

El Modelo del Negocio como base del Modelo de Requisitos¹

María José Ortín, Jesús García Molina, Begoña Moros, Joaquín Nicolás
Grupo de Investigación de Ingeniería del Software
Departamento de Informática y Sistemas
Facultad de Informática. Universidad de Murcia
C.P. 30071 Campus de Espinardo, Murcia, Spain

Resumen. En este trabajo se presenta una estrategia para obtener de modo sistemático el modelo de casos de uso y el modelo conceptual, a partir del modelado del negocio basado en diagramas de actividades UML. Después de determinar los procesos del negocio de la organización bajo estudio, y de describir sus flujos de trabajo mediante diagramas de actividades, los casos de uso son identificados y estructurados a partir de las actividades de cada proceso, mientras que los conceptos del modelo conceptual se obtienen a partir de los datos que fluyen entre las actividades. Además, las reglas del negocio son identificadas e incluidas en un glosario, como parte de la especificación de datos y actividades. Un aspecto destacable de nuestra propuesta es el hecho de que el modelado conceptual y el de casos de uso son realizados en paralelo, haciendo más fácil la identificación y especificación de casos de uso adecuados. Tanto el modelado de casos de uso como el modelado conceptual forman parte de la fase de análisis de requisitos de un modelo de proceso completo en cuya definición estamos trabajando y cuya primera etapa es el modelado del negocio. Este proceso está siendo completado y adaptado a la tecnología web dentro de un proyecto PROFIT en cooperación con una empresa de desarrollo de software.

1 Introducción

Desde que UML [1] fue adoptado por el OMG como el lenguaje estándar para el modelado, se ha definido un buen número de modelos de proceso para el desarrollo de aplicaciones orientadas a objetos (OO), que utilizan este lenguaje como medio de expresión de los diferentes modelos que se crean durante el desarrollo. Estas propuestas suelen estar *guiadas por los casos de uso*, de manera que éstos se emplean para definir los requisitos funcionales del sistema, y todas las etapas del proceso (planificación de las iteraciones, análisis, diseño y pruebas) se articulan en torno a los casos de uso identificados.

Actualmente, en muchas discusiones sobre casos de uso se coincide en señalar que con frecuencia son mal interpretados, y que no hay guías precisas para resolver los

¹ Esta ponencia es una revisión del trabajo “*De los procesos de negocio a los casos de uso*”, presentado en las *V Jornadas de Ingeniería del Software y Bases de Datos* celebradas en Valladolid en noviembre de 2000.

La revisión ha sido subvencionada por el Proyecto PROFIT FIT-070000-2000-411 y por la *Red Temática de Investigación en Ingeniería del Software TIC 2000-2052-E*.

aspectos que tienen que ver con su organización. En este sentido, se han publicado diferentes propuestas (por ejemplo [2, 5, 6]) en las que se discuten cuestiones tales como la granularidad de los casos de uso, el nivel de detalle en que deben describirse, o la conveniencia de crear una jerarquía de casos de uso.

En la actualidad trabajamos en la definición de un proceso basado en UML orientado a sistemas de información de gestión y en su adaptación al desarrollo de aplicaciones web, dentro de un proyecto PROFIT² en cooperación con la empresa de consultoría y desarrollo de software *Sinergia Tecnológica*. Este proceso incluye una fase inicial de modelado del negocio, que describe los procesos del negocio de la organización bajo estudio de manera que se puedan construir, de forma sencilla y directa, versiones iniciales de los modelos conceptual y de casos de uso, propios de la etapa de modelado de requisitos. En este trabajo describimos cómo realizar el modelado del negocio y su conexión con el modelo de requisitos.

La estructura del trabajo es la siguiente: en el apartado 2 comentamos brevemente la problemática asociada a la utilización del concepto de caso de uso, y ofrecemos una visión general de nuestra propuesta; en el apartado 3 presentamos la manera de abordar el modelado del negocio; en el apartado 4 mostramos cómo realizar la transición desde el modelo del negocio a los modelos de casos de uso y conceptual; finalmente, en la sección 5 exponemos nuestras conclusiones.

2 Motivación

2.1 Problemas en la Utilización de los Casos de Uso

Actualmente, la mayor parte de los modelos de proceso propuestos para UML se definen como *guiados por los casos de uso*. Un caso de uso puede ser definido como *una secuencia de acciones, incluyendo variaciones, que el sistema puede ejecutar y que produce un resultado observable de valor para un actor que interactúa con el sistema* [1]. Aunque el éxito de los casos de uso se suele justificar con el hecho de que constituyen una técnica simple e intuitiva, algunos autores (ver por ejemplo [2, 5, 6]) señalan las dificultades que entraña la obtención y la especificación de casos de uso verdaderamente útiles, y la falta de consenso sobre cómo organizarlos y manejarlos. Estas son las razones que nos llevan a pensar que es necesario establecer un conjunto de guías para la identificación, descripción y organización de los casos de uso.

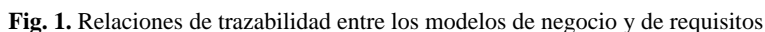
Algunas discusiones interesantes acerca del manejo de casos de uso son las proporcionadas por T. Korson y A. Cockburn. Korson [5] defiende que los requisitos (y por tanto los casos de uso) han de ser organizados jerárquicamente y establece que i) cada nivel de casos de uso no debe añadir nuevos requisitos, sino refinar los del nivel superior, y ii) la jerarquía de casos de uso no debe ser el resultado de una descomposición funcional, y ha de ser desarrollada de manera iterativa e incremental.

Por otro lado, Cockburn [2] utiliza el concepto de objetivo (*goal*) para organizar jerárquicamente los casos de uso. Distingue básicamente entre *objetivos estratégicos* (los *procesos del negocio* de la organización) y *objetivos de usuario* (las funciones del sistema). Los objetivos estratégicos se corresponden con un conjunto de objetivos de

² Proyecto PROFIT “Definición y Aplicación de un Proceso Software basado en UML para el desarrollo de Aplicaciones web” FIT-070000-2000-411.

Otra cuestión importante es la ubicación del modelado de casos de uso dentro del modelo de proceso. Normalmente se concibe el modelado de casos de uso como un paso previo al modelado conceptual. Sin embargo, Korson [6] argumenta que no es posible crear casos de uso adecuados y útiles (ni implementarlos correctamente) sin comprender el dominio, y por tanto, el modelado de casos de uso y el modelado conceptual deben ser actividades realizadas en paralelo.

Normalmente, los casos de uso son elicitados de forma intuitiva a partir de la especificación del sistema y, posteriormente, las entidades del modelo conceptual se extraen a partir de las especificaciones de los casos de uso. En las siguientes secciones presentamos una propuesta para obtener de forma sistemática tanto el modelo de casos de uso como el modelo conceptual, a partir de un modelo del negocio, de acuerdo con el esquema mostrado en la Fig. 1.



Una vez identificados los procesos de negocio de la organización, y descritos sus flujos de trabajo mediante diagramas de actividades, los casos de uso se elicitán y estructuran a partir de las actividades de cada proceso, mientras que las entidades del modelo conceptual se obtienen de los datos que fluyen entre tales actividades. Además, se identifican las reglas del negocio y se incluyen en un glosario como parte de la especificación de los datos y las actividades. Un aspecto notable de nuestra propuesta es que el modelado de casos de uso y el modelado conceptual se realizan al mismo tiempo, facilitando, por tanto, la identificación y especificación de los casos de uso adecuados.

3 Modelado del Negocio

Para conseguir sus objetivos, una empresa organiza su actividad por medio de un conjunto de *procesos de negocio*. Cada uno de ellos se caracteriza por una colección de *datos* que son producidos y manipulados mediante un conjunto de *tareas*, en las que ciertos *agentes* (por ejemplo, trabajadores o departamentos) participan de acuerdo a un *flujo de trabajo* determinado. Además, estos procesos se hallan sujetos a un conjunto de *reglas de negocio*, que determinan la estructura de la información y las políticas de la empresa. Por tanto, la finalidad del modelado del negocio es describir cada proceso del negocio, especificando sus datos, actividades (o tareas), roles (o agentes) y reglas de negocio.

3.1 Identificación de Procesos de Negocio

El primer paso del modelado del negocio consiste en capturar los procesos de negocio de la organización bajo estudio. La definición del conjunto de procesos del negocio es una tarea crucial, ya que define los límites del proceso de modelado posterior.

Nos basamos en el concepto de *objetivo estratégico*, introducido por Cockburn [2], para identificar de manera adecuada los diferentes procesos de negocio de una organización a partir de la determinación y estructuración de sus objetivos.

En primer lugar, por tanto, identificamos los *objetivos estratégicos* de la empresa. Teniendo en cuenta que estos objetivos van a ser muy complejos y de un nivel de abstracción muy alto, cada uno de ellos puede ser descompuesto en un conjunto de subobjetivos más concretos, que deberán cumplirse para conseguir el objetivo estratégico original. Estos subobjetivos pueden a su vez ser descompuestos en otros, de manera que se defina una jerarquía de objetivos. En nuestro estudio, hemos experimentado que dos o tres niveles de descomposición son suficientes. Para cada objetivo que no ha sido descompuesto en otros, definimos un proceso del negocio cuyo propósito será dar soporte a dicho objetivo, es decir lograrlo o realizarlo. Representamos cada proceso del negocio mediante un *caso de uso del negocio*.

En el resto del trabajo, ilustramos el proceso mediante el ejemplo de una compañía que fabrica productos bajo demanda (siguiendo un esquema *just in time*). Los objetivos estratégicos de dicha compañía podrían incluir *Satisfacer un pedido de un cliente*, *Incrementar en un 25% las ventas*, o *Disminuir el tiempo de fabricación en un 15%*. El objetivo *Satisfacer un pedido de un cliente* puede ser dividido en subobjetivos tales como *Registrar Pedido de Cliente*, *Fabricar Producto Pedido*, *Gestionar Almacén* y *Realizar Pedidos a Proveedores*. Éstos serán los objetivos que utilizaremos para definir nuestros procesos del negocio.

Un enfoque muy similar, presentado posteriormente a nuestra propuesta, es utilizado por H. Eriksson y M. Penker [4], donde se propone la construcción de un *modelo de objetivo/problema* que facilita la identificación de los procesos de negocio. En [4] también se define el patrón de negocio *Business Goal Decomposition* que puede ser utilizado como guía para la descomposición de los objetivos de la organización.

3.2 Identificación de Roles del Entorno del Negocio

Una vez se han identificado los procesos de negocio, es preciso encontrar los agentes involucrados en su realización. Cada uno de estos agentes o actores del negocio desempeña cierto papel (*juega un rol*) cuando colabora con otros para llevar a cabo las actividades que conforman dicho caso de uso del negocio. De hecho, identificaremos los roles que son jugados por agentes de la propia empresa (que incluyen trabajadores, departamentos y dispositivos físicos) o agentes externos (como clientes u otros sistemas). Por el momento nos centraremos en este último tipo de roles, con los que la organización interactúa para llevar a cabo sus procesos de negocio. En nuestro ejemplo tenemos los roles *Cliente* y *Proveedor*, claramente externos al sistema.

Para tener una visión general de los diferentes procesos de negocio de la organización, puede construirse un *diagrama de casos de uso del negocio*, en el cual aparece cada proceso del negocio como un caso de uso. Este diagrama permite mostrar los límites y el entorno de la organización bajo estudio. Por esta razón, sólo aparecerán en este diagrama los actores del negocio correspondientes a los roles externos al sistema, de forma que los procesos de negocio en los que sólo tomen parte roles internos a la organización no estarán conectados a ningún actor. En la Fig. 2 se muestra el diagrama de casos de uso del negocio para nuestro ejemplo; es un diagrama de casos de uso UML formado por casos de uso del negocio y actores. En el diagrama se muestra además que el agente *Cliente* arranca la realización del caso de uso relacionado, mientras que *Proveedor* simplemente participa en el caso de uso asociado.

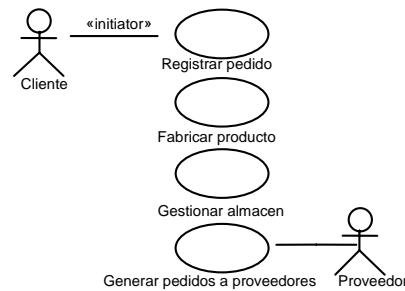


Fig. 2. Diagrama de casos de uso del negocio para el sistema de producción *just in time*

3.3 Descripción de los Casos de Uso del Negocio

El siguiente paso dentro del modelado del negocio es introducirse en cada uno de los casos de uso del negocio identificados, para describirlo en detalle. Inicialmente se rellena una plantilla de descripción, y después, a partir de la información reflejada en dicha plantilla, se construye un conjunto de diagramas que describen completamente el caso de uso del negocio. Nos centraremos en uno de los casos de uso del negocio de nuestro ejemplo, *Registrar Pedido*, cuya descripción se muestra en la Fig. 3. La *plantilla de descripción de casos de uso del negocio* –inspirada en el conjunto de valores etiquetados utilizados en [4] para describir un proceso de negocio–, contiene los campos *objetivo* (qué se intenta conseguir), *descripción* (especificación informal de lo que hace el proceso), *prioridad* (importancia del proceso, por ejemplo si es fundamental o básico, de administración, o de soporte), *riesgos* (por ejemplo errores o

fallos que pueden ocurrir al ejecutar este proceso), *posibilidades* (cambios o mejoras futuras del proceso), y por último, *tiempo* y *coste* aproximados de ejecución. Esta descripción puede ser validada fácilmente por los usuarios.

A continuación, hemos de determinar los agentes internos que juegan un rol en el caso de uso del negocio. Hasta el momento hemos identificado los roles que pertenecen al entorno de la organización. Ahora es necesario estudiar la *Descripción* (ver Fig. 3) de cada caso de uso del negocio, y observar el conjunto completo de roles involucrados, tanto externos como internos a la organización. Los roles del caso del uso del negocio *Registrar pedido* son *Cliente*, *Comercial*, *JefeTecnico*, y *JefeProduccion* (siendo los tres últimos internos al sistema).

| | |
|---------------------------|---|
| Proceso de Negocio | Registrar Pedido |
| Objetivo | Registrar Pedido de Cliente |
| Descripción | <ol style="list-style-type: none"> 1. El cliente envía una orden de pedido, que debe incluir la fecha de solicitud, datos del cliente y productos solicitados. Es posible que sea un empleado del departamento comercial quien introduzca el pedido, a petición de un cliente que realizó su pedido por teléfono o lo envió por fax o correo ordinario al dpto. comercial de la empresa. 2. El empleado revisa el pedido (completándolo, si es necesario), y comienza su procesamiento enviándolo al jefe técnico, encargado de su análisis. 3. El jefe técnico analiza la viabilidad de cada producto pedido por separado: <ul style="list-style-type: none"> • Si el producto pedido está en el catálogo, su fabricación es aceptada. • En caso contrario, es considerado un <i>producto especial</i>, y el jefe técnico estudia su producción: <ul style="list-style-type: none"> - Si es viable, la fabricación del producto especial es aceptada; - Si no es viable, el producto especial no será fabricado. 4. Una vez estudiado el pedido completo, el jefe técnico... <ul style="list-style-type: none"> • Informa al departamento comercial de la aceptación o rechazo de cada producto pedido; • Si todos los productos de un pedido han sido aceptados, se crea una orden de trabajo para cada producto, a partir de una plantilla de fabricación (la estándar si el producto estaba catalogado, o una nueva, específicamente diseñada para el producto, si éste no estaba en el catálogo). Cada orden de trabajo es enviada al jefe de producción, y queda pendiente de su lanzamiento. 5. El comercial comunica al cliente el resultado final del análisis de su pedido. |
| Prioridad | Básico |
| Riesgos | ... |
| Posibilidades | ... |
| Tiempo Ejecución | ... |
| Coste de Ejecución | ... |

Fig. 3. Descripción parcial del caso de uso del negocio *Registrar pedido*

El aspecto estructural de la colaboración entre los roles para llevar a cabo un caso de uso del negocio, puede ser representado en un *diagrama de roles*, en el que cada rol (una clase UML estereotipada) aparece asociado con los roles con los que puede colaborar (ver Fig. 4). Por tanto, este diagrama permite expresar el conocimiento que unos roles tienen de otros, así como las características (como la multiplicidad) de cada relación entre roles. Además, este diagrama permite también mostrar las características de los roles identificados, tales como sus atributos y responsabilidades. Ortín y García Molina discuten con más detalle el modelado de roles con UML en [8].

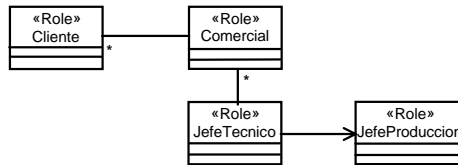


Fig. 4. Diagrama de roles para el caso de uso del negocio *Registrar Pedido*

Después, podemos crear escenarios para mostrar el aspecto de comportamiento de la colaboración. Para ello utilizaremos *diagramas de secuencia* UML (ver Fig. 5), donde los objetos denotan las instancias de los roles que intervienen en la interacción.

En cada proceso podemos distinguir entre el flujo básico o normal de la interacción (en nuestro ejemplo, *solicitud de un pedido que es aceptado*) y los posibles flujos alternativos (por ejemplo, *rechazo o cancelación de un pedido*). Para mejorar la legibilidad, es conveniente asociar varios escenarios a un mismo caso de uso del negocio, en lugar de mostrar en una única secuencia todas las posibilidades.

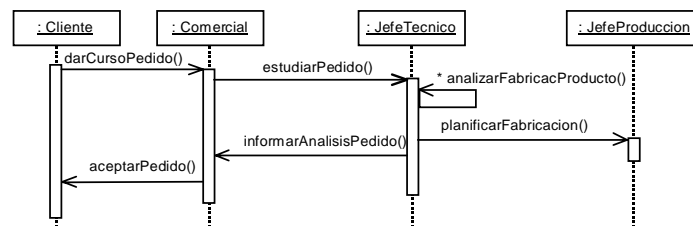


Fig. 5. Diagrama de secuencia para el caso de uso del negocio *Registrar Pedido*

Para mostrar de forma más detallada el flujo de trabajo que realiza cada proceso del negocio, utilizaremos diagramas de actividades con calles (*swimlanes*), que llamaremos *diagramas de proceso*.

La Fig. 6 muestra el diagrama de proceso que incluye el escenario de la Fig. 4. Existe una calle por cada rol participante en el escenario, que incluye las actividades que realiza dicho rol. El diagrama también muestra la información que necesita y produce cada actividad, y la sincronización requerida entre las diferentes actividades. Los datos aparecen como objetos que fluyen entre las actividades y pueden tener un estado. Por ejemplo, la actividad *Cursar pedido* recibe un pedido propuesto e inicia su revisión (ver Fig. 6). Nos referimos a estos objetos como *objetos de información*.

Durante la descripción de un proceso del negocio mediante un diagrama de proceso, es posible encontrar una actividad cuya complejidad sea tal que sea necesario describirla mediante otro diagrama de proceso adicional. Por tanto, este nuevo diagrama de proceso describirá un subobjetivo en relación con el objetivo ligado al proceso del negocio original. De este modo los procesos de negocio se organizan jerárquicamente. También es posible mostrar en diferentes diagramas de proceso el flujo normal y los flujos alternativos.

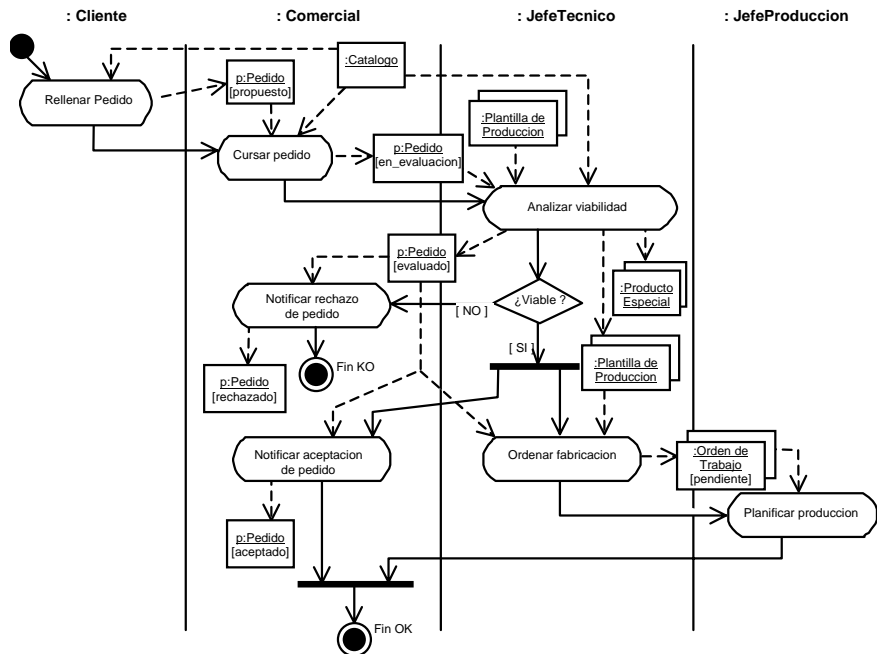


Fig. 6. Diagrama de proceso para el caso de uso del negocio *Registrar Pedido*

3.4 Especificación de Reglas del Negocio

En una organización, tanto los procesos como los datos que estos manejan, están restringidos por las reglas del negocio. Estas reglas aseguran que la actividad de la empresa se lleva a cabo de acuerdo a restricciones impuestas desde el entorno (leyes o normas) o desde dentro de la propia organización.

Como afirma B. Whitenack [9], las reglas del negocio rara vez son capturadas de forma explícita durante el desarrollo del producto, a pesar de que suelen ser importantes restricciones sobre el comportamiento del sistema. El hecho de que no exista un marco de trabajo bien definido en el que situar las reglas, unido a la existencia de una gran variedad de tipos de reglas de difícil comprensión, hace que a menudo las reglas del negocio sean ignoradas hasta la fase de implementación.

Con el fin de tener en cuenta todos los tipos de reglas que aparecen en la especificación de requisitos, hemos utilizado la clasificación descrita por J. Odell [7]. Esta clasificación es sencilla pero completa, cubriendo prácticamente todos los tipos de reglas del negocio. Las categorías de reglas del negocio son las siguientes:

- **Reglas de Restricción:** especifican políticas o condiciones que restringen la estructura y comportamiento de los objetos y procesos. Estas reglas pueden ser divididas en *reglas de estímulo-respuesta* (que restringen el comportamiento y especifican las condiciones que deben cumplirse para activar una operación), *reglas de restricción de operación* (que especifican condiciones que deben cumplirse antes y después de ejecutarse una operación) y *reglas estructurales* (que especifican restricciones sobre los tipos de objetos y las asociaciones).

- **Reglas de Derivación:** especifican políticas y condiciones para inferir o calcular hechos (información) a partir de otros hechos existentes en el negocio.

Por otro lado, Eriksson y Penker [4] introducen otro tipo de reglas del negocio, las *Reglas de Existencia*, que establecen cuándo puede existir un determinado objeto.

De acuerdo con esta clasificación, recogemos de manera explícita cada tipo de regla en el modelo del negocio mediante la especificación de las actividades y objetos de información que aparecen en los diagramas de proceso. Estas especificaciones se reúnen en un glosario. La Fig. 7 muestra la especificación del objeto de información *Pedido* y de las actividades *Ordenar fabricación* y *Notificar aceptación de pedido*.

| | |
|--|---|
| <p>...</p> <p>Objeto de Información: <u>Pedido</u></p> <p>Atributos</p> <p>Código de pedido</p> <p>Fecha de solicitud</p> <p>Fecha de creación</p> <p>Fecha máxima de entrega</p> <p>Conjunto de {Producto}</p> <p>Cliente</p> <p>Importe total</p> <p>Estado actual</p> <p>Restricciones</p> <ul style="list-style-type: none"> - El código de pedido identificará unívocamente el pedido, y será asignado automáticamente por el sistema. - Las fechas de solicitud y de creación serán previas a la fecha máxima de entrega. - Un pedido contendrá al menos un producto; no existe límite máximo de productos. - Un pedido siempre será solicitado por un y solamente un cliente - El importe total del pedido será calculado a partir del precio y unidades pedidas de cada producto incluido. <p>...</p> <p>Clase del Dominio: -pendiente de especificar-</p> <p>...</p> | <p>...</p> <p>Actividad: <u>Ordenar fabricación</u></p> <p>Origen: Analizar viabilidad</p> <p>Agente: Jefe Técnico</p> <p>Precondiciones:</p> <ul style="list-style-type: none"> - El estado del pedido es evaluado - La fabricación de todo producto en el pedido es viable - Existe una plantilla de fabricación para cada uno de dichos productos. <p>Postcondiciones:</p> <ul style="list-style-type: none"> - Ha sido creada una orden de trabajo para cada producto solicitado; - El estado de cada orden de trabajo es pendiente. - Cada orden de trabajo ha sido enviada al jefe de producción para su planificación. <p>Caso de Uso del sistema: -pendiente de especificar-</p> <p>Actividad: <u>Notificar aceptación de pedido</u></p> <p>Origen: Analizar viabilidad</p> <p>Agente: Comercial</p> <p>Precondiciones:</p> <ul style="list-style-type: none"> - La fabricación de todos sus productos es viable. <p>Postcondiciones:</p> <ul style="list-style-type: none"> - Se ha comunicado al cliente la aceptación de su pedido. - El estado del pedido es aceptado. <p>Caso de Uso del Sistema: -pendiente de especificar-</p> <p>...</p> |
|--|---|

Fig. 7. Extracto del glosario: objetos de información y actividades

Cada objeto de información se describe mediante un conjunto de atributos y sus restricciones de integridad (si las tuviera); por tanto, establecemos explícitamente las reglas estructurales, de derivación y de existencia. Por otro lado, la especificación de la semántica de cada actividad contendrá: *origen* (actividades que la preceden), *agente* (responsable de llevar a cabo la actividad), y *pre* y *post-condiciones* (que establecen qué tiene que cumplirse antes y después de la actividad). Estos dos últimos campos recogen las reglas de operación, mientras que las reglas de estímulo-respuesta quedan reflejadas mediante el *origen*, donde se expresa el orden entre las actividades, así como mediante aquellas precondiciones que representan las condiciones necesarias para ejecutar la actividad. El glosario tendrá una estructura de hipertexto (referencias-cruzadas) con el objeto de mantener las relaciones de trazabilidad entre los procesos del negocio y las clases y los casos de uso que especifican la funcionalidad del sistema.

4 Modelado de Requisitos: Modelos Conceptual y de Casos de Uso Iniciales

A partir del modelo del negocio descrito en la sección anterior, es posible obtener de manera sistemática y directa, tanto la colección inicial de casos de uso del sistema como el modelo conceptual preliminar. A continuación vamos a describir de manera separada cómo obtener cada modelo.

Los requisitos elicitados y especificados en esta fase serán incluidos en un documento de especificación de requisitos del software (ERS). Recomendamos el uso de una plantilla de ERS estándar, como la IEEE 830-1998, pero siempre particularizada al contexto de trabajo, marcado por el tipo de proyecto y la cultura de equipo de desarrollo.

4.1 Obtención del Modelo Inicial de Casos de Uso del Sistema

Según nuestra experiencia, las actividades del diagrama de proceso tienen el nivel de granularidad adecuado para ser asociadas a un caso de uso del sistema. De esta manera, crearemos un caso de uso del sistema por cada actividad del diagrama de proceso que deba ser soportada por el sistema software. Por tanto, el rol que lleva a cabo la actividad será el actor principal del caso de uso. Nótese que, de acuerdo con la definición de caso de uso, no todas las actividades del diagrama de proceso serán consideradas como casos de uso, sino solamente aquellas que sean de valor para algún actor.

Por ejemplo, supongamos que el rol *Cliente* no rellenara él mismo el pedido (mediante un formulario web, por ejemplo), sino que comunicara todos los datos por fax, teléfono, o cualquier otro medio, como resultado de la actividad *Rellenar pedido* (ver Fig. 6). Como esta actividad se llevaría a cabo fuera del sistema software, no aparecerían en el diagrama de casos de uso del sistema ni la actividad *Rellenar pedido*, ni el rol *Cliente* (puesto que no interactuaría con el sistema software). La Fig. 8 muestra el diagrama de casos de uso del sistema obtenido para el proceso del negocio *Registrar Pedido*, correspondiente al diagrama de proceso de la Fig. 6, considerando que todas las actividades serán soportadas por el sistema software.

Debemos señalar que algunos casos de uso no se obtendrán directamente a partir de los diagramas de proceso. Estos nuevos casos de uso se detectarían al describir los casos de uso identificados y adquirir un mayor conocimiento sobre los requisitos que deben ser soportados, y representarían funciones que debe llevar a cabo el sistema para lograr algún objetivo asociado con algún caso de uso ya existente. Normalmente los casos de uso detectados de esta manera serán *casos de uso de soporte*, puesto que no surgen directamente de la descripción de los procesos de negocio. En nuestro ejemplo, para *Analizar viabilidad* es necesario buscar en el catálogo de productos si un producto solicitado existe y, por tanto, este catálogo debe mantenerse actualizado. En consecuencia, añadimos el caso de uso *Mantener Productos del Catálogo*. Otro ejemplo de un nuevo caso de uso sería *Mantener Plantillas de Fabricación*.

De este modo, el modelo del negocio permite obtener los casos de uso más importantes dentro de cada proceso del negocio, y además facilita la determinación del conjunto adecuado de pasos incrementales en el proceso iterativo de desarrollo, tal y como defiende Korson [5].

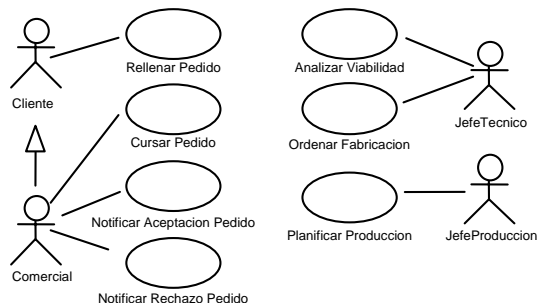


Fig. 8. Diagrama inicial de casos de uso del sistema

Los casos de uso se pueden organizar en varios niveles (recomendamos dos o tres como máximo) de acuerdo con la descomposición jerárquica propuesta en el modelo del negocio.

Cada caso de uso se describirá mediante una plantilla que puede rellenarse a partir de la especificación de la actividad asociada, que se encuentra recogida en el glosario como ya hemos visto. La plantilla que proponemos está basada en la de Coleman [3], a la que hemos añadido un campo para las postcondiciones del caso de uso, tal como se muestra en la Fig. 9.

| | |
|----------------------------|--|
| Caso de Uso | Ordenar Fabricación |
| Descripción | Crear órdenes de trabajo para cada producto solicitado en el pedido, y enviarlas al jefe de producción para dar comienzo a la planificación de su fabricación. |
| Actores | Jefe técnico |
| Asunciones | <ul style="list-style-type: none"> - El estado del pedido es evaluado. - Es viable la fabricación de cada producto solicitado en el pedido. - Existe una plantilla de fabricación para cada producto solicitado. |
| Postcondiciones | <ul style="list-style-type: none"> - Ha sido creada una orden de trabajo para cada producto solicitado. - El estado de cada orden de trabajo es pendiente. - Cada orden de trabajo ha sido enviada al jefe de producción |
| Pasos | 1 REPETIR 1.1 Obtener un producto del pedido. 1.2 Buscar la plantilla de fabricación asociada al producto. 1.3 Crear la orden de trabajo. 1.4 Almacenar la orden de trabajo con el estado <i>pendiente</i> . 1.5 Enviar la orden de trabajo al jefe de producción |
| Variaciones | -- |
| Req. No Funcionales | -- |
| Cuestiones | -- |

Fig. 9. Descripción del caso de uso del sistema *Ordenar Fabricación*

Una vez descrito el caso de uso, se conectará a la especificación de la actividad asociada en el glosario, con el objeto de mantener la trazabilidad entre los casos de uso del negocio y los del sistema.

También podrían encontrarse relaciones entre los casos de uso, tales como *include*, si se detectan aspectos comunes entre varios casos de uso, y *extend*, para expresar caminos opcionales o alternativos en un caso de uso. No obstante, estamos de acuerdo con las recomendaciones ampliamente extendidas de no abusar de estas relaciones y no mostrarlas en los diagramas de casos de uso.

4.2 Obtención del Modelo Conceptual Inicial

Los objetos de información que fluyen entre las actividades de un caso de uso del negocio representan datos del dominio, por lo que suponen una buena base para crear el modelo conceptual inicial. Este modelo incluirá los conceptos y sus relaciones y se describirá mediante un diagrama de clases UML, en el que los conceptos se representan mediante clases (clases del dominio). La Fig. 10 muestra el diagrama de clases que describe el primer modelo conceptual de nuestro ejemplo.

El modelo conceptual inicial será refinado posteriormente gracias a la experiencia del modelador. Creemos que este es un buen punto de partida, en lugar de abordar la construcción del modelo conceptual partiendo de la nada.

Así, cada objeto de información del diagrama de proceso se convertirá ahora en un concepto (y en la etapa de diseño dará lugar a una clase si el sistema software debe dar soporte a dicho concepto). A partir de la especificación de cada objeto de información obtendremos la definición del concepto asociado, es decir, sus atributos, relaciones con otras clases y restricciones. Por ejemplo, a partir de la especificación de *Pedido* mostrada en la Fig. 7, podríamos obtener: i) los atributos *codigo*, *fechaSolicitud*, *fechaCreacion*, *fechaMaxEntrega*, *importeTotal*, *estadoActual*; ii) las relaciones *Cliente-Pedido* y *Pedido-Producto*, y iii) restricciones que podrían ser expresadas textualmente o bien mediante OCL (*Object Constraint Language*), como $\{fechaMaxEntrega > fechaCreacion\}$.

Nótese además que cuando un modelo conceptual evoluciona hacia un diagrama de clases, algunas responsabilidades se pueden obtener a partir de ciertas restricciones ya especificadas en el glosario. Por ejemplo, la clase *Pedido* podría tener responsabilidades como *obtenerProductos*, *calcularFechaMaxEntrega*, *calcularImporteTotal* o *cambiarEstado*.

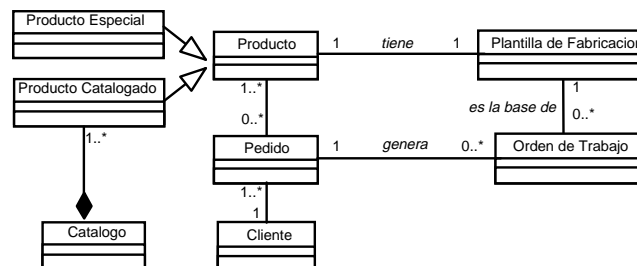


Fig. 10. Modelo conceptual inicial para el caso de uso del negocio *Registrar Pedido*

En esta etapa del desarrollo, es aconsejable centrarse más en la identificación y especificación de los conceptos que en las relaciones entre ellos, puesto que éstas serán refinadas en fases posteriores. Por el momento, podemos concentrarnos en las asociaciones del tipo *debe-conocer* y en la *generalización* para identificar las relaciones entre conceptos más importantes. Por ejemplo, a partir del glosario podemos establecer que un pedido *debe conocer* al cliente que lo realiza y los productos que lo componen (ver Fig. 7).

Por otro lado, alguno de los roles identificados en el modelo del negocio, y por tanto especificado en el modelo de roles, podría ser incluido como una clase en el modelo conceptual. Es el caso de la clase *Cliente* en nuestro ejemplo.

De igual forma que conectamos en el glosario las actividades con los casos de uso del sistema, vincularemos cada objeto de información con la clase del dominio que lo representa en el sistema.

El modelo del negocio permite además identificar aquellas clases cuyo comportamiento depende de un conjunto no trivial de estados alcanzables. En estos casos, sería interesante definir una máquina de estados mediante un *diagrama statechart* UML. Estas clases se detectan con facilidad en los diagramas de proceso, puesto que se corresponden con objetos de información etiquetados con varios estados diferentes. En nuestro ejemplo, *Pedido* sería candidato para construir una máquina de estados que mostrase los estados de un pedido (*propuesto*, *en evaluación*, *evaluado*, *aceptado* y *rechazado*) y los eventos que producen los cambios entre estados.

4.3 Representación de las Reglas del Negocio

En este apartado comentaremos con más detalle cómo son llevadas al modelo de requisitos las diferentes reglas del negocio que han sido recogidas en el glosario durante el modelado del negocio.

Las reglas de negocio estructurales, de derivación y de existencia quedan plenamente expresadas en el modelo conceptual. La propia sintaxis del diagrama de clases de UML permite representar los atributos de los conceptos y sus relaciones, mediante los atributos de las clases correspondientes y asociaciones entre éstas. Las reglas de derivación pueden ser especificadas mediante expresiones OCL dentro de *constraints* de UML colocadas en el diagrama cerca del elemento derivado, o bien dentro de notas conectadas a dicho elemento. La multiplicidad de las asociaciones permite por ejemplo representar las reglas del tipo de *un pedido será solicitado por uno y sólo un cliente*. Otras restricciones pueden expresarse en OCL utilizando *constraints* cercanas a los elementos a los que restringen, como $\{fechaMaxEntrega > fechaCreacion\}$ para la clase *Pedido*. También es posible utilizar OCL o lenguaje natural para expresar una restricción dentro de una nota conectada a los elementos afectados por dicha restricción. Las reglas de existencia están implícitas en el modelo conceptual, por ejemplo en el caso de un objeto agregado, como *Catálogo*, cuya existencia no es posible a menos que existan los objetos que lo componen.

Por otro lado, las reglas de negocio de operación quedan expresadas en la plantilla de descripción de los casos de uso, puesto que las precondiciones y postcondiciones de las operaciones especificadas en el glosario se recogen respectivamente en los campos *Asunciones* y *Postcondiciones*.

Por último, las reglas de estímulo/respuesta, expresadas mediante los campos *origen* y *precondiciones* de la especificación de las actividades, quedarán recogidas en el campo *Asunciones* de las plantillas de casos de uso.

4.4 Identificación de Requisitos No Funcionales

Para completar esta fase debemos establecer los requisitos no funcionales, relacionados por ejemplo con el rendimiento, la disponibilidad o la seguridad. Cuando estén asociados a un caso de uso, podrán especificarse en la plantilla de caso de uso propuesta. Los requisitos no funcionales que afecten a varios casos de uso o sean globa-

les a toda la organización, serán recogidos en el apartado correspondiente de la plantilla de ERS elegida.

El modelo del negocio puede ayudar a encontrar requisitos no funcionales, tal y como se indica en [4], pues por ejemplo los campos *tiempo de ejecución* y *coste de ejecución* de la plantilla de descripción de casos de uso del negocio expresan necesidades no funcionales que deben ser trasladadas a la plantilla de ERS y asociadas a los correspondientes casos de uso del sistema. Por otro lado, el que los procesos del negocio sean el resultado del análisis de los objetivos de la organización, permite que los requisitos (funcionales y no funcionales) del sistema puedan ser validados y verificados contra los objetivos.

5 Conclusiones

Este trabajo presenta una estrategia para abordar el modelado del negocio y el análisis de requisitos, en la que una primera versión del modelo de casos de uso y del modelo conceptual se obtienen de forma sencilla, a partir del modelo del negocio basado en el uso de diagramas de actividades UML.

Con las guías proporcionadas, el modelador dispone de un modo sistemático de identificar y organizar casos de uso, y de identificar y definir las clases del modelo conceptual. Los procesos de negocio de la organización se identifican partiendo de sus propios objetivos, y se describen mediante flujos de actividades que se representan mediante diagramas de actividades UML. De este modo, los casos de uso del sistema se obtienen a partir de las actividades de los procesos del negocio, se organizan jerárquicamente y se facilita su desarrollo iterativo e incremental, de acuerdo con lo indicado por Korson [5].

Las clases del modelo conceptual se obtienen a partir de los objetos de información que fluyen entre las actividades. Nos gustaría subrayar, como una característica importante de nuestro enfoque, que el modelado de los casos de uso del sistema y el modelado conceptual se realizan en paralelo, de acuerdo con Korson [6], quien establece que esto es crucial para obtener casos de uso correctos, puesto que es necesario entender bien el dominio para poder escribir casos de uso que sean realmente útiles.

A la vez que se realiza el modelado del negocio y de los requisitos, las reglas del negocio de la organización se recogen en un glosario, en forma de especificación de las actividades y de los casos de uso asociados, así como de los objetos de información y de las clases que los implementan. Esto permite mantener las correspondientes relaciones de trazabilidad entre los diferentes artefactos del modelado.

En este trabajo hemos expuesto cómo el modelado del negocio puede facilitar la identificación de los requisitos tanto funcionales como no funcionales del sistema. Además, el hecho de que tales requisitos surjan de la descripción de los procesos del negocio, y que éstos sean el resultado del análisis de los objetivos de la organización, posibilita que los requisitos del sistema sean validados y verificados contra los objetivos del negocio.

Referencias

1. Booch, G., Rumbaugh, J., Jacobson, I.: The Unified Modeling Language User Guide. Addison-Wesley (1999)
2. Cockburn, A.: Using Goal-Based Use Cases. JOOP, Vol. 10, No. 7 (Nov/Dec 1997) 56-62
3. Coleman, D.: A Use Case Template: Draft for discussion.
http://www.bredemeyer.com/use_case.pdf. (1998)
4. Eriksson, H.E., Penker, M.: Business Modeling with UML. Business Patterns at Work. John Wiley & Sons, Inc. (2000)
5. Korson, T.: Misuse of Use Cases.
<http://software-architects.com/publications/korson/korson9803om.htm>. (1998)
6. Korson, T.: Constructing Useful Use Cases.
<http://software-architects.com/publications/korson/usecase3>. (1999)
7. Martin, J. Odell, J.J.: Object-Oriented Methods: A Foundation. Prentice Hall. (1997)
8. Ortín, M.J., García-Molina, J.: Modelado con Roles en UML. IV Jornadas de Ingeniería del Software y Bases de Datos. Cáceres, Spain (1999)
9. Whitenack, B.: RAPPeL: A Requirements Analysis Process Pattern Language for Object-Oriented Development. In: Coplien, J.O., Schmidt, D.C. (eds.): Pattern Languages of Program Design. Addison-Wesley (1995) 259-291