

Acercando modelos de negocio y de proceso para el diseño de servicios

Francisco J. Pérez-Blanco, Juan M. Vara, Cristian Gómez,
Valeria De Castro, David Granada, Esperanza Marcos

Grupo de Investigación Kybele, Universidad Rey Juan Carlos
Calle Tulipán S/N, 28933 Móstoles, Madrid

{francisco.perez, juanmanuel.vara, cristian.gomez,
valeria.decastro, david.granada, esperanza.marcos}@urjc.es

Resumen. Actualmente existe gran cantidad de técnicas o notaciones para el modelado de negocio y el modelado de procesos que permiten representar una organización con mayor o menor nivel de detalle a la vez que ayudan a entender, conceptualizar y representar los servicios que aportan valor a la compañía. Esas técnicas tienen diferencias y similitudes, pero en muchos casos son complementarias. Sin embargo, no existe una solución que permita trabajar de manera integrada con varias de estas notaciones, acortando la distancia entre las áreas de negocio y tecnología. Este artículo presenta las últimas funcionalidades incorporadas en un entorno de modelado para el diseño de servicios que soporta 5 notaciones diferentes (*Canvas*, *e³value*, *Service Blueprint*, *Process Chain Network* y *BPMN*) y que permite generar vistas parciales de modelos basados en una determinada notación a partir de modelos elaborados con otra de las notaciones soportadas, además del modelo de relaciones correspondiente.

Keywords: Business Modeling, Business Process Modeling, Model Driven Engineering.

1 Introducción

Históricamente, el concepto de modelo de negocio se ha considerado desde tres perspectivas diferentes: orientado a la tecnología, orientado a la estrategia y orientado a las organizaciones [19]. Por ejemplo, en el contexto de las tecnologías de la información los modelos de negocio se identifican inmediatamente con modelos de procesos, mientras que, en el contexto de las teorías de la organización, se asocian más a representaciones abstractas de la estructura o arquitectura de una compañía. Algunos autores incluso distinguen cuatro categorías de modelos de negocios, a saber: modelos de procesos de negocio, modelos de motivación empresarial, modelos de organización empresarial y modelos de reglas de negocio [2].

En cualquier caso, no hay duda del interés que generan los modelos de negocio tanto en la academia como entre los profesionales, ni del hecho de que una comprensión clara

y una visión compartida de los modelos de negocio de un proyecto parecería algo deseable. Sin embargo, esta visión compartida se antoja complicada, más aún si tenemos en cuenta de que los implicados probablemente hablan *idiomas* diferentes, lo que en este contexto implica el uso de diferentes notaciones [4].

Es posible no obstante identificar relaciones entre modelos expresados con notaciones distintas. Por ejemplo, la mayoría de las técnicas existentes establecen claramente quiénes son los consumidores o proveedores de un servicio. Es interesante identificar y registrar estas relaciones, ya que no es inhabitual que los socios que colaboran para brindar un servicio determinado (o incluso diferentes equipos dentro de la misma organización) utilicen una o más de estas técnicas, con los consiguientes problemas de comunicación.

Si bien en la actualidad podemos encontrar herramientas de soporte para la mayoría de estas notaciones, no existe una solución integrada que permita trabajar con varias de ellas, debiendo por tanto recurrir a diferentes herramientas específicas para cada notación o, en algunos casos, al uso de herramientas *genéricas*, como MS-Visio o Lucidchart, que si bien pueden ser buenas opciones para realizar bocetos rápidos, no se diseñaron para facilitar el procesamiento de la información recopilada en los modelos elaborados [16].

Para tratar de paliar estos problemas, este trabajo presenta las últimas funcionalidades incorporadas en INNoVaServ¹, un entorno de modelado para el diseño de servicios que soporta varias notaciones, que ofrecen niveles de detalle diferentes y complementarios entre sí. Estas novedades pasan por: a) la incorporación de nuevas notaciones, con lo que actualmente la herramienta soporta ya cinco notaciones relacionadas con el modelado de negocio y el modelado de procesos: *Canvas* [10], *e3value* [7], *Service Blueprint* [1], *Process Chain Network* [13] y *BPMN* [21]; b) la generación de modelos parciales para una notación a partir de un modelo expresado con otra notación; c) la generación de modelos de trazas entre ambos modelos. Además, INNoVaServ soporta la validación de modelos *Service Blueprint* y *PCN* mediante técnicas formales [22] [23].

El resto del artículo está estructurado de la siguiente manera: en la sección 2 se ilustran las notaciones soportadas por la herramienta con un caso de estudio; la sección 3 resume conceptualmente las correspondencias entre las diferentes notaciones; la sección 4 presenta la arquitectura de la solución tecnológica y los detalles de implementación; la sección 5 analiza el rendimiento de los puentes tecnológicos construidos entre notaciones; la sección 6 analiza los trabajos relacionados y finalmente la sección 7 concluye resumiendo las principales aportaciones del trabajo y algunas líneas de trabajo futuro.

2 Un entorno de modelado para el diseño de servicios

Esta sección presenta una breve descripción de las cinco notaciones soportadas actualmente por INNoVaServ, utilizando *Deliveroo*², como caso de estudio. *Deliveroo* es un

¹ <http://www.kybele.es/innovaserv/>

² <https://deliveroo.co.uk/>

servicio de entrega de comida a domicilio que tiene una serie de repartidores ubicados a lo largo de grandes ciudades, de modo que los usuarios pueden pedir comida a través de una aplicación móvil, y los repartidores recogerán el pedido en los restaurantes y lo llevarán al domicilio de los clientes.

2.1 Canvas

Canvas es probablemente la notación más popular entre los profesionales de las áreas de negocio. Este modelo ayuda a visualizar y evaluar una propuesta de valor concreta que combina la oferta de productos y servicios, representando la lógica que permite a la organización generar beneficio en función de 9 módulos o dimensiones.

La Figura 1 muestra el modelo Canvas de *Deliveroo* desarrollado con INNoVaServ, donde, entre otra información, podemos ver los socios principales de la empresa (restaurantes y repartidores), actividades clave (entrega de alimentos, proporcionar una plataforma IT adecuada), gastos (pagos a trabajadores), ingresos (comisiones), etc.

En resumen, el Canvas proporciona una visión global de la empresa, sin profundizar en el detalle de las cadenas de valor o las operaciones de servicio que subyacen. Una de sus principales limitaciones es que sólo permite visualizar la información referente a una única propuesta de valor, mientras que disponer de una visión integrada de todas las actividades de producción de una empresa, es especialmente útil para diseñar ofertas de productos y servicios [3].

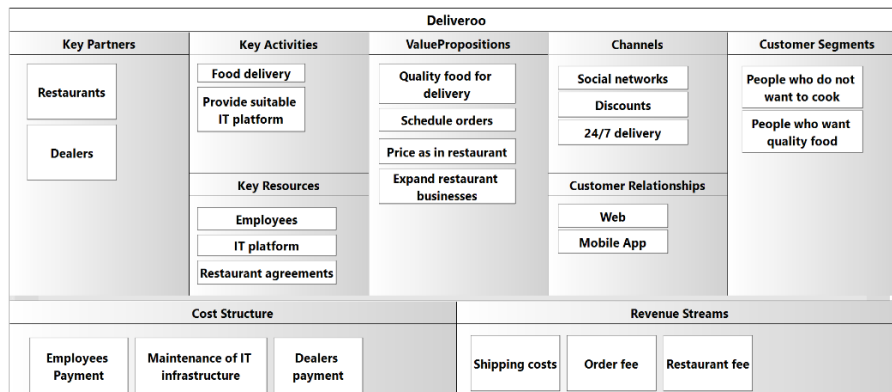


Figura 1. Modelo Canvas elaborado con INNoVaServ – *Deliveroo*.

2.2 e³value

e³value es un modelo de negocio que permite la representación gráfica de una idea de negocio, sin entrar en los detalles de los procesos que componen los servicios ofrecidos. Así, el modelo e³value se centra en representar las actividades de valor, que son las actividades realizadas por un actor para obtener un cierto beneficio, y los intercambios de valor donde los actores (unitarios o segmentos de mercado) intercambian diferentes objetos de valor (bienes, dinero o incluso servicios).

Respecto al modelo e³value para el caso de estudio de *Deliveroo* (Figura 2), nótese que contiene elementos que coinciden con algunos de los incluidos en el Canvas, como

los actores (*restaurantes y repartidores*) y algunos objetos de valor (*comisión del pedido, entrega de alimentos, etc.*).

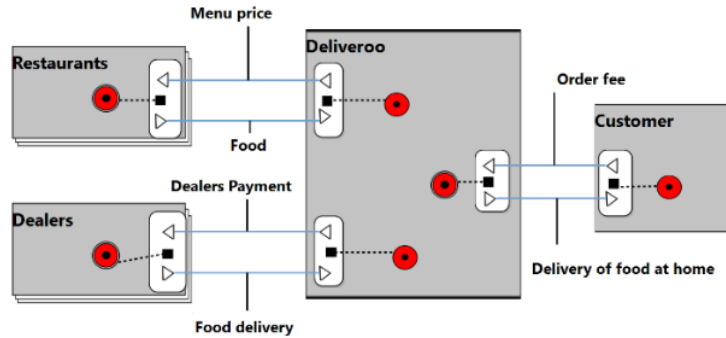


Figura 2. Modelo e³value elaborado con INNoVaServ – Deliveroo.

El modelo e³value no proporciona una vista detallada de las operaciones de servicio subyacentes, para lo que se debe recurrir al Service Blueprint.

2.3 Service Blueprint

Service Blueprint es una notación gráfica para el diseño de operaciones de servicio, que se centra en detallar la interacción cliente-proveedor en la prestación de un servicio determinado.

Los modelos Canvas o e³value pueden contener varias operaciones de servicio, cuyos detalles se pueden visualizar en diferentes Service Blueprints.

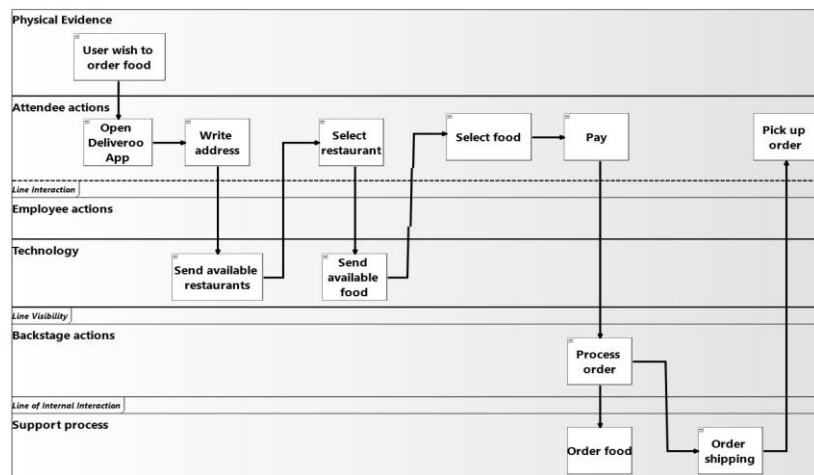


Figura 3. Service Blueprint elaborado con INNoVaServ – Servicio de entrega de Deliveroo.

La Figura 3 muestra el proceso de entrega de Deliveroo: una vez que el usuario tiene hambre (*evidencia física*), se produce una serie de interacciones entre el cliente y el proveedor a través de la *línea de interacción*. El pedido se administra en el *back stage*, mientras que se solicita a entidades externas cocinar la comida y enviarla.

Como puede apreciarse, el Service Blueprint se centra en representar las interacciones entre el cliente y el proveedor, dejando a un lado los intercambios de valor para centrarse en los procesos internos que se dan para la provisión del servicio. Por otro lado, el Service Blueprint no permite representar nodos de decisión, lo que dificulta la visualización de procesos basados en condiciones del contexto y, además, solo admite la representación de dos entidades (cliente y proveedor). Para paliar estas limitaciones se pueden utilizar modelos PCN, como muestra la siguiente sección.

2.4 Process Chain Network

Process Chain Network (PCN) es una técnica de modelado de servicios propuesta por Scott E. Sampson, que permite visualizar procesos de provisión de servicios.

La Figura 4 muestra el diagrama PCN para el proceso de entrega de *Deliveroo*, en el que *Deliveroo*, *Restaurante*, *Repartidor* y *Cliente* son las entidades participantes. Nótese que cada entidad contiene 3 tipos de regiones. Las regiones de los laterales corresponden a las áreas de **interacción directa** (*Dir.*), donde se representan las relaciones directas entre entidades. A continuación, las áreas de **interacción subrogada** (*Sur.*) contienen los pasos del proceso que involucran alguna acción de la entidad sobre recursos materiales de otra entidad. Finalmente, la región de **proceso independiente** (*Ind.*) es donde se representan las tareas internas de cada entidad.

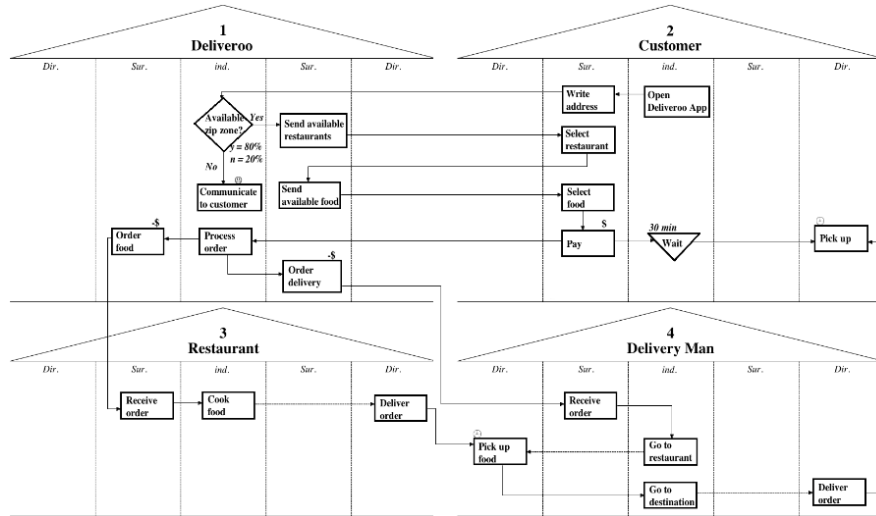


Figura 4. Diagrama PCN elaborado con INNoVaServ – Servicio de entrega de *Deliveroo*.

Al igual que BPMN o UML, la sintaxis concreta de PCN gira en torno a un elemento básico, en el que pequeñas variaciones gráficas, implican significados diferentes. Sin embargo, por razones de espacio se remite al lector a [13] para conocer los detalles de esta notación.

PCN ofrece por tanto mayor nivel de detalle que Service Blueprint, que no distingue las acciones internas del cliente ni refleja el resto de las entidades participantes en el

proceso, mientras que PCN si puede hacerlo debido a que fue concebido para representar redes de procesos u operaciones de servicio.

2.5 BPMN

Finalmente, debido al nivel de adopción y popularidad de BPMN, recientemente se integró en INNoVaServ un DSL gráfico que implementa esta notación, que permite representar modelos de proceso utilizando fundamentalmente elementos gráficos agrupados en cuatro categorías: *objetos de flujo*, *enlaces*, *swimlanes* y *artefactos*. Así, como las PCN, el concepto de *lanes* permite modelar operaciones de servicio que representan a todos los participantes en un mismo BPMN.

La Figura 5 muestra un extracto del modelo BPMN para el proceso de entrega de *Deliveroo* realizado con INNoVaServ. Como pasaba con el PCN, este diagrama muestra las entidades participantes en el proceso: *Deliveroo*, *Repartidor*, *Restaurante* y *Cliente*, además de las diferentes interacciones entre ellos.

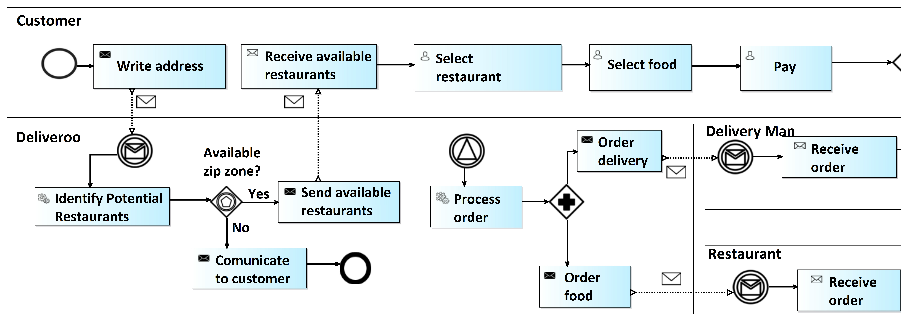


Figura 5. Diagrama BPMN elaborado con INNoVaServ – Servicio de entrega de *Deliveroo*.

A modo de conclusión, podría decirse que ninguna de las notaciones soportadas por INNoVaServ es mejor o peor que el resto. De algún modo, todas son complementarias y cualquiera de ellas se puede utilizar en cierto momento, en función del objetivo que se persiga en ese momento, del nivel de detalle con el que se deban representar las operaciones de servicio, o de las necesidades y experiencia de la organización. Por todo ello, la posibilidad de identificar y registrar las relaciones entre modelos elaborados con estas notaciones se antoja útil e interesante. Como un primer paso en esa dirección, la siguiente sección analiza las relaciones conceptuales que pueden establecerse entre las notaciones soportadas por INNoVaServ.

3 Análisis de correspondencias

Dado que en el contexto del diseño de los servicios que presta una organización pueden utilizarse varios de los modelos soportados por INNoVaServ, se ha llevado a cabo un análisis de las relaciones entre las notaciones o lenguajes de modelado soportadas por la herramienta en la actualidad.

Este estudio se ha dividido en tres bloques, en función de la naturaleza de estas notaciones: relaciones entre notaciones para el modelado de negocio (Canvas y e³value);

relaciones entre notaciones para el modelado de procesos (PCN, Service Blueprint y BPMN); relaciones entre los 2 grupos anteriores.

La Figura 6 muestra un resumen de un hipotético modelo de relaciones entre los elementos principales del modelo Canvas (izquierda) y el modelo e³value (derecha).

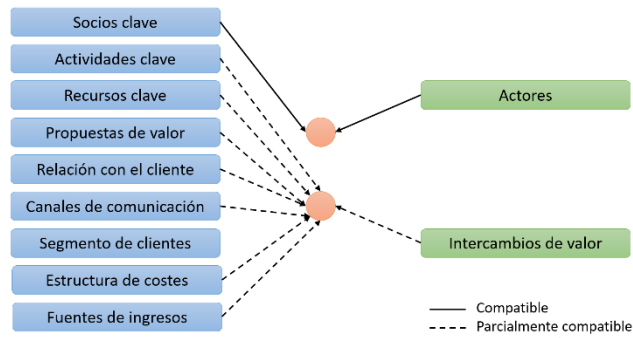


Figura 6. Correspondencias entre los modelos Canvas y e³value.

Tras analizar diferentes casos de estudio, se ha llegado a la conclusión de que los socios clave de un Canvas se corresponden en su mayoría con los actores de un modelo e³value (ver Figura 1 y Figura 2). Además, los valores de todas las secciones del Canvas, excepto *Cliente* y *Socios Clave* se corresponden con los objetos de valor que se representan en el modelo e³value.

Por otro lado, la Figura 7 resume las correspondencias encontradas entre los elementos principales de los modelos PCN, BPMN y Service Blueprint. El número de similitudes identificadas es en realidad mucho mayor ya que estas notaciones se componen de muchos más elementos de modelado.

Por ejemplo, las entidades de un modelo PCN se pueden traducir directamente a elementos *lane* de un BPMN. Sin embargo, esto no es posible en el caso de Service Blueprint ya que esta notación solo representa dos entidades: el cliente y el proveedor del servicio. Algunas relaciones más complejas, podrían ser las correspondencias entre la amplia variedad de acciones y dependencias disponibles tanto en PCN como en BPMN (acciones normales, acciones de espera, nodos de decisión, dependencias normales, dependencias con retraso, etc.). En el caso de caso de Service Blueprint, esto es mucho más simple ya que tan solo contempla un tipo de acciones y dependencias.

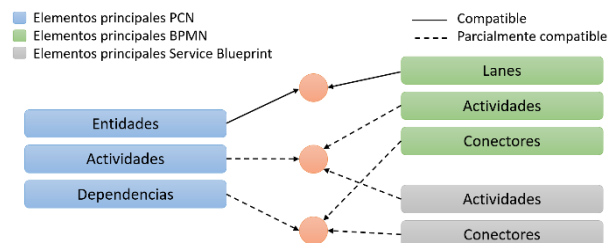


Figura 7. Representación del modelo de relaciones entre PCN, BPMN y Service Blueprint.

Finalmente, el tercer paso de este estudio ha consistido en analizar las relaciones entre ambos tipos de modelos (modelos de negocio y de proceso). Dado que los prime-

ros se centran en representar la estructura de una organización, mientras que los segundos se centran en la descripción de las operaciones de servicio que subyacen, la principal relación reside en que los modelos de proceso *explotan* la información recogida en los modelos de negocio. De este modo, a partir de los datos contenidos en un Canvas (como las actividades clave) o en un modelo e³value (como los intercambios de valor), podemos identificar cuantos modelos de proceso serán necesario para representar las principales operaciones de servicio. Por ejemplo, si observamos el modelo de valor para *Deliveroo* (Figura 2), sabremos que necesitamos tres modelos Service Blueprint para representar los 3 intercambios de valor presentes en la provisión de servicios de *Deliveroo*. Las entidades participantes en estos modelos de proceso se corresponderán con los actores del modelo de valor. Obviamente, no es posible deducir las acciones que implementarán la interacción entre esas entidades, únicamente a partir de la información recogida en el modelo de valor.

4 Solución tecnológica

En esta sección se presenta la solución tecnológica desarrollada para dar soporte al diseño de servicios, describiendo primero su arquitectura, y detallando luego algunos aspectos técnicos de su construcción.

4.1 Arquitectura conceptual

La arquitectura de INNoVaServ (Figura 8), que dota de un alto nivel de modularización a la herramienta, puede describirse de acuerdo a dos dimensiones ortogonales.

Atendiendo a la dimensión vertical de la figura, podemos ver INNoVaServ como un conjunto de cinco DSLs integrados, uno por cada notación soportada por la herramienta. De este modo, la Figura 8 permite distinguir cinco módulos: Canvas, e³value, Service Blueprint, PCN y BPMN.

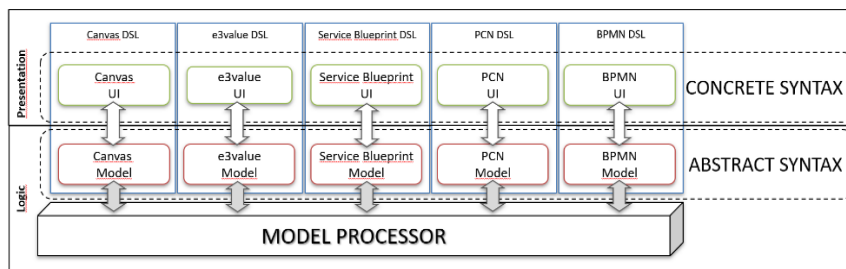


Figura 8. Arquitectura conceptual de INNoVaServ.

Por otro lado, siguiendo el principio de *separation of concerns* [11], INNoVaServ distingue la representación de cada modelo del modelo en sí mismo y esta distinción corresponde a la separación tradicional entre las sintaxis concreta y abstracta de cualquier lenguaje de modelado. De este modo, la capa de presentación incluye los elementos necesarios para la edición y representación del modelo, mientras que los modelos son gestionados por la capa lógica.

En la parte inferior de la Figura 8 se muestra una capa o bus denominado *procesador de modelos* [16], que conecta los cinco DSLs actuando como contenedor para diferentes componentes que soportarán diferentes tareas de gestión de modelos: transformación, validación, *weaving*, etc.

4.2 Detalles de implementación

En un principio, cada uno de los DSL incluidos en INNoVaServ se construyó sobre Eclipse EMF/GMF, siguiendo el proceso de desarrollo descrito en [16] para la construcción de entornos de modelado.

Sin embargo, la falta de soporte para GMF obligó a migrar los DSLs ya desarrollados a Sirius [18] (3 en aquel momento, con el consiguiente impacto en términos de esfuerzo). Sirius también se basa en EMF, y el proceso de desarrollo de DSL gráficos con Sirius es similar al proceso con GMF: diseño de un metamodelo, diseño de la sintaxis concreta (Sirius permite ver resultados en tiempo real, además de crear fácilmente diferentes *puntos de vista* para la misma sintaxis abstracta), creación de los modelos de relaciones entre sintaxis abstracta y concreta, creación de la paleta de herramientas y, finalmente, refinamiento del resultado mediante la modificación del código generado.

Para soportar la validación (y la corrección automática) de modelos se utilizó Acceleo, mientras que para la gestión de las relaciones comentadas en la Sección 3 se ha utilizado la familia de lenguajes Epsilon³. En particular, ETL ha sido el lenguaje utilizado para la implementación de las transformaciones que permiten generar modelos parciales y modelos de trazas a partir de un modelo elaborado con otra notación. El hecho de que ETL permita combinar bloques imperativos con la definición de reglas declarativas ha resultado especialmente útil, dado que la mayor parte de las transformaciones distan de ser directas y requieren ciertos niveles de interacción con el usuario, soportados fundamentalmente en forma de cuadros de diálogo implementados con EOL.

Por otro lado, para recoger las relaciones entre los modelos involucrados en cada transformación, se ha definido el metamodelo de trazas (o relaciones) mostrado en la Figura 9. De este modo, cuando se ejecuta cualquiera de las posibles transformaciones, además del modelo de destino, se genera el modelo de trazas correspondiente.

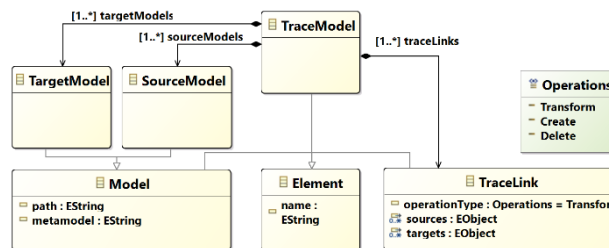


Figura 9. Metamodelo de trazas genérico.

³ <https://www.eclipse.org/epsilon/>

Para visualizar y gestionar estos modelos de relaciones se utiliza Modelink⁴, un editor multipanel que, como muestra la Figura 10, permite visualizar los modelos origen y destino junto a de trazas, además de permitir editar las relaciones existentes.

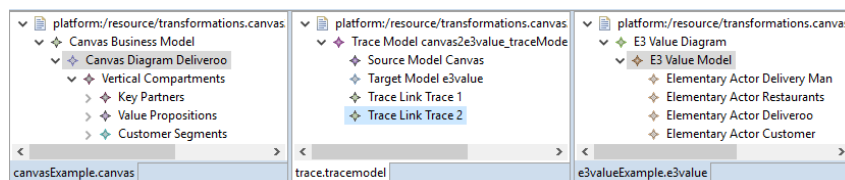


Figura 10. Modelo de relaciones entre los modelos Canvas y e³value (Modelink).

Finalmente, dado que INNoVaServ sigue siendo básicamente una herramienta EMF/GMF, es interoperable con cualquier otra herramienta construida sobre este marco de metamodelado, que viene siendo el estándar de-facto para el desarrollo de herramientas basadas en modelos desde hace más de una década. A modo de ejemplo, apoyándonos en Papyrus resultaría casi inmediato combinar el uso de modelos UML con los modelos soportados por INNoVaServ para el diseño de servicios.

5 Análisis de completitud

Dado que actualmente INNoVaServ soporta 5 notaciones diferentes, habría lugar a realizar 20 transformaciones que las conecten (combinaciones 2 a 2). Para reducir este número y hacerlo más manejable, se ha optado por seleccionar un subconjunto de esas transformaciones que, encadenándolas si fuera necesario, permita generar modelos en cualquiera de las 5 notaciones contempladas, con la menor pérdida de información posible. Así, las transformaciones implementadas son las que aparecen en la Figura 11.

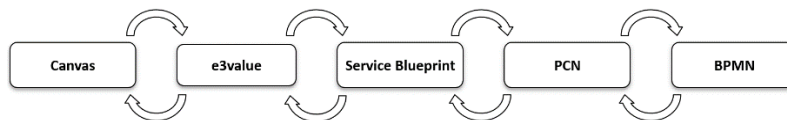


Figura 11. Transformaciones directas implementadas.

Para evaluar la utilidad de estas transformaciones, se ha llevado a cabo un análisis de completitud que, en términos coloquiales, permite conocer qué porcentaje del modelo destino pueden generar. Conviene recordar que, si bien las diferentes notaciones contempladas son complementarias, no son equivalentes, por tanto es imposible que todos los modelos generables sean completos, puesto que en algunos casos el modelo origen no tiene toda la información necesaria para elaborar el modelo destino.

Así, se analizan ciertos datos, como el porcentaje de elementos generados por cada transformación, respecto al número total de elementos del modelo elaborado por el diseñador de forma manual. De esta forma se dispone de un indicador que intuitivamente determina cuán completos (en términos cuantitativos) serán los modelos generados.

Para este análisis se han utilizado 5 casos de estudio diferentes: el que se utiliza en este artículo (*Deliveroo*), la empresa de alquiler de vehículos *Car2go*, la empresa de

⁴ <https://www.eclipse.org/epsilon/doc/modelink/>

alquiler de bicicletas *Bicimad*, una casa de apuestas online y un instituto de neuro-psicología y psicopedagogía. Nótese que este análisis está condicionado por la cantidad de información recogida en los modelos de origen: si éstos fueran incompletos, las transformaciones serían menos efectivas.

Con todo ello, la Tabla 1 muestra el porcentaje de los modelos generados con la herramienta para cada uno de los 5 casos de estudio. Los datos respaldan la intuición inicial y lógica: en la generación de modelos de negocio a partir de modelos de negocio (respectivamente modelos de proceso) el porcentaje de modelo generado es más alto. Por ejemplo, la generación de PCNs a partir de Service Blueprints es bastante efectiva, ya que ambos contienen en esencia las mismas acciones. De la misma forma, en el paso de BPMN a PCN, se conserva prácticamente toda la información.

	Canvas desde e3value	E3value desde canvas	E3value desde SBP	SBP desde e3value	BPM desde PCN	PCN desde SBP	PCN desde BPMN	BPMN desde PCN
Caso 1	32,00	68,97	3,45	8,70	100,00	44,64	85,71	60,24
Caso 2	38,71	76,56	3,13	8,00	100,00	45,16	93,55	75,00
Caso 3	47,83	80,77	3,85	9,52	95,24	45,65	91,30	73,17
Caso 4	28,13	79,41	2,94	6,67	93,33	49,18	90,16	73,53
Caso 5	37,14	72,22	2,78	5,88	100,00	40,28	87,50	55,36
MEDIA	36,76	75,59	3,23	7,75	97,71	44,98	89,65	67,46

Tabla 1. Porcentaje de modelos generados por la herramienta en los cinco casos de estudio.

A la luz de los datos recogidos en la tabla se puede concluir que en general las vistas parciales generadas por la herramienta se aproximan bastante a la realidad: de ocho posibles escenarios, en cuatro de ellos el porcentaje medio generado supera el 67% y otros dos están entre el 36% y el 50%. Además, hay dos más que están por debajo del 8%, pero estos corresponden a la transición entre modelos de negocio y de proceso. Cabe destacar no obstante que la automatización de estas transiciones es, aún así, una de las contribuciones más interesantes en el contexto del proceso de diseño, pues guiarán el paso del modelado de negocio a la implementación del modelo en forma de operaciones de servicio. Haciendo una analogía con la concepción más tradicional del proceso de desarrollo de software, estas transiciones guían el paso de la fase de análisis a la fase de diseño.

6 Trabajos relacionados

De entre las áreas relacionadas con este trabajo, un vistazo rápido a la gran cantidad de revisiones sistemáticas disponibles sobre el tema, permite identificar el modelado de procesos de negocio como el que más interés ha despertado por la comunidad IT hasta la fecha. De hecho, dada la madurez del área, las revisiones más recientes no se centran ya en caracterizar las propuestas existentes, sino en los mecanismos disponibles para evaluar su calidad [5] o complejidad [12].

Sin embargo, a pesar de la cantidad de propuestas existentes, todavía surgen nuevos enfoques para el modelado de procesos de negocio [6] y nuevos dialectos BPMN [14]. Muchos de estos trabajos tienen por objetivo acortar la distancia entre los profesionales de las áreas de negocio y las notaciones para el modelado de procesos de negocio [15].

Por otro lado, a pesar del creciente interés en este área que ha provocado el reciente aumento de las ofertas producto-servicio [3], el modelado de inteligencia de negocio [9] y otras disciplinas relacionadas, la investigación sobre el modelado de negocio orientado a la estrategia y a la organización todavía está una etapa temprana, probablemente debido a que el *boom* por los modelos de proceso precedió al de los modelos de negocio [4].

En este contexto, el objetivo de INNoVaServ, no es definir nuevos lenguajes para el modelado de negocio o de procesos, si no ofrecer un entorno integrado que soporte los modelos de procesos ya existentes, como Service Blueprint, BPMN o PCN y las notaciones para modelado de negocio orientadas a la estrategia y a la organización como Canvas o e³value. La idea de fondo es recortar la distancia que históricamente ha separado a las áreas de IT y de negocio. Un entorno como INNoVaServ permite que los modelos definidos y manejados por los profesionales de negocio se conecten directamente a los modelos utilizados por los profesionales de IT, expresados típicamente en términos de UML y/o BPMN.

Respecto a soluciones tecnológicas, y dado que no existe ninguna herramienta que soporte las cinco notaciones integradas en INNoVaServ, a continuación se presentan brevemente algunas de las herramientas existentes que soportan al menos dos de ellas.

*Canvanizer*⁵ y *Real Time Board*⁶ son dos aplicaciones web que permiten la edición colaborativa de diagramas Canvas y Service Blueprint. Poseen una interfaz gráfica simple e intuitiva, pero no están basadas en el uso de modelos, por lo que la información representada es meramente gráfica. No ofrecen capacidades de exportación en formatos adecuados para el posprocesamiento (como XML), por lo que la salida se reduce a una imagen. Ambas son soluciones comerciales y ofrecen una edición gratuita limitada.

Por otro lado, hasta la aparición de INNoVaServ, el soporte tecnológico para e³value, se limitaba a e³editor [8], una aplicación que permite representar diagramas e³value y persistirlos en formato RDF, lo que simplifica las tareas de exportación/importación.

Respecto a PCN, no se ha encontrado ninguna herramienta que soporte esta notación. La única forma de definir diagramas PCN hasta la fecha era utilizar aplicaciones de diagramación genéricas o incluso editores de imágenes, como MS Visio o Lucidchart.

Finalmente, existen multitud de herramientas BPMN, como Bonita Studio, Signavio, BizAgi o IBM WebSphere, cada una de ellas con diferentes capacidades.

7 Conclusiones y trabajos futuros

En este trabajo se han presentado las últimas funcionalidades incorporadas en INNoVaServ, un entorno de modelado para el diseño de servicios que soporta la elaboración de modelos *Canvas*, *e³value*, *Service Blueprint*, *PCN* y *BPMN*.

Los avances más recientes y relevantes respecto a la herramienta pasan por la incorporación de nuevas notaciones y la generación automática de modelos de relaciones. Para ellos se han desarrollado varios puentes tecnológicos basados en técnicas de procesamiento de modelos [16] que permiten generar vistas parciales de un modelo a partir

⁵ <https://canvanizer.com/>

⁶ <https://realtimeboard.com/>

de otro modelo expresado con una notación diferente y el correspondiente modelo de relaciones [17] que recoge las correspondencias identificadas entre el modelo origen y el modelo destino.

A medio plazo, el objetivo pasa por conectar los modelos de negocio con los datos generados por la operativa diaria de las organizaciones. Enriquecer los modelos con información de alto nivel generada a partir de estos datos, facilitará la identificación de fortalezas y debilidades, y podría facilitar la toma de decisiones estratégicas.

Finalmente, este trabajo supone un paso más en la línea que comenzamos hace algún tiempo, en el que nuestro objetivo no es tanto innovar en el área de la Ingeniería de Modelos, sino mostrar su utilidad en otras áreas. En este sentido, pensamos que las técnicas MDE han alcanzado ciertos niveles de madurez, pero necesitan un punto de vista más aplicado o realista [20]. Un movimiento en esta dirección pasa por ampliar su ámbito de aplicación a otros campos, como las áreas de negocio, donde tanto el modelado como la automatización pueden ayudar de manera decisiva a resolver problemas de interés para los expertos de esas áreas.

Agradecimientos. Este trabajo ha sido financiado parcialmente por el Gobierno de la Comunidad de Madrid, mediante el proyecto FORTE-CM (S2018/TCS-4314) y por el Ministerio de Economía y Empresa Español, mediante el proyecto MADRID (TIN2017-88557-R).

Referencias

1. Bitner, M.J., Ostrom, A.L., Morgan, F.N. Service blueprinting: A practical technique for service innovation (2008) *California Management Review*, 50 (3), pp. 66-94.
2. Bridgeland, D. M., & Zahavi, R. (2009). *Business modeling: a practical guide to realizing business value*. Morgan Kaufmann/Elsevier.
3. Cavalieri, S., & Pezzotta, G. (2012). Product–Service Systems Engineering: State of the art and research challenges. *Computers in industry*, 63(4), 278-288.
4. DaSilva, C. M., & Trkman, P. (2014). Business model: What it is and what it is not. *Long range planning*, 47(6), 379-389.
5. de Oca, I. M. M., Snoeck, M., Reijers, H. A., & Rodríguez-Morffi, A. (2015). A systematic literature review of studies on business process modeling quality. *Information and Software Technology*, 58, 187-205.
6. Estañol, M., Sancho, M. R., & Teniente, E. (2015, June). Verification and validation of UML artifact-centric business process models. In *Proceedings of CAiSE 2015*, 434-449. Springer.
7. Gordijn, J. E-business value modelling using the e3-value ontology. In W.L. Curry editor, *Value creation from e-business models*, pp. 98--127, Oxford, UK, 2004.
8. Gronback, R. C. (2009). *Eclipse modeling project: a domain-specific language (DSL) toolkit*. Pearson Education.
9. Horkoff, J., Barone, D., Jiang, L., Yu, E., Amyot, D., Borgida, A., Borgida, A. (2014). Strategic business modeling: representation and reasoning. *Software & Systems Modeling*, 13(3), 1015–1041.
10. Osterwalder, A., & Pigneur, Y. (2010). *Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers*. John Wiley and Sons.
11. Parnas, D.L. On the criteria to be used in decomposing systems into modules. *Communications of the ACM*, 15 (12) (1972), pp. 1053–1058.
12. Polančič, G., & Cegnar, B. (2017). Complexity metrics for process models – A systematic literature review. *Computer Standards & Interfaces*, 51, 104–117.

13. Sampson, S.E. (2012). Visualizing service operations. *Journal of Service Research*, 15(2), 182-198.
14. Solís-Martínez, J., Espada, J. P., Pelayo, C., -Bustelo, G., & Cueva Lovelle, J. M. (2014). BPMN MUSIM: Approach to improve the domain expert's efficiency in business processes modeling for the generation of specific software applications. *Expert Systems with Applications*, 41(4), 1864–1874.
15. Umuhoza, E., Brambilla, M., Ripamonti, D., & Cabot, J. (2015). An empirical study on simplification of business process modeling languages. *Proceedings of SLE 2015*, 13–24. ACM.
16. Vara, J. M., & Marcos, E. (2012). A framework for model-driven development of information systems: Technical decisions and lessons learned. *Journal of Systems and Software*, 85(10), 2368-2384.
17. Vara, J. M., Bollati, V. A., Jiménez, Á., & Marcos, E. (2014). Dealing with traceability in the MDD of model transformations. *IEEE Trans. on Software Engineering*, 40(6), 555-583.
18. Viyović, V., Maksimović, M., & Perisić, B. (2014, July). Sirius: A rapid development of DSM graphical editor. In *IEEE Intelligent Engineering Systems (INES), 2014* (pp. 233-238).
19. Wirtz, B. W., Pistoia, A., Ullrich, S., & Göttel, V. (2015). *Business models: origin, development and future research perspectives*. Long Range Planning.
20. Whittle, J., Hutchinson, J., & Rouncefield, M. (2014). The State of Practice in Model-Driven Engineering. *IEEE Software*, 31(3), 79–85.
21. Chinosi, M., & Trombetta, A. (2012). BPMN: An introduction to the standard. *Computer Standards & Interfaces*, 34(1), 124-134.
22. Estañol M., Marcos E., Oriol X., Pérez F.J., Teniente E., Vara J.M. (2017) Validation of Service Blueprint Models by Means of Formal Simulation Techniques. *ICSOC 2017*.
23. Elena Gómez-Martínez, Francisco Pérez-Blanco, Juan de Lara, Juan Manuel Vara, Esperanza Marcos. (2019) Formal Support of Process Chain Networks using Model-driven Engineering and Petri nets. *SAC 2019*.