, M.; Jacobson, M.; Fiksdahl-King, I. and Angel, S. (1989). A pattern language. Oxford University Press, New York.