



TESIS DOCTORAL

**INNoVaServ: un entorno de modelado
para el diseño de servicios**

Autor:

D. Francisco Javier Pérez Blanco

Director:

Dr. Juan Manuel Vara Mesa

Programa de Doctorado en
Tecnologías de la Información y las Comunicaciones

Escuela Internacional de Doctorado

2022



El Dr. D. Juan M. Vara, profesor del Departamento de Ciencias de la Educación, Lenguaje, Cultura y Artes, Ciencias Histórico-Jurídicas y Humanísticas y Lenguas Modernas de la Universidad Rey Juan Carlos, director de la Tesis Doctoral: “INNoVaServ: un entorno de modelado para el diseño de servicios” realizada por el doctorando D. Francisco Javier Pérez Blanco,

HACE CONSTAR QUE:

Esta tesis doctoral reúne los requisitos para su defensa y aprobación

En Madrid, a 22 de abril de 2022

Fdo: Juan Manuel Vara Mesa

*“No figura en ningún mapa;
los verdaderos lugares nunca lo están.”*

– Herman Melville

Resumen

Actualmente nos encontramos en un entorno cambiante en el que continuamente aparecen nuevas oportunidades de negocio, a la vez que otras se vuelven obsoletas mucho más rápido de lo que venía siendo tradicional. Responder a esta volatilidad pasa por el desarrollo o rediseño de productos y/o servicios que permitan reaccionar a las necesidades del mercado en todo momento. En gran medida, esta respuesta pasa por introducir o revisar constantemente el modelo y los procesos de negocio de la organización, para lo cual es clave que todos los estamentos de la misma dispongan de una visión clara, entendible y compartida de éstos, aun cuando utilicen técnicas diferentes para su representación.

Hoy en día existe una gran variedad de técnicas o notaciones para el modelado de negocio y proceso que permiten representar las formas en que una organización crea, proporciona y obtiene valor, siendo además pieza indispensable en la toma de decisiones estratégicas a corto, medio y largo plazo. En esencia, las técnicas para el modelado negocio y proceso permiten representar de forma gráfica la propuesta de valor de una compañía, así como los productos y servicios que ofrece, detallando las acciones y pasos necesarios para la provisión de esos servicios. Muchas de las técnicas presentan diferencias y similitudes entre sí, pero en la mayoría de los casos, la información que aportan es complementaria.

Sin embargo, aunque existen ciertas herramientas que permiten trabajar de manera aislada con alguna de estas notaciones, no existe un entorno que posibilite trabajar, de manera integrada y eficaz, con modelos elaborados mediante diferentes técnicas.

Por todo ello, el objetivo de la presente tesis doctoral es el desarrollo, mediante técnicas de ingeniería dirigida por modelos, de un entorno para el diseño de servicios que permita trabajar de manera integrada con varias notaciones para el modelado de negocio y proceso. En particular, el entorno soportará la elaboración de modelos con las notaciones *Business Model Canvas*, *e³value*, *Service Blueprint*, *Process Chain Network (PCN)* y *Business Process Model and Notation (BPMN)*, y la automatización de varias tareas con estos modelos, como la validación, la simulación o la generación de modelos parciales en una determinada notación, a partir de modelos elaborados con otra técnica diferente. Además, el entorno ofrecerá un cuadro de mando que facilitará la gestión e identificación de todos los modelos de negocio y proceso elaborados en cualquier proyecto de diseño de servicios.

Abstract

We are currently in a changing environment in which new business opportunities are continually appearing, while others are becoming obsolete much faster than ever. Dealing with this volatility requires the development or redesign of products and/or services that allow us to constantly react to market needs at all times. To a large extent, this involves introducing or permanently revising the organization's business model and processes. To that end, it is essential that all parts of the organization have a clear, understandable, and shared vision of these models, even if they use different techniques for their representation.

Nowadays there is indeed a wide variety of techniques or notations for business and process modelling that make it possible to represent the ways in which an organization creates, provides, and obtains value, as well as being an indispensable part of strategic decision-making in the short, medium, and long term. In essence, business and process modelling techniques allow the value proposition of a company to be represented graphically, as well as the products and services it offers, detailing the actions and steps necessary for the provision of these services. Many of these techniques have differences and similarities between them, but in most cases the information they provide is complementary.

However, although there are certain tools that allow working in an isolated manner with some of these notations, there is not an environment that makes it possible to work, in an integrated and efficient way, with business and process models developed with different techniques.

Therefore, the main goal of this thesis is the development, using model-driven engineering techniques, of an environment for service design that allows working in an integrated way with several notations for business and process modelling. In particular, the environment will support the elaboration of models with the following notations: *Business Model Canvas*, *e³value*, *Service Blueprint*, *Process Chain Network (PCN)* and *Business Process Model and Notation (BPMN)*. It will also support the automation of several tasks with these models, such as validation, simulation, or the generation of partial models in a certain notation, from models elaborated with a different one. In addition, the environment will provide a dashboard that will facilitate the management of the business and process models developed in the context of any given design project.

Agradecimientos

A Juan Manuel Vara, el capitán de este trabajo. Gracias por haber confiado en mí desde el primer momento. Gracias por todos tus consejos, incluido el primero de ellos: *“si entras aquí, que sea porque te guste de verdad”*. Gracias por tener soluciones para todo, por el tiempo y esfuerzo que has dedicado a este proyecto, por haber corregido mi rumbo en más de una ocasión. Es una verdadera suerte haberte tenido como director de esta tesis.

A Esperanza Marcos. Gracias por tu confianza y haberme permitido formar parte de la aventura Kybele. Gracias por ponerme a prueba con tus comentarios y hacerme ver con ellos diferentes puntos de vista de este trabajo.

A David Granada. Nada más embarcarme en este viaje pude ayudarte en algún pequeño detalle para el depósito de tu tesis. Tú no has parado de ayudarme en todo este tiempo. Has sido una gran fuente de inspiración. Gracias, de verdad.

Al resto de miembros de Kybele, especialmente a Valeria de Castro y a mis compañeros de laboratorio Cristian Gómez y Maricela Salgado, con quienes he compartido tantas charlas tratando de modelar y arreglar el mundo. Ya mismo comprobaréis la presión de incluir a tantas personas en tan pocas líneas. Gracias también a Javier Sevilla, quien me ha prestado todo su apoyo desde que le conocí, especialmente durante mi estancia predoctoral.

A todos los miembros del grupo DISE, quienes no pudieron acogerme mejor durante mi travesía por Ámsterdam. Gracias especialmente a Jaap Gordijn por su profesionalidad y por enseñarme tantas buenas prácticas. Gracias también a Fadime Kaya por su enorme esfuerzo en nuestras colaboraciones, por mostrarme la inutilidad de buscar el modelo perfecto y la satisfacción de casi encontrarlo.

A mamá y papá, quienes me han apoyado en todo, siempre, incluso cuando el viento me ha llevado al otro lado del mundo sin billete de vuelta. Gracias a vosotros, esta aventura, y cualquier otra, ha resultado muchísimo más sencilla.

A mi hermana Estefanía, a mi cuñado Carlos, a la pequeña Daniela y a Carmen, quien está a punto de llegar a este mundo de cosas por descubrir. Ellos me han aguantado desvariar en los días difíciles, los no tan fáciles, y los días buenos: es decir, en todo momento. Gracias por tantas cenas y tantas cervezas...

A mi abuelo, a quien tengo que contarle que esto no es sino el inicio de mis estudios. Gracias por habernos complacido tanto desde que éramos pequeños, por aquellos paseos al sol los fines de semana por la mañana.

Shūshu, Āyí, gracias también a vosotros por vuestra confianza y apoyo durante todo este tiempo, de manera especial durante mi estancia en Ámsterdam. Fēi cháng gǎn xiè.

Después de que mi compañero David leyera su tesis, escuché a Esperanza dar la enhorabuena a su novia porque “*no es fácil ser la pareja de un estudiante de doctorado*”. A Victoria, Chen Xiao Fan, mi compañera de viaje; la vela, el ancla, el timón... la quilla del barco. De no ser por tu empujón inicial, todo esto no habría comenzado. Gracias por todo, todo este tiempo.

A todos los que hemos sido y forman parte de lo que soy: Gracias.

Índice General

1. INTRODUCCIÓN	25
1.1 Planteamiento del problema y enfoque.....	25
1.2 Hipótesis y objetivos	29
1.3 Marco de investigación.....	31
1.3.1 <i>Proyectos de investigación y estancias</i>	32
1.3.1.1 Proyectos de investigación	33
1.3.1.2 Estancias	36
1.4 Método de investigación.....	37
1.4.1 <i>Design Science Research</i>	37
1.4.2 <i>Revisiones sistemáticas</i>	41
1.4.3 <i>Casos de estudio</i>	42
1.4.4 <i>Experimentos</i>	44
1.5 Estructura de la tesis	45
2. ESTADO DEL ARTE	49
2.1 Modelado de negocio y proceso	49
2.1.1 <i>Diseño de servicios</i>	50
2.1.2 <i>Notaciones para el modelado de negocio y proceso</i>	54
2.1.2.1 Business Model Canvas	54
2.1.2.2 E ³ value.....	59
2.1.2.3 Service Blueprint.....	61
2.1.2.4 Process Chain Network (PCN).....	64
2.1.2.5 Business Process Model and Notation (BPMN).....	68
2.2 Herramientas para el modelado de negocio y proceso.....	71
2.2.1 <i>Método</i>	72
2.2.2 <i>Cuestiones de investigación</i>	73
2.2.3 <i>Fuentes de información y cadenas de búsqueda</i>	74
2.2.4 <i>Criterios de inclusión y exclusión</i>	77
2.2.5 <i>Búsqueda y selección de estudios primarios</i>	78
2.2.6 <i>Extracción de datos</i>	79

2.2.7	<i>Análisis de las propuestas</i>	80
2.2.8	<i>Discusión</i>	85
2.3	Ingeniería Dirigida por Modelos (MDE)	87
2.3.1	<i>Desarrollo del Software Dirigido por Modelos (MDD)</i>	89
2.3.2	<i>Arquitectura Dirigida por Modelos (MDA)</i>	90
2.3.3	<i>Modelado y metamodelado</i>	91
2.3.4	<i>Lenguajes de Dominio Específico (DSL)</i>	93
2.3.5	<i>Transformaciones de modelos</i>	94
2.3.6	<i>Trazabilidad</i>	95
2.4	Tecnologías MDE.....	97
2.4.1	<i>Eclipse IDE</i>	98
2.4.2	<i>Eclipse Modeling Framework (EMF)</i>	100
2.4.3	<i>Graphical Modeling Framework (GMF)</i>	101
2.4.4	<i>Sirius</i>	103
2.4.5	<i>Epsilon</i>	104
3.	SOLUCIÓN TECNOLÓGICA	109
3.1	Introducción.....	109
3.2	Especificación de la solución.....	112
3.2.1	<i>Arquitectura conceptual</i>	112
3.2.2	<i>Sintaxis abstracta</i>	113
3.2.2.1	Business Model Canvas	114
3.2.2.2	E ³ value.....	115
3.2.2.3	Service Blueprint.....	116
3.2.2.4	Process Chain Network (PCN).....	117
3.2.2.5	Business Process Model and Notation (BPMN).....	117
3.2.3	<i>Correspondencias entre notaciones</i>	118
3.2.3.1	Modelos de negocio	119
3.2.3.2	Modelos de proceso	121
3.2.3.3	Modelos de negocio y proceso	123
3.2.4	<i>Diseño técnico</i>	125
3.3	Implementación de la solución	126
3.3.1	<i>Sintaxis concreta</i>	127

3.3.1.1	Editores GMF.....	128
3.3.1.2	Editores Sirius.....	132
3.3.2	<i>Puentes tecnológicos</i>	134
3.3.2.1	Transformaciones de modelos.....	134
3.3.2.2	Panel para la gestión de modelos	136
3.3.3	<i>Validación de modelos</i>	137
4.	VALIDACIÓN.....	141
4.1	Protocolo de validación	141
4.2	Caso de estudio.....	142
4.2.1	<i>Definición del caso de estudio</i>	143
4.2.2	<i>Protocolo de recogida de datos</i>	143
4.2.3	<i>Ejecución del caso de estudio</i>	144
4.2.4	<i>Análisis del caso de estudio</i>	149
4.3	Estudio empírico.....	153
4.3.1	<i>Planificación</i>	154
4.3.2	<i>Sujetos</i>	154
4.3.3	<i>Material</i>	155
4.3.4	<i>Análisis y resultados</i>	156
4.4	Conclusiones de la validación	161
5.	CONCLUSIONES	167
5.1	Análisis de consecución de objetivos	167
5.2	Principales contribuciones	172
5.3	Resultados científicos	175
5.4	Trabajos futuros.....	178
5.4.1	<i>Construcción de una versión web de INNoVaServ</i>	178
5.4.2	<i>Mejorar la integración entre notaciones de negocio y proceso</i>	179
5.4.3	<i>Conexión de modelos con datos reales</i>	179
5.4.4	<i>Integración de un DSL para Smart Contracts</i>	179
APÉNDICE A.	RECURSOS DEL ESTUDIO EMPÍRICO.....	183
A.1	Cuestionario inicial.....	183
A.2	Cuestionario final	185

A.3	Enunciado del caso de estudio	187
A.3.1	<i>Contextualización</i>	187
A.3.2	<i>Situación actual</i>	187
A.3.3	<i>Problema</i>	187
APÉNDICE B.	MANUAL DE USUARIO DE INNOVASERV	191
B.1	Descarga e instalación	191
B.2	Crear un nuevo proyecto.....	193
B.3	Trabajar con modelos Canvas, e ³ value, SBP y PCN	193
B.4	Trabajar con modelos BPMN	194
B.5	Validar modelos.....	196
B.6	Generar vistas parciales de modelos.....	196
B.7	Visualización de modelos de trazas	197
APÉNDICE C.	CONCLUSIONS	201
C.1	Analysis of achievements	201
C.2	Main contributions.....	205
C.3	Scientific Results	208
C.4	Future work	211
C.4.1	<i>Building a web version of INNoVaServ</i>	211
C.4.2	<i>Improving integration between business and process notations</i>	212
C.4.3	<i>Connecting models with real data</i>	212
C.4.4	<i>Integration of a DSL for Smart Contracts</i>	213
REFERENCIAS	215
ACRÓNIMOS	225

Índice de Figuras

Figura 1-1. Visión general de un modelo Canvas (izq.) y un modelo e ³ vaue (der.) elaborados con INNoVaServ.	27
Figura 1-2. Visión general (y simplificada) de un modelo Service Blueprint (sup. izq.), un modelo PCN (sup. der.) y un modelo BPMN (inf.) elaborados con INNoVaServ.	28
Figura 1-3. Proyectos de investigación y estancias de la tesis doctoral.	32
Figura 1-4. Actividades de DSR. Adaptado de [52].	38
Figura 1-5. Ciclos de DSR. Extraído de [54].	38
Figura 1-6. Instanciación de DSR durante la presente tesis doctoral.	40
Figura 1-7. Proceso de revisión sistemática propuesto por Biolchini [56].	41
Figura 1-8. Fases de validación mediante casos de estudio. Adaptado de [43].	43
Figura 2-1. Modelo del doble diamante para el diseño de servicios.	52
Figura 2-2. Secciones de un modelo Canvas. Extraído de [10].	55
Figura 2-3. Diagrama Canvas para el caso de estudio Deliveroo.	58
Figura 2-4. Diagrama e ³ value para el caso de estudio Deliveroo.	60
Figura 2-5. Diagrama Service Blueprint para el caso de estudio Deliveroo.	63
Figura 2-6. Diagrama PCN de un restaurante. Extraído de [24].	65
Figura 2-7. Diagrama PCN para el caso de estudio Deliveroo.	67
Figura 2-8. Extracto de diagrama BPMN para el caso de estudio Deliveroo.	70
Figura 2-9. Paradigmas MDE y terminología. Adaptado de [105].	88
Figura 2-10. Proceso MDD aplicado al modelado de negocio y proceso.	89
Figura 2-11. Jerarquía de modelado según [116].	92
Figura 2-12. Vista de diagrama (izq.) y árbol (der.) de un modelo e ³ value.	93
Figura 2-13. Proceso de transformación de modelos.	95
Figura 2-14. Relaciones de trazabilidad y enlaces de traza.	96
Figura 2-15. Arquitectura básica de Eclipse.	99
Figura 2-16. Vista en forma de árbol de un modelo EMF e ³ value.	101
Figura 2-17. Desarrollo de un diagramador GMF.	102
Figura 2-18. Panel de control de Eclipse GMF.	102

Figura 2-19. Diagramador GMF para la notación e ³ value.....	103
Figura 3-1. Pantalla de carga de INNoVaServ.	111
Figura 3-2. Arquitectura conceptual de INNoVaServ.	112
Figura 3-3. Metamodelo de la notación Canvas.	114
Figura 3-4. Resumen del metamodelo de la notación e ³ value.	115
Figura 3-5. Metamodelo de la notación Service Blueprint.	116
Figura 3-6. Resumen del metamodelo de la notación PCN.	117
Figura 3-7. Resumen del metamodelo de la notación BPMN.	118
Figura 3-8. Resumen de correspondencias entre Canvas y e ³ value.	119
Figura 3-9. Resumen de correspondencias entre SBP, PCN y BPMN.	121
Figura 3-10. Relación entre notaciones de modelado de negocio y proceso.	124
Figura 3-11. Arquitectura técnica de INNoVaServ.	125
Figura 3-12. Proceso genérico de desarrollo de los editores.	127
Figura 3-13. Metamodelo Ecore para PCN elaborado con EMF.....	129
Figura 3-14. Definición gráfica de acciones PCN con GMF.....	130
Figura 3-15. Definición de la paleta de herramientas del editor PCN con GMF.	130
Figura 3-16. Definición del modelo <i>.gmfmap</i> para el editor PCN con GMF.....	131
Figura 3-17. Diagramador GMF para la notación PCN.....	131
Figura 3-18. Extracto del metamodelo Ecore para BPMN elaborado con Sirius.	132
Figura 3-19. Definición gráfica de acciones BPMN con Sirius.....	133
Figura 3-20. Creación de la paleta de herramientas en Sirius.....	133
Figura 3-21. Diagramador Sirius para la notación BPMN.	134
Figura 3-22. Transformaciones directas implementadas en INNoVaServ.....	135
Figura 3-23. Metamodelo de trazas genérico.....	135
Figura 3-24. Detalle de implementación ETL.	136
Figura 3-25. Uso de Modelink para la gestión de modelos.	136
Figura 3-26. Panel de control de modelos de INNoVaServ.....	137
Figura 4-1. Modelo Canvas del caso de estudio Unidad TFG elaborado con INNoVaServ.	145
Figura 4-2. Modelo e ³ value del caso de estudio Unidad TFG elaborado con INNoVaServ.	146

Figura 4-3. Modelo de trazas generado por INNoVaServ en la conversión de Canvas a e ³ value.	146
Figura 4-4. Modelo Service Blueprint del caso de estudio Unidad TFG elaborado con INNoVaServ.	147
Figura 4-5. Modelo PCN del caso de estudio Unidad TFG elaborado con INNoVaServ.	148
Figura 4-6. Modelo de trazas generado por INNoVaServ en la conversión de Service Blueprint a PCN.	148
Figura 4-7. Modelo BPMN del caso de estudio Unidad TFG elaborado con INNoVaServ.	149
Figura 4-8. Modelo de trazas generado por INNoVaServ en la conversión de PCN a BPMN.	149
Figura C-1. Configuración del JDK en el archivo <i>eclipse.ini</i>	192
Figura C-2. Estructura del proyecto <i>finalThesis_examples</i>	192
Figura C-3. Creación de un nuevo proyecto en INNoVaServ.	193
Figura C-4. Edición en INNoVaServ de un diagrama Service Blueprint.	194
Figura C-5. Creación de modelos BPMN en INNoVaServ.	195
Figura C-6. Creación de diagramas BPMN en INNoVaServ.	195
Figura C-7. Validación de modelos con INNoVaServ.	196
Figura C-8. Carga de modelos en el panel de control de INNoVaServ.	197

Índice de Tablas

Tabla 1-1. Detalles del proyecto SICOMORo-CM.	33
Tabla 1-2. Detalles del proyecto ELASTIC.	34
Tabla 1-3. Detalles del proyecto MADRID.	34
Tabla 1-4. Detalles del proyecto FORTE-CM.	35
Tabla 1-5. Detalles del proyecto SerDigital.	35
Tabla 1-6. Detalles de la Red MDE.	36
Tabla 2-1. Elementos gráficos de la notación Canvas.	57
Tabla 2-2. Elementos gráficos de la notación e ³ value.	59
Tabla 2-3. Elementos gráficos de la notación Service Blueprint.	62
Tabla 2-4. Elementos gráficos de la notación PCN.	66
Tabla 2-5. Elementos gráficos básicos de la notación BPMN.	69
Tabla 2-6. Librerías digitales para la SLR.	74
Tabla 2-7. Bloques de las cadenas de búsqueda para la SLR.	75
Tabla 2-8. Cadenas de búsqueda para la SLR.	75
Tabla 2-9. Distribución de estudios clasificados por fuente de información.	79
Tabla 2-10. Ejemplo de extracción de datos de cada estudio primario.	79
Tabla 2-11. Herramientas analizadas.	81
Tabla 2-12. Vista general de herramientas de modelado de negocio y proceso. ...	84
Tabla 2-13. Herramientas de desarrollo de editores gráficos. Adaptada de [143].	97
Tabla 3-1. Correspondencias entre Canvas y e ³ value.	120
Tabla 3-2. Correspondencias entre BPMN, PCN y Service Blueprint.	122
Tabla 3-3. Elementos gráficos adicionales de la notación PCN.	138
Tabla 4-1. Tiempos empleados (en segundos) para la elaboración de los modelos del caso de estudio.	151
Tabla 4-2. Porcentaje de elementos generados por INNoVaServ en cada transformación.	152
Tabla 4-3. Resumen de resultados del cuestionario previo.	157
Tabla 4-4. Resultados medios del cuestionario final a preguntas sobre características específicas.	159

Tabla 4-5. Resultados del cuestionario final a preguntas sobre características globales.....	160
--	-----

Introducción

En la presente tesis doctoral se aborda, a grandes rasgos, el análisis de diferentes notaciones para el modelado de negocio y proceso, así como las relaciones o correspondencias entre ellas, con el objetivo último de construir un entorno integrado de modelado para el diseño de servicios.

La sección 1.1 de este capítulo describe el contexto y el problema que ha motivado la realización de esta tesis. A continuación, la sección 1.2 define la hipótesis de partida y los objetivos derivados de ésta. En la sección 1.3 se hace referencia al contexto en el que se ha desarrollado este trabajo, así como a los proyectos de investigación en los que ha participado el doctorando y las estancias de investigación realizadas durante el desarrollo de la presente tesis doctoral. Finalmente, la sección 1.4 describe el método de investigación seguido, y la sección 1.5 presenta la estructura del resto de contenidos de la tesis.

1.1 Planteamiento del problema y enfoque

La literatura especializada en el ámbito del modelado de negocio revela la existencia de un amplio número de definiciones acerca de qué son estos modelos [1]–[8]. Podemos encontrar algunas más académicas, como las proporcionadas por Teece en [9], donde se indica que *“los modelos de negocio nos permiten identificar las formas en que una organización provee valor a sus clientes, los atrae de tal modo que éstos pagan por dicho valor, y finalmente, convierte ese pago en beneficio para la compañía”*. Otra definición es la proporcionada por Osterwalder y Pigneur en [10], quienes argumentan que *“los modelos de negocio describen las bases desde las que las organizaciones crean, proveen y capturan valor”*. Otra, sin embargo más informal, es la proporcionada por Lewis en [11], donde se indica que *“un modelo de negocio es una representación de un plan para obtener un beneficio”*.

Independientemente de la definición utilizada, el concepto de modelo de negocio se ha considerado históricamente desde tres perspectivas diferentes: orientado a la tecnología, orientado a la estrategia y orientado a las organizaciones [12]. Por ejemplo, en el contexto de las tecnologías de la información, los modelos de negocio se identifican inmediatamente con modelos de proceso, mientras que, en el contexto de las teorías de la organización, se asocian en mayor medida con representaciones abstractas de la estructura o arquitectura de una compañía. Algunos autores distinguen incluso cuatro categorías de modelos de negocio, a

saber: modelos de procesos de negocio, modelos de motivación empresarial, modelos de organización empresarial y modelos de reglas de negocio [13][14].

Dos de estas perspectivas (modelos de organización empresarial y modelos de procesos de negocio), constituyen el núcleo del diseño de servicios [15], una disciplina tomada como referencia en esta tesis doctoral, y que tiene como objetivo ayudar en el desarrollo o mejora de propuestas para ofrecer servicios centrados en el usuario, poniendo el foco en las interacciones que se producen entre el proveedor y el consumidor [16]. Sus principios fundamentales consisten en la creación conjunta de valor, focalización en el usuario del servicio, naturaleza basada en procesos, visión holística del servicio diseñado, y búsqueda de evidencias tangibles durante el proceso de diseño [17], [18].

Habiendo nacido en el contexto de la investigación sobre marketing de servicios, el diseño de servicios evolucionó y ganó popularidad de la mano de IDEO¹ [19], hasta finalmente convertirse en la puerta de entrada al desarrollo de servicios para cualquier organización seriamente preocupada por la experiencia de usuario, la transformación digital, y una serie de factores relacionados [20]. Véanse, por ejemplo, los esfuerzos del gobierno británico en esta línea, materializados en la iniciativa *Government Digital Service*². Dicho proyecto dio pie, en abril de 2011, a la creación de un departamento dependiente del gabinete de gobierno, para el diseño y mantenimiento de nuevos servicios digitales gubernamentales centrados en los ciudadanos, logrando desde su creación un ahorro al gobierno de más de mil millones de libras.

El modelado de negocio y proceso juega un papel clave en el diseño de servicios, ya que los modelos nos ayudan a comprender, conceptualizar y representar los servicios que aportan valor a una organización. Por ejemplo, cualquier organización que desee adoptar un enfoque centrado en la provisión de servicios, necesita rediseñar su modelo de negocio constantemente [21]. Para ello, es fundamental que todos los miembros o departamentos de dicha organización, tengan una visión clara y compartida del modelo, incluso cuando utilicen notaciones diferentes para su representación [1], lo cual no es infrecuente.

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. Por lo tanto, resulta interesante identificar y registrar estas correspondencias, ya

¹ <https://www.ideo.com/>

² <https://gds.blog.gov.uk/>

que es habitual que los socios que colaboran para brindar un servicio determinado (o incluso diferentes equipos dentro de una misma organización) utilicen una o más de estas técnicas, con los consiguientes problemas de comunicación [14].

De este modo, en el contexto del diseño de servicios, algunas notaciones como el modelo de negocio *Canvas* (*Business Model Canvas*) [10] o el modelo de valor *e³value* [22], sirven para proporcionar una visión general y estratégica de la organización (ver Figura 1-1). Por otro lado, técnicas de modelado de procesos de negocio como *Service Blueprint* [19], *Business Process Model and Notation* (BPMN) [23] o *Process Chain Network* (PCN) [24], resultan clave para mostrar los detalles de la provisión de una oferta de servicios en particular (ver Figura 1-2). Conviene señalar que las notaciones mencionadas (que son las soportadas por el entorno de modelado construido en esta tesis doctoral), son tan solo algunas de las técnicas de modelado de negocio y proceso existentes. En la sección 2.1 se exponen los motivos que llevaron a la elección de este conjunto de notaciones para el presente trabajo.

En cuanto a la relación entre modelado de negocio y modelado de proceso, en esta tesis doctoral se adopta un enfoque que afirma que el modelado de negocio debería preceder al modelado de proceso [25]. Por lo tanto, antes de abordar la tarea de elaborar los modelos de proceso de negocio que permitirán representar las operaciones de servicio, las organizaciones deberían identificar en qué puntos se genera valor, siendo aquí donde el modelo de negocio resulta de gran utilidad [26].

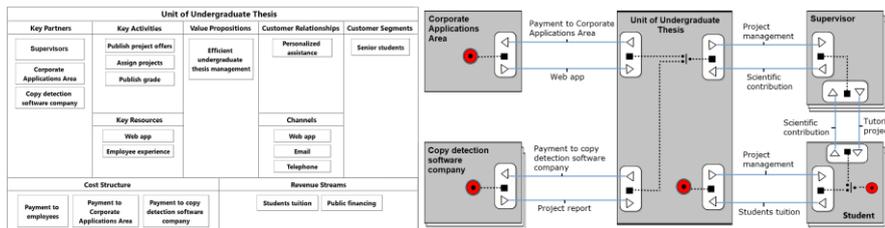


Figura 1-1. Visión general de un modelo Canvas (izq.) y un modelo e³vaue (der.) elaborados con INNoVaServ.

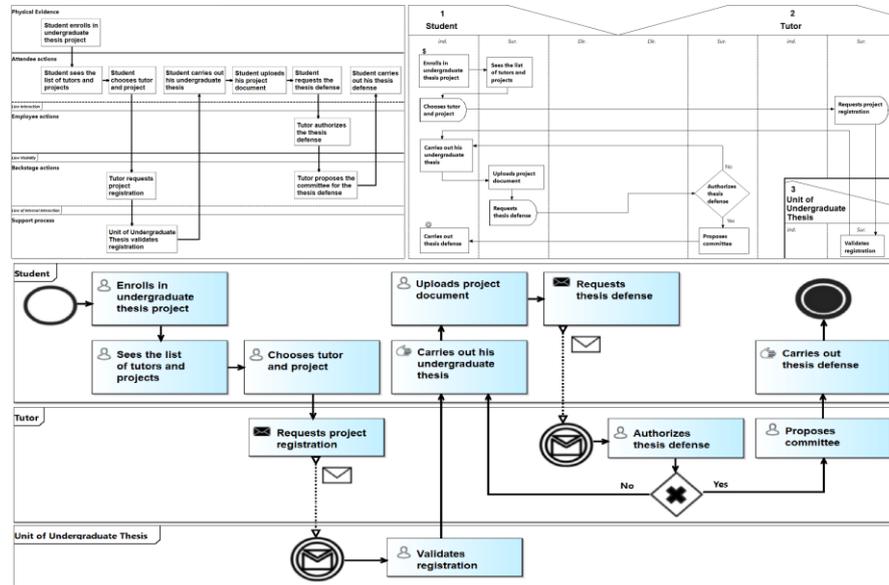


Figura 1-2. Visión general (y simplificada) de un modelo Service Blueprint (sup. izq.), un modelo PCN (sup. der.) y un modelo BPMN (inf.) elaborados con *INNoVaServ*.

Por otro lado, aun cuando existen herramientas que soportan algunas de las notaciones mencionadas, lo cierto es que, en el momento de la realización de esta tesis doctoral, no existe ninguna solución basada en modelos que permita trabajar de manera integrada y eficaz con todas ellas. Por el contrario, tendríamos que recurrir a diferentes herramientas específicas para cada notación, o a diagramadores genéricos como MS Visio³ o Lucidchart⁴. Aunque estas soluciones pueden ser buenas alternativas para realizar bocetos rápidos, lo cierto es que al no adoptar un enfoque centrado en modelos, no fueron diseñadas con la intención de facilitar el trabajo con los modelos que contienen esos diagramas, o para llevar a cabo un posprocesamiento de la información recogida en los mismos [27].

Por contra, la presente tesis doctoral adopta un enfoque centrado en modelos y explota los beneficios de la Ingeniería Dirigida por Modelos (MDE, *Model-Driven Engineering*) [28]. Para ello, se apuesta por la creación de un DSL (Lenguaje de Dominio Específico) gráfico para cada una de las notaciones de modelado de negocio y proceso mencionadas. Un DSL es un lenguaje creado específicamente para dar respuesta a los problemas de un dominio concreto, de

³ <https://www.microsoft.com/es-ww/microsoft-365/visio/flowchart-software>

⁴ <https://www.lucidchart.com/>

modo que proporciona mayor expresividad y facilidad de uso para ese dominio, frente a un lenguaje de propósito general (GPL) [29].

Una vez construidos los DSLs para cada una de las notaciones a soportar, es posible utilizar técnicas MDE para automatizar una serie de tareas relacionadas con el procesamiento de la información recogida en los modelos elaborados con esos DSLs [30], como su validación o la construcción de puentes tecnológicos que permitan conectarlos, para lo que se hace imprescindible un análisis previo de las correspondencias existentes entre los elementos de estos lenguajes.

Como resultado, esta tesis doctoral cubre la falta de soporte tecnológico para utilizar de manera integrada modelos de negocio y proceso en el contexto del diseño de servicios, mediante la construcción de *INNoVaServ*: un prototipo de código abierto que proporciona un conjunto de herramientas de modelado con las que es posible trabajar, de manera integrada, con distintas notaciones de modelado de negocio y proceso. Para ello, *INNoVaServ* integra un conjunto de DSLs, además de las herramientas necesarias para registrar y gestionar las relaciones entre los modelos elaborados con esos DSLs en el contexto de un proyecto de diseño de servicios. De este modo, además de cinco DSLs gráficos (Business Model Canvas, e³value, Service Blueprint, PCN y BPMN), *INNoVaServ* proporciona un panel de control que facilita la generación semiautomática de modelos parciales a partir de modelos expresados con una notación diferente, incluyendo un modelo de trazas para recoger las relaciones entre los modelos de entrada y salida. Adicionalmente, *INNoVaServ* permite validar y corregir modelos de manera automática, y ofrece capacidades de validación formal de modelos Service Blueprint y PCN.

Finalmente, cabe mencionar que el objetivo de esta tesis doctoral no pasa tanto por innovar en el área de la MDE, sino por mostrar su utilidad en el campo del diseño de servicios, ya que se parte de la idea de que las técnicas MDE han alcanzado un alto nivel de madurez, pero necesitan un punto de vista más aplicado o realista [31]. Por lo tanto, en esta tesis doctoral se pretende 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.

1.2 Hipótesis y objetivos

La **hipótesis** de partida de esta tesis doctoral es la siguiente: “*Es factible el desarrollo de un entorno de modelado que permita utilizar de forma integrada*

varias notaciones para el modelado de negocio y proceso en el contexto del diseño de servicios”.

Por su parte, el **objetivo principal** de esta tesis doctoral, derivado directamente de la hipótesis de partida, es el siguiente: *“Diseñar, construir y evaluar un entorno de modelado que permita utilizar de forma integrada varias notaciones para el modelado de negocio y proceso en el contexto del diseño de servicios”.*

Para la consecución del objetivo principal, se han establecido los siguientes **objetivos parciales**:

O1. Estudio de trabajos previos:

O1.1. Estado del arte de herramientas de modelado de negocio y proceso.

O1.2. Estudio de tecnologías MDE.

O2. Estudio y selección de las notaciones a soportar:

O2.1. Selección de un conjunto de notaciones para el modelado de negocio y proceso en el contexto del diseño de servicios.

O2.2. Estudio y análisis de las notaciones seleccionadas.

O3. Construcción de un entorno de modelado de negocio y proceso para el diseño de servicios:

O3.1. Especificación de la sintaxis abstracta de cada notación seleccionada.

O3.2. Especificación de la sintaxis concreta de cada notación seleccionada.

O3.3. Implementación de mecanismos de validación y corrección automática para cada DSL.

O3.4. Integración de los diferentes DSLs en un único entorno de modelado.

O4. Construcción de puentes tecnológicos que permitan conectar las distintas notaciones seleccionadas:

O4.1. Análisis de correspondencias entre las notaciones seleccionadas.

O4.2. Implementación de transformaciones semiautomáticas entre los DSLs, basadas en el análisis de correspondencias entre notaciones.

- O4.3.** Especificación de un metamodelo de trazas para recoger las relaciones entre los modelos de origen y destino de cada transformación.
 - O4.4.** Integración en el entorno de modelado de una solución multipanel para facilitar la visualización y análisis de los modelos involucrados en cada transformación.
 - O4.5.** Construcción de un panel de control para la gestión unificada de los modelos involucrados en un proyecto de diseño de servicios.
- O5.** Evaluación de la propuesta:
- O5.1.** Desarrollo de un caso de estudio para validar el correcto funcionamiento de la propuesta construida.
 - O5.2.** Desarrollo de un estudio empírico para evaluar las características de la propuesta construida.

1.3 Marco de investigación

La actividad investigadora del grupo Kybele de la Universidad Rey Juan Carlos, del que forma parte el doctorando, ha estado históricamente ligada a la provisión de soluciones metodológicas y tecnológicas basadas en modelos y la orientación a servicios.

Tiempo atrás se trabajó en la definición de MIDAS [32]–[36], una metodología centrada en la arquitectura para el desarrollo dirigido por modelos de Sistemas de Información (SIs). Dentro del marco MIDAS, se desarrollaron diferentes soluciones a problemas específicos en el contexto del desarrollo de SIs, como por ejemplo: SOD-M [37], una aproximación metodológica basada en Arquitecturas Dirigidas por Modelos (MDA, *Model-Driven Architecture*) [38] para el desarrollo orientado a servicios de SIs; PISA [39], una arquitectura de integración de portales web basados en servicios web semánticos; MIDAS MDA Tool (M2DAT) [40], la herramienta MDA que soporta cada uno de los métodos propuestos en MIDAS; MeTAGem [41], una meta herramienta para semi automatizar el desarrollo de transformaciones de modelos para M2DAT; ArchiMeDeS [42], un marco de trabajo metodológico para la especificación de arquitecturas de servicios construido sobre los principios del paradigma de la computación orientada a servicios; MeTAGeM-Trace [43], un marco de trabajo que automatizaba la generación de transformaciones de modelos que soportaban la producción de modelos de trazas; iTrace [44], un marco para la generación y gestión de trazas en proyectos de MDE, y CEViNEdit [45], una solución

tecnológica para la generación semiautomática de editores gráficos para DSLs visuales.

En los últimos años la actividad del grupo se ha volcado más en el mundo de los servicios, como pone de manifiesto la puesta en marcha en el 2014 del primer Grado en Ciencia, Gestión e Ingeniería de Servicios⁵, una titulación impulsada y pilotada por el grupo desde su concepción, y que se centra en el diseño, construcción, operación, mantenimiento y gestión de servicios durante todo su ciclo de vida. En este contexto, la actividad investigadora del grupo se ha centrado en el desarrollo de metodologías para el diseño de servicios [46], la evaluación de estos [47], o como en el caso de esta tesis doctoral, la provisión de soporte tecnológico para el diseño de servicios, lo cual permite poner en práctica todo el conocimiento que atesora el grupo sobre MDE para proporcionar soluciones en el ámbito de los servicios. En esta misma línea, recientemente también se está trabajando en la construcción de herramientas para la creación de servicios digitales mediante el uso de *Smart Contracts* [48].

1.3.1 Proyectos de investigación y estancias

La investigación llevada a cabo en esta tesis doctoral se ha realizado en el grupo de investigación Kybele⁶ de la Universidad Rey Juan Carlos.

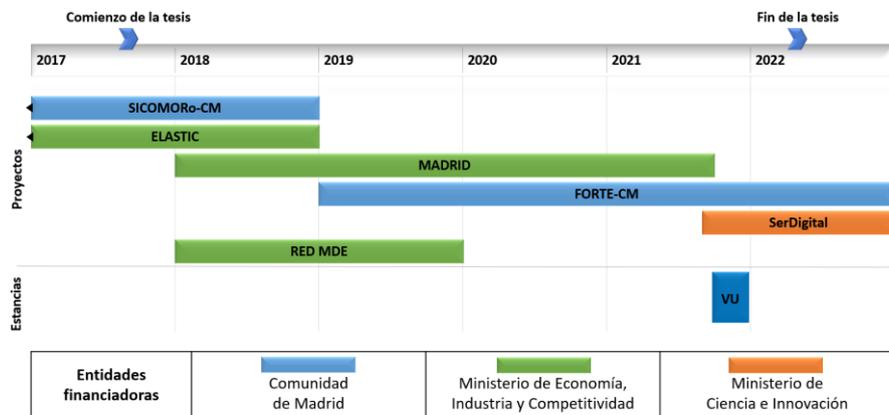


Figura 1-3. Proyectos de investigación y estancias de la tesis doctoral.

Tal y como muestra la Figura 1-3, este trabajo se enmarca en el contexto de los siguientes proyectos de investigación: SICOMORo-CM y su continuación,

⁵ <http://ise.edu.es/>

⁶ <http://www.kybele.es/es/>

FORTE-CM, ambos financiados por la Comunidad de Madrid; ELASTIC y su continuación, MADRID, financiados por el Ministerio de Economía, Industria y Competitividad; SerDigital (continuación de MADRID), financiado por el Ministerio de Ciencia e Innovación, y la Red de Excelencia en Ingeniería del Software Dirigida por Modelos (Red MDE), financiada por el Ministerio de Economía, Industria y Competitividad.

1.3.1.1 Proyectos de investigación

El proyecto SICOMORo-CM, financiado por la Comunidad de Madrid, tuvo como objetivo principal la construcción de metodologías, apoyadas en herramientas, que permitiesen el desarrollo de software confiable y de alta calidad utilizando un proceso riguroso que cubriese todas sus fases. En el contexto de este proyecto, el doctorando trabajó en el desarrollo del entorno *INNoVaServ*, así como en la integración de algunas herramientas desarrolladas por los diferentes grupos de investigación participantes en el proyecto. La Tabla 1-1 muestra los detalles de este proyecto.

Tabla 1-1. Detalles del proyecto SICOMORo-CM.

<i>Título</i>	SICOMORo-CM: Desarrollo de sistemas confiables mediante modelos y herramientas avanzadas
<i>Entidad financiadora</i>	Comunidad de Madrid
<i>Entidades participantes</i>	Universidad Complutense de Madrid, Universidad Rey Juan Carlos, Universidad Autónoma de Madrid, Universidad Politécnica de Madrid
<i>Duración</i>	Desde: 01/10/2014, Hasta: 31/12/2018
<i>Investigador principal</i>	Dr. Manuel Núñez

El proyecto ELASTIC fue financiado por el Ministerio de Economía, Industria y Competitividad, y tuvo por objetivo crear una metodología ligera y basada en modelos para la sistematización del proceso de servitización, lo cual supone un problema para las organizaciones que se enfrentan a un cambio de enfoque que pasa por la transformación desde una organización tradicional orientada a producto, a una organización moderna enfocada a servicios. En el contexto de este proyecto, el doctorando trabajó en el desarrollo de herramientas tecnológicas de soporte para las metodologías de servitización. La Tabla 1-2 muestra los detalles del proyecto ELASTIC.

Tabla 1-2. Detalles del proyecto ELASTIC.

<i>Título</i>	ELASTIC: proceso ligero de servitización basado en modelos: aplicación a las TIC
<i>Entidad financiadora</i>	Ministerio de Economía, Industria y Competitividad (TIN2014-52938-C2-1-R)
<i>Entidades participantes</i>	Universidad Rey Juan Carlos, IBM
<i>Duración</i>	Desde: 01/01/2015, Hasta: 31/12/2018
<i>Investigador principal</i>	Dra. Esperanza Marcos

El proyecto MADRID, continuación de ELASTIC y también financiado por el Ministerio de Economía, Industria y Competitividad, tuvo como objetivo la definición de una metodología centrada en el usuario para el desarrollo de servicios, identificando una serie de técnicas y prácticas, adaptadas de métodos como *design thinking* o la gamificación, orientadas a potenciar la creatividad y la participación del usuario en el diseño temprano de la solución. En este caso, el trabajo del doctorando también ha pasado por el desarrollo de un entorno de modelado para dar soporte a las diferentes fases del proceso propuesto. La Tabla 1-3 muestra los detalles del proyecto MADRID.

Tabla 1-3. Detalles del proyecto MADRID.

<i>Título</i>	MADRID: Marco metodológico centrado en el usuario para el diseño de servicios en la nueva sociedad digital
<i>Entidad financiadora</i>	Ministerio de Economía, Industria y Competitividad (TIN2017-88557-R)
<i>Entidades participantes</i>	Universidad Rey Juan Carlos
<i>Duración</i>	Desde: 01/01/2018, Hasta: 30/09/2021
<i>Investigador principal</i>	Dr. Juan M. Vara y Dra. Valeria de Castro

El proyecto FORTE-CM, continuación de SICOMORo-CM e igualmente financiado por la Comunidad de Madrid, tiene como objetivo principal definir y aplicar metodologías, apoyadas con herramientas, que permitan la creación de aplicaciones emergentes de alta calidad utilizando técnicas formales en todas las fases de su desarrollo. El trabajo del doctorando en este proyecto, ha consistido en el desarrollo de entornos de modelado de negocio y proceso, y su integración con otra herramienta diferente para el modelado de *Smart Contracts*. En la Tabla 1-4 pueden consultarse los detalles del proyecto FORTE-CM.

Tabla 1-4. Detalles del proyecto FORTE-CM.

<i>Título</i>	FORTE-CM: FORMal models and Technologies for Emerging application
<i>Entidad financiadora</i>	Comunidad de Madrid
<i>Entidades participantes</i>	Universidad Complutense de Madrid, Universidad Rey Juan Carlos, Universidad Autónoma de Madrid
<i>Duración</i>	Desde: 01/01/2019, Hasta: 31/12/2022
<i>Investigador principal</i>	Dr. Manuel Núñez

El proyecto SerDigital, continuación de MADRID y financiado por el Ministerio de Ciencia e Innovación, tiene como objetivo la especificación de un marco que permita guiar a las organizaciones en el proceso de transformación de sus modelos de negocio y gestión, a modelos de servicio o producto-servicio digitalizados. En este sentido, el trabajo del doctorando ha consistido en proveer apoyo tecnológico como soporte al modelado de procesos de la organización.

Tabla 1-5. Detalles del proyecto SerDigital.

<i>Título</i>	SerDigital: Marco para la servitización digital de organizaciones
<i>Entidad financiadora</i>	Ministerio de Ciencia e Innovación (PID2020-117244RB-I00)
<i>Entidades participantes</i>	Universidad Rey Juan Carlos
<i>Duración</i>	Desde: 01/09/2021, Hasta: 31/08/2025
<i>Investigador principal</i>	Dra. Esperanza Marcos y Dra. María Luz Martín

Por último, el doctorando también ha formado parte de la Red de Excelencia en Ingeniería del Software Dirigida por Modelos (Red MDE), financiada por el Ministerio de Economía, Industria y Competitividad, cuyos objetivos pasan por impulsar la excelencia científica de la comunidad MDE y consolidar la adopción de tecnologías MDE a nivel industrial. En el contexto de la red, el doctorando ha tenido la oportunidad de asistir a diferentes cursos cuyas temáticas estaban directamente relacionadas con el interés de esta tesis como, por ejemplo, técnicas de transformaciones de modelos o minería de procesos. La Tabla 1-6 muestra los detalles de la red.

Tabla 1-6. Detalles de la Red MDE.

<i>Título</i>	Red de Excelencia en Ingeniería del Software Dirigida por Modelos
<i>Entidad financiadora</i>	Ministerio de Economía, Industria y Competitividad (TIN2016-81836-REDT)
<i>Entidades participantes</i>	Universidad de Castilla-La Mancha, Universidad de Málaga, Universidad de Almería, Universidad Rey Juan Carlos, Universidad Autónoma de Madrid, Universidad de Murcia, Universidad Politécnica de Catalunya, Universidad de Extremadura, Universitat Oberta de Catalunya, Universitat Politècnica de València
<i>Duración</i>	Desde: 01/01/2018, Hasta: 31/12/2019
<i>Investigador principal</i>	Dr. Robert Clarisó

1.3.1.2 Estancias

Durante el desarrollo de la presente tesis doctoral, el doctorando llevó a cabo una estancia de investigación de tres meses en el grupo DISE⁷ de la Vrije Universiteit (VU) de Ámsterdam (Países Bajos). Dicha estancia se realizó entre septiembre y diciembre de 2021, ya que fue retrasándose debido a las restricciones causadas por la pandemia de la COVID-19, hasta que finalmente pudo llevarse a cabo con garantías.

Los intereses del grupo de investigación DISE, giran fundamentalmente en torno a la representación y modelado conceptual de sistemas de gobierno descentralizados. Además, el Prof. Jaap Gordijn, director del grupo y supervisor del doctorando durante su estancia, fue quien junto a Hans Akkermans, definió la especificación de la notación e³value, que ha sido objeto de estudio en esta tesis doctoral, y es de hecho una de las notaciones soportadas por el entorno de modelado construido.

Durante su estancia, el doctorando colaboró activamente en varias líneas de investigación con diferentes miembros del grupo, integrándose de pleno en la dinámica de trabajo del grupo receptor, como un miembro más del equipo. En particular, trabajó en el desarrollo de guías de derivación para generar modelos e³value a partir de modelos BPMN, la búsqueda de correspondencias entre notaciones de modelado de negocio y proceso, y la especificación de una ontología para la representación de sistemas descentralizados [49].

⁷ <https://dise-lab.nl/>

Cabe mencionar que las colaboraciones anteriores continúan una vez finalizada la estancia predoctoral, habiéndose materializado ya en el envío de varios trabajos a distintos foros internacionales y en la elaboración de nuevos trabajos en curso. No en vano, próximamente otro miembro del grupo Kybele realizará una nueva estancia de investigación en el grupo DISE. En definitiva, la estancia predoctoral del doctorando ha resultado realmente fructífera y será, muy probablemente, clave en el desarrollo de su carrera investigadora en el futuro.

1.4 Método de investigación

Este apartado describe los métodos de investigación utilizados durante el desarrollo de la presente tesis doctoral. Para guiar la investigación, se ha utilizado la metodología *Design Science Research* (DSR). Durante las primeras etapas, se ha realizado una revisión sistemática de la literatura (SLR, *Systematic Literature Review*) para conocer el estado del arte de las herramientas de modelado de negocio y operaciones de servicio. Por último, se han empleado casos de estudio, experimentos y encuestas durante las fases de validación y evaluación de la propuesta.

1.4.1 Design Science Research

Durante el desarrollo de esta tesis doctoral, se ha empleado la metodología *Design Science Research*. A grandes rasgos, DSR puede describirse como la generación de conocimiento a partir del diseño, construcción y evaluación de un artefacto, entendiendo por este término cualquier cosa artificial o sintética, desde una metodología, hasta un producto software [50]. Además, dicho artefacto debe ser una solución innovadora a un problema no trivial [51]. En el caso de esta tesis doctoral, el artefacto en cuestión consiste en la herramienta a construir.

Actualmente existen varios enfoques para DSR que se complementan entre sí. Por ejemplo, Wieringa [52] defiende que DSR consiste en el diseño e investigación de artefactos en un contexto determinado, de tal modo que dichos artefactos son diseñados para interactuar con ese contexto y mejorar ciertos aspectos del mismo. Como ya se ha dicho, por artefacto podemos entender desde una metodología hasta una herramienta software, mientras que por contexto podemos suponer una organización, un método, un proceso, etc. Según Wieringa, el artefacto por sí solo no soluciona ningún problema, sino que es la interacción entre éste y el contexto lo que contribuye a resolver el problema, de tal forma que

un mismo artefacto podría interactuar de manera distinta en contextos diferentes y, por tanto, resolver problemas también distintos.

Para Wieringa, por tanto, el objeto de estudio de DSR es un artefacto en un contexto determinado, mientras que las dos actividades principales son el diseño de la solución y el estudio del artefacto en su contexto. De este modo, como muestra la Figura 1-4, la metodología DSR itera en torno a dos ciclos o actividades: ciclo de diseño y ciclo empírico. En el primero se llevan a cabo las tareas de diseño y construcción del artefacto para dar solución a un problema del contexto. En el segundo se estudia el artefacto en su contexto, lo que proporciona nuevos conocimientos o descubrimiento de nuevos problemas que implican volver a la primera etapa.

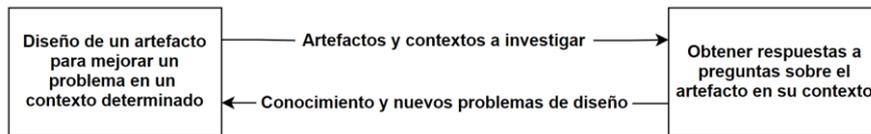


Figura 1-4. Actividades de DSR. Adaptado de [52].

Por otro lado, algunos autores como Hevner distinguen hasta tres ciclos en DSR, que no siguen un orden concreto y puede iterarse entre ellos durante la investigación [53]. Como muestra la Figura 1-5, dichos ciclos son: ciclo de relevancia, ciclo de rigor y ciclo de diseño. Nótese que, según esta especificación, el núcleo de DSR (parte central de la Figura 1-5), corresponde parcialmente con la visión de Wieringa.

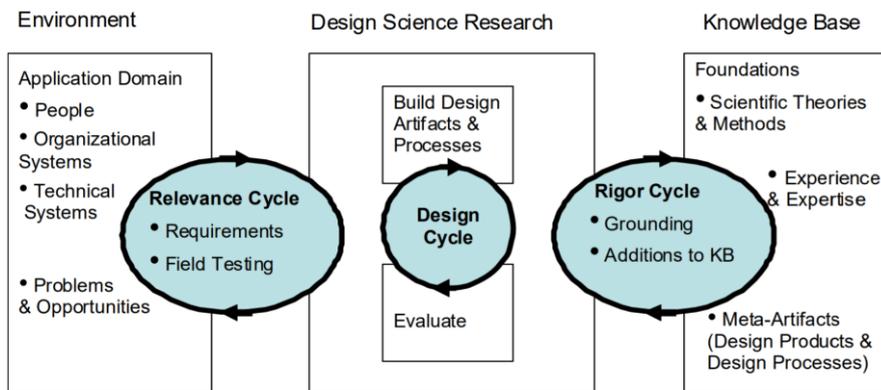


Figura 1-5. Ciclos de DSR. Extraído de [54].

- **Ciclo de relevancia:** esta etapa implica una toma de contacto con el contexto para el que se va a desarrollar el artefacto. Al igual que en

Wieringa [52], en este contexto podemos encontrar personas, sistemas tecnológicos, sistemas organizacionales o procesos, que interactúan para lograr un objetivo común [54]. El estudio del contexto, por lo tanto, permite identificar oportunidades y problemas en el ámbito de aplicación del artefacto. Pero DSR no se limita a la observación de un contexto, sino que busca interactuar con él a través del artefacto a construir. Por lo tanto, en esta etapa se definen también las preguntas y objetivos de investigación, así como los requisitos que deberá cumplir el artefacto en cuestión.

- **Ciclo de rigor:** en este ciclo, DSR conecta con una base de conocimientos científicos que incluye teorías, métodos ingenieriles, experiencias previas, artefactos, procesos existentes para el ámbito de aplicación de la investigación, etc. Por lo tanto, DSR se sirve de todo este conocimiento previo para asegurar que el artefacto desarrollado supone una contribución científica relevante, a la par que una innovación en su contexto. Pero, además, la investigación llevada a cabo puede aportar nuevas teorías, modelos, métodos, artefactos o experiencias adquiridas a las mismas bases de conocimiento, lo que a su vez puede afectar al ciclo de diseño del artefacto.
- **Ciclo de diseño:** este ciclo corresponde al núcleo de DSR. En él se itera entre las tareas de diseño del artefacto, su construcción y su evaluación, volviendo, si es necesario, a la fase de diseño a partir de los resultados de la evaluación. Algunos autores defienden que en esta etapa, se deben diseñar diferentes alternativas para posteriormente analizarlas y seleccionar la que cumpla en mayor medida con los requisitos estipulados [55]. Nótese que el ciclo de diseño está conectado con el resto de ciclos. Por ejemplo, los requisitos que marcan el diseño se extraen del ciclo de relevancia. A su vez, este diseño utiliza como base las teorías, técnicas, tecnologías, etc. extraídas del ciclo de rigor. Por su parte, el ciclo de diseño aporta nuevos componentes al contexto y a la base de conocimiento, lo que implica replantear los ciclos de relevancia y rigor.

Con todo lo anterior, la Figura 1-6 presenta cómo se ha instanciado la metodología DSR durante el desarrollo de la presente tesis doctoral. Como puede observarse, determinadas experiencias previas permitieron identificar la necesidad de construir un artefacto como el desarrollado a lo largo de este trabajo.

Estas experiencias previas se enmarcan en dos grandes grupos. Por un lado, los profesores de la asignatura de Ingeniería de Procesos de Negocio, del Grado en

Ciencia, Gestión e Ingeniería de Servicios de la Universidad Rey Juan Carlos, advirtieron que no existía un entorno que permitiese trabajar con varias notaciones de modelado de proceso, y en particular, con aquellas más populares en el contexto del diseño de servicios. Por otro lado, en el contexto del desarrollo de la tesis doctoral de D^a Maricela Salgado, el grupo de investigación al que pertenece el doctorando había comenzado a trabajar en la especificación de una propuesta metodológica para el diseño de servicios. Ya durante las primeras discusiones en torno a cómo debía abordarse la especificación de esa propuesta, resultó evidente que sería deseable disponer de un soporte tecnológico que permitiese utilizar modelos de negocio y proceso para el diseño de servicios. Estos dos hallazgos relacionados, dieron lugar al desarrollo de una SLR con la que comprobar si, efectivamente, no existía una solución tecnológica que reuniese estas características. En términos de DSR, esta revisión conforma, junto a las experiencias previas mencionadas, el contexto identificado durante el **ciclo de relevancia** de esta tesis doctoral.

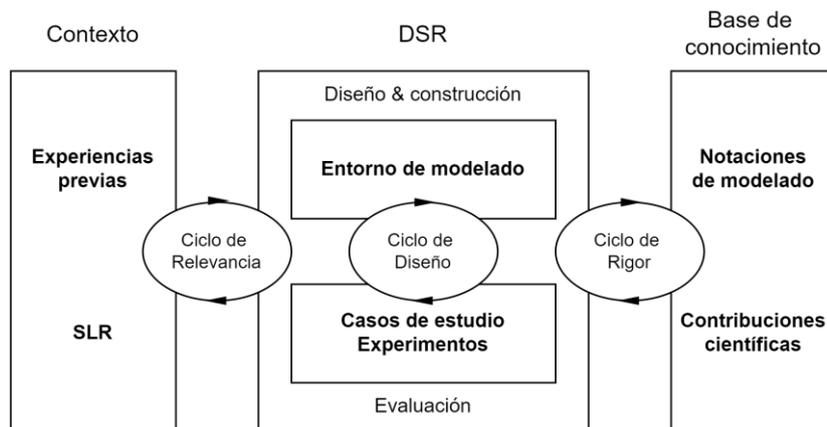


Figura 1-6. Instanciación de DSR durante la presente tesis doctoral.

Una vez identificado el problema a resolver y el ámbito de aplicación, durante el **ciclo de rigor** se recurrió a la base de conocimiento existente para analizar varios trabajos sobre notaciones de modelado de negocio y proceso, y otros centrados en el estudio de las correspondencias entre estas notaciones. Posteriormente, los resultados conseguidos a lo largo de esta tesis doctoral, permitieron añadir a esta base de conocimiento varias contribuciones científicas en forma de comunicaciones en distintos foros nacionales e internacionales.

Durante el **ciclo de diseño** se llevó a cabo la especificación y construcción del artefacto, aplicando la propuesta de [27] para el desarrollo de entornos de

modelado. Finalmente, para validar la funcionalidad del artefacto construido se llevó a cabo un caso de estudio de laboratorio, mientras que para evaluarlo se realizó un experimento en el que participaron diversos investigadores y profesionales del ámbito de los servicios.

1.4.2 Revisiones sistemáticas

Dado que parte del objetivo principal de esta tesis doctoral pasa por la construcción de un entorno de modelado, el cual permita trabajar de manera integrada con modelos de negocio y proceso elaborados mediante técnicas diferentes, resulta necesario, en primer lugar, determinar la situación actual en lo que respecta a este tipo de herramientas, para lo que se optó por realizar una revisión sistemática de la literatura.

La SLR es una metodología de investigación que tiene por objeto recopilar y evaluar las evidencias disponibles sobre un tema concreto, una cuestión de investigación o un fenómeno de interés. Para ello, las SLRs se desarrollan de un modo formal y sistemático, por lo que han de seguir una serie de pasos metodológicos que garanticen el valor y rigor científico de los resultados obtenidos, y permitan que otros profesionales puedan reproducir el procedimiento [56].

Para el desarrollo de la revisión sistemática de esta tesis doctoral, se han seguido las guías propuestas por Biolchini [56] y Bárbara Kitchenham [57]. De acuerdo con estas ideas, una revisión sistemática en el ámbito de la Ingeniería del Software, debe dividirse en actividades independientes agrupadas en las fases de **planificación, ejecución, análisis de resultados y empaquetado**. La Figura 1-7 muestra el proceso de la SLR seguido en esta tesis doctoral.

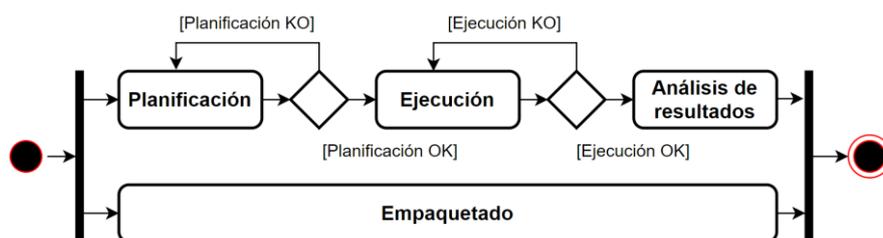


Figura 1-7. Proceso de revisión sistemática propuesto por Biolchini [56].

A continuación se resumen los principales aspectos de cada una de las etapas que componen el proceso de una SLR:

- **Planificación:** en esta etapa se han de definir los objetivos de la investigación y el protocolo para llevar a cabo la SLR [56]. En este protocolo se debe especificar diversa información, como por ejemplo cuáles son las preguntas de investigación a las que se pretende dar respuesta mediante la SLR, qué estrategias de búsqueda se utilizarán, en qué fuentes se realizará la búsqueda de información y mediante qué términos clave, cuáles son los criterios de inclusión y exclusión para seleccionar los trabajos que formarán parte de la SLR, o qué información se extraerá de cada estudio [57]. En definitiva, en esta fase se definirá el proceso a seguir durante las siguientes etapas.
- **Ejecución:** en esta fase se lleva a cabo el procedimiento definido durante la planificación. Para ello, en el caso de realizar una SLR utilizando fuentes digitales, se introducen las cadenas de búsqueda diseñadas en las distintas fuentes que se han seleccionado. A continuación, se recopilan los estudios obtenidos y se analizan y evalúan de acuerdo con los criterios de inclusión y exclusión definidos.
- **Análisis de resultados:** en esta etapa se sintetiza la información obtenida durante la fase de ejecución, con el objetivo de responder a las preguntas de investigación especificadas en la planificación.
- **Empaquetado:** de forma paralela al resto de etapas, se lleva a cabo el empaquetado de la información generada con el objetivo de almacenar los datos obtenidos en cada fase.

Además de las etapas principales, como puede apreciarse en la Figura 1-7, existen unos puntos de control para garantizar que estas fases se llevan a cabo de manera correcta. De este modo, si se detectasen incongruencias al finalizar la etapa de planificación, se debería volver atrás para solventar los errores encontrados. De igual forma, si hubiese problemas con los resultados obtenidos al finalizar la etapa de ejecución, se debería volver a dicha fase para corregir tales errores.

1.4.3 Casos de estudio

Para comprobar y validar una propuesta tecnológica existen diversas alternativas, como las investigaciones empíricas definidas por Kish [58]: encuestas, experimentos e investigaciones simples o casos de estudio.

En el caso concreto de esta tesis doctoral, se han llevado a cabo casos de estudio de laboratorio para validar el artefacto diseñado y construido. Los casos de estudio son investigaciones de tipo observacional que permiten analizar una propuesta en su contexto real, “*en las que no existe aleatoriedad de variables que no puedan controlarse, ni representatividad de los sujetos que componen la muestra de estudio*” [59]. Tampoco hay un control absoluto de la ejecución, aunque sí permiten controlar la medición de los datos obtenidos [60].

De acuerdo con [61]–[63], una investigación mediante casos de estudio comienza con la necesidad de analizar distintos aspectos de una propuesta, como por ejemplo su comportamiento en un contexto real. En nuestro caso, esa necesidad pasaba por validar el correcto funcionamiento del artefacto construido durante el diseño y modelado de un servicio concreto.

La Figura 1-8 muestra el resto de fases principales de una validación mediante casos de estudio.

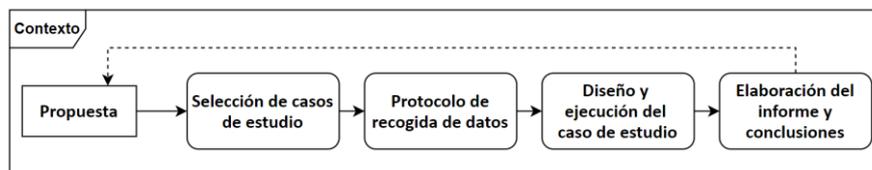


Figura 1-8. Fases de validación mediante casos de estudio. Adaptado de [43].

- **Selección de casos de estudio:** en este paso se definen los casos de estudio a utilizar, para lo cual es imprescindible conocer ampliamente la propuesta a validar, su contexto, y qué aspectos son los que se quieren analizar [62]. En este punto es recomendable elegir casos de estudio representativos, completos, y que permitan obtener un buen conjunto de lecciones aprendidas a partir de su desarrollo y ejecución [64].
- **Protocolo de recogida de datos:** el siguiente paso consiste en la definición de un protocolo que permita obtener una visión global del caso de estudio, así como de los procedimientos de ejecución, recolección de datos y documentación del caso. Para la recolección de datos es posible identificar distintos tipos de fuentes, como registro de archivos, entrevistas, observaciones directas, artefactos, etc. [61].
- **Diseño y ejecución del caso de estudio:** en este paso se diseña el caso de estudio, para lo cual, según [61] es importante definir las preguntas de investigación y los aspectos que se analizarán para darles respuesta, así como los criterios para interpretar los resultados. Una vez diseñado el caso

de estudio, se ejecuta y se recopilan los datos especificados en el protocolo de recogida de datos.

- **Elaboración del informe y conclusiones:** en esta última etapa se redacta el informe final sobre el desarrollo del caso de estudio. En esta memoria deben constar datos como el público al que se dirige el caso de estudio, si forma parte o no de un experimento más grande, el tipo de caso de estudio, etc. [61], [63], [64]. Una vez elaborado el informe, se deben analizar los resultados obtenidos para obtener conclusiones con las que validar la propuesta inicial. Si fuese necesario, en base a estas conclusiones se podrían atacar las debilidades detectadas en la propuesta [61].

1.4.4 Experimentos

Un experimento es un proceso llevado a cabo en un orden determinado con el objetivo de verificar, refutar o establecer la validez de una o varias hipótesis. De este modo, los experimentos proporcionan información sobre relaciones causa-efecto al demostrar qué resultado se produce cuando se manipula una o varias de las variables que posiblemente son su causa [65].

En esta tesis doctoral se ha llevado a cabo un experimento para evaluar el artefacto construido. En concreto, el experimento utilizado en esta investigación, se ha planificado y llevado a cabo de acuerdo a la metodología propuesta en [66] para la realización y presentación de informes de investigación empírica en la Ingeniería del Software. Las principales etapas de este método de investigación son:

- **Introducción:** planteamiento del problema y explicación de la solución.
- **Trabajos relacionados:** descripción del artefacto que está siendo investigado, así como de otros estudios relacionados.
- **Planificación:** definición de la información necesaria para replicar el experimento (objetivos, sujetos implicados, material utilizado, tareas, hipótesis, variables, diseño y procedimiento del experimento).
- **Ejecución:** descripción de cómo se llevó a cabo el experimento.
- **Análisis:** preparación de la información, estadística descriptiva y pruebas de las hipótesis.
- **Discusión:** explicación de los resultados del experimento e implicaciones.
- **Conclusiones:** presentación del impacto del experimento.

De manera general, en el experimento llevado a cabo en esta tesis doctoral participaron diferentes investigadores y profesionales del ámbito de los servicios, a los que se les entregó un cuestionario inicial con preguntas sobre experiencias previas de modelado y preferencias personales en lo que respecta a herramientas de soporte. A continuación, se les facilitó diversa documentación sobre las notaciones de modelado de negocio y proceso soportadas por el entorno construido en este trabajo, así como un caso de estudio a desarrollar con la herramienta y un pequeño manual de ésta. Por último, los participantes completaron un cuestionario final en el que expresaron su opinión sobre diferentes características del entorno. Los resultados obtenidos, permitieron no solo evaluar distintos aspectos del artefacto, sino también determinar cuáles eran sus puntos fuertes y aquellos en los que existía margen de mejora.

1.5 Estructura de la tesis

A continuación se ofrece una visión general del resto de contenidos de la presente tesis doctoral:

- El **capítulo 2** corresponde al estado del arte. En primer lugar, la sección 2.1 describe, en el contexto del diseño de servicios, las notaciones de modelado de negocio y proceso soportadas por el entorno desarrollado, y justifica la elección de estas. A continuación, la sección 2.2 revisa las soluciones tecnológicas existentes para soportar de forma integrada el modelado de negocio y proceso. Finalmente, para introducir las herramientas utilizadas para dar solución a los problemas identificados en los puntos anteriores, las secciones 2.3 y 2.4, describen los principios de la Ingeniería Dirigida por Modelos y algunas de las tecnologías que se han utilizado a lo largo de esta tesis doctoral, como determinados *frameworks* para la construcción de herramientas de modelado, DSLs para la gestión de modelos, etc.
- El **capítulo 3** presenta la solución tecnológica desarrollada en esta tesis doctoral: *INNoVaServ*, un entorno de modelado que permite utilizar de forma integrada varias notaciones para el modelado de negocio y proceso en el contexto del diseño de servicios. La sección 3.1 ofrece una visión general del entorno, mientras que la sección 3.2 presenta la especificación de la solución, comentando los aspectos más relevantes de su arquitectura conceptual, los metamodelos que definen la sintaxis abstracta de las diferentes notaciones soportadas, el análisis de correspondencias entre dichas notaciones y el diseño técnico del entorno. A continuación, la sección 3.3 presenta los detalles de implementación más relevantes de la

construcción de los distintos editores gráficos, el desarrollo de transformaciones que implementan los puentes tecnológicos con los que conectar las notaciones, o los mecanismos de validación formal para algunas notaciones, desarrolladas en colaboración con otros grupos de investigación.

- El **capítulo 4** describe la validación y evaluación del entorno de modelado construido en esta tesis doctoral. La sección 4.1 define el protocolo aplicado. La sección 4.2 muestra el desarrollo de un caso de estudio de laboratorio, llevado a cabo con el objetivo de comprobar el correcto funcionamiento de la herramienta. La sección 4.3 presenta un estudio empírico centrado en evaluar diferentes características del entorno. Por último, la sección 4.4 describe las principales conclusiones extraídas.
- El **capítulo 5** concluye resumiendo las principales contribuciones de esta tesis doctoral. En la sección 5.1 se analiza la consecución de los objetivos fijados al inicio del presente trabajo. La sección 5.2 enumera las principales contribuciones de la tesis, y en la sección 5.3 se listan las publicaciones científicas que sirven para contrastar estos resultados. Finalmente, la sección 5.4 presenta una serie de líneas de investigación para el trabajo futuro, que surgen a partir del trabajo desarrollado en esta tesis doctoral.

Adicionalmente, este documento incluye los siguientes apéndices:

- **Apéndice A:** cuestionarios y enunciado del caso de estudio empleado durante el experimento llevado a cabo en la fase de evaluación del entorno de modelado construido.
- **Apéndice B:** manual de usuario sobre las principales características del entorno de modelado construido.
- **Apéndice C:** conclusiones en inglés de la presente tesis doctoral.

Estado del Arte

El presente capítulo ofrece una visión global del estado del arte en lo que respecta a las dos grandes áreas temáticas relacionadas con la cuestión de investigación de esta tesis doctoral: el modelado de negocio y proceso (secciones 2.1 y 2.2), y la Ingeniería Dirigida por Modelos (MDE) (secciones 2.3 y 2.4).

De este modo, la sección 2.1 presenta las notaciones *Business Model Canvas*, *e³value*, *Service Blueprint*, *Process Chain Network* (PCN) y *Business Process Model and Notation* (BPMN) en el contexto del diseño de servicios. A continuación, la sección 2.2 presenta una revisión de la literatura en lo que a herramientas tecnológicas para el modelado de negocio y proceso se refiere.

Por su parte, dado que en esta tesis doctoral se propone la aplicación de principios y técnicas MDE para la construcción de un entorno de modelado de negocio y proceso, la sección 2.3 se centra en presentar algunos conceptos relacionados con la disciplina: Desarrollo del Software Dirigido por Modelos (MDD), Arquitectura Dirigida por Modelos (MDA), modelado y metamodelado, transformaciones entre modelos, trazabilidad y Lenguajes de Dominio Específico (DSL). Por último, la sección 2.4 presenta algunas de las tecnologías MDE que se han utilizado a lo largo de esta tesis doctoral: Eclipse, *Eclipse Modeling Framework* (EMF), *Graphical Modeling Framework* (GMF), Sirius y el conjunto de lenguajes y herramientas de Epsilon.

2.1 Modelado de negocio y proceso

Como se ha visto en la sección 1.1, existen multitud de definiciones sobre el concepto de modelo de negocio, como la de Osterwalder y Pigneur en [10], que afirman que un modelo de negocio describe las bases desde las que una organización crea, provee y captura valor.

Históricamente el concepto de modelo de negocio ha sido considerado desde tres perspectivas distintas: orientados a la tecnología, orientados a la estrategia y orientados a las organizaciones [12]. Por ejemplo, en el contexto de las tecnologías de la información, los modelos de negocio se identifican normalmente con modelos de proceso. Otros autores, sin embargo, distinguen hasta cuatro categorías de modelos de negocio: modelos de procesos de negocio, modelos de motivación empresarial, modelos de organización empresarial y modelos de reglas de negocio [13].

En cualquier caso, dos de las áreas donde este tipo de modelos han jugado tradicionalmente un papel más relevante, son la representación de la actividad de una organización (modelado de negocio) y la especificación de las operaciones que rigen la operativa diaria de esa organización (modelado de procesos). La presente tesis doctoral asume esta clara separación de funciones entre los modelos de negocio y los modelos de proceso.

En el contexto del diseño de servicios, el modelado de negocio permite ofrecer una visión rápida y estratégica de la organización, identificando las actividades que proporcionan valor a la misma. Para ello, es habitual utilizar notaciones como Business Model Canvas [10] o e³value [22].

Los modelos de proceso, por su parte, permiten mostrar el detalle de las operaciones que soportan los servicios identificados como generadores de valor durante el modelado de negocio. Además de BPMN [23], Service Blueprint [20] o Process Chain Network (PCN) [24] son notaciones especialmente populares para el modelado de procesos en el contexto del diseño de servicios.

Además de las notaciones mencionadas, existen muchas otras técnicas para el modelado de negocio y proceso como *Lean Canvas* [67], *Product Canvas* [68], *Resource Event Agent* (REA) [69], UML [70], Redes de Petri [71], etc. Sin embargo, en el contexto del diseño de servicios, Business Model Canvas, e³value, Service Blueprint, PCN o BPMN, son algunas de las notaciones más populares para el modelado de negocio y proceso [72]–[75].

Para tratar de ilustrar cómo encajan los modelos elaborados con estas notaciones en el diseño de servicios, se discuten a continuación las fases habituales en cualquier proyecto de esta disciplina, y cómo pueden utilizarse en cada una de las etapas las notaciones de modelado de negocio y proceso soportadas por el entorno construido en la presente tesis doctoral.

2.1.1 *Diseño de servicios*

Tomando como objetivo el seleccionar un conjunto de notaciones para el modelado de negocio y proceso que se adapte al contexto del diseño de servicios, se hace necesario en primer lugar acotar el ámbito de aplicación a una propuesta metodológica concreta. El diseño de servicios es una disciplina cuyo propósito es el de ayudar en el desarrollo o mejora de servicios centrados en el usuario, poniendo el foco en las interacciones que se producen entre el proveedor del servicio y el consumidor del mismo [16].

Los principios básicos del diseño de servicios son la orientación hacia las personas (se trata de servicios centrados en el usuario del mismo), la creación conjunta de valor (pretende crear valor mediante la interacción de diferentes actores), una naturaleza basada en procesos (no sólo se define la estructura y relaciones de una organización, sino también los procesos que debe llevar a cabo para la generación de valor), la búsqueda de evidencias tangibles (trata de buscar soluciones reales a problemas reales) y una visión holística del conjunto del servicio diseñado (visión del servicio como un todo distinto de la suma de cada uno de sus componentes) [17], [18].

Nacido en el contexto de la investigación sobre marketing de servicios, el diseño de servicios evolucionó y cobró relevancia de la mano de IDEO⁸ [19], para establecerse como referencia en el desarrollo de servicios para cualquier organización seriamente interesada en la experiencia de usuario, la transformación digital, o la servitización [20]. Un buen ejemplo de la importancia que ha cobrado la disciplina, es la iniciativa *Government Digital Service*⁹ del gobierno británico, en un esfuerzo por institucionalizar el diseño de servicios como práctica habitual e imprescindible en la administración, probablemente el mayor proveedor de servicios en cualquier país.

En el marco del diseño de servicios, el modelado de negocio y de procesos es, efectivamente, un factor esencial para lograr los resultados esperados, ya que las organizaciones deben rediseñar sus modelos de negocio constantemente. En este sentido, resulta necesario que todos los departamentos de una organización compartan una misma visión del modelo de negocio, incluso cuando utilicen notaciones diferentes para su representación. Estas notaciones, como ya se ha mencionado, pueden estar enfocadas a proveer una visión rápida y estratégica de la organización (modelos de negocio), o centradas en mostrar los detalles de una oferta de servicios en particular (modelos de proceso).

Para acotar el ámbito de aplicación de la propuesta de esta tesis doctoral, se ha seleccionado el modelo del doble diamante del *UK Design Council*¹⁰ por ser ampliamente reconocido como uno de los modelos más utilizados para el diseño de servicios (ver Figura 2-1), si bien es posible encontrar otras opciones como las que propone IDEO [19] o el Instituto Hasso-Plattner [76].

⁸ <https://www.ideo.com/>

⁹ <https://gds.blog.gov.uk/>

¹⁰ <https://www.designcouncil.org.uk/>

El modelo del doble diamante define un proceso de diseño que comprende dos fases con sendas etapas de razonamiento convergente y divergente. El objetivo de la primera fase es entender el contexto del problema de una manera global, descubriendo las necesidades de los usuarios para más tarde definir el problema que será abordado. Por lo tanto, en esta fase los modelos de negocio pueden resultar de gran ayuda. Por su parte, el objetivo de la segunda fase consiste en el diseño de múltiples soluciones para el problema identificado, y la especificación de los detalles de la solución seleccionada. En este caso, los modelos de proceso pueden ayudar en el diseño de ese conjunto de soluciones. Con todo ello, la Figura 2-1 muestra el modelo del doble diamante para el diseño de servicios, ilustrando cómo encajan las notaciones de modelado de negocio y proceso que se han seleccionado en cada una de sus fases.

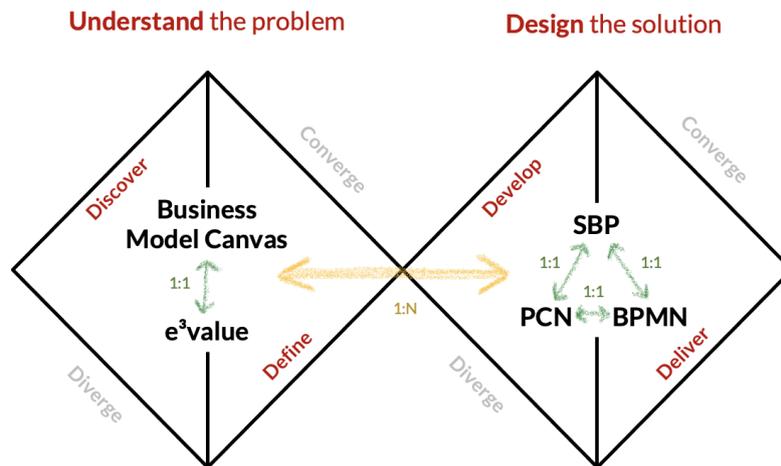


Figura 2-1. Modelo del doble diamante para el diseño de servicios.

Dado que la primera fase del proceso (comprensión del problema) precisa de una visión global del contexto, se han seleccionado dos notaciones que pueden ser utilizadas para la representación de la estructura global de un servicio desde diferentes puntos de vista: Business Model Canvas [10] y e³value [22]. Mientras que la primera ofrece una representación genérica de una organización, la segunda se centra en detallar las actividades e intercambios de valor que se producen entre los diferentes actores participantes en un servicio. De este modo, las notaciones Canvas y e³value pueden ayudar en esta primera fase, ya que permiten entender el contexto del negocio, lo que constituye el objetivo principal de las etapas de descubrimiento del problema y definición de este.

Una vez que se ha identificado el problema en la fase anterior, la segunda etapa del proceso requiere de modelos que permitan diseñar e implementar la solución, lo que implica la especificación del proceso que soporta la provisión del servicio. Para llevar a cabo esta tarea, se han seleccionado tres notaciones que, de manera similar al caso anterior, ofrecen diferentes puntos de vista de un mismo proceso. Por ejemplo, mientras que Service Blueprint [20] ofrece una visión simplificada, BPMN [23] permite especificar muchos más detalles y PCN [24] soporta la definición de elementos no representados en BPMN, como las pérdidas o ganancias monetarias o no monetarias relacionadas con cada acción. Así, los modelos elaborados con estas notaciones, juegan un papel fundamental durante las etapas de desarrollo y puesta en marcha de la segunda fase del proceso, ya que ayudan en el diseño de prototipos y la evaluación de la solución. En concreto, durante la etapa de desarrollo, es frecuente la utilización por parte de las áreas de negocio de modelos como Service Blueprint o PCN, con los que es posible discutir los detalles de las operaciones de servicio que deben ser implementados para poder ofrecer los servicios que dan respuesta a los problemas identificados en la fase anterior. A continuación, para la puesta en marcha, teniendo en cuenta que los profesionales TIC están más familiarizados con la notación BPMN, los modelos Service Blueprint y PCN pueden ser utilizados como entrada para generar los modelos BPMN que implementen la solución diseñada.

En resumen, es evidente que existen otras muchas técnicas para el modelado de negocio y proceso que también podrían haber sido seleccionadas para ser incluidas en el entorno a construir. La elección de las cinco notaciones soportadas viene determinada fundamentalmente por la gran popularidad de algunas de ellas, especialmente en el contexto del diseño de servicios, y por la posibilidad de producir una herramienta que facilite la comunicación entre los diferentes tipos de profesionales implicados en el diseño de servicios. En el caso particular de PCN, dado que esta notación no es actualmente de uso extendido, los motivos de su elección recaen más en el potencial de esta técnica para cubrir aspectos que otras notaciones no recogen, y en su utilización por los expertos en la gestión de operaciones [24].

En cualquier caso, el objetivo era soportar un conjunto de notaciones entre las que el profesional implicado en el diseño de servicios pueda elegir la que mejor encaje con su especialidad, experiencia o necesidades. Además, la herramienta construida en esta tesis doctoral ha sido diseñada para que pueda ser extendida, de forma que sea posible añadir todas aquellas notaciones que en un futuro se estimen necesarias.

La siguiente sección presenta las principales características de cada una de las cinco notaciones para el modelado de negocio y proceso soportadas.

2.1.2 Notaciones para el modelado de negocio y proceso

Una vez visto en la sección anterior el modelo del doble diamante para el diseño de servicios, y cómo encajan en él las notaciones seleccionadas para el modelado de negocio y proceso, a continuación se presentan las principales características de estas notaciones. Para ello, se utilizarán como ejemplos los modelos elaborados para un caso de estudio de laboratorio sobre la compañía de reparto de comida a domicilio *Deliveroo*¹¹.

A modo de resumen, *Deliveroo* es un servicio de entrega de comida a domicilio que tiene una serie de repartidores a lo largo de las ciudades en las que opera. Los usuarios pueden pedir comida de diferentes establecimientos a través de una aplicación móvil, y los repartidores recogerán el pedido y lo llevarán al domicilio de los clientes.

Antes de comenzar, conviene mencionar que ninguna de las notaciones que se presentan a continuación podría considerarse mejor o peor que otra. Todas ellas son de algún modo complementarias, y cualquiera de ellas puede ser utilizada en un momento dado dependiendo del objetivo que se persiga, y en función de múltiples factores, como por ejemplo el nivel de detalle con el que se deban representar las operaciones de servicio, o las necesidades y experiencia previa de la organización.

2.1.2.1 Business Model Canvas

Business Model Canvas (al que también nos referimos como Canvas a lo largo de esta tesis doctoral) es una técnica para la representación de propuestas de valor, ideada por el experto en negocios, innovación y marketing Alexander Osterwalder [10]. Canvas es probablemente la notación más popular entre los profesionales de las áreas de negocio para la representación de una idea o propuesta. 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 nueve módulos o dimensiones, que pueden verse representados en la Figura 2-2. A continuación se describen cada uno de estos nueve módulos:

¹¹ <https://deliveroo.co.uk/>

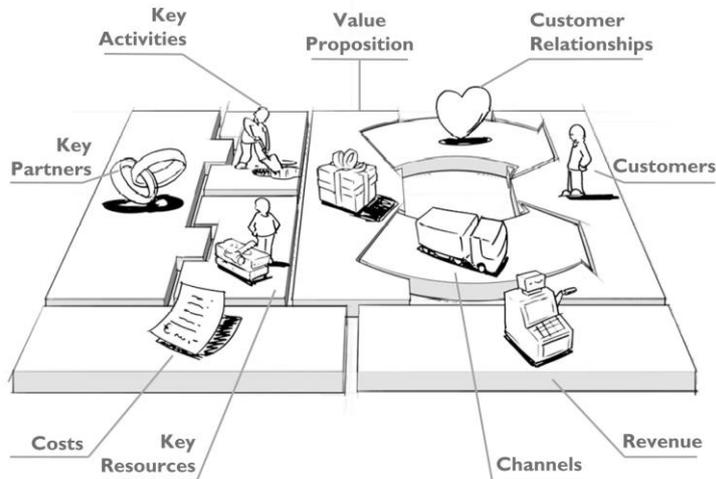


Figura 2-2. Secciones de un modelo Canvas. Extraído de [10].

- **Segmento de clientes (*Customer Segments*):** este módulo define los diferentes grupos de personas u organizaciones a las que la empresa pretende servir. Ninguna compañía puede sobrevivir mucho tiempo sin clientes. Por este motivo, para mejorar la satisfacción de estos, la empresa puede clasificarlos en diferentes segmentos que compartan necesidades e intereses comunes, presenten comportamientos similares, respondan a los mismos perfiles, etc. Así, en este módulo es posible especificar uno o varios segmentos de clientes hacia los que se enfocará el servicio. Una vez tomada esta decisión, el resto del modelo puede ser diseñado en base a las necesidades de ese conjunto de clientes, ya que cuanto más información se tenga sobre estos, mejor se podrá adaptar el servicio ofrecido a esas necesidades.
- **Propuesta de valor (*Value Proposition*):** este módulo define los productos/servicios que la compañía oferta a los segmentos de clientes definidos en el módulo anterior. En definitiva, se trata de qué valor ofrece la empresa a sus clientes, pudiendo ser esto algo cuantitativo (por ejemplo, velocidad de un servicio, cantidad de solicitudes disponibles, etc.) o cualitativo (como la experiencia de usuario).
- **Canales (*Channels*):** este módulo especifica qué medios utiliza la compañía para llegar a sus clientes y comunicarse con ellos con el objetivo de hacerles llegar su propuesta de valor. Puede decirse que estos canales de

comunicación, distribución y ventas, hacen de interfaz entre los clientes y la organización.

- **Relaciones con los clientes (*Customer Relationships*):** este módulo define las relaciones que se establecen y mantienen entre la compañía y sus clientes, con el objetivo de hacer tangible dicha relación y materializarla en beneficios a través de las propuestas de valor. Entre otras cosas, esta sección permite definir cómo fidelizar a los segmentos de clientes.
- **Fuentes de ingreso (*Revenue Streams*):** este módulo permite especificar los ingresos que la compañía espera generar por cada segmento de mercado. Sin embargo, para obtener el beneficio, habría que restar los gastos de la organización. Al igual que los clientes, los ingresos son un elemento fundamental para cualquier organización. Cualquier compañía debería preguntarse por qué propuestas de valor los clientes estarían dispuestos a pagar. Respondiendo a esta pregunta, la organización puede generar una o varias fuentes de ingreso para cada segmento de clientes.
- **Recursos clave (*Key Resources*):** este módulo identifica los principales activos que la compañía necesita para hacer funcionar el conjunto del modelo de negocio. En este sentido, toda compañía precisa de una serie de recursos que le permiten crear y ofrecer la propuesta de valor, llegar a los distintos segmentos de clientes, mantener los canales y relaciones con estos, o generar flujos de ingreso. Estos recursos pueden ser tanto activos materiales (materias primas, servidores, oficinas, etc.) como no materiales (conocimiento, capital humano, etc.). Además, los recursos pueden ser propiedad de la compañía, estar alquilados, o haber sido adquiridos por los socios clave.
- **Actividades clave (*Key Activities*):** en este módulo se especifican las actividades necesarias para poder ofrecer la propuesta de valor a los distintos segmentos de clientes. Algunas de estas actividades pueden estar relacionadas con el funcionamiento interno del negocio, con intercambios de valor con socios clave, etc.
- **Socios clave (*Key Partnerships*):** este módulo especifica la red de proveedores y socios que hacen posible el funcionamiento del modelo de negocio. Las alianzas entre diferentes socios o actores son cada vez más necesarias en los modelos de negocio actuales, ya que permiten generar valor de manera conjunta entre varias organizaciones.
- **Estructura de costes (*Cost Structure*):** en este módulo se especifican los costes derivados del funcionamiento del modelo de negocio. Estos costes

pueden ser estimados después de definir el resto de elementos del modelo, como los recursos clave, las actividades clave o los socios clave. Esta especificación de costes permite identificar, por ejemplo, en qué puntos se producen más gastos a lo largo de la provisión de la propuesta de valor.

A continuación, para mejorar la comprensión del modelo, la Tabla 2-1 ilustra los principales elementos gráficos de cualquier diagrama que represente un modelo Canvas:

Tabla 2-1. Elementos gráficos de la notación Canvas.

<i>Elemento</i>	<i>Representación</i>	<i>Descripción</i>
Lienzo		Panel dividido en nueve secciones correspondientes a los nueve módulos de la Figura 2-2.
Anotación		Etiqueta posicionable en cada una de las nueve secciones del lienzo para representar las entidades de cada módulo. Cada sección puede contener de 0 a N anotaciones.

Como ejemplo de modelo Canvas, la Figura 2-3 muestra un diagrama para el caso de estudio de *Deliveroo*. Puede observarse que la compañía tiene dos socios clave: restaurantes y repartidores. Su propuesta de valor está enfocada en el servicio de entrega de comida de calidad, manteniendo los precios del restaurante, pudiendo programar envíos, y prometiendo expandir el negocio de los restaurantes participantes. Para llevar a cabo esta propuesta de valor, sus actividades clave son el reparto de comida a domicilio y el mantenimiento de una aplicación móvil con la que proveer el servicio. Los recursos de los que dispone son sus empleados, la plataforma tecnológica y los acuerdos con restaurantes. El segmento de clientes al que se dirige son personas que no quieran/puedan cocinar y que deseen comida de calidad. Para comunicarse con sus clientes, la compañía hace uso de su aplicación móvil y página web, mientras que la relación con ellos se establece por medio de redes sociales donde se ofrecen descuentos, o mediante su disponibilidad las veinticuatro horas del día, los siete días de la semana. Por último, sus principales costes vienen determinados por el pago a empleados y repartidores, y el mantenimiento de la infraestructura tecnológica, mientras que sus principales

flujos de ingreso provienen de las comisiones cobradas a restaurantes, y las tasas de gestión y gastos de envío cobradas a los usuarios.

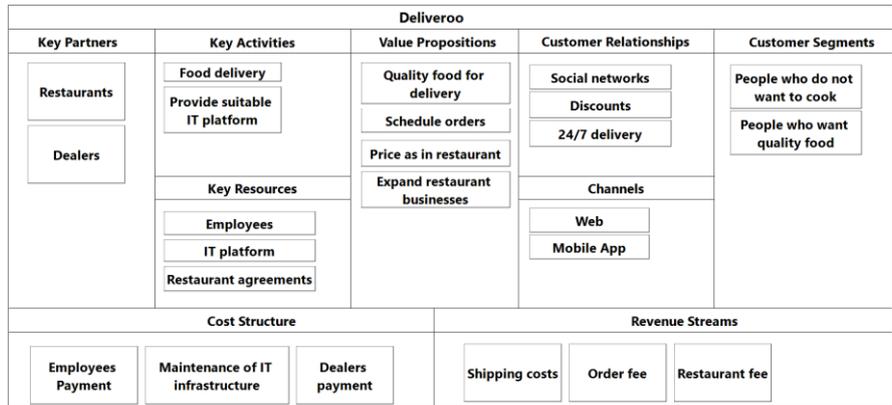


Figura 2-3. Diagrama Canvas para el caso de estudio Deliveroo.

Entre las ventajas del modelo de negocio Canvas, destaca la posibilidad de definir de manera global un negocio, visualizando como un todo los diferentes componentes implicados en la generación de valor para los clientes. Por lo tanto, es posible especificar, de una manera resumida, los detalles importantes de un modelo de negocio, lo que permite tener una idea clara del mismo, aunque sin un alto nivel de detalle. Además, permite vislumbrar qué relaciones existen con los diferentes actores, pero no permite detallar cómo se llevan a cabo estas relaciones ni entre qué participantes.

Respecto a las desventajas, nos encontramos con que, debido a su simplicidad, no es posible profundizar en los elementos que componen el modelo, por lo que únicamente resulta de utilidad para identificar las partes fundamentales del negocio. Además, para complementar la información que ofrece, debemos recurrir a modelos de proceso que permitan representar las operaciones de servicio que integran la propuesta de valor. Sin embargo, es cierto que esta simplicidad forma parte de las características del modelo Canvas.

En resumen, el modelo de negocio Canvas proporciona una visión global de una propuesta de negocio, 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 [10].

2.1.2.2 E³value

E³value es una técnica para el modelado de negocio y valor propuesta por los profesores Jaap Gordijn y Hans Akkermans de la Vrije Universiteit de Ámsterdam [77], que permite representar gráficamente una idea de negocio. Así, 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) [22].

Si Canvas ofrecía una visión global del conjunto de una propuesta de valor, e³value pone el foco en los actores y segmentos de mercado participantes en la propuesta, las actividades de valor que lleva a cabo cada uno, y los intercambios que se producen entre ellos. La Tabla 2-2 ilustra los principales elementos gráficos que componen un modelo e³value.

Tabla 2-2. Elementos gráficos de la notación e³value.

<i>Elemento</i>	<i>Representación</i>	<i>Descripción</i>
Actor		Un actor es una entidad individual que participa en el modelo de negocio. Por ejemplo: una compañía, un organismo público, etc.
Segmento de mercado		Cuando un actor representa a un grupo de entidades, se denomina segmento de mercado. Por ejemplo: clientes, repartidores, restaurantes.
Actividad de valor		Las actividades de valor se posicionan dentro de los actores o segmentos de mercado para representar actividades que generan beneficios.
Necesidad de usuario		Representa la necesidad de usuario que da inicio al intercambio de valor.
Elemento de límite		Representa el límite a partir del cual no se muestran más detalles sobre cómo se lleva a cabo la prestación del servicio.
Interfaz de valor		Representa el punto en el que se producen intercambios de valor entre actores o entre actividades de valor de un mismo actor.
Puerto de valor		Representa la dirección de un objeto de valor en un intercambio de valor.

Intercambio de valor		Representa los canales a través de los que se lleva a cabo un intercambio de objetos de valor.
Objeto de valor		Representa los bienes, productos o servicios intercambiados.
Conexión		Permiten enlazar los diferentes elementos contenidos en los actores.
Dependencia AND		Permite representar el operador booleano AND cuando se combinan varias actividades en un actor.
Dependencia OR		Permite representar el operador booleano OR en la realización de varias actividades de un actor.
Implesión / Explosión		Permite representar que un actor lleva a cabo más de un intercambio de valor, utilizando la misma interfaz de valor.

La Figura 2-4 muestra un diagrama e³value elaborado para el caso de estudio *Deliveroo*. Como se observa, se ha modelado el actor *Deliveroo* y los segmentos de mercado clientes, restaurantes y repartidores. La provisión del servicio comienza con una necesidad del cliente, lo que da lugar a un intercambio de valor entre éste y *Deliveroo*, siendo los objetos de valor intercambiados la entrega de comida a domicilio y la tasa por el pedido. Cuando *Deliveroo* recibe el pedido, se activan los mecanismos que inician nuevos intercambios de valor entre la compañía y los restaurantes (comida a cambio del coste de ésta), y la compañía y los repartidores (servicio de entrega a cambio de un importe determinado).

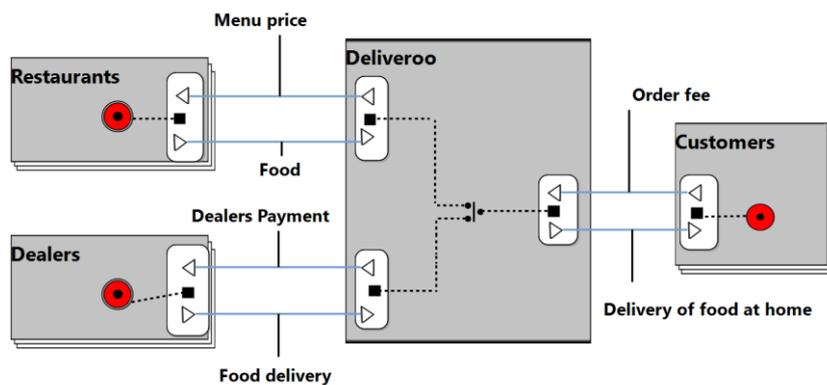


Figura 2-4. Diagrama e³value para el caso de estudio Deliveroo.

Respecto a las ventajas de la notación e³value, se trata de un modelo que de un modo sencillo permite identificar los distintos flujos de negocio que se producen en la provisión de un servicio, así como los actores que participan en él, quiénes actúan como consumidores y quiénes como proveedores, o qué bienes y/o servicios se intercambian. Además, también permite mostrar las actividades más importantes de cada actor, así como las relaciones que ocurren entre estas.

Como desventajas, e³value no permite descomponer actividades en otras más sencillas, por lo que habrían de tratarse como actividades diferenciadas. Además, si bien e³value se centra en representar los intercambios de valor que se producen entre los actores involucrados en la provisión de un servicio, lo cierto es que no detalla cómo se llevan a cabo esos intercambios, para lo que habría que recurrir a modelos de proceso.

En este punto, vale la pena incidir de nuevo en la complementariedad entre Canvas y e³value, y mencionar ciertas correspondencias que pueden observarse entre ambos modelos en el caso de estudio de *Deliveroo*. Por ejemplo, los socios clave del modelo de la Figura 2-3, aparecen como actores o segmentos de mercado en el modelo e³value. A su vez, los objetos de valor del modelo e³value, pueden observarse como anotaciones del modelo Canvas en diferentes módulos, como la estructura de gastos, flujos de ingreso o actividades clave.

2.1.2.3 Service Blueprint

Service Blueprint es una notación gráfica propuesta por G. Lynn Shostack para el diseño de operaciones de servicio, que se centra en detallar la interacción cliente-proveedor en la prestación de un servicio determinado [19]. Para ello, estructura las acciones que se llevan a cabo durante la provisión de un servicio en cinco segmentos distintos (o zonas de interacción), lo que permite diferenciar las actividades que corresponden al proveedor, de aquellas que lleva a cabo el consumidor. A continuación se describen cada una de las cinco zonas de interacción:

- **Evidencia física (*Physical Evidence*):** en esta zona pueden situarse las evidencias, hechos o acciones que desencadenan la interacción entre un cliente y un determinado proveedor, y aquellos elementos tangibles asociados a cada acción que tienen el potencial de influir en la percepción del servicio por parte del cliente.
- **Acciones del cliente (*Attendee Actions*):** en esta región se sitúan las acciones que lleva a cabo el cliente durante la provisión del servicio.

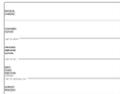
- **Acciones del proveedor (*Employee Actions*):** en esta zona tienen lugar las actividades del proveedor del servicio que interactúan directamente con el cliente. Por este motivo, esta región también puede denominarse *Front-of-Stage Interactions*. Esta zona tiene además una doble interacción, ya que también refleja las actividades del *front-end* que se relacionan con las tareas internas del servicio o *back-end*.
- **Acciones internas (*Backstage Actions*):** en esta región se llevan a cabo las actividades del proveedor que se realizan al margen del cliente y que son necesarias para la provisión del servicio.
- **Acciones de soporte (*Support Processes*):** en esta zona se detallan las acciones que son llevadas a cabo por terceros actores, pero que igualmente resultan necesarias para la provisión del servicio.

Además, entre algunas de estas regiones también podemos encontrar las siguientes líneas divisorias:

- **Línea de interacción (*Line of Interaction*):** frontera que separa las acciones del consumidor del servicio de las acciones correspondientes al proveedor de este.
- **Línea de visibilidad (*Line of Visibility*):** división entre las acciones del *front-end* del servicio (las que interactúan directamente con el consumidor) y del *back-end* (acciones internas del servicio).
- **Línea de interacción interna (*Line of Internal Interaction*):** división entre las acciones internas del servicio y las acciones que son llevadas a cabo por terceros actores.

A continuación, la Tabla 2-3 ilustra los principales elementos gráficos de un modelo Service Blueprint:

Tabla 2-3. Elementos gráficos de la notación Service Blueprint.

<i>Elemento</i>	<i>Representación</i>	<i>Descripción</i>
Tablero		Lienzo que contiene cada una de las líneas divisorias y regiones sobre las que se posicionarán las acciones correspondientes.

Acción	<div style="border: 1px solid black; padding: 2px; display: inline-block;">Llamar por teléfono</div>	Elemento para representar acciones llevadas a cabo por el consumidor, proveedor o terceros actores. Posicionables en cada una de las regiones del tablero.
Conexión		Elemento para conectar y definir la secuencia entre las diferentes acciones.

La Figura 2-5 muestra un diagrama elaborado para el caso de estudio de *Deliveroo*. En concreto, el proceso que se representa es el envío de comida a domicilio, que comienza cuando el consumidor desea pedir comida y abre la aplicación correspondiente. A continuación, escribe su dirección y *Deliveroo* le envía la lista de restaurantes disponibles en su zona. El cliente selecciona entonces un restaurante, y *Deliveroo* responde informando sobre la comida disponible en ese restaurante. El usuario selecciona la comida, realiza el pago, y mediante acciones internas, *Deliveroo* procesa el pedido y encarga al restaurante la comida y al repartidor la recogida y entrega. Por último, el proceso finaliza cuando el cliente recoge el pedido en la dirección especificada.

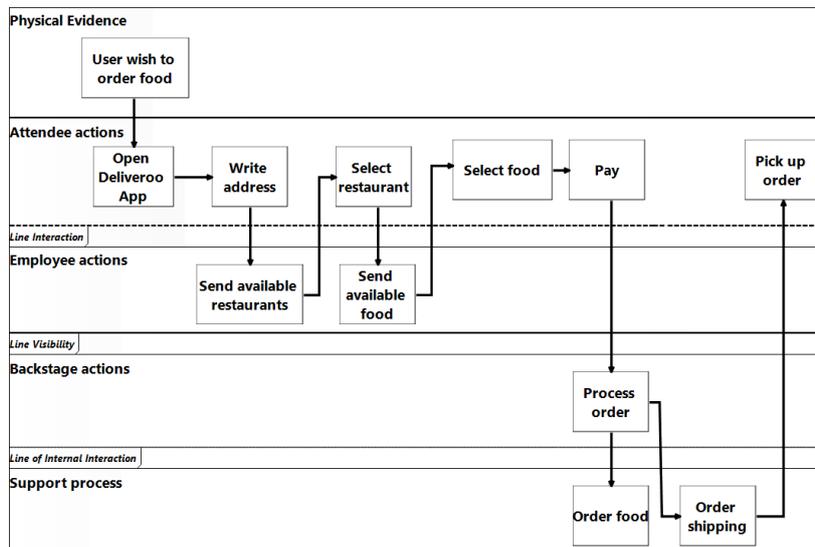


Figura 2-5. Diagrama Service Blueprint para el caso de estudio Deliveroo.

Entre las ventajas de Service Blueprint, nos encontramos con que es una notación que permite representar las interrelaciones que se llevan a cabo entre los actores cuando se produce un intercambio de valor, definiendo también un flujo

direccional entre las acciones producidas. Además, permite categorizar las diferentes actividades en función del grado de interacción con el cliente, gracias a la línea de visibilidad del tablero.

Respecto a las limitaciones, el modelo sólo representa al consumidor del servicio y al proveedor de este, dejando de lado terceros actores que podría ser interesante o incluso necesario contemplar. Tampoco permite representar nodos de decisión en los flujos de acciones, lo que dificulta la visualización de procesos basados en condiciones del contexto. Para paliar estas limitaciones, es posible utilizar modelos de proceso más detallados como los que se describen en las siguientes secciones.

En resumen, Service Blueprint es una notación que se centra en representar las interacciones entre el cliente y el proveedor de un servicio, dejando a un lado los intercambios de valor que se producen para enfocarse en los procesos internos que se dan durante la provisión del servicio. Es posible, no obstante, establecer relaciones entre este modelo y las notaciones para el modelado de negocio vistas en secciones anteriores. Por ejemplo, Service Blueprint puede ser utilizado para detallar las acciones llevadas a cabo en cada intercambio de valor de un modelo e³value.

2.1.2.4 Process Chain Network (PCN)

Process Chain Network (PCN) es una técnica de modelado propuesta por Scott E. Sampson para la representación de los procesos que se dan en la provisión de servicios [24], [78].

Las principales abstracciones del lenguaje son los **procesos** (secuencia de pasos que son llevados a cabo por entidades que actúan sobre recursos), las **cadenas de procesos** (secuencia de pasos con un objetivo claro, que se representan mediante flechas simbolizando el flujo de información o elementos materiales), las **entidades** (actores que participan en un proceso y tienen la capacidad de tomar decisiones que alteran el flujo del mismo), y los **dominios** (conjunto de pasos que son iniciados y/o controlados por una entidad determinada). Aunque en un diagrama PCN se pueden representar diferentes entidades, a título ilustrativo, la Figura 2-6 muestra el diagrama de un simple restaurante.

Para cada entidad representada se distinguen tres regiones bien diferenciadas. En las regiones que corresponden a los extremos laterales, encontramos las zonas de **interacción directa** (*Dir.*) con otras entidades, donde se representan las relaciones persona – persona, como un paciente que comenta sus

síntomas con un facultativo, o dos intervinientes inmersos en la negociación de un contrato.

Junto a estas se sitúan las zonas de **interacción subrogada** (*Sur.*), donde se representan los pasos del proceso que implican alguna actuación de la entidad sobre los recursos materiales de otra entidad. Por ejemplo, al realizar una compra por internet, la entidad usuario interactúa con la página web de la entidad vendedor, es decir, interactúa con un recurso material del vendedor, pero no con el vendedor en sí mismo.

Por último, en el centro de la parte del diagrama dedicada a cada entidad se representa la región de **proceso independiente** (*Ind.*), donde se representan las tareas realizadas por la entidad que no implican interacción con otras entidades o con los recursos de aquellas. Podría decirse que nos referimos a actividades internas, como realizar el inventario de una tienda, o diseñar nuevos menús en un restaurante.

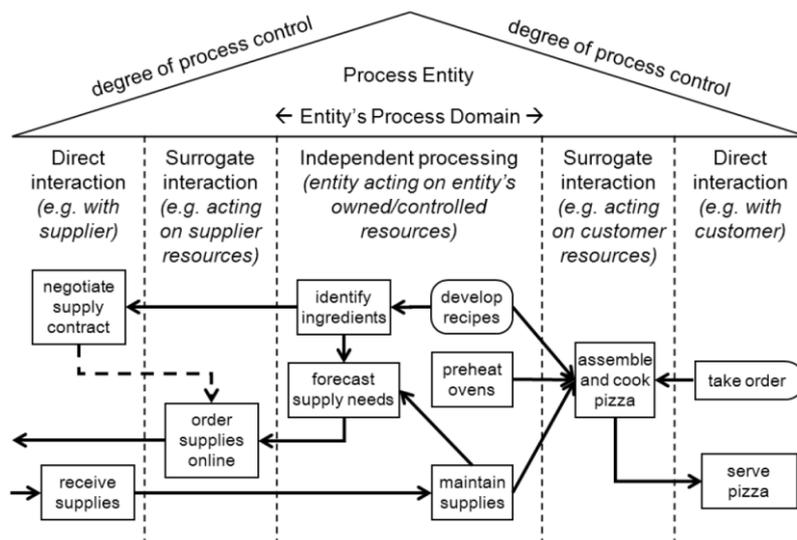


Figura 2-6. Diagrama PCN de un restaurante. Extraído de [24].

Además, en la parte superior del diagrama, un triángulo simboliza el grado de control de la entidad sobre las actividades de su dominio. La mayor altura del triángulo en la zona de procesamiento interno refleja el control casi total de la entidad sobre las actividades de esa región, control que disminuye al movernos hacia las regiones de interacción subrogada y directa.

Respecto a la representación gráfica de los diferentes conceptos, PCN es similar a otras notaciones como BPMN, que giran en torno a un elemento gráfico básico, en el que ligeras variaciones implican diferentes semánticas. La sintaxis concreta propone diferentes elementos gráficos que permiten modelar casi cualquier escenario. La Tabla 2-4 muestra los principales elementos gráficos de un modelo PCN:

Tabla 2-4. Elementos gráficos de la notación PCN.

<i>Elemento</i>	<i>Representación</i>	<i>Descripción</i>
Entidad		Figura que representa cada entidad participante en un proceso. En su interior se encuentran las cinco zonas de interacción.
Paso estándar		Acción llevada a cabo por una entidad. Posicionable en cualquiera de las cinco zonas de una entidad.
Acción y espera		Acción llevada a cabo por una entidad. Una vez terminada, la entidad permanece a la espera.
Acción externa		Acción llevada a cabo por una entidad externa no representada en el diagrama.
Paso innovador		Acción llevada a cabo por una entidad que supone una innovación respecto a anteriores modelos.
Paso de espera		Acción de espera llevada a cabo por una entidad.
Paso de decisión		Nodo de decisión con la capacidad de alterar el flujo del proceso.
Referencia		Enlace a otra entidad. Útil cuando en diagramas de gran tamaño es necesario establecer una dependencia con otra entidad alejada físicamente.
Dependencia estándar		Transición entre dos acciones.
Dependencia con retardo		Transición entre dos acciones con retardo esperado: la transición no se completa de manera inmediata.

Ganancias / Pérdidas monetarias		Etiquetas situadas junto a las acciones para indicar que en ese paso se producen ganancias o pérdidas monetarias para la entidad.
Ganancias / Pérdidas no monetarias		Etiquetas situadas junto a las acciones para indicar que en ese paso se producen ganancias o pérdidas no monetarias para la entidad (p.ej. satisfacción).
Anotación textual		Etiqueta textual situada junto a las acciones para indicar anotaciones sobre la misma.

Una vez vistas las características del modelo PCN, la Figura 2-7 muestra el diagrama elaborado para el proceso de envío de comida a domicilio del caso de estudio *Deliveroo*. Como puede observarse, las cuatro entidades que se ven involucradas en el proceso son el cliente, *Deliveroo*, el restaurante y el repartidor. El proceso comienza cuando el cliente abre la aplicación de *Deliveroo*. Dado que las siguientes acciones las realiza a través de la aplicación (recurso de *Deliveroo*), se posicionan en la zona de interacción subrogada. Por ejemplo, puede apreciarse el descontento del cliente si *Deliveroo* le comunica que su zona postal está fuera del alcance del servicio. Además, se produce una ganancia monetaria para *Deliveroo* cuando el cliente realiza el pago del pedido. También es posible comprobar que la recogida del pedido por parte del repartidor en el restaurante, se define en la zona de interacción directa.

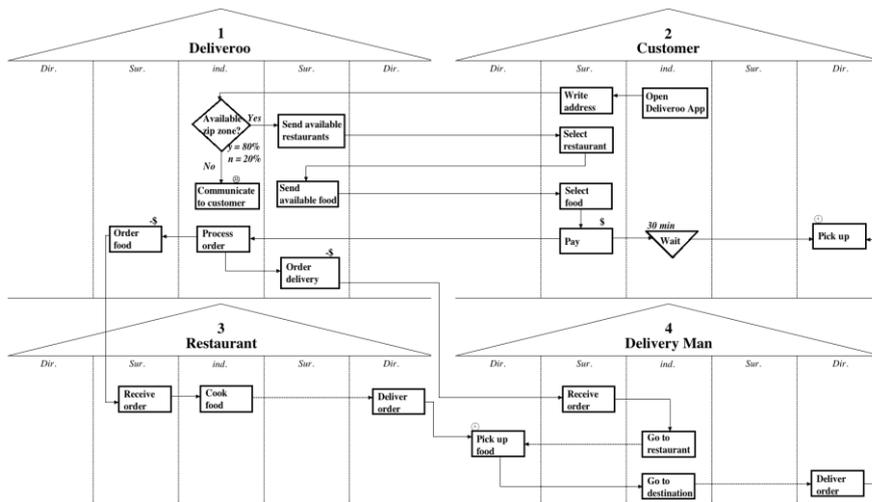


Figura 2-7. Diagrama PCN para el caso de estudio Deliveroo.

Entre las ventajas de la notación PCN, destaca el hecho de ser una técnica que permite representar a todos los actores participantes en un proceso, así como el gran nivel de detalle con el que es capaz de definirlo, incluyendo elementos como las ganancias o pérdidas monetarias y no monetarias, que no recogen otras técnicas como BPMN. Además, permite clasificar las acciones en función del grado de control de la entidad sobre las mismas.

En cuanto a las desventajas, probablemente el mayor inconveniente de esta técnica sea la disposición gráfica de las entidades, que en diagramas de cierto tamaño puede perjudicar su usabilidad. Además, si bien permite definir eventos de espera temporal, no tiene una riqueza léxica tan grande como BPMN respecto a otros tipos de eventos.

Finalmente, al igual que con las notaciones anteriores, es posible definir correspondencias entre PCN y otras técnicas como Service Blueprint, e³value y Canvas. Por ejemplo, si Service Blueprint permitía representar de manera sencilla cómo se llevan a cabo los intercambios de valor especificados en un modelo e³value, PCN permite ahondar en los detalles de dichos intercambios, mostrando además a todos los actores que en ellos participan.

2.1.2.5 Business Process Model and Notation (BPMN)

Business Process Model and Notation (BPMN) es una notación gráfica para el modelado de procesos de negocio propuesta en 2004 por la *Business Process Modeling Initiative*, y adoptada como estándar del OMG en 2006¹². Originalmente se tomaron los diagramas de actividad UML como inspiración, y se trató de ofrecer una notación fácilmente entendible por todas las partes involucradas en una organización, desde los analistas de negocio, hasta los desarrolladores encargados de implementar las soluciones diseñadas [23].

El estándar ha sufrido constantes revisiones a lo largo de su historia, siendo publicada la versión actual en 2013 (v2.0.2). Aunque dicha versión incluye más de 85 elementos, lo cierto es que algunos estudios aseguran que durante el uso habitual de la notación, se utiliza un subconjunto de elementos mucho más reducido [79]. De este modo, la Tabla 2-5 muestra los principales elementos de BPMN, mientras que la especificación completa puede ser consultada en la documentación oficial [80].

¹² <https://www.bpmn.org/>

Tabla 2-5. Elementos gráficos básicos de la notación BPMN.

<i>Elemento</i>	<i>Representación</i>	<i>Descripción</i>
Pool		Representa a un actor o participante del proceso. Cada <i>pool</i> puede contener uno o más <i>lanes</i> .
Lane		Representa una actividad llevada a cabo por un <i>pool</i> .
Evento inicial		Elemento disparador del proceso. El icono de su interior informa del tipo de evento capturado (mensaje, tiempo, etc.) que dispara el proceso.
Evento intermedio		Representar que algo ha ocurrido durante el proceso. El icono de su interior informa del tipo de evento.
Evento final		Indica el final del proceso. El icono de su interior informa del tipo de evento lanzado.
Tarea		Unidad de trabajo no divisible llevada a cabo por un pool en un <i>lane</i> determinado. Puede incluir un icono en su interior, indicando el tipo de tarea.
Subproceso		Actividad divisible en varias tareas o subprocessos. Permite encapsular detalles del proceso.
Compuerta		Representa una alteración del flujo del proceso. El icono de su interior representa el tipo de compuerta (AND, OR, XOR, etc.).
Secuencia		Conectando varias tareas o subprocessos, representa el orden en el que estas se llevan a cabo.
Flujo de mensaje		Representa un flujo comunicativo o de interacción entre diferentes <i>pools</i> .
Asociación		Permite conectar las tareas, subprocessos, compuertas o eventos con artefactos, que son elementos para aportar información adicional al modelo.

Una vez vistos los elementos básicos de la notación BPMN, la Figura 2-8 muestra un extracto de un modelo para representar el proceso de entrega de comida a domicilio de *Deliveroo*. Como puede observarse, los cuatro actores participantes en el proceso (cliente, *Deliveroo*, repartidor y restaurante) se ven representados por sus correspondientes *pools*. Aunque en este caso no aparecen *lanes* por tratarse de una sola actividad en cada *pool*, puede observarse el paso de mensajes que se produce entre los diferentes participantes, lo que da lugar a diferentes eventos intermedios de recepción de mensajes. Por otro lado, puede apreciarse la compuerta XOR que bifurca el flujo de tareas en función de si el código postal introducido por el usuario está cubierto, o no, por el servicio de *Deliveroo*.

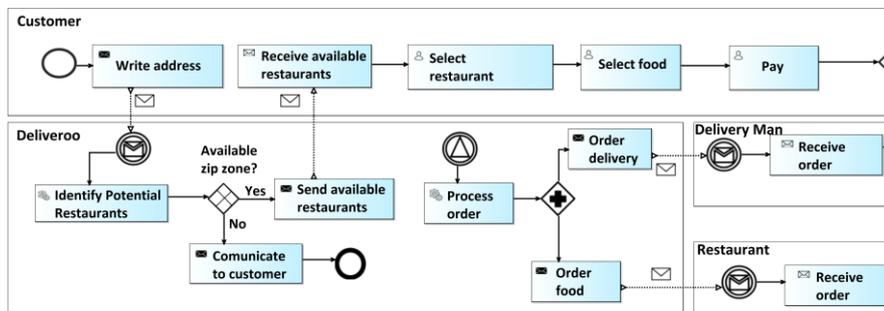


Figura 2-8. Extracto de diagrama BPMN para el caso de estudio Deliveroo.

Entre las ventajas de BPMN, destaca el hecho de ser una notación que soporta diferentes niveles de detalle, siendo comprensible por diferentes estamentos de una misma organización, si bien su uso está especialmente extendido en los departamentos de tecnologías de la información. Permite representar además a todos los actores participantes en un proceso, detallando incluso las diferentes actividades que llevan a cabo en forma de distintos *lanes*. También permite especificar una gran cantidad de tipos de tareas y eventos.

Como contrapartida, la riqueza semántica que hace tan expresiva a la notación, puede hacer que su curva de aprendizaje sea mayor que la de otras técnicas de modelado de procesos [79].

Finalmente, una vez más es posible encontrar correspondencias entre la notación BPMN y el resto de técnicas vistas en esta tesis doctoral. Por ejemplo, si PCN detallaba en profundidad los procesos representados en Service Blueprint (que a su vez explotaban los intercambios de valor del modelo e^3 value), con BPMN podemos acercar el nivel de detalle de la representación de esos procesos a los profesionales de las tecnologías de la información. Todas estas

correspondencias se estudian con mayor detalle en la sección 3.2.3 de la presente tesis doctoral.

2.2 Herramientas para el modelado de negocio y proceso

Una vez visto en la sección anterior el modelo del doble diamante para el diseño de servicios, así como la descripción de diversas notaciones para el modelado de negocio y proceso que encajan en dicha metodología, la presente sección pretende conocer el estado del arte en lo que se refiere a herramientas tecnológicas que den soporte al modelado de negocio y proceso.

Para ello y a modo de introducción, como viene expresando esta memoria, en la actualidad los modelos de negocio se han convertido en una herramienta fundamental para el diseño de nuevos servicios y propuestas de valor. Estos modelos permiten representar de forma gráfica los productos y servicios que ofrece una empresa, definiendo las acciones y pasos necesarios para conseguir sus objetivos [81].

Como ya se ha visto, existen diferentes técnicas para el modelado de negocio y proceso, algunas utilizadas tradicionalmente y otras más recientes y novedosas. Además, cada una permite visualizar una propuesta de valor desde perspectivas diferentes. Por ejemplo, si con un modelo Canvas podemos representar el negocio desde un punto de vista global, donde todos los elementos se reflejan como un todo [82], con un modelo Service Blueprint podemos centrarnos en una operación de servicio concreta, detallando las interacciones que se producen entre el cliente y el proveedor del servicio [83]. En esta misma línea, podemos encontrar otros modelos de negocio y proceso que es posible emplear según el punto de vista y el nivel de detalle con el que se quiera representar una propuesta de valor, como por ejemplo e³value, Process Chain Network o BPMN [23], [24], [84].

La representación de estos modelos de negocio y proceso puede realizarse utilizando herramientas software [85]. Actualmente encontramos algunos programas específicos que nos permiten trabajar con determinados tipos de notaciones. Por ejemplo, con la herramienta *Real Time Board* podemos diseñar diagramas Service Blueprint, con *Canvanaizer* diagramas Canvas, o con *e³value Editor* diagramas e³value. Sin embargo, estas soluciones software, aparte de escasas, en muchos casos tan solo nos permiten trabajar con diagramas de tipos específicos de manera aislada, siendo más difícil encontrar una herramienta que permita diseñar diferentes modelos de negocio y proceso de una manera integrada.

También podemos encontrar diagramadores genéricos, 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 [27].

Con todo ello, dado que los modelos de negocio y proceso se han convertido en activos de valor para la mayoría de empresas de servicios, conviene disponer de herramientas informáticas que permitan la gestión de los mismos, para lo cual resulta imprescindible que dichas herramientas almacenen la información recogida en los modelos con los formatos adecuados para poder procesarlos más adelante.

En este punto, también resulta interesante la posibilidad de establecer correspondencias entre distintas notaciones [86]. Como ya se ha mencionado, las diferentes técnicas para el modelado de negocio y proceso se centran en representar el negocio desde distintos puntos de vista. Es por ello por lo que pueden tener ciertos elementos en común que nos permitan formar un modelo en una determinada notación, a partir de uno o varios modelos elaborados con técnicas diferentes.

Por lo tanto, el objetivo de esta revisión del estado del arte, consiste en el análisis de la literatura en lo que respecta a soporte técnico para el modelado de negocio y proceso, que permita trabajar de manera integrada con diferentes notaciones para el diseño de servicios, y que pueda además llevar a cabo tareas complejas como validaciones o transformaciones automáticas o semiautomáticas de modelos.

2.2.1 Método

Este estudio sobre estado del arte se ha llevado a cabo en forma de revisión sistemática de la literatura (SLR) siguiendo las especificaciones de Biolchini [56] y Bárbara Kitchenham [57] descritas en la sección 1.4.2. De acuerdo con esta visión, una revisión sistemática se compone de varias actividades agrupadas en las fases de planificación, ejecución, análisis de resultados y empaquetado.

Respecto a la fase de planificación, la sección 2.2.2 presenta las cuestiones de investigación a las que se pretende dar respuesta por medio de esta SLR. A continuación, la sección 2.2.3 especifica las fuentes de información y las cadenas de términos empleadas para llevar a cabo las consultas, y la sección 2.2.4 define los criterios de inclusión y exclusión a tener en cuenta durante la selección de estudios primarios.

En cuanto a la fase de ejecución, la sección 2.2.5 describe el proceso de búsqueda y filtrado de estudios primarios, mientras que la sección 2.2.6 describe cómo se llevó a cabo la extracción de información a partir de dichos trabajos.

Por último, en lo que respecta al análisis de resultados, la sección 2.2.7 realiza un análisis de las herramientas de modelado encontradas durante la SLR y la sección 2.2.8 concluye con una reflexión general sobre algunos trabajos relacionados.

2.2.2 *Cuestiones de investigación*

El objetivo principal de esta revisión sistemática, es el de conocer el estado del arte en lo que respecta a herramientas para el modelado de negocio y proceso que permitan trabajar de manera integrada con distintas notaciones para el diseño de servicios, y que además soporten tareas complejas como validaciones o transformaciones automáticas o semiautomáticas de modelos. Para lograr este objetivo, se han definido las siguientes cuestiones de investigación:

- **RQ1:** ¿Qué herramientas software hay disponibles para el modelado de negocio y proceso?
- **RQ2:** ¿Permiten estas herramientas trabajar de manera integrada con diferentes notaciones para el modelado de negocio y proceso?
- **RQ3:** ¿Almacenan estas herramientas los modelos con formatos adecuados para permitir su posprocesamiento y tareas complejas como validaciones o transformaciones?
- **RQ4:** ¿En qué grado cumplen los modelos generados mediante estas herramientas con las especificaciones de las notaciones correspondientes?
- **RQ5:** ¿Existen herramientas de modelado para cada una de las técnicas seleccionadas?

Las preguntas RQ1, RQ2 y RQ3, tienen el objetivo de conocer las herramientas informáticas que actualmente hay disponibles para el modelado de negocio y proceso, así como si las que ya existen son capaces de trabajar de manera conjunta e integrada con diferentes notaciones para el modelado de negocio y proceso, y si utilizan los formatos adecuados para su gestión, posprocesamiento y ejecución de tareas complejas como validaciones o transformaciones de modelos.

En cuanto a RQ4, esta pregunta trata de evaluar en cierto modo la calidad de las herramientas disponibles. Para ello, se compararían las características de los diagramas generados por las herramientas, con las especificaciones oficiales de las técnicas de modelado correspondientes.

Por último, la pregunta RQ5, busca analizar si existen herramientas software que soporten todas las notaciones seleccionadas para el modelado de negocio y proceso (Canvas, e³value, Service Blueprint, PCN, BPMN), de cara a evaluar el impacto que pudiera tener esta tesis doctoral en caso de no existir herramientas para algunas de las notaciones seleccionadas.

2.2.3 Fuentes de información y cadenas de búsqueda

Como se ha descrito anteriormente, de acuerdo a [56] y [57], en la fase inicial de una SLR se deben seleccionar las fuentes de información en las que se llevará a cabo la búsqueda de estudios, así como la definición de un conjunto de cadenas de búsqueda adaptadas a cada fuente de información.

En este sentido, la búsqueda de estudios primarios se ha realizado en una selección de librerías digitales. La Tabla 2-6 muestra el listado de librerías utilizadas con su nombre, acrónimo y dirección web:

Tabla 2-6. Librerías digitales para la SLR.

Nombre	Acrónimo	Dirección web
ACM Digital Library	ACM	http://dl.acm.org
IEEEExplore	IEEE	http://ieeexplore.ieee.org
Science Direct	SD	http://www.sciencedirect.com
Scopus	SCOPUS	http://www.scopus.com
SpringerLink	SL	http://link.springer.com/

Para la elaboración de las cadenas de búsqueda se han utilizado términos relacionados con el objetivo de esta revisión sistemática. En concreto, podemos dividir la consulta en tres bloques, donde el primero de ellos contiene términos relacionados con modelos de negocio y proceso, el segundo técnicas y notaciones de modelado, y el tercero palabras relacionadas con el soporte tecnológico. Los términos contenidos en cada bloque están relacionados con el operador lógico OR, mientras que el conjunto de bloques se encuentra unido con el operador lógico

AND. La Tabla 2-7 muestra un resumen de los bloques que conforman las cadenas de búsqueda:

Tabla 2-7. Bloques de las cadenas de búsqueda para la SLR.

business model	AND	ontology	AND	tool
business analysis		notation		tooling
business process model		diagram		technological support
business management		design view		technical support
business design		blueprint		diagrammer
		pcn		application
		process chain network		environment
		bpmn		ide
		canvas		
		e ³ value		

Dado que cada librería utiliza una sintaxis distinta del resto, se han tenido que construir cadenas de búsqueda acordes a las características de cada una. Además, en algunos buscadores, se ha debido reducir la consulta por razones técnicas. Así, la restricción en el número de términos de IEEE, nos ha obligado a desechar algunas palabras, mientras que en SpringerLink se ha modificado ligeramente la consulta dado que, con la cadena original, se obtenía un número excesivamente elevado de resultados no relevantes. La Tabla 2-8 muestra las cadenas de búsqueda definidas para cada una de las librerías empleadas:

Tabla 2-8. Cadenas de búsqueda para la SLR.

Librería	Cadena de búsqueda	Objetivo
ACM	("business model" OR "business analysis" OR "business process model" OR "business management" OR "business design") AND ("ontology" OR "notation" OR "diagram" OR "design view" OR "blueprint" OR "pcn" OR "process chain network" OR "bpmn" OR "canvas" OR "e ³ value") AND ("tool" OR "tooling" OR "technological support" OR "technical support" OR "diagrammer" OR "application" OR "environment" OR "ide")	Todo

IEEE	("business model" OR "business management" OR "business process model" OR "business design") AND (ontology OR notation OR diagram OR blueprint OR pcn OR bpmn OR canvas OR e ³ value) AND (tool OR "technological support" OR diagrammer OR application OR environment)	Todo
SD	(Title-Abstr-Key({business model}) OR Title-Abstr-Key({business analysis}) OR Title-Abstr-Key({business process model}) OR Title-Abstr-Key({business management}) OR Title-Abstr-Key({business design})) AND (Title-Abstr-Key({ontology}) OR Title-Abstr-Key({notation}) OR Title-Abstr-Key({diagram}) OR Title-Abstr-Key({design view}) OR Title-Abstr-Key({blueprint}) OR Title-Abstr-Key({pcn}) OR Title-Abstr-Key({process chain network}) OR Title-Abstr-Key({bpmn}) OR Title-Abstr-Key({canvas}) OR Title-Abstr-Key({e ³ value})) AND (Title-Abstr-Key({tool}) OR Title-Abstr-Key({tooling}) OR Title-Abstr-Key({technological support}) OR Title-Abstr-Key({technical support}) OR Title-Abstr-Key({diagrammer}) OR Title-Abstr-Key({application}) OR Title-Abstr-Key({environment}) OR Title-Abstr-Key({ide}))	Título, abstract y palabras clave
SCOPUS	(TITLE-ABS-KEY({business model}) OR TITLE-ABSKEY({business analysis}) OR TITLE-ABSKEY({business process model}) OR TITLE-ABSKEY({business management}) OR TITLE-ABSKEY({business design})) AND (TITLE-ABSKEY({ontology}) OR TITLE-ABSKEY({notation}) OR TITLE-ABSKEY({diagram}) OR TITLE-ABSKEY({design view}) OR TITLE-ABSKEY({blueprint}) OR TITLE-ABSKEY({pcn}) OR TITLE-ABSKEY({process chain network}) OR TITLE-ABSKEY({bpmn}) OR TITLE-ABSKEY({canvas}) OR TITLE-ABSKEY({e ³ value})) AND (TITLE-ABSKEY({tool}) OR TITLE-ABSKEY({tooling}) OR TITLE-ABSKEY({technological support}) OR TITLE-ABSKEY({technical support}) OR TITLE-ABSKEY({diagrammer}) OR TITLE-ABSKEY({application}) OR TITLE-ABSKEY({environment}) OR TITLE-ABSKEY({ide}))	Título, abstract y palabras clave
SL	("business design") AND ("ontology" OR "notation" OR "diagram" OR "design view" OR "blueprint" OR "pcn" OR "process chain network" OR "bpmn" OR "canvas" OR "e ³ value") AND ("tool" OR "tooling" OR "technological support" OR "technical support" OR "diagrammer" OR "ide")	Todo

2.2.4 Criterios de inclusión y exclusión

Aunque la cadena de búsqueda ha sido diseñada para tratar de obtener resultados relacionados con el objetivo de esta revisión, es preciso considerar que, en cualquier SLR, es posible que algunos de los resultados extraídos no estén relacionados completamente con las cuestiones de investigación definidas. Por este motivo, según [57], resulta necesario plantear una serie de criterios de inclusión y exclusión a partir de las preguntas de investigación de la SLR, con el objetivo de poder descartar trabajos no relacionados o que no cumplan con un conjunto de condiciones.

Según lo anterior, los criterios de inclusión para la selección de estudios relacionados se han basado, en el caso de esta SLR, en la lectura y análisis del título, resumen y palabras clave de cada trabajo recuperado en la fase de búsqueda. De este modo, los trabajos relevantes que han sido incluidos en esta SLR habrían de cumplir con el siguiente criterio de inclusión:

- *Artículos cuyo título, resumen y palabras clave dejen claro que el objeto de estudio versa sobre herramientas software para el modelado de negocio y/o procesos de negocio.*

En esta SLR se llevó a cabo, por tanto, una lectura y análisis preliminar del título, resumen y palabras clave de cada resultado obtenido, con el objetivo de determinar si su contenido estaba relacionado con herramientas software para el modelado de negocio y/o procesos de negocio. Aquellos estudios que cumplían con este criterio de inclusión, fueron seleccionados como trabajos relevantes de la SLR. Además, este proceso se llevó a cabo para todos los trabajos devueltos por las cadenas de búsqueda en cada librería digital.

Una vez filtrados los trabajos relevantes, se desecharon los artículos duplicados con el objetivo de mantener una única copia aunque se hubiesen encontrado en diferentes librerías digitales. A continuación, se llevó a cabo una lectura detallada de cada trabajo relevante para conocer si dichos artículos contenían información directamente relacionada con el objetivo de esta SLR. Para ello, se definieron los siguientes criterios de exclusión:

- *Trabajos cuyo objetivo principal sea el de clasificar otras revisiones del estado del arte.*
- *Trabajos cuyo objetivo principal no esté directamente relacionado con el objetivo de esta SLR, aunque mencionen conceptos afines en el título, resumen o palabras clave.*

Mediante esta última fase fue posible desechar los trabajos relevantes correspondientes del conjunto de estudios primarios, esto es, los trabajos que cumplen con los criterios de inclusión y no así con los criterios de exclusión.

2.2.5 Búsqueda y selección de estudios primarios

Una vez finalizados los preparativos de la SLR en la fase de planificación, la siguiente etapa es la de ejecución. En concreto, el primer paso de esta etapa consiste en la búsqueda y selección de estudios primarios mediante la introducción de las cadenas de búsqueda de la Tabla 2-8 en las distintas librerías digitales de la Tabla 2-6. Tras esto, se obtuvo un total de 1195 resultados cuya distribución puede observarse en las dos primeras columnas de la Tabla 2-9.

Tras lanzar las búsquedas en las diferentes librerías, se realizó una primera criba en función de los criterios de inclusión. De este modo, se filtraron 129 estudios relevantes que se aproximaban al área del objetivo de esta revisión. La distribución de este dato puede verse en detalle en la tercera columna de la Tabla 2-9.

Posteriormente, se descartaron los artículos duplicados y se aplicaron los criterios de exclusión sobre los estudios relevantes. De este modo, se seleccionaron un total de 50 artículos primarios cuya distribución puede observarse en la cuarta columna de la Tabla 2-9.

La quinta columna de la Tabla 2-9 muestra, para cada buscador, el porcentaje de estudios primarios obtenidos sobre el total de resultados encontrados. Observando estos datos, podemos apreciar que el buscador Science Direct, es el que aporta un mayor porcentaje de estudios primarios (12.19%), siendo además el que menos resultados en bruto devuelve tras la búsqueda inicial. Por este motivo es posible afirmar que, en el caso de esta SLR, Science Direct ha resultado ser el buscador más eficiente de cara a los objetivos perseguidos.

Finalmente, la sexta y última columna de la Tabla 2-9, muestra el porcentaje de estudios primarios que aporta cada buscador al conjunto del total. Analizando los datos de dicha columna, observamos que la librería que más artículos primarios aporta es SCOPUS (42%), si bien teniendo en cuenta los datos del resto de columnas, se trata de la librería que más resultados en bruto devolvió tras la búsqueda inicial.

Tabla 2-9. Distribución de estudios clasificados por fuente de información.

Librería	Resultados	Relevantes	Primarios	% Primarios	% del total de primarios
ACM	202	11	6	2,97	12
IEEE	180	27	8	4,44	16
SD	41	8	5	12,19	10
SCOPUS	469	55	21	4,47	42
SL	303	28	10	3,30	20
Total	1195	129	50	4,18	100

2.2.6 Extracción de datos

Una vez finalizada la búsqueda y selección de estudios primarios, la siguiente etapa de la fase de ejecución, según [56] y [57], consiste en la extracción de datos a partir de dichos estudios primarios. De este modo, tal y como muestra el ejemplo de la Tabla 2-10, se extrajeron una serie de datos a partir de cada trabajo seleccionado.

Tabla 2-10. Ejemplo de extracción de datos de cada estudio primario.

Title	How Software Can Support Innovating Business Models: A Taxonomy of Functions of Business Model Development Tools
Author	Daniel Szopinski, Thorsten Schoormann, Thomas John, R. Knackstedt, Dennis Kundisch
Abstract	The interest in business model innovation has risen rapidly in recent years, and software tools for business model development hold great promise for supporting business model innovation. Nonetheless, virtually no design-relevant knowledge exists concerning the functions that such tools should possess. Therefore, we develop a comprehensive taxonomy that identifies characteristic functions of software-based business model development tools. For developing the taxonomy, we draw on prior research on business model innovation, process modeling, and creativity support systems, and we analyze software tools for business model development that have been proposed in practice. The resulting taxonomy can support practitioners in their tool (re-)design and investment decisions, and for researchers can serve as a preliminary step towards more advanced theories for software tools for business model development.
Date	August, 2017

Journal/Conf.	23 rd Americas Conference on Information Systems (AMCIS 2017)
Relevant info.	Description of some features of different tools for business modelling such as Canvanizer, Strategyzer or Realtime Board.

A partir de los datos recopilados, ya fue posible proporcionar una respuesta preliminar a la quinta cuestión de investigación (RQ5) de esta SLR: *¿Existen herramientas de modelado para cada una de las técnicas seleccionadas?* A este respecto, si bien se encontraron diferentes herramientas de modelado para las notaciones Canvas, e³value, Service Blueprint y BPMN, lo cierto es que no se descubrió ninguna para la notación PCN, debiendo recurrir a diagramadores genéricos como única alternativa para la edición de este tipo de diagramas. Del mismo modo, fue posible apreciar también que la notación para la que actualmente existen más herramientas de modelado, es BPMN.

2.2.7 *Análisis de las propuestas*

Una vez finalizada la extracción de datos de la fase de ejecución, la siguiente etapa consiste en el análisis de las propuestas encontradas correspondiente a la fase de análisis de resultados. Cabe mencionar, sin embargo, que esta sería una tarea sin fin debido a que continuamente surgen nuevas herramientas para el modelado de negocio y/o proceso, por lo que este análisis se ciñe al soporte tecnológico encontrado en el momento de realizar la SLR.

A continuación se describen los criterios utilizados para el análisis de las propuestas, que han sido obtenidos directamente a partir de las cuestiones de investigación descritas en la sección 2.2.2:

- **Notaciones para el modelado de negocio y/o proceso soportadas:** con este criterio se busca conocer el ámbito general de cada herramienta en el contexto del diseño de servicios. Además, este criterio también permite dar respuesta a la quinta cuestión de investigación (RQ5): *¿Existen herramientas de modelado para cada una de las técnicas seleccionadas?*
- **Posibilidad de trabajar de manera integrada con modelos de negocio y proceso:** este criterio pretende dar respuesta a la segunda cuestión de investigación (RQ2): *¿Permiten estas herramientas trabajar de manera integrada con diferentes notaciones para el modelado de negocio y proceso?* El objetivo de este criterio es doble: por un lado, busca analizar si las herramientas encontradas posibilitan trabajar con un conjunto de notaciones diferentes para el modelado de negocio y proceso. Pero además,

persigue analizar si dichas herramientas permiten trabajar de manera integrada con estas notaciones en el contexto de un proyecto de diseño de servicios, por ejemplo mediante el empleo de relaciones o trazas entre los distintos modelos.

- **Formato de modelos adecuado para permitir su posprocesamiento:** este criterio busca dar respuesta a la primera parte de la tercera cuestión de investigación (RQ3): *¿Almacenan estas herramientas los modelos con formatos adecuados para permitir su posprocesamiento y tareas complejas como validaciones o transformaciones?* De este modo, se busca analizar si las herramientas encontradas almacenan los modelos diseñados con formatos adecuados para el posprocesamiento de la información contenida, o si por el contrario se limitan a guardarlos como una simple imagen.
- **Soporte para operaciones complejas como validaciones o transformaciones de modelos:** este criterio pretende dar respuesta a la segunda parte de la tercera cuestión de investigación (RQ3): *¿Almacenan estas herramientas los modelos con formatos adecuados para permitir su posprocesamiento y tareas complejas como validaciones o transformaciones?* Por lo tanto, en este caso se analizará si las herramientas encontradas permiten realizar tareas como validaciones, simulaciones o transformaciones de modelos, o si por el contrario tan solo permiten la edición de diagramas visuales.
- **Cumplimiento con la especificación de las notaciones soportadas:** este último criterio busca responder a la cuarta pregunta de investigación (RQ4): *¿En qué grado cumplen los modelos generados mediante estas herramientas con las especificaciones de las notaciones correspondientes?* Dado que existen herramientas de modelado con elementos genéricos que permiten realizar modelos para diversas notaciones, se hace necesario comprobar si los modelos construidos con dichas herramientas cumplen con las características de las notaciones utilizadas.

Una vez descritos los criterios empleados para el análisis de las propuestas, la Tabla 2-11 muestra el conjunto de herramientas sobre las que se han aplicado:

Tabla 2-11. Herramientas analizadas.

<i>Herramienta</i>	<i>Sitio Web</i>
<i>Bizagi</i>	https://www.bizagi.com/es

<i>BonitaSoft</i>	https://es.bonitasoft.com/
<i>bpmn.io</i>	https://bpmn.io/
<i>Canvanizer</i>	https://canvanizer.com/
<i>Draw.io</i>	https://app.diagrams.net/
<i>e³value editor</i>	https://research.e3value.com/tools/
<i>Gliffy</i>	https://www.gliffy.com/
<i>Lucidchart</i>	https://www.lucidchart.com/pages/es
<i>MS Visio</i>	https://www.microsoft.com/es-es/microsoft-365/visio/
<i>Miro</i>	https://miro.com/es/
<i>Signavio</i>	https://www.signavio.com/
<i>Strategyzer</i>	https://www.strategyzer.com/

Respecto al primer criterio de este análisis (notaciones soportadas), lo primero que llama la atención es la gran cantidad de soluciones disponibles específicamente para la notación BPMN. Véanse, por ejemplo, las herramientas *Bizagi* [87], *BonitaSoft* [88], *bpmn.io* [89], o *Signavio* [90]. También destaca el hecho de que ninguna de las herramientas analizadas dé soporte específico a la notación PCN, si bien podrían realizarse diagramas de este tipo con herramientas genéricas para el modelado de procesos como *Draw.io* [91], *Gliffy* [92], *Lucidchart* [93] o *MS Visio* [94]. En cualquier caso, al no cumplir específicamente con los requisitos de PCN, dichas soluciones no pueden asegurar que los modelos construidos cumplan con la especificación formal de dicha notación. Por otro lado, se ha encontrado una herramienta que da soporte específico a la notación Canvas (*Strategyzer* [95]), y otras dos que, además de Canvas, también ofrecen soporte a Service Blueprint (*Canvanizer* [95] y *Miro* – antigua *Realtime Board* [96]). Por último, como soporte específico para la notación e³value, tan solo se ha encontrado la herramienta *e³value editor* [22].

Como conclusión tras analizar las distintas propuestas en base al primer criterio, es posible dar una respuesta a la quinta pregunta de investigación (RQ5). De este modo, en el momento de realización de esta SLR, existen herramientas que dan soporte tecnológico específico a todas las notaciones seleccionadas, con la excepción de PCN.

Respecto al segundo criterio para el análisis de las propuestas (gestión integrada de modelos de negocio y proceso), como se ha visto, existen diversas

herramientas que soportan más de una notación: aquellas que permiten modelar diagramas Canvas y Service Blueprint (*Canvanizer* y *Miro*) y aquellas que sirven como diagramadores genéricos de procesos, que pueden servir para modelar diagramas BPMN o UML entre otros (*Draw.io*, *Gliffy*, *Lucidchart*, *MS Visio*). Sin embargo, según se ha podido comprobar, a pesar de soportar distintas notaciones, estas herramientas tratan de manera aislada a cada uno de los modelos involucrados en un determinado proyecto, sin establecer ningún tipo de relación o trazabilidad entre ellos. Por este motivo, y como respuesta a la segunda cuestión de investigación, es posible concluir que no se ha encontrado ninguna herramienta que permita trabajar de manera integrada con diferentes notaciones para el modelado de negocio y proceso.

En cuanto al tercer criterio del análisis (formato adecuado para el posprocesamiento), todas las herramientas analizadas permiten almacenar los modelos construidos con los formatos adecuados para llevar a cabo la edición posterior de los mismos, siendo posible también exportar los diagramas con distintos formatos de imagen. Además, algunas herramientas poseen los formatos adecuados para llevar a cabo un trabajo colaborativo sobre los modelos construidos, como por ejemplo *Canvanizer*, *Lucidchart*, *Miro*, *Signavio*, *Strategyzer* o *Gliffy*, permitiendo también este último exportar los modelos en formatos compatibles con otras herramientas como *MS Visio*.

Respecto al cuarto criterio del análisis de las propuestas (operaciones complejas sobre los modelos), se han identificado una serie de herramientas que permiten llevar a cabo simulaciones o validaciones de modelos. Por ejemplo, con *Bizagi*, *BonitaSoft* o *Lucidchart*, es posible llevar a cabo simulaciones de procesos BPMN definiendo la cantidad de instancias del proceso o las probabilidades de bifurcación en cada compuerta de decisión. *MS Visio* permite además la validación de modelos con el fin de asegurar que los modelos construidos cumplen estrictamente con la especificación de la notación correspondiente. Por último, *e³value editor* posee un sistema de simulación que permite estimar las ganancias de los actores participantes en diferentes intercambios de valor definiendo el número de intercambios y el valor intercambiado en cada uno. De este modo, como respuesta a la tercera cuestión de investigación (RQ3), es posible concluir que sí existen herramientas de modelado que permiten llevar a cabo tareas complejas como validaciones o simulaciones con los modelos de negocio y proceso construidos, si bien no se ha encontrado ninguna que permita realizar transformaciones entre diferentes notaciones, esto es, generar modelos en una determinada notación a partir de otro modelo elaborado con una técnica diferente.

Finalmente, con respecto al quinto criterio del análisis de propuestas (cumplimiento con la especificación de las notaciones soportadas), se ha comprobado que la mayoría de herramientas afirman seguir fielmente los principios de las notaciones a las que dan soporte, no siendo así en tres casos de diagramadores genéricos para procesos (*Draw.io*, *Gliffy* y *Lucidchart*). Como casos destacados del análisis de este criterio, merece la pena comentar que tanto *Strategyzer* como *e³value editor* son las herramientas desarrolladas por los creadores de las notaciones correspondientes (Canvas y *e³value* respectivamente), siendo además *e³value editor* la única solución específica para el modelado de *e³value*.

Como resumen de la información generada en este análisis de propuestas, la Tabla 2-12 muestra el listado de herramientas analizadas junto al cumplimiento o no de cada uno de los criterios comentados:

Tabla 2-12. Vista general de herramientas de modelado de negocio y proceso.

<i>Herramienta</i>	<i>Notaciones</i>	<i>Integración</i>	<i>Formato</i>	<i>Op. Complejas</i>	<i>Especificación</i>
<i>Bizagi</i>	BPMN	No	Sí (edición)	Sí (simulación)	Sí
<i>BonitaSoft</i>	BPMN	No	Sí (edición)	Sí (simulación)	Sí
<i>bpmn.io</i>	BPMN	No	Sí (edición)	No	Sí
<i>Canvanizer</i>	Canvas, Service Blueprint	No	Sí (edición, trabajo colaborativo)	No	Sí
<i>Draw.io</i>	Genérico (procesos)	No	Sí (edición)	No	No
<i>e³value editor</i>	<i>e³value</i>	No	Sí (edición)	Sí (simulación)	Sí
<i>Gliffy</i>	Genérico (procesos)	No	Sí (edición, compatibilidad MS Visio)	No	No
<i>Lucidchart</i>	Genérico (procesos)	No	Sí (edición, trabajo colaborativo)	Sí (simulación)	No

<i>MS Visio</i>	Genérico (procesos)	No	Sí (edición)	Sí (validación)	Sí
<i>Miro</i>	Canvas, Service Blueprint	No	Sí (edición, trabajo colaborativo)	No	Sí
<i>Signavio</i>	BPMN	No	Sí (edición, trabajo colaborativo)	Sí (simulación)	Sí
<i>Strategyzer</i>	Canvas	No	Sí (edición, trabajo colaborativo)	No	Sí

2.2.8 *Discusión*

Como conclusión de la presente sección, a continuación se hace una reflexión más global sobre los trabajos relacionados con esta tesis doctoral, incluyendo en ella ámbitos como el modelado de negocio y proceso, o el análisis de herramientas visto en esta SLR.

De entre las áreas relacionadas con esta tesis doctoral, 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 [97] o complejidad [98].

Sin embargo, a pesar de la cantidad de propuestas existentes, todavía surgen nuevos enfoques para el modelado de procesos de negocio [99] y nuevos dialectos BPMN [100]. 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 [101].

Por otro lado, a pesar del creciente interés en este área que ha provocado el reciente aumento de las ofertas producto-servicio [20], el modelado de inteligencia de negocio [102] 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 [1].

En este contexto, el objetivo de la herramienta a construir en esta tesis doctoral, no es definir nuevos lenguajes para el modelado de negocio o de

procesos, sino 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 el desarrollado en esta tesis doctoral 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, como ya se ha visto, en el momento de realizar esta SLR no se ha encontrado ninguna herramienta que permita trabajar de manera integrada con las diferentes notaciones seleccionadas. Sí existen, sin embargo, algunas herramientas que dan soporte aislado a distintas notaciones, o diversos diagramadores genéricos.

Algunas de las herramientas que dan soporte a Canvas y/o Service Blueprint son *Canvanizer* [95], *Miro* [96] o *Strategyzer* [95]. Estas soluciones web permiten trabajar de manera colaborativa, si bien en ningún caso soportan operaciones complejas como validaciones o simulaciones de modelos y trabajan con los distintos modelos de manera aislada.

Respecto a la notación e³value, la única herramienta encontrada ha sido *e³value editor* [22], que fue desarrollada por los mismos creadores de la técnica e³value, lo que asegura que los modelos generados cumplen con la especificación de la notación. *e³value editor* permite además realizar simulaciones de los modelos a partir de datos cuantitativos de los distintos intercambios de valor. Sin embargo, una vez más nos encontramos ante una herramienta específica para el trabajo aislado con una única notación.

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 *Draw.io* [91], *Gliffy* [92], *Lucidchart* [93] o *MS Visio* [94].

Por último, la notación para la que más herramientas de modelado se han encontrado es BPMN, siendo algunas de ellas *Bizagi* [87], *BonitaSoft* [88], *bpmn.io* [89], o *Signavio* [90].

Con todo lo anterior, hasta donde se ha podido conocer en el momento de realizar esta SLR, la propuesta de esta tesis doctoral es la primera en considerar las notaciones para el modelado de negocio y proceso discutidas en la sección 2.1.2 y, junto con el aprovechamiento de sus relaciones y correspondencias, proporcionar soporte tecnológico integrado para su uso en el contexto del diseño de servicios.

2.3 Ingeniería Dirigida por Modelos (MDE)

Como se ha visto en la sección anterior, actualmente existe una falta de soporte tecnológico respecto a herramientas basadas en modelos, que permitan trabajar de manera integrada con distintas notaciones para el modelado de negocio y proceso. Como respuesta a este problema, la presente tesis doctoral propone el uso de técnicas de Ingeniería Dirigida por Modelos (MDE) para la construcción de un entorno de modelado que cumpla con las características mencionadas.

A grandes rasgos, MDE provee de un mayor nivel de abstracción en el desarrollo del software. En este sentido, la aparición de los primeros lenguajes de programación ya supuso un gran avance para la construcción del software, ya que permitió a los programadores abstraerse de los complejos desarrollos en código máquina haciendo uso de instrucciones de alto nivel equivalentes a varias instrucciones máquina [103]–[105]. De este modo, el proceso de compilación o interpretación de un programa consiste, en mayor o menor medida, en la transformación de un lenguaje de programación a código máquina.

MDE, por tanto, surge como un paso natural en esa tendencia histórica por abstraer cada vez más el proceso de construcción de un software [43]. En particular, MDE permite partir de modelos de alto nivel expresados mediante algún lenguaje de modelado o Lenguaje de Dominio Específico (DSL), para generar el código de la aplicación en un determinado lenguaje de programación [105]. De manera similar a lo que ocurre con los procesos de compilación, en este caso se ha de llevar a cabo una transformación de modelo a código.

Las ventajas que proporciona el paradigma MDE incrementan las que en su día aparecieron con los lenguajes de programación: permite reducir el salto semántico entre el espacio del problema y el espacio de la solución, es decir, entre la forma en que los desarrolladores piensan las soluciones y la forma en la que deben expresarlas, lo que resulta en una reducción del esfuerzo necesario para la tarea de programar, mayor productividad, mantenimiento menos costoso, etc. [105]

Tal y como puede apreciarse en la Figura 2-9, MDE engloba otros paradigmas que pueden utilizarse según el ámbito de aplicación en el que se trabaje. De este modo, es posible distinguir entre ingeniería directa para la creación de nuevas aplicaciones (Desarrollo del Software Dirigido por Modelos, MDD), Reingeniería Basada en Modelos (MDR) y modelos en tiempo de ejecución (Models@runtime). A su vez, dentro de MDD destacan la Arquitectura Dirigida por Modelos (MDA), el Desarrollo Específico de Dominio (DSM) y las factorías de software.

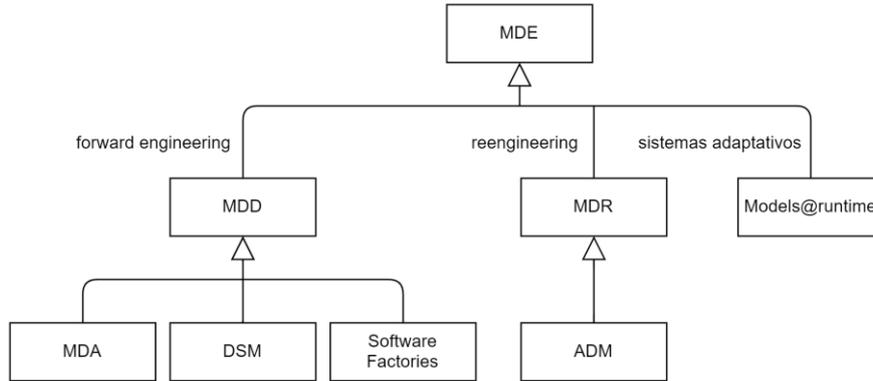


Figura 2-9. Paradigmas MDE y terminología. Adaptado de [105].

A pesar de encontrar tantos paradigmas diferentes dentro de MDE, todos ellos comparten unos principios básicos comunes, a saber [105]:

1. Cada modelo representa total o parcialmente un aspecto de un sistema software.
2. Dichos modelos son representados mediante DSLs o lenguajes de modelado.
3. Los DSLs anteriores son representados formalmente mediante metamodelos.
4. La automatización, normalmente se logra mediante transformaciones de modelos a código.

Si bien el uso que se hace en esta tesis doctoral del paradigma MDE puede encuadrarse dentro de MDD y MDA, lo cierto es que conviene hacer unas matizaciones sobre los principios básicos para comprender su aplicación en este trabajo. En primer lugar, el objetivo que se persigue en este caso mediante MDE no consiste en el desarrollo de software, sino en el desarrollo de modelos de negocio y proceso. De esta forma, cada modelo representará total o parcialmente un aspecto de una organización (su estructura global, un proceso, etc.). En segundo lugar, y como consecuencia del primer punto, los DSLs construidos consistirán en lenguajes para el modelado de negocio y proceso. En tercer lugar, se diseñarán tantos metamodelos como DSLs a soportar. Por último, la automatización no solo consistirá en transformaciones modelo a código, sino también en transformaciones modelo a modelo (entre DSLs diferentes). En lo sucesivo se describen distintos aspectos relacionados con la aplicación del paradigma MDE en esta tesis doctoral.

2.3.1 Desarrollo del Software Dirigido por Modelos (MDD)

Como se ha visto en la Figura 2-9, MDE engloba varios paradigmas diferentes según el ámbito de aplicación. Cuando hablamos de ingeniería directa para el desarrollo de software, nos encontramos ante MDD.

MDD, por tanto, pretende adaptar los principios de MDE al desarrollo de software mediante ingeniería directa [106]. Al igual que MDE, el objetivo de MDD pasa por incrementar el nivel de abstracción, pero también por lograr mayores cotas de automatización en el desarrollo de software. Para lograr este fin, se sirve del empleo de modelos que especifican un sistema con diferentes niveles de abstracción, de tal modo que los modelos de más alto nivel son transformados, mediante el uso de modelos de transformaciones, a otros modelos de menor nivel de abstracción hasta alcanzar modelos ejecutables mediante la generación de código o interpretación de modelos. Por lo tanto, las transformaciones entre modelos con diferentes niveles de abstracción, son un factor fundamental del paradigma MDD [107]–[111].

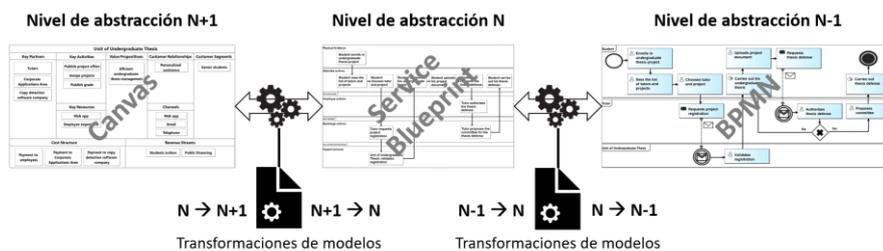


Figura 2-10. Proceso MDD aplicado al modelado de negocio y proceso.

Tal y como se ha hecho con el modelado de negocio y proceso a lo largo de esta tesis doctoral, MDD puede aplicarse a diferentes contextos, lo que da lugar a un conjunto de metodologías que se componen de un proceso de desarrollo, una serie de modelos y metamodelos que guían el proceso, y un conjunto de relaciones o transformaciones entre esos modelos y metamodelos. Nótese que al igual que sucede con los modelos que representan un sistema software, los modelos de negocio y proceso también pueden tener diferentes niveles de abstracción respecto a la implementación del servicio correspondiente. Por ejemplo, mientras que un modelo Canvas tiene un alto nivel de abstracción de la organización, un modelo BPMN posee un nivel mucho más cercano a la implementación del servicio. Para ejemplificar esto, la Figura 2-10 muestra, de manera simplificada, un proceso MDD aplicado al modelado de negocio y proceso.

2.3.2 *Arquitectura Dirigida por Modelos (MDA)*

Dentro de MDD es posible distinguir el paradigma MDA, que puede considerarse como la encarnación o puesta en marcha de MDE [112], ya que supone la aplicación más fiel de sus principios [38], [113]–[119]. A grandes rasgos, MDA consiste en la separación de la funcionalidad de un software (lógica de negocio) de los detalles de implementación en una plataforma determinada, de manera que el desarrollador diseña modelos de alto nivel independientes de la plataforma y, por medio de transformaciones modelo a modelo, genera los modelos dependientes de la plataforma [105].

MDA, por tanto, surge como una iniciativa del consorcio OMG (*Object Management Group*) con el objetivo de separar la especificación de un sistema software de los detalles relacionados con el uso de la plataforma por parte del sistema [38], para lo cual se define una arquitectura basada en modelos clasificados en tres niveles de abstracción:

- Modelos independientes de computación (CIM, *Computation Independent Model*): en estos modelos se recogen los requisitos del sistema. Sirven de conexión entre el personal de negocio y los desarrolladores, ya que recogen los detalles del ámbito de aplicación (dominio del negocio) por lo que pueden calificarse como modelos de dominio.
- Modelos independientes de la plataforma (PIM, *Platform Independent Model*): estos modelos representan la funcionalidad y estructura del sistema (dominio del sistema), si bien no recogen los detalles tecnológicos propios de la plataforma en el que se integrará el sistema.
- Modelos específicos de la plataforma (PSM, *Platform Specific Model*): una vez definidas en los modelos anteriores las características del sistema independientes de la plataforma, en los modelos PSM se combinan estas especificidades con los detalles de dicha plataforma. Por lo tanto, si se utilizan distintos modelos PSM, es posible generar diferentes implementaciones del mismo sistema.

Además, para llevar a la práctica el paradigma MDA, la OMG define el uso de diferentes estándares [38], como el lenguaje UML (*Unified Modeling Language*), que es un lenguaje de modelado para la especificación de sistemas software [70]; MOF (*Meta-Object Facility*), un lenguaje para la creación de metamodelos [120]; OCL (*Object Constraint Language*), un lenguaje para la definición de restricciones sobre modelos mediante expresiones formales [121]; y XMI (*XML Metadata Interchange*), un estándar independiente de la plataforma para el intercambio de metadatos y que mapea de MOF a XML [122].

Si bien el paradigma MDA está claramente enfocado al desarrollo de sistemas software, lo cierto es que resulta posible establecer paralelismos entre su definición y la aplicación que se ha hecho de él en esta tesis doctoral. Así, por ejemplo, mientras que MDA realiza una clasificación de modelos en función de su nivel de abstracción, cuando hablamos de modelos para el diseño de servicios también es posible clasificarlos según este parámetro. Prueba de ello, es la diferenciación constante que se hace a lo largo de esta tesis entre modelos de negocio (mayor nivel de abstracción) y modelos de proceso (menor nivel de abstracción), precisando bajar el nivel para acercarnos a los detalles de implementación del servicio en un contexto concreto. Conviene mencionar también, que durante esta investigación se ha comenzado a experimentar con la generación de modelos de *Smart Contracts* a partir de los modelos de negocio y proceso como e³value o BPMN [48], por lo que estaríamos hablando de un nuevo nivel de modelo mucho más cercano a la implementación de servicios en sistemas de información. Además, el entorno de modelado construido en esta tesis doctoral, hace uso, entre otras, de las tecnologías especificadas en la definición del OMG [38] para la creación de metamodelos, la definición de restricciones de modelos y el intercambio de metadatos.

2.3.3 *Modelado y metamodelado*

Como se ha visto a lo largo de las secciones anteriores, los modelos son un elemento fundamental de MDE y de todos los paradigmas que engloba. Pero también resultan clave en multitud de disciplinas científicas, por lo que podemos encontrar diferentes definiciones según el campo en el que nos encontremos [123], [124]. A continuación, por tanto, se describe qué se entiende por modelo y metamodelo en el contexto de esta tesis doctoral.

Un **modelo** es básicamente una simplificación de la realidad, resultado de un proceso de abstracción, que ayuda a entender dicha realidad (también denominada sistema en estudio). Para ello, los modelos ocultan cierta información y muestran solamente la que resulte relevante para un propósito determinado, lo que facilita la comprensión del sistema en estudio [125]. La realidad que representan los modelos puede consistir en cualquier tipo de sistema: un sistema software, un sistema mecánico, un sistema humano, etc. [105]. A lo largo de esta tesis doctoral, se utilizan modelos para representar sistemas de negocio y procesos para la provisión de servicios.

Históricamente, las ingenierías tradicionales han utilizado modelos como un paso previo en la construcción de sistemas, de tal forma que se sirven de planos

o mapas que especifican y describen el comportamiento y la estructura de los sistemas que representan. Incluso en otras áreas, como en el diseño de servicios, se utilizan modelos para ayudar en el diseño y comprensión de nuevos servicios [126]. En esta misma línea, la Ingeniería del Software comenzó a utilizar el modelado con el objetivo de definir los sistemas a construir [127], [128]. De esta forma, una definición de modelo que puede darse en el contexto de MDE, es la siguiente: un modelo es una descripción de un sistema (o parte de un sistema) en un lenguaje bien formado [116].

En este sentido, los modelos se expresan mediante lenguajes de modelado que especifican los elementos que pueden formar parte de un modelo, así como las relaciones entre dichos elementos. Esta información normalmente se describe por medio de un **metamodelo**, que es un modelo que contiene las “normas” que deben cumplir los modelos válidos para un determinado lenguaje [129]–[131]. Cuando un metamodelo se construye utilizando el lenguaje que él mismo define, entonces hablamos de **metametamodelo**. La Figura 2-11 representa la jerarquía existente entre los modelos, metamodelos y metametamodelos.

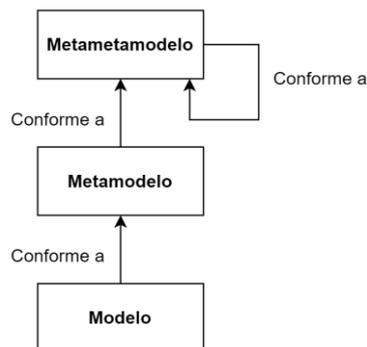


Figura 2-11. Jerarquía de modelado según [116].

Por otro lado, resulta importante distinguir entre los términos “modelo” y “diagrama”, ya que aunque en muchas ocasiones se utilizan como sinónimos, lo cierto es que poseen significados diferentes. El modelo es un concepto más amplio que el diagrama, ya que el diagrama es una representación gráfica del modelo, pero este último puede tener otro tipo de representaciones diferentes al diagrama, como por ejemplo una vista en forma de árbol o incluso una representación textual (por ejemplo, mediante XMI). Sirva para ilustrar esta idea lo mostrado en la Figura 2-12, donde se observan dos posibles vistas (diagrama y árbol) de un

modelo e³value. Por lo tanto, podemos decir que el modelo es independiente de su representación gráfica o textual [105].

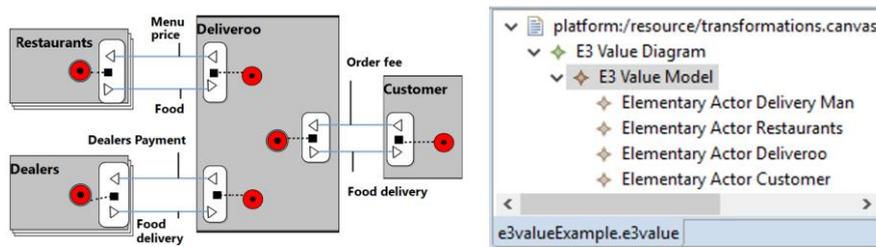


Figura 2-12. Vista de diagrama (izq.) y árbol (der.) de un modelo e³value.

2.3.4 Lenguajes de Dominio Específico (DSL)

En la sección anterior se ha hablado de los lenguajes de modelado utilizados para expresar modelos, si bien el término DSL se está imponiendo para referirnos a dichos lenguajes [105].

Un DSL es un lenguaje desarrollado específicamente para trabajar con los elementos y características de un dominio concreto. De esta forma, se consigue una mayor expresividad y facilidad de uso frente a un lenguaje de propósito general, ya que un DSL se acerca más al dominio del problema para un ámbito determinado. Al igual que los lenguajes de programación en su momento, los DSLs surgieron como una capa de abstracción adicional. Para una mejor comprensión de esta idea, es posible establecer un símil entre los DSLs y tales lenguajes de programación: un DSL puede ser visto como la gramática de un lenguaje de programación, mientras que los modelos construidos conforme a dicho DSL, corresponderían a las aplicaciones desarrolladas haciendo uso del lenguaje.

Los DSLs normalmente se componen de tres elementos principales: sintaxis abstracta, sintaxis concreta y semántica. La sintaxis abstracta especifica los elementos del lenguaje, las relaciones entre estos y las reglas que determinan los modelos bien formados. Normalmente se define por medio de un metamodelo junto a una serie de reglas para establecer las restricciones que deben cumplir los modelos correctos. La sintaxis concreta corresponde a la representación de los elementos de la sintaxis abstracta. Como se ha visto en la sección 2.3.3, esta representación puede ser textual o gráfica. Por último, la semántica generalmente se consigue por medio de la transformación a conceptos de otro lenguaje cuya semántica ya es conocida.

A lo largo de esta tesis doctoral se deben definir tantos DSLs como notaciones para el modelado de negocio y proceso a soportar. En concreto, se deberán construir DSLs para al menos las notaciones Canvas, e³value, Service Blueprint, PCN y BPMN. Cada uno de estos DSLs se compondrá de una sintaxis abstracta en forma de metamodelo, una serie de reglas de validación, una sintaxis concreta gráfica para la realización de diagramas, y unas reglas de transformación con las que poder transformar un modelo acorde a un DSL concreto, a otro DSL diferente.

2.3.5 Transformaciones de modelos

Como se ha visto en la sección 2.3.1, uno de objetivos del MDD consiste en proveer mayor nivel de abstracción en el desarrollo del software. En esta línea, MDA define diferentes niveles de modelos software en función de su dependencia o independencia de la plataforma de ejecución. Por lo tanto, para lograr este objetivo, MDD hace uso de transformaciones entre modelos que permitan desplazarse entre diferentes niveles de abstracción.

También resulta de gran importancia el segundo de los objetivos de MDD: la automatización del desarrollo del software. Si las transformaciones anteriores tuvieran que llevarse a cabo manualmente, los desarrolladores terminarían por descartar la solución MDD y concentrarían sus esfuerzos en las tareas de codificación [105]. Por lo tanto, para lograr el conjunto de los objetivos de MDD, es necesaria la automatización de dichas transformaciones.

Podemos encontrar diferentes definiciones sobre transformaciones de modelos. Por ejemplo, en [132] se describen como procesos automáticos que tienen uno o varios modelos de entrada y generan uno o varios modelos de salida, utilizando para ello un conjunto de reglas de transformaciones. Por su parte, en [38] se hace referencia a las transformaciones como procesos para convertir un modelo en otro modelo del mismo sistema.

En el contexto de esta tesis doctoral, también nos encontramos con diferentes niveles de modelos (negocio y proceso) que abarcan diferentes notaciones para el diseño de servicios. Por lo tanto, adoptando los principios de MDD para este cometido, también resultan necesarias transformaciones automáticas de modelos que permitan movernos entre diferentes niveles de abstracción. Para ello, se han utilizado las definiciones anteriores sobre transformaciones, y se han establecido una serie de relaciones entre elementos de diferentes notaciones a nivel de metamodelo que permitan generar, a partir de un modelo de entrada en una determinada notación, el correspondiente modelo de

salida en una notación diferente. La Figura 2-13 ofrece una visión general de este proceso de transformación entre modelos.

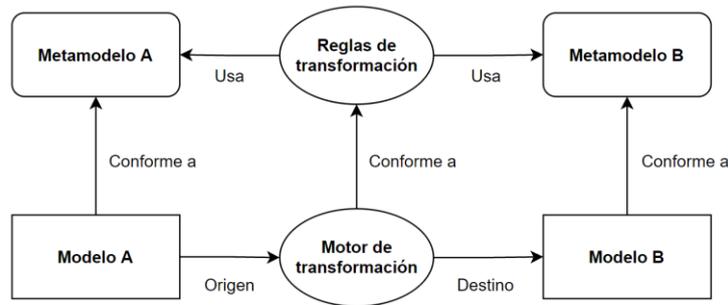


Figura 2-13. Proceso de transformación de modelos.

Por otro lado cabe mencionar que, dada la naturaleza de este trabajo, en ocasiones no resultará posible ni deseable una completa automatización del proceso de transformación de modelos por varios motivos. El primero de ellos es que, en determinadas transformaciones, no será posible disponer de suficiente información para generar elementos concretos del modelo destino. La segunda causa consiste en la pérdida de creatividad durante el proceso de diseño de servicios provocada por una excesiva automatización. De este modo, en el presente trabajo, ciertas transformaciones requerirán la asistencia del diseñador de servicios para completar información faltante o realizar pequeñas modificaciones sobre el modelo destino, pero siempre teniendo en cuenta los principios MDD para conseguir que el proceso de transformación no sea más complejo ni costoso que crear el modelo de destino de manera completamente manual.

2.3.6 Trazabilidad

Relacionado con las transformaciones de modelos vistas en la sección anterior, encontramos el término de trazabilidad. Este concepto, sin embargo, aparece originalmente en la Ingeniería del Software para hacer referencia a la gestión de requisitos, de tal forma que sea posible describir la vida de un requisito con el objetivo de asegurar que el software desarrollado cumple con los requerimientos del cliente. Para ello, mediante trazabilidad se ha de comprobar que se han satisfecho todos los requisitos y que cada componente del sistema satisface al menos uno de ellos [133]–[136].

Posteriormente, el concepto de trazabilidad fue evolucionando hasta hacer referencia a las relaciones que existen entre los artefactos involucrados en el ciclo

de vida de un software [133]. De acuerdo con esta nueva definición, la información aportada por la trazabilidad puede utilizarse para facilitar otras tareas del ciclo de vida del software, como por ejemplo valorar el impacto de cambios en el sistema, guiar la toma de decisiones, mantenimiento, comprobación de requisitos, etc. [133], [134], [137]–[139].

En el contexto de MDE, y por ende en MDD y MDA, la gestión de la trazabilidad cobra una importancia vital. Como ya se ha visto, estos paradigmas basan la construcción de un sistema en la especificación de modelos de alto nivel (independientes de la plataforma) que son transformados a modelos de bajo nivel (dependientes de la plataforma). Debido a que los diseñadores de los modelos iniciales y los desarrolladores de las transformaciones pueden ser personas distintas, resulta imprescindible poder conocer de dónde procede y por qué se genera cada componente del sistema [140], [141].

Trasladando lo anterior al ámbito de esta tesis doctoral y considerando un servicio como el sistema a construir, resulta igualmente necesario el mantenimiento de la trazabilidad con el fin de asegurar que cada modelo de un proyecto es coherente con el resto de modelos: por ejemplo, que un modelo BPMN contenga al menos todas las acciones que se especificaron en otro modelo Service Blueprint para el mismo servicio, o que ambos describan procesos relacionados con las propuestas de valor de un Canvas o los intercambios de valor de un e³value. Por supuesto, esto permite también realizar un seguimiento de los requisitos especificados en modelos de negocio (alto nivel) hacia modelos de proceso y viceversa, o facilitar la gestión de cambios en todos los modelos involucrados en un proyecto determinado.

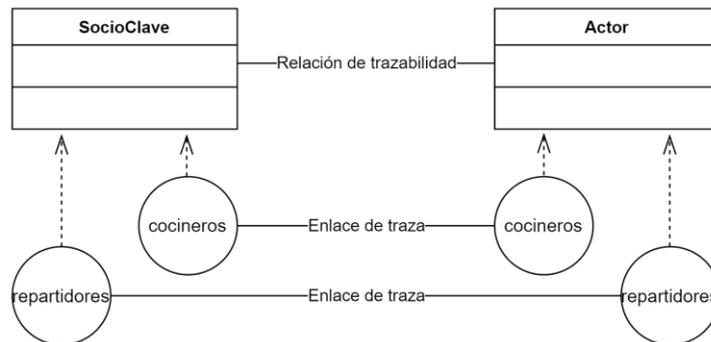


Figura 2-14. Relaciones de trazabilidad y enlaces de traza.

Finalmente, conviene aclarar que las relaciones de trazabilidad, al igual que las reglas de transformación, como muestra la Figura 2-14, se deben especificar a

nivel de clases del metamodelo (por ejemplo, entre los socios clave de un Canvas y los actores de un e³value). Cuando se realiza una transformación de modelos y a partir de instancias concretas de una clase (por ejemplo, repartidores y cocineros) se generan instancias concretas de otra clase en el modelo destino, se establecerán los enlaces de traza acordes a la relación de trazabilidad entre ambas clases.

2.4 Tecnologías MDE

Para dar soporte a las técnicas y metodologías MDE descritas en la sección anterior, actualmente existen multitud de herramientas tecnológicas con altos niveles de madurez [142], que permiten la creación de entornos de modelado gráfico capaces de llevar a cabo tareas complejas con los modelos generados.

La tabla Tabla 2-13, extraída de [143], muestra una clasificación de herramientas para el desarrollo de editores gráficos en función del *framework* que utilizan, los lenguajes empleados para la construcción de las sintaxis concreta y abstracta, capacidades de edición, empleo de modelos durante el proceso de desarrollo, automatización, usabilidad, y la posesión o carencia de una base metodológica que asegure la calidad del editor construido.

Tabla 2-13. Herramientas de desarrollo de editores gráficos. Adaptada de [143].

<i>Tool</i>	<i>Frame- work</i>	<i>Abs. syntax</i>	<i>Conc. syntax</i>	<i>Editing</i>	<i>Models</i>	<i>Auto- mation</i>	<i>Usabi- lity</i>	<i>Meth. basis</i>
Diagen	Eclipse	Ecore/ UML	DiaMeta Design	✓✓	✓✓	✓✓	✓✓✓	-
Eugenia	Eclipse	Ecore	EOL	✓✓	✓✓✓	✓✓✓	✓✓✓	-
GMF	Eclipse	Ecore	Draw2D	✓✓✓	✓✓✓	✓✓	✓✓	-
Graphiti	Eclipse	Ecore/ Java	Draw2D	✓✓✓	✓✓✓	✓✓✓	✓✓	-
MetaEdit+	Own	GOP- PRR	Internal API	✓✓✓	✓✓	✓✓✓	✓✓✓	✓
Sirius	Eclipse	Ecore	Odesign	✓✓✓	✓✓✓	✓✓✓	✓✓✓	✓
Tiger	Eclipse	AGG	Shape- Fig	✓	✓✓	✓	✓✓✓	-

Para el desarrollo de los DSLs y editores gráficos de esta tesis doctoral, por varios motivos se ha optado por la utilización de Eclipse GMF y Sirius. El primero de ellos es que ambas son herramientas basadas en el entorno de desarrollo Eclipse, lo que aporta fiabilidad, familiaridad y compatibilidad con otras herramientas por el hecho de ser un entorno muy extendido en el campo del

desarrollo software. Además, las dos herramientas emplean el lenguaje *Ecore* para la definición del metamodelo, siendo este un estándar el OMG [38], y librerías gráficas de fácil utilización para el desarrollo de la sintaxis concreta (*Draw2D* en el caso de GMF y *ODesign* en el caso de Sirius). Ambas herramientas poseen excelentes capacidades de personalización de los editores, siendo esto algo indispensable para lograr una fiel representación de los modelos de negocio y proceso a soportar, y las dos hacen uso de diferentes modelos gráficos para facilitar el trabajo de construcción de los editores. Las dos herramientas cuentan con un buen nivel de automatización durante el proceso de desarrollo de los editores y una buena usabilidad. Si bien en el caso de Eclipse GMF la curva de aprendizaje es más pronunciada, la experiencia previa del doctorando con este *framework* hace que esto no sea un inconveniente. Por último, Sirius cuenta con una base metodológica para asegurar la calidad de los editores gráficos construidos, algo de lo que no dispone Eclipse GMF.

Por otro lado, de acuerdo con las características mostradas en la Tabla 2-13, Graphiti también podría resultar una excelente opción para el desarrollo de los editores gráficos de esta tesis doctoral. Sin embargo, su uso quedó descartado durante una puesta en marcha comparativa en la que se observó su dificultad de uso respecto a Eclipse GMF, *framework* en el que además de poseer experiencia previa el doctorando, todo el proceso de desarrollo puede llevarse a cabo de manera visual mediante el uso de modelos gráficos.

Con todo lo anterior, a continuación se realiza un repaso de las tecnologías necesarias para la construcción de los diferentes DSLs y editores gráficos de esta tesis doctoral.

2.4.1 *Eclipse IDE*

Eclipse IDE¹³ (*Integrated Development Environment*) es un entorno de desarrollo integrado de código abierto, universal y portable [144]. Eclipse proporciona un marco de trabajo para la construcción de aplicaciones y herramientas, y es ampliamente reconocido y aceptado en el campo del desarrollo software por la facilidad con la que es posible ampliar su funcionalidad por medio de extensiones. Dichas extensiones son conocidas como *plug-ins*, y consisten en aplicaciones que aportan funcionalidades adicionales añadiéndolas al núcleo principal de Eclipse.

¹³ <https://www.eclipse.org/>

De esta forma, tal y como muestra la Figura 2-15, podemos visualizar Eclipse como una aplicación con un núcleo principal desarrollado en Java (*Platform Runtime*) al que se conectan un conjunto de *plug-ins* disponibles en tiempo de ejecución. Es posible conectar tantos *plug-ins* como funcionalidades necesitemos utilizar en Eclipse. Sin embargo, existen dos que resultan indispensables para el funcionamiento del entorno y que servirán de enlace con el resto de *plug-ins* del sistema: *Workspace* y *Workbench*. El primero de ellos hace referencia al espacio de trabajo, y se utiliza para gestionar todos los recursos mostrados por el entorno de desarrollo que tienen correspondencia con carpetas o archivos del sistema, como los proyectos, paquetes, clases, etc. Por su parte, el *plug-in Workbench*, sirve para proveer una interfaz gráfica de usuario que puede extenderse o personalizarse mediante puntos de extensión, que pueden aportar o modificar las barras de herramientas, menús, etc.

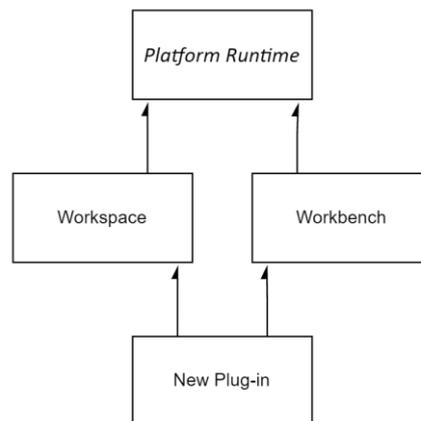


Figura 2-15. Arquitectura básica de Eclipse.

A partir de lo anterior, es posible afirmar que el entorno de modelado construido en esta tesis doctoral, será desarrollado en Eclipse y podrá ser ejecutado en Eclipse. En primer lugar, como ya se ha indicado, los editores gráficos de este trabajo se construirán utilizando las herramientas Eclipse GMF y Sirius, que se sirven de Eclipse IDE como base. Por lo tanto, para el desarrollo, se partirá de una versión de Eclipse a la que se habrán conectado los *plug-ins* necesarios para poder trabajar con ambas herramientas. En segundo lugar, todos los editores gráficos y funcionalidades adicionales construidas, se desarrollarán en forma de nuevos *plug-ins* de Eclipse, por lo que su utilización deberá llevarse a cabo instalando tales *plug-ins* en una nueva versión de Eclipse. Para más

información sobre la arquitectura técnica del entorno de modelado objeto de esta tesis doctoral, se remite al lector a la sección 3.2.4.

2.4.2 *Eclipse Modeling Framework (EMF)*

Si bien el entorno de modelado para el diseño de servicios será construido mediante los *frameworks* Eclipse GMF y Sirius, conviene aclarar qué es Eclipse EMF (*Eclipse Modeling Framework*), ya que sirve de base a los editores gráficos desarrollados mediante GMF.

Eclipse EMF¹⁴ es un proyecto de Eclipse que provee facilidades para la construcción de herramientas basadas en modelos de datos estructurados. De este modo, a partir de una especificación de modelo mediante XMI, EMF permite el desarrollo de un DSL completo. Para ello, EMF genera una serie de clases Java para describir el modelo, y otras con las que permitir su visualización y edición mediante una vista en forma de árbol [144].

La principal ventaja de EMF es la interoperabilidad entre herramientas diferentes basadas en el mismo *framework*. De este modo, es factible importar y trabajar con modelos especificados en otras herramientas, lo que posibilita que el entorno de modelado sea fácilmente ampliable con nuevas notaciones desarrolladas por terceros.

En concreto, durante el desarrollo del entorno de modelado, EMF se utilizará para la especificación de los metamodelos de cada DSL. En este sentido, con EMF es posible definir modelos *Ecore* que son instancias del meta metamodelo definido por EMF con el mismo nombre. A continuación, se añade información sobre el dominio de la solución y se genera un nuevo modelo llamado *Genmodel*, a partir del que pueden obtenerse diferentes *plug-ins* que sirven de editores de modelos con vista de árbol.

En definitiva, en este trabajo se crearán tantos proyectos EMF como notaciones a soportar, lo que proveerá no solo los metamodelos correspondientes, sino también la funcionalidad necesaria para crear y gestionar modelos acordes a dichos metamodelos mediante una vista con un esquema en forma de árbol (véase Figura 2-16). A continuación, a partir de estos proyectos y mediante el uso de Eclipse GMF, se construirá la capa gráfica (editor gráfico) para cada una de las notaciones.

¹⁴ <https://www.eclipse.org/modeling/emf/>

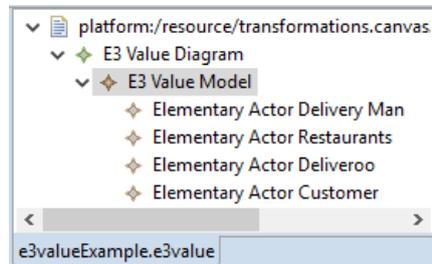


Figura 2-16. Vista en forma de árbol de un modelo EMF e³value.

2.4.3 Graphical Modeling Framework (GMF)

Eclipse GMF¹⁵ (*Graphical Modeling Framework*) es un proyecto de Eclipse que proporciona una componente generativa y una plataforma de ejecución para editores gráficos a partir de proyectos EMF [144].

GMF se basa en el desarrollo de una serie de modelos para la definición de la sintaxis visual de un DSL, así como de las relaciones entre elementos del metamodelo y elementos gráficos. Dichos modelos son los siguientes:

- **.ecore**: metamodelo del DSL elaborado previamente mediante EMF y que servirá de base para la construcción del editor gráfico.
- **.gmfgraph**: modelo que contiene la sintaxis concreta (elementos visuales) que empleará el editor gráfico.
- **.gmftool**: modelo que define los elementos disponibles en la paleta de herramientas del editor gráfico, así como sus iconos correspondientes.
- **.gmfmap**: modelo que define las relaciones entre los elementos de los modelos anteriores (qué clase del metamodelo corresponde a qué elemento gráfico y a qué herramienta de la paleta).
- **.gmfgen**: modelo que recoge toda la información anterior y a partir del cual se genera el código del editor gráfico.

Como puede observarse en la Figura 2-17, a partir del metamodelo, de la representación gráfica y de la paleta de herramientas, GMF construye, con la asistencia del desarrollador, un modelo que recoge las relaciones entre los tres modelos anteriores. A partir de este modelo de relaciones, se genera el modelo que sirve de base para la obtención del código Java del editor gráfico, que podrá ser empaquetado y utilizado como un nuevo *plug-in* de Eclipse. Todo este trabajo es

¹⁵ <https://www.eclipse.org/gmf-tooling/>

llevado a cabo mediante un panel de control o *dashboard* (ver Figura 2-18) que permite tener un gran control sobre el conjunto del proceso.

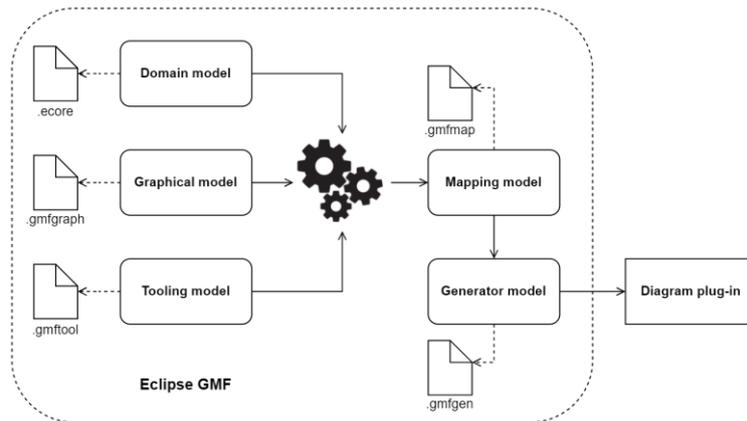


Figura 2-17. Desarrollo de un diagramador GMF.

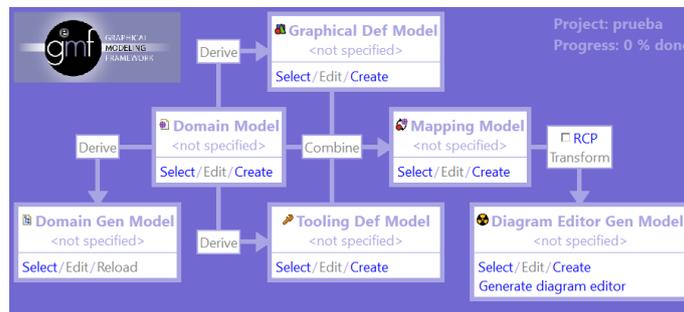


Figura 2-18. Panel de control de Eclipse GMF.

Teniendo en cuenta lo anterior, a lo largo de esta tesis doctoral se crearán tantos diagramadores gráficos como notaciones a soportar. A su vez, por cada diagramador se habrán de definir los modelos correspondientes (ecore, representación gráfica, paleta, mapeo) y generar los *plug-ins* finales. La Figura 2-19 muestra el resultado del proceso de desarrollo de un editor gráfico GMF para la notación e³value.

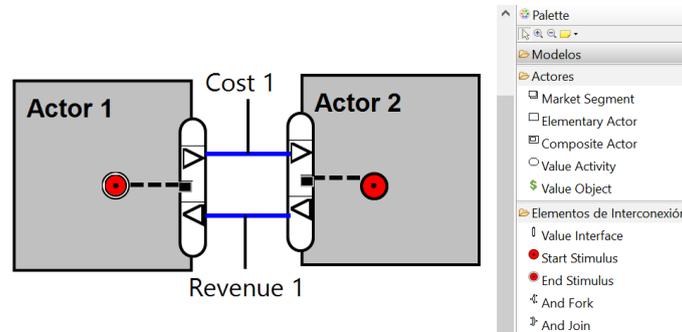


Figura 2-19. Diagramador GMF para la notación e^3 value.

2.4.4 Sirius

Aunque en un primer momento se seleccionó Eclipse GMF para el desarrollo de los editores gráficos de esta tesis doctoral, la herramienta Sirius¹⁶ ha sufrido en los últimos años importantes avances en materia de usabilidad [145], [146], compatibilidad con el lenguaje *Xtext* para la especificación de DSLs, e integración con el entorno de desarrollo Obeo Designer¹⁷. Por este motivo, y para demostrar la posibilidad de integrar en el conjunto del entorno de modelado de servicios otros editores gráficos también basados en Eclipse EMF, se optó por construir el editor BPMN utilizando el *framework* de Sirius.

Sirius es un proyecto de Eclipse desarrollado por Obeo¹⁸ y Thales¹⁹ para la generación de editores gráficos para DSLs. Mediante una aproximación basada en puntos de vista o *viewpoints*, Sirius facilita el trabajo con arquitecturas complejas de dominios específicos [147].

Al igual que con GMF, el desarrollo de un editor gráfico mediante Sirius parte de la especificación de un DSL elaborado con EMF. Sin embargo, a diferencia del primero, en el que para visualizar el resultado construido se ha de generar el código Java a partir de los modelos intermedios, Sirius provee *feedback* instantáneo en todo momento sin necesidad de generar el código de la aplicación [147]. Una vez finalizado el trabajo de desarrollo, el editor puede ser empaquetado y distribuido en forma de *plug-in* de Eclipse, lo que permite una rápida integración en el conjunto de la herramienta de modelado construida en esta tesis. Además,

¹⁶ <https://www.eclipse.org/sirius/>

¹⁷ <https://www.obeodesigner.com/>

¹⁸ <https://www.obeosoft.com/>

¹⁹ <https://www.thalesgroup.com/>

como ya se ha mencionado, el hecho de ser una herramienta basada en EMF, permite una total compatibilidad con el resto de modelos del entorno para llevar a cabo tareas como validaciones de modelos, transformaciones, etc. Cabe mencionar que Sirius utiliza GMF como base de funcionamiento.

También resulta destacable la integración de Sirius con el lenguaje Xtext, lo que permite especificar reglas de validación y corrección durante el desarrollo del editor gráfico, o el hecho de que Sirius permita construir fácilmente diferentes *viewpoints* para un mismo DSL, de tal forma que en tiempo de ejecución el usuario puede seleccionar la representación gráfica que desee (entre las que estén disponibles) para el modelo con el que esté trabajando.

2.4.5 *Epsilon*

Las herramientas descritas en las secciones anteriores permiten definir DLSs y construir editores gráficos utilizándolos como base. Esto nos permite trabajar con diagramas y realizar operaciones básicas con los modelos subyacentes, como la edición de los elementos que contienen. Sin embargo, cuando se requiere de operaciones más complejas, como validaciones, correcciones, transformaciones, etc., se debe recurrir a lenguajes específicos para la gestión de modelos.

En esta línea, Epsilon²⁰ es una familia de lenguajes de scripting basada en Java para la automatización de tareas con modelos EMF en el contexto de MDE. De esta forma, Epsilon posee lenguajes para llevar a cabo generaciones de código, transformaciones entre modelos, validaciones, comparaciones, etc. Además, Epsilon ofrece una serie de editores, depuradores de código e intérpretes para cada uno de sus lenguajes, así como herramientas para la visualización y gestión de modelos, pudiendo instalar todas estas características como *plug-ins* adicionales de Eclipse [148]–[150]. Algunos de los lenguajes más utilizados de Epsilon son los siguientes:

- ***Epsilon Object Language (EOL)***: lenguaje considerado como el núcleo de Epsilon, que sirve de base para el resto de lenguajes. EOL puede ser utilizado como un lenguaje de propósito general para llevar a cabo tareas con modelos. Se trata de un lenguaje imperativo que permite crear, consultar y modificar modelos EMF [151].
- ***Epsilon Validation Language (EVL)***: lenguaje utilizado para especificar y evaluar restricciones para la validación de modelos, pudiendo ofrecer

²⁰ <https://www.eclipse.org/epsilon/>

además mensajes de error al usuario o incluso propuestas de solución a los problemas de validación.

- ***Epsilon Transformation Language (ETL)***: lenguaje utilizado para definir reglas de transformación (ver Figura 2-13) con las que automatizar transformaciones desde uno o varios modelos de entrada en una determinada notación, a uno o varios modelos de salida (en una notación diferente). Además, ETL permite combinar bloques de código imperativo con la definición de reglas declarativas, lo que permite llevar a cabo transformaciones que requieren la asistencia del usuario por medio de cuadros de diálogo [152].
- ***Epsilon Wizard Language (EWL)***: lenguaje empleado para llevar a cabo transformaciones en las que además de generar el o los modelos de destino, también se modifica el o los modelos de origen [153].
- ***Epsilon Comparison Language (ECL)***: lenguaje cuya finalidad es la de comparar modelos para, por ejemplo, buscar elementos que aparezcan duplicados o definir relaciones entre ellos [154].
- ***Epsilon Generation Language (EGL)***: lenguaje para la conversión de modelos a texto para generar, por ejemplo, código en un determinado lenguaje de programación a partir de un modelo concreto [155].
- ***Epsilon Merging Language (EML)***: lenguaje para la unificación de dos o más modelos en uno solo, aunque provengan de distintos metamodelos [156].

Como se ha indicado, además de diferentes lenguajes de scripting con distintas funcionalidades, Epsilon también proporciona un conjunto de utilidades que complementan la interacción con proyectos EMF y GMF, como por ejemplo la herramienta Eugenia para el desarrollo de editores gráficos basados en GMF, cuyas características pueden observarse en la Tabla 2-13, o el editor de modelos EMF Modelink, que permite visualizar en un panel dos o tres modelos al mismo tiempo, lo que puede resultar útil para gestionar enlaces de traza entre los diferentes modelos.

En el contexto de esta tesis doctoral, se utilizarán especialmente los lenguajes EOL y ETL para implementar las transformaciones entre modelos de negocio y proceso, así como la herramienta Modelink para gestionar los enlaces de traza entre los modelos involucrados en una transformación (modelo de origen, modelo de destino y modelo de trazas). El principal motivo para la elección de los lenguajes ETL y EOL para la implementación de transformaciones se debe a que,

como ya se ha especificado, la combinación de ambos lenguajes permite construir transformaciones en las que se intercala código imperativo con código declarativo, resultando esto algo imprescindible ya que tal y como se describía en el último párrafo de la sección 2.3.5, muchas de las transformaciones de este trabajo requerirán la asistencia del usuario por medio de cuadros de diálogo.

Solución tecnológica

Una de las principales conclusiones de las dos primeras secciones del capítulo anterior, consiste en la falta de soporte tecnológico integrado para el modelado de negocio y proceso en el contexto del diseño de servicios. Como solución a este problema, la presente tesis doctoral propone el uso de técnicas de la Ingeniería Dirigida por Modelos (MDE) para la construcción de INNoVaServ: un entorno integrado para el modelado y gestión eficaz de modelos de negocio y proceso elaborados mediante distintas notaciones útiles en el proceso de diseño de servicios.

El presente capítulo realiza, por tanto, un recorrido por el proceso de construcción de la herramienta. Para ello, en primer lugar, la sección 3.1 presenta el entorno de INNoVaServ de una manera global. A continuación, la sección 3.2 describe la especificación de la solución, definiendo su arquitectura conceptual, el proceso de metamodelado de las cinco notaciones soportadas, el análisis de relaciones entre dichas notaciones y el diseño técnico de la herramienta. Posteriormente, la sección 3.3 presenta el proceso de desarrollo de INNoVaServ, describiendo la construcción de los diferentes editores gráficos, la implementación de puentes tecnológicos entre modelos basados en el análisis teórico de relaciones, así como distintos trabajos relacionados con la herramienta sobre validaciones formales de algunas de las notaciones soportadas.

3.1 Introducción

Como se ha visto en la sección 2.2, si bien actualmente existen multitud de herramientas tecnológicas que dan soporte aislado a determinadas notaciones, lo cierto es que no se ha encontrado ninguna solución que permita trabajar, de manera integrada, con diferentes técnicas para el modelado de negocio y proceso en el contexto del diseño de servicios. Además, la mayoría de herramientas disponibles no están basadas en modelos, lo que significa que más allá de la edición de diagramas, en muchos casos no es posible llevar a cabo tareas complejas como validaciones o transformaciones automáticas de modelos con la información que contienen.

Para tratar de solventar esta falta de soporte tecnológico y demostrar la viabilidad de la solución planteada, la presente tesis doctoral propone el uso de técnicas MDE aplicadas al modelado de negocio y proceso con el objetivo de

construir un entorno que permita trabajar, de manera integrada y eficaz, con diferentes notaciones en el contexto de proyectos de diseño de servicios.

Por lo tanto, durante este proyecto se ha trabajado en la construcción del entorno de modelado INNoVaServ: un prototipo de código abierto basado en Eclipse para tareas de investigación, que provee un conjunto de herramientas de modelado para tratar de solventar la falta de soporte tecnológico en lo que se refiere al modelado integrado mediante distintas notaciones de negocio y proceso para el diseño de servicios. Para ello, INNoVaServ integra una serie de DSLs visuales además de las herramientas necesarias para registrar y gestionar las relaciones entre los modelos elaborados con dichos DSLs, en el contexto de un proyecto de diseño de servicios. De este modo, además de los DSLs gráficos para las notaciones Canvas, e³value, Service Blueprint, PCN y BPMN, INNoVaServ contiene un panel de control o *dashboard* que permite la generación parcial de modelos en una determinada notación, a partir de otros modelos elaborados mediante técnicas diferentes, así como la generación de modelos de traza que recopilan las relaciones entre los modelos de entrada y salida. Además, INNoVaServ también soporta validaciones y correcciones automáticas de los modelos construidos.

Dado que INNoVaServ se ha construido en torno a una distribución de Eclipse IDE, su uso no requiere de ningún tipo de instalación más allá de su descarga y la configuración de Java en el equipo anfitrión. Sin embargo, también resulta posible integrarlo en cualquier otra distribución de Eclipse descargando en ella los distintos *plug-ins* que componen INNoVaServ.

Cabe mencionar que las notaciones soportadas por INNoVaServ no han sido seleccionadas de manera aleatoria, sino que cada una de ellas provee diferentes puntos de vista empleados en el proceso de diseño de servicios, tal y como se mostró en la sección 2.1. Antes de poder llevar a cabo la implementación del soporte para cada notación, resulta imprescindible el diseño de un metamodelo que recoja los elementos de cada técnica de modelado, así como las relaciones entre estos. Además, como ya se ha mencionado, el soporte para cada una de las notaciones integradas consiste en un DSL gráfico que emplea EMF de base y es construido mediante los *frameworks* Eclipse GMF y Sirius. De este modo, es posible incorporar de manera sencilla nuevos DSLs a la herramienta siempre y cuando utilicen EMF como base.

Para poder conectar las diferentes notaciones de modelado de negocio y proceso soportadas por INNoVaServ, se ha llevado a cabo un análisis de relaciones a nivel de metamodelo entre las distintas técnicas, que ha servido de

guía para la implementación de transformaciones semiautomáticas mediante los lenguajes EOL y ETL de Epsilon. Además, con el objetivo de mantener la trazabilidad entre los modelos involucrados en una determinada transformación, se ha diseñado un metamodelo de trazas genérico, a partir del cual es posible obtener un modelo de trazas que recoge las relaciones entre los modelos de entrada y salida de cada transformación. Para poder visualizar y gestionar de manera sencilla todos los modelos involucrados en una transformación, INNoVaServ integra el visor de modelos EMF Modelink de Epsilon, que permite mostrar en un mismo panel el modelo de entrada, el modelo de salida, y el correspondiente modelo de trazas.

Como también se ha mencionado, INNoVaServ provee un pequeño panel de control para la gestión eficaz de los modelos de negocio y proceso utilizados en el contexto de un determinado proyecto de diseño de servicios. Dicho panel permite la creación y selección de modelos para, una vez cargados en el sistema, habilitar las opciones que permiten lanzar las transformaciones y generar modelos basados en notaciones diferentes, así como los correspondientes modelos de trazas, a partir del modelo de entrada.

Por último, gracias a determinados trabajos llevados a cabo en colaboración con diferentes grupos de investigación durante la construcción del entorno, INNoVaServ también da soporte a la validación formal de modelos Service Blueprint y PCN [157], [158]. Del mismo modo, actualmente se están llevando a cabo otros trabajos de manera paralela a esta tesis doctoral, que exploran la posibilidad de integrar un DSL para la generación de *Smart Contracts* a partir de modelos e³value [48].



Figura 3-1. Pantalla de carga de INNoVaServ.

3.2 Especificación de la solución

Una vez vistas en la sección anterior las características generales del entorno de modelado INNoVaServ, la presente sección realiza un repaso a la especificación de la herramienta comentando las decisiones de diseño relacionadas con su arquitectura conceptual, el metamodelado de las cinco notaciones soportadas, el análisis de relaciones llevado a cabo entre dichas notaciones con el objetivo de implementar transformaciones semiautomáticas entre modelos, o la arquitectura técnica propuesta para la construcción del entorno.

3.2.1 Arquitectura conceptual

Durante el diseño de la arquitectura conceptual de INNoVaServ, uno de los objetivos ha sido el dotar a la herramienta de un alto nivel de modularización en la medida de lo posible. De este modo, tal y como muestra la Figura 3-2, dicha arquitectura puede describirse de acuerdo a dos dimensiones ortogonales.

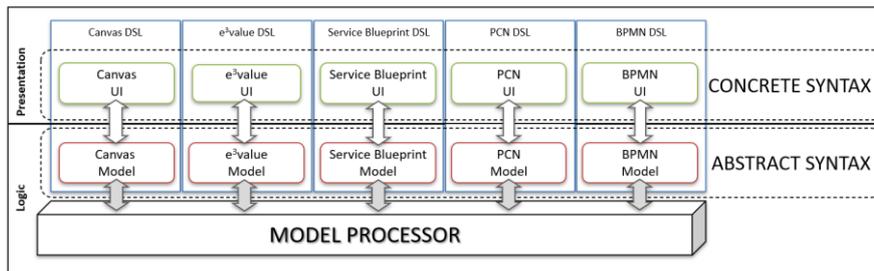


Figura 3-2. Arquitectura conceptual de INNoVaServ.

Observando la dimensión vertical de la Figura 3-2, es posible entender INNoVaServ como un conjunto de DSLs para dar soporte a diversas notaciones de modelado de manera aislada. En concreto, la herramienta consta de cinco DSLs para las técnicas Canvas, e³value, Service Blueprint, PCN y BPMN, si bien esta separación por DSLs permite, en principio, integrar nuevas notaciones de manera sencilla en el conjunto de la herramienta.

Por otro lado, para el diseño de la dimensión horizontal de la Figura 3-2, se ha tenido en cuenta el principio de separación de problemáticas de [159]. De esta manera, la arquitectura conceptual de INNoVaServ distingue, en cada DSL, la capa de representación de cada modelo del modelo en sí mismo. Dicha separación corresponde además a la división tradicional entre la sintaxis concreta y la sintaxis abstracta de cualquier lenguaje de modelado, tal y como se describió en la sección 2.3.4. Por lo tanto, la capa de presentación abarca la sintaxis concreta de cada

DSL, por lo que contiene los elementos necesarios para la edición y representación de cada modelo, como son las figuras gráficas que representan a cada elemento o la paleta de herramientas. Por su parte, la capa lógica abarca la sintaxis abstracta de cada DSL, por lo que contiene los metamodelos de cada notación, así como los modelos acordes a dichos metamodelos que se elaboran con la herramienta. Nótese que esta separación permite además modificar las representaciones de las notaciones (capa de presentación) sin que los modelos y metamodelos se vean afectados (capa lógica).

A pesar de la separación en capas de la dimensión horizontal de la arquitectura conceptual, INNoVaServ también permite al usuario trabajar con los modelos evitando la capa de presentación, si bien esta funcionalidad no está contemplada en el diseño de la arquitectura conceptual. Esto puede llevarse a cabo visualizando los modelos con formato texto, lo cual puede ser útil para realizar un análisis exhaustivo de los mismos, o corregir determinados errores durante la edición de un modelo.

Por último, dado que por la dimensión vertical las distintas notaciones se encuentran aisladas unas de otras, la capa lógica también abarca el procesador de modelos [27], que es el elemento encargado de conectar los diferentes DSLs integrados en la herramienta. Para ello, el procesador de modelos contiene diferentes componentes necesarios para llevar a cabo tareas de gestión avanzadas como validaciones de modelos, correcciones, transformaciones, mantenimiento de la trazabilidad, etc. En definitiva, el procesador de modelos es el elemento que permite integrar los distintos DSLs y, de este modo, dar soporte al trabajo conjunto mediante diferentes notaciones para el diseño de servicios.

3.2.2 *Sintaxis abstracta*

Como se ha visto en la sección anterior, la arquitectura conceptual de INNoVaServ distingue cinco DSLs para dar soporte a las notaciones Canvas, e³value, Service Blueprint, PCN y BPMN. Cada uno de estos DSLs consta principalmente de una sintaxis abstracta, que define los elementos del lenguaje, las relaciones entre estos y las reglas que determinan los modelos bien formados, y una sintaxis concreta, que especifica la representación gráfica que tendrá cada elemento de la sintaxis abstracta.

Según lo anterior, resulta razonable que la definición de la sintaxis abstracta de un DSL debe preceder a la construcción de su sintaxis concreta. Por este motivo, a continuación se presentan los metamodelos que dan forma a las sintaxis abstractas de cada una de las notaciones integradas en INNoVaServ. Para

el diseño de cada uno de estos metamodelos, se han tenido en cuenta las especificaciones mostradas en la sección 2.1.2 para cada una de las técnicas de modelado de negocio y proceso.

3.2.2.1 Business Model Canvas

La Figura 3-3 muestra el metamodelo de la notación Canvas. Su diseño se ha llevado a cabo en base a la especificación de esta técnica, que se describió en la sección 2.1.2.1, así como el listado de cada uno de sus elementos gráficos mostrado en la Tabla 2-1.

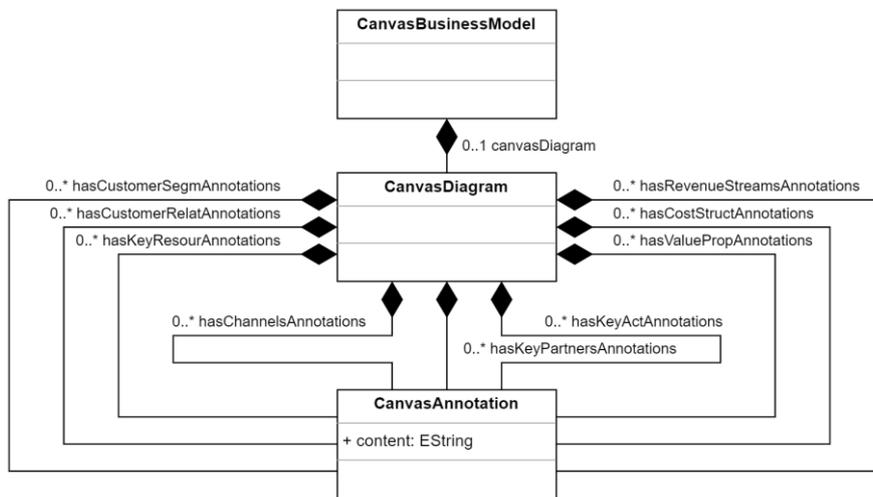


Figura 3-3. Metamodelo de la notación Canvas.

Como puede observarse en la Figura 3-3, el elemento raíz del metamodelo es *CanvasBusinessModel*, que representa el conjunto del modelo Canvas. Este elemento se compone de cero o un *CanvasDiagram*, clase utilizada para representar el lienzo de trabajo del modelo. A su vez, *CanvasDiagram* se compone de un número indeterminado de elementos *CanvasAnnotation*, que simbolizan las distintas anotaciones que pueden posicionarse en el interior del lienzo. Nótese que *CanvasAnnotation* posee un atributo de tipo cadena de texto para representar el contenido de la anotación correspondiente.

Como se vio en la sección 2.1.2.1, el lienzo del modelo Canvas posee nueve secciones diferentes, en cada una de las cuales es posible posicionar distintas anotaciones. Por este motivo, resulta necesario proporcionar en el metamodelo un mecanismo para distinguir qué anotaciones contiene cada una de las nueve regiones del lienzo. Esta distinción se ha modelado por medio de nueve

relaciones de composición que unen el elemento *CanvasDiagram* con *CanvasAnnotation*. De este modo, la representación gráfica del lienzo estará dividida en nueve secciones (contenedores) cada una de las cuales hará uso de exactamente una relación de composición, y de este modo todas las anotaciones de cada sección se encontrarán relacionadas con el lienzo por medio de la relación de composición correspondiente.

3.2.2.2 E³value

La Figura 3-4 muestra un resumen del metamodelo diseñado para la notación e³value en base a la especificación comentada en la sección 2.1.2.2 y la Tabla 2-2, en la que se recogen los principales elementos gráficos de esta técnica. Sin embargo, dado el gran número de elementos, relaciones entre estos y particularidades de la notación e³value, el metamodelo completo posee una complejidad mayor que la mostrada en la Figura 3-4. Por lo tanto, a continuación se describen las principales características de este metamodelo sin entrar en otros detalles de menor relevancia.

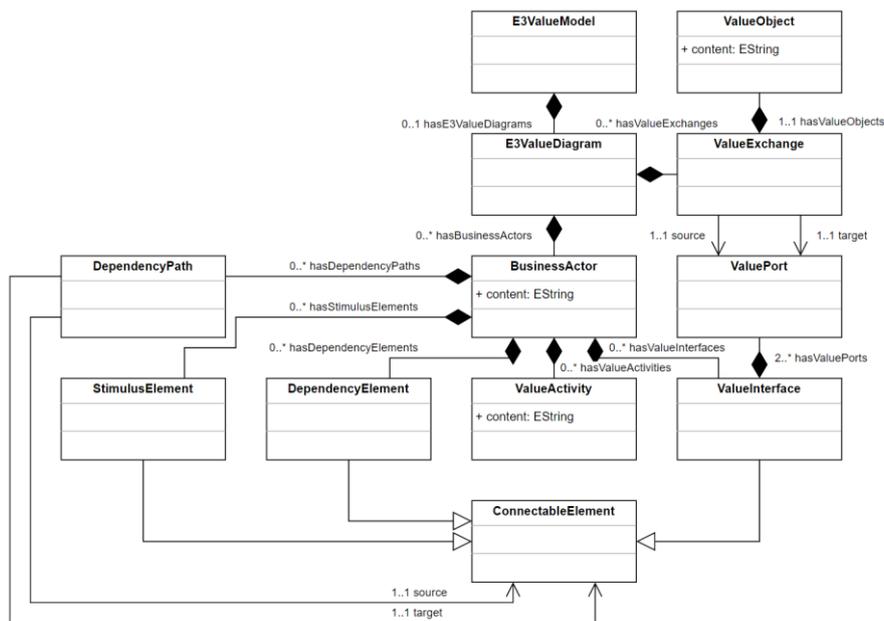


Figura 3-4. Resumen del metamodelo de la notación e³value.

Como puede observarse en la Figura 3-4, el elemento raíz del metamodelo e³value es *E3ValueModel*, que representa el conjunto del modelo y se compone de cero o un *E3ValueDiagram*, que simboliza el lienzo sobre el que se posicionarán

el resto de elementos del modelo. Cada *E3ValueDiagram* a su vez se compone de cero o más *BusinessActor*, que pueden ser actores individuales o segmentos de mercado. Estos *BusinessActor* pueden poseer además actividades de valor (*ValueActivity*), elementos de inicio o finalización de un intercambio de valor (*StimulusElement*), elementos de dependencia AND y OR (*DependencyElement*), o interfaces de valor (*ValueInterface*). Este último elemento posee además al menos dos puertos de valor (*ValuePort*) que permiten establecer intercambios de valor (*ValueExchange*) con otra interfaz de valor. Por último, estos intercambios de valor tendrán un objeto de valor asociado (*ValueObjetc*) para representar los objetos y/o servicios intercambiados.

3.2.2.3 Service Blueprint

La Figura 3-5 muestra el metamodelo diseñado para la notación Service Blueprint en base a la especificación de la sección 2.1.2.3 y el resumen de elementos gráficos de la Tabla 2-3.

Como puede observarse en la Figura 3-5, el nodo raíz del metamodelo es *ServiceBlueprintModel*, que representa el conjunto del modelo y se compone de cero o un *ServiceBlueprintDiagram* (lienzo del modelo con las cinco regiones de Service Blueprint). El lienzo se compone a su vez de acciones (*Action*), que se unen a éste por medio de cinco relaciones de composición diferentes (una por cada región del diagrama), y de conexiones (*Connection*), que tienen una acción de origen y una acción de destino.

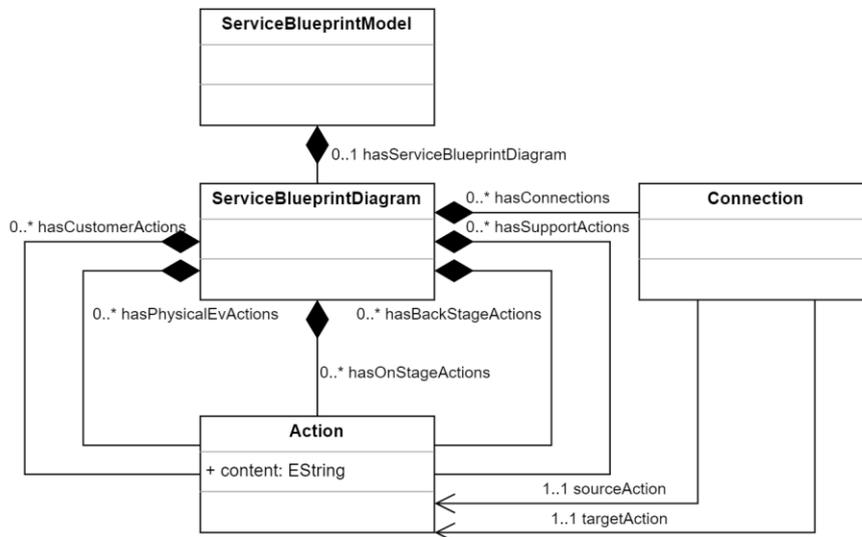


Figura 3-5. Metamodelo de la notación Service Blueprint.

3.2.2.4 Process Chain Network (PCN)

La Figura 3-6 muestra un resumen del metamodelo elaborado para la notación PCN tomando como referencia la especificación de la sección 2.1.2.4 y el resumen de elementos gráficos de la Tabla 2-4. De nuevo, el metamodelo completo posee una elevada complejidad, por lo que a continuación se describen los detalles más relevantes.

Como puede comprobarse en la Figura 3-6, la raíz del metamodelo (*PCNModel*) se compone esencialmente de diferentes entidades (*PCNProcessEntity*). A su vez, cada entidad está compuesta de diferentes acciones (*PCNProcessStep*) conectadas a ésta por medio de cinco relaciones de composición diferentes (una por cada región de interacción de la entidad). Las acciones tienen tantas implementaciones distintas como tipos de acciones diferentes posee la notación PCN, y pueden conectarse entre ellas mediante las relaciones *PCNStandardDependency*, *PCNDelayedDependency*, o las correspondientes a los nodos de decisión (*PCNDecisionStep*). Por último, las acciones se componen de etiquetas (*PCNTag*), teniendo de nuevo tantas implementaciones diferentes como tipos de etiquetas especifica la notación PCN.

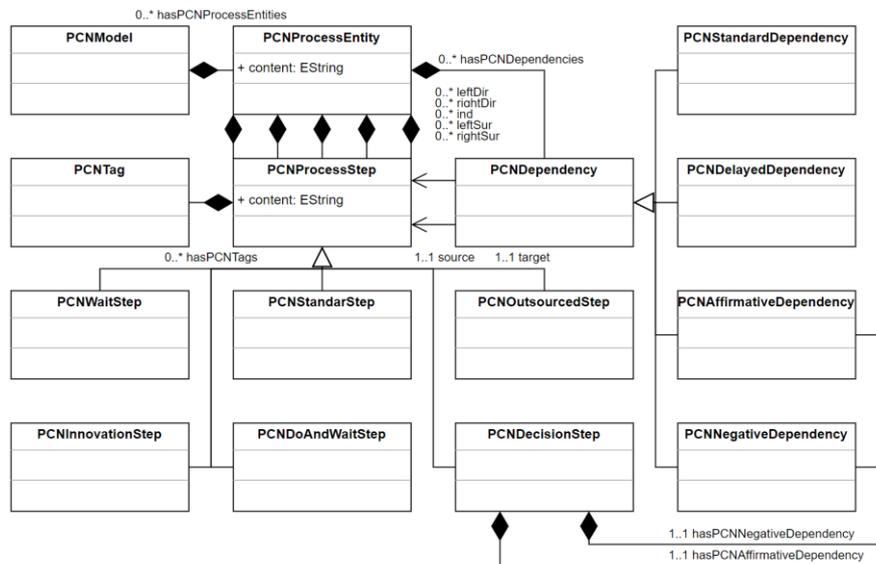


Figura 3-6. Resumen del metamodelo de la notación PCN.

3.2.2.5 Business Process Model and Notation (BPMN)

La Figura 3-7 muestra el resumen del metamodelo elaborado para la notación BPMN teniendo en cuenta la especificación de la sección 2.1.2.5 y el

resumen de elementos gráficos de la Tabla 2-5. Una vez más, debido al alto número de elementos de esta técnica de modelado, el metamodelo completo resultante posee una complejidad mayor que la mostrada en la figura, por lo que a continuación tan solo se describen los detalles más relevantes del mismo.

La clase *BPMNModel* representa la raíz del metamodelo, que se compone de un número indeterminado de elementos *BPMNPool*, que a su vez poseen al menos un *BPMNLane*. Además, cada *BPMNLane* posee diferentes elementos *BPMNFigure*, que pueden ser actividades, compuertas, eventos o representaciones de datos, y elementos *BPMNConnection* para establecer conexiones entre los componentes *BPMNFigure*, pudiendo estas conexiones representar paso de mensajes, secuencia, o relaciones de asociación.

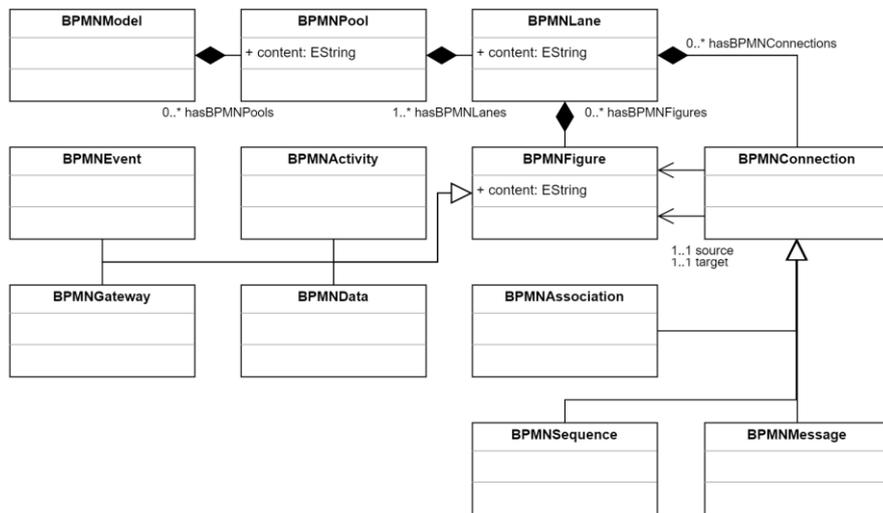


Figura 3-7. Resumen del metamodelo de la notación BPMN.

3.2.3 Correspondencias entre notaciones

Uno de los hitos de esta tesis doctoral, pasa por la creación de puentes tecnológicos basados en técnicas de procesamiento de modelos, que conecten las distintas notaciones soportadas por el entorno desarrollado. En este sentido y como se menciona a lo largo de la presente memoria, la información que proporcionan los modelos de negocio y proceso elaborados con diferentes notaciones, suele ser complementaria. Por este motivo, es posible identificar relaciones entre modelos expresados con diferentes notaciones. Por lo tanto, se ha llevado a cabo un análisis de relaciones entre las notaciones soportadas por INNoVaServ, que servirá de base para, a partir de un modelo expresado con una

determinada notación, generar automáticamente vistas parciales de modelos expresados con otras notaciones y los correspondientes modelos de relaciones.

Sin embargo, dada la diferente naturaleza y enfoque de las distintas notaciones soportadas por INNoVaServ, el estudio de correspondencias se ha dividido en tres bloques. De este modo, en primer lugar se analizarán las correspondencias entre los modelos de negocio Canvas y e³value. En segundo lugar, las correspondencias entre los modelos de proceso Service Blueprint, PCN y BPMN. Por último, se estudiarán posibles relaciones entre los dos bloques anteriores.

3.2.3.1 Modelos de negocio

En un intento por simplificar las relaciones encontradas entre las notaciones para el modelado de negocio soportadas por INNoVaServ, la Figura 3-8 muestra un resumen de un hipotético modelo de relaciones entre los elementos principales de un modelo Canvas (izquierda) y un modelo e³value (derecha).

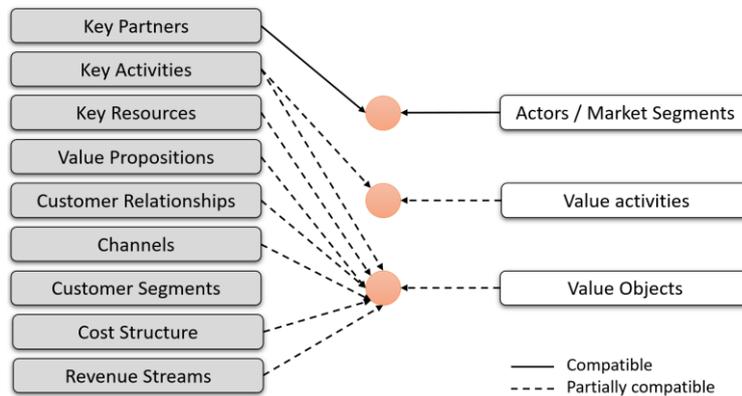


Figura 3-8. Resumen de correspondencias entre Canvas y e³value.

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 2-3 y Figura 2-4 por ejemplo). 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.

Profundizando más en estas relaciones, la Tabla 3-1 muestra el detalle de las correspondencias entre las notaciones Canvas y e³value. Para su elaboración se ha tomado como base la Tabla 2-2 en la que se especificaban los elementos gráficos de un modelo e³value, estableciendo las distintas correspondencias

respecto a las anotaciones posicionadas en las diferentes regiones de un tablero Canvas.

Tabla 3-1. Correspondencias entre Canvas y e³value.

<i>e³value</i>	<i>Canvas</i>
Actor	Socios clave
Segmento de mercado	Socios clave
Actividad de valor	Actividades clave, propuesta de valor
Necesidad de usuario	-
Elemento de terminación	-
Interfaz de valor	-
Puerto de valor	-
Intercambio de valor	-
Objeto de valor	Estructura de costes, fuentes de ingreso, recursos clave, actividades clave, propuestas de valor, relación con el cliente, canales de comunicación.
Conexión	-
Dependencia AND	-
Dependencia OR	-

Como puede observarse, tanto los *actores* como los *segmentos de mercado* de un e³value, podrían obtenerse a partir de las anotaciones colocadas en los *socios clave* del Canvas y viceversa, si bien no es posible determinar de manera completamente automática cuándo se trataría de actores individuales y cuándo de segmentos de mercado. En cualquier caso, al resultado obtenido habría que añadir dos actores adicionales: el cliente y el proveedor del servicio.

Del mismo modo, las *actividades de valor* de cada actor o segmento de mercado del e³value, podrían deducirse a partir de las anotaciones posicionadas en las *actividades clave* y *propuesta de valor*, si bien esto no se cumple en todos los casos.

Respecto a los *objetos de valor* intercambiados en un e³value, éstos pueden extraerse a partir de las anotaciones posicionadas en diferentes regiones del Canvas: *estructura de costes, fuentes de ingreso, recursos clave, actividades clave, propuestas de valor, relación con el cliente* o *canales de comunicación*. De

nuevo, aunque todas estas regiones poseen elementos potencialmente convertibles en objetos de valor, lo cierto es que no siempre es así y se requeriría de un proceso manual para su confirmación.

Por otro lado, aunque no se ha encontrado relación directa con los elementos de e³value *interfaz de valor*, *puerto de valor* e *intercambio de valor*, lo cierto es que pueden deducirse a partir de los objetos de valor previamente confirmados, ya que cada uno de ellos tendrá asociado un intercambio, puerto e interfaz de valor.

Finalmente, no se ha encontrado ninguna relación con los elementos *necesidad de usuario*, *elemento de terminación*, *conexiones* y *dependencias* de la notación e³value, así como con las anotaciones posicionadas en el *segmento de clientes* de la notación Canvas.

3.2.3.2 Modelos de proceso

La Figura 3-9 muestra una simplificación de las correspondencias encontradas entre las notaciones para el modelado de proceso soportadas por INNoVaServ. A modo de resumen, puede decirse que las *entidades* de un PCN serían equivalentes a los *pools* de BPMN, no teniendo correspondencia directa con ningún elemento de Service Blueprint, ya que éste último tan solo representa las entidades equivalentes al cliente y el proveedor del servicio. Por otro lado, dado que tanto PCN como BPMN y Service Blueprint poseen elementos para representar acciones y dependencias, en mayor o menor medida existirían correspondencias entre las tres notaciones con este tipo de elementos.

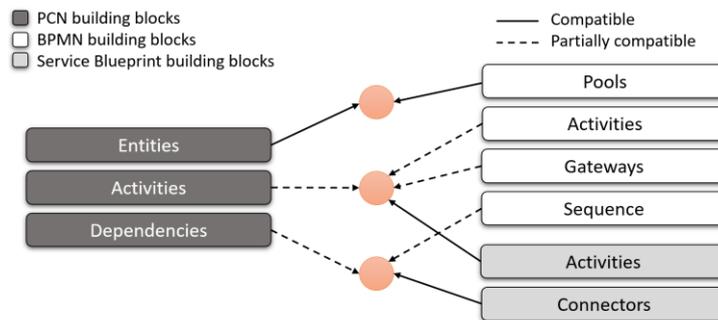


Figura 3-9. Resumen de correspondencias entre SBP, PCN y BPMN.

Dado, sin embargo, el alto número de elementos de las notaciones PCN y BPMN, la realidad es mucho más compleja que la representada por la Figura 3-9. En este sentido, la Tabla 3-2 muestra con mayor detalle las correspondencias encontradas entre las notaciones BPMN, PCN y Service Blueprint. Para su

elaboración, se han tomado como referencia los elementos básicos de dichas notaciones mostrados en la Tabla 2-3, Tabla 2-4 y Tabla 2-5.

Tabla 3-2. Correspondencias entre BPMN, PCN y Service Blueprint.

<i>BPMN</i>	<i>PCN</i>	<i>Service Blueprint</i>
Pool	Entidad	-
Tarea	Paso estándar	Acción
Tarea + Evento I. tiempo	Acción y espera	Acción
Tarea	Acción externa	Acción
Tarea	Paso innovador	Acción
Evento I. tiempo	Paso de espera	Acción
Compuerta OR	Paso de decisión	Acción
Secuencia / Flujo mensaje	Referencia	Conexión
Secuencia / Flujo mensaje	Dependencia estándar	Conexión
Secuencia / Flujo mensaje + Evento I. tiempo	Dependencia con retardo	Conexión
-	Ganancias / Pérdidas monetarias y no monetarias	-
-	Anotación textual	-

Como ya se ha indicado, existe una relación directa entre los *pools* de BPMN y las *entidades* de PCN. En el caso de Service Blueprint, sería necesario indicar manualmente cuáles son las dos entidades que se desean representar.

Por otro lado, cualquier *tarea* de BPMN o *paso* de PCN es directamente convertible a una *acción* de Service Blueprint, ya que éste tan solo posee un elemento para representar acciones. Sin embargo, existen ciertas particularidades entre BPMN y PCN. Por ejemplo, mientras que los *pasos estándar* de PCN son directamente convertibles a *tareas* BPMN, cuando se trata de *pasos de acción y espera*, la correspondencia se establecería con una *tarea* BPMN junto a un *evento intermedio de temporalización*.

También es posible establecer una correspondencia entre las *compuertas OR* de BPMN y los *nodos de decisión* de PCN, relacionándose estos elementos con *acciones* de Service Blueprint, ya que esta notación no posee elementos para representar decisiones.

En cuanto a las *dependencias* de PCN, éstas podrían asociarse con *secuencias* o *flujo de mensajes* de BPMN, en función de si la transición se produce dentro de una misma entidad o no, a pesar de que en un sentido estricto, los flujos de mensajes BPMN no representan transición entre acciones. Además, si se trata de una *dependencia con retardo* de PCN, a lo anterior habría que añadir un *evento intermedio de temporalización* BPMN. En cuanto a Service Blueprint, todas las dependencias o secuencias mencionadas tendrían correspondencia directa con las *conexiones* de esta notación. Cabe mencionar además que las *referencias* de PCN, pueden tratarse como *secuencias* o *flujo de mensajes* BPMN y *conexiones* Service Blueprint, ya que su función es igualmente la de conectar dos acciones.

Por otro lado, no se ha encontrado correspondencia directa con las *anotaciones* de la notación PCN, ni con los *lanes*, *eventos iniciales* o *finales*, *subprocesos*, *compuertas AND* y *asociaciones* de BPMN, si bien en algunos casos es posible deducir estos elementos mediante un procesamiento de la información recogida en un modelo PCN. Por ejemplo, si un *paso* PCN no posee dependencias de entrada, es posible asumir que antes de dicha tarea existiría un *evento de inicio* en BPMN. De forma equivalente, sería posible deducir los *eventos finales*. También resulta posible obtener *compuertas AND* para BPMN, a partir de acciones con dos o más dependencias de salida en PCN.

Por último, dado que tanto las entidades PCN como el lienzo Service Blueprint poseen diferentes regiones en las que posicionar acciones, es posible establecer ciertas correspondencias entre ellas. Por ejemplo, las acciones situadas en la *zona de interacción directa* de PCN, deberían posicionarse sobre las zonas adyacentes a la *línea de interacción* de Service Blueprint, mientras que las acciones posicionadas sobre la *región de proceso independiente* de la entidad correspondiente al proveedor en PCN, deberían posicionarse en la *zona de acciones internas* de Service Blueprint. Finalmente, las *acciones externas* (llevadas a cabo por terceros) de PCN, sería posible situarlas en la *zona de acciones de soporte* del modelo Service Blueprint.

3.2.3.3 Modelos de negocio y proceso

Una vez identificadas las relaciones entre los modelos de negocio y entre los modelos de proceso soportados por INNoVaServ, el tercer paso de este estudio ha consistido en el análisis de las relaciones entre ambos tipos de modelos. Dado que los primeros se centran en representar la estructura de una organización, mientras que los segundos ponen el foco 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 caminos de dependencia), podemos identificar cuántos modelos de proceso serán necesarios para representar las principales operaciones de servicio.

Si bien la aproximación inicial de esta tesis doctoral consistía en que el número de intercambios de valor de un e³value corresponde al número de modelos de proceso necesarios para representar la propuesta de valor [14], a raíz de la estancia predoctoral llevada a cabo en el grupo de investigación DISE²¹ de la Vrije Universiteit de Ámsterdam, ha sido posible conocer otros enfoques [160], [161] que junto a distintos casos de estudio, han permitido redefinir la propuesta inicial. De esta forma, para estimar a partir de un modelo e³value el número de modelos de proceso necesarios para representar una propuesta de valor, se ha de analizar cuántos caminos de dependencia contiene el primero. Para ello, cada camino de dependencia ha de empezar con al menos una necesidad de usuario y finalizar con al menos un elemento de terminación.

Por ejemplo, dado que el modelo e³value de la Figura 3-10 posee dos caminos de dependencia diferenciados, necesitaremos dos modelos de proceso para representar la propuesta de valor correspondiente. Las entidades participantes en estos modelos de proceso corresponderán a los actores o segmentos de mercado del modelo de valor. Sin embargo, más allá de eventos de inicio y finalización, no es posible deducir las acciones que implementarán la interacción entre esas entidades únicamente a partir de la información recogida en este modelo.

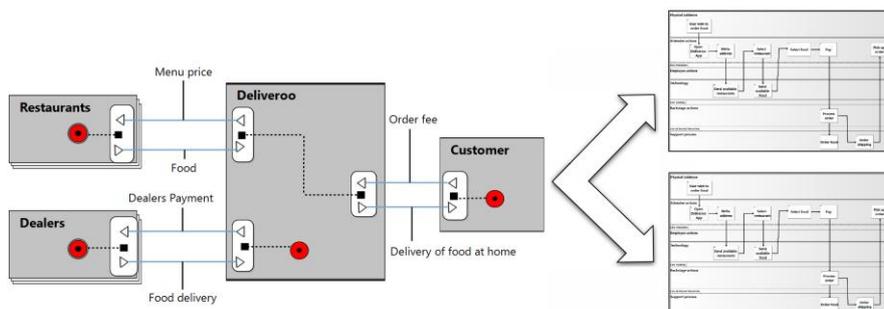


Figura 3-10. Relación entre notaciones de modelado de negocio y proceso.

²¹ <https://dise-lab.nl/>

3.2.4 Diseño técnico

Una vez vista la arquitectura conceptual del entorno de modelado a construir, así como el diseño de los metamodelos de cada DSL y el análisis de correspondencias entre los mismos, a continuación se presenta la arquitectura técnica que servirá de base para la implementación de la herramienta.

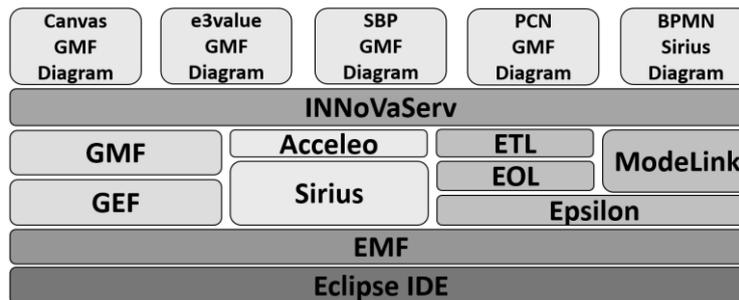


Figura 3-11. Arquitectura técnica de INNoVaServ.

Como puede apreciarse en la Figura 3-11, la base sobre la que se integran el resto de componentes es el entorno de desarrollo Eclipse, que fue descrito en la sección 2.4.1. De esta forma, el entorno de modelado es construido sobre una distribución de Eclipse, y es posible distribuirlo como una nueva versión de este entorno de desarrollo, por lo que no requeriría de más instalación que la asociada a la máquina virtual de Java.

El segundo gran componente de la arquitectura técnica de INNoVaServ, es el *framework* de Eclipse EMF, cuyas características se presentaron en la sección 2.4.2. A modo de resumen, EMF permite el diseño, construcción y administración de DSLs, de tal forma que posibilita trabajar con modelos textuales o en forma de árbol acordes a dichos DSLs. El motivo de situar EMF en este nivel de la arquitectura, se debe a que es un componente necesario para los elementos que se describirán a continuación. Además, también posibilita integrar en INNoVaServ cualquier DSLs adicional basado en EMF. Por otro lado, su integración en la plataforma de Eclipse se lleva a cabo mediante la instalación del *plug-in* correspondiente.

En el siguiente nivel de la arquitectura técnica de INNoVaServ, se encuentran distintos componentes que permiten trabajar con sintaxis concretas visuales sobre los DSLs basados en EMF. De este modo, los *frameworks* GEF y GMF (descrito en la sección 2.4.3), permiten trabajar con editores gráficos que utilizan como base los DSLs construidos sobre EMF. De manera análoga, Sirius

(sección 2.4.4) también se encuentra integrado en la arquitectura técnica para permitir la utilización de diagramadores construidos mediante esta solución. Junto a Sirius se encuentra el componente Acceleo²², que es el generador de código utilizado por esta herramienta. Cabe mencionar que sería posible integrar en esta arquitectura herramientas para soportar otro tipo de editores gráficos como las mostradas en la Tabla 2-13, siempre y cuando utilicen EMF como base.

En el mismo nivel anterior se encuentran diferentes componentes de Epsilon (sección 2.4.5), que permiten trabajar con la información recogida en los distintos modelos de un proyecto de diseño de servicios. En concreto, se encuentran integrados los lenguajes EOL y ETL para la implementación y ejecución de transformaciones entre modelos, así como la herramienta Modelink para la gestión de los modelos involucrados en una determinada transformación.

El conjunto de componentes descrito en los párrafos anteriores, constituye la funcionalidad elemental del entorno de modelado INNoVaServ, sobre el que es posible integrar distintos DSLs construidos mediante diferentes herramientas para la generación de editores gráficos basados en EMF. En concreto, la versión desarrollada en esta tesis doctoral, integra cuatro DSLs desarrollados mediante Eclipse GMF para dar soporte a las notaciones Canvas, e³value, Service Blueprint, y PCN. Adicionalmente, con el objetivo de demostrar la viabilidad de integrar en la herramienta otros DSLs construidos mediante diferentes herramientas basadas en EMF, se ha incorporado un DSL elaborado mediante Sirius para la edición y gestión de modelos BPMN.

3.3 Implementación de la solución

Una vez vista la especificación de INNoVaServ y algunas decisiones de diseño en la sección anterior, a continuación se realiza un repaso por algunos detalles técnicos de la solución construida, como son los referentes al proceso de desarrollo de los editores gráficos con GMF y Sirius, la implementación de las transformaciones semiautomáticas entre modelos y de un panel de control para la gestión de los mismos, o las validaciones formales de las notaciones PCN y Service Blueprint llevadas a cabo en colaboración con otros grupos de investigación.

²² <https://www.eclipse.org/acceleo/>

3.3.1 Sintaxis concreta

Como se vio en la arquitectura conceptual de la Figura 3-2, INNoVaServ integra un conjunto de DSLs para dar soporte a las notaciones Canvas, e³value, Service Blueprint, PCN y BPMN. Los editores gráficos para las cuatro primeras notaciones, se construyeron mediante el *framework* de Eclipse GMF debido al alto nivel de personalización que ofrece esta herramienta y al resto de ventajas discutidas en la sección 2.4. Sin embargo, debido a los recientes avances de Sirius, y para demostrar la viabilidad de integrar en el conjunto de INNoVaServ editores gráficos construidos con otras soluciones basadas en EMF, se decidió elaborar el DSL correspondiente a la notación BPMN mediante Sirius.

Si bien en las siguientes subsecciones se detallan algunos aspectos concretos del proceso de desarrollo de los editores mediante Eclipse GMF y Sirius, lo cierto es que en ambos casos se ha seguido un patrón común para la generación de editores gráficos en forma de DSLs como los descritos en [27], cuyas etapas se muestran en la Figura 3-12 y se describen a continuación:

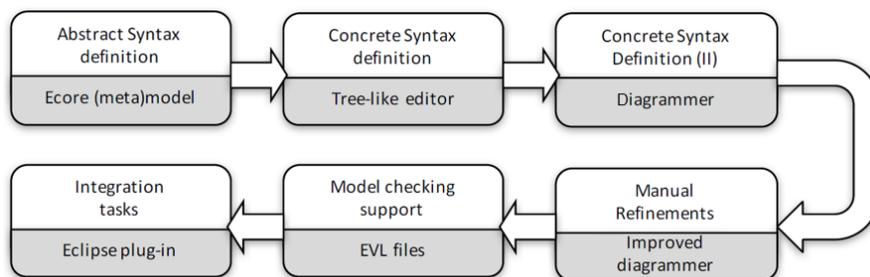


Figura 3-12. Proceso genérico de desarrollo de los editores.

- El primer paso del proceso de desarrollo consiste en la especificación de la sintaxis abstracta del DSL que se desea modelar, lo que fue llevado a cabo en la sección 3.2.2. A continuación, estos metamodelos deben ser expresados en términos del lenguaje Ecore [162].
- Para proveer al DSL de una sintaxis concreta, se utiliza EMF para generar un editor en forma de árbol completamente funcional, que permite diseñar modelos conformes al metamodelo definido previamente. La funcionalidad básica implementada por este editor, que es generada prácticamente de forma automática gracias a las características de EMF, es más que suficiente en este punto del proceso de desarrollo para realizar pruebas: se definen algunos modelos para comprobar que la sintaxis abstracta definida en el punto anterior es correcta y completa.

- Sin embargo, las características del editor en forma de árbol definido con EMF, no se corresponden con las características del editor gráfico que se desea desarrollar. De este modo, se desarrolla una sintaxis concreta visual generando un diagramador básico utilizando las herramientas de Eclipse GMF o Sirius. Dicho diagramador estará construido en realidad sobre el editor en forma de árbol generado anteriormente.
- Al igual que con EMF, la naturaleza genérica de GMF o Sirius supone ciertos límites en el nivel de personalización que se pretende conseguir con el diagramador obtenido. Por lo tanto, se han de realizar manualmente algunos refinamientos sobre el código Java generado, con el fin de implementar funcionalidades complejas que la generación automática de código no es capaz de soportar. Ejemplos de tales funcionalidades, son el redimensionamiento uniforme de figuras y posicionamiento de estas.
- Dado que cabe la posibilidad de que los usuarios de la herramienta no sean expertos de cada una de las notaciones, una tarea crucial es la de identificar qué modelos son válidos y cuáles no. Cabe destacar que los modelos generados por los usuarios, podrían ser procesados más tarde por un administrador automático de modelos que dé por hecho que las entradas siempre serán modelos válidos, por lo que resulta necesario comprobar que no haya errores en los mismos. Por lo tanto, el siguiente paso del proceso de desarrollo es el de implementar un validador de modelos utilizando el lenguaje de Epsilon EVL.
- Por último, se genera el *plug-in* que integra todo lo desarrollado hasta este punto, como el metamodelo, editores y validadores de modelos. Este *plug-in* proporciona una interfaz de usuario común, incluyendo diferentes controles como menús contextuales, ventanas de configuración de las diferentes tareas automatizadas en la herramienta, etc. El resultado de todo esto, es un entorno integrado que puede ser descargado o integrado como un único *plug-in*.

3.3.1.1 Editores GMF

El proceso de generación de un diagramador mediante Eclipse GMF, se reduce a la construcción de los distintos modelos mostrados en la Figura 2-17. De este modo, el primer paso consiste en la implementación del metamodelo correspondiente con el objetivo de obtener el metamodelo Ecore. Gracias a las características de EMF, es posible llevar a cabo este proceso de una manera completamente visual. A modo de ejemplo, la Figura 3-13 muestra el metamodelo Ecore para la notación PCN.

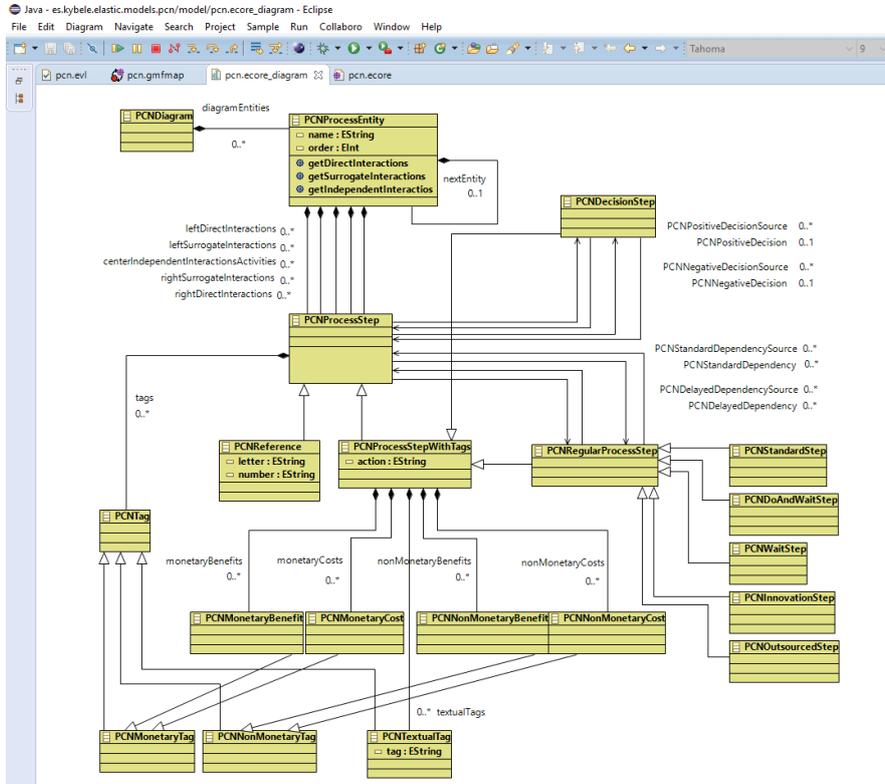


Figura 3-13. Metamodelo Ecore para PCN elaborado con EMF.

Una vez cargado el metamodelo Ecore en el panel de control de la Figura 2-18, el siguiente paso consiste en la construcción del modelo *.gmfgraph*, que contiene la definición de la sintaxis concreta del DSL, es decir, qué representación gráfica desea asociarse a cada elemento del metamodelo. De nuevo, para llevar a cabo este proceso, gracias a la funcionalidad de GMF no es necesario realizar una implementación en un determinado lenguaje de programación, sino que tan solo se requiere la definición de una serie de descriptores. La Figura 3-14 muestra el ejemplo del descriptor correspondiente a las acciones normales de la notación PCN, que consta de un rectángulo exterior invisible para colocar las anotaciones asociadas, y un rectángulo interior que representa la acción en sí.

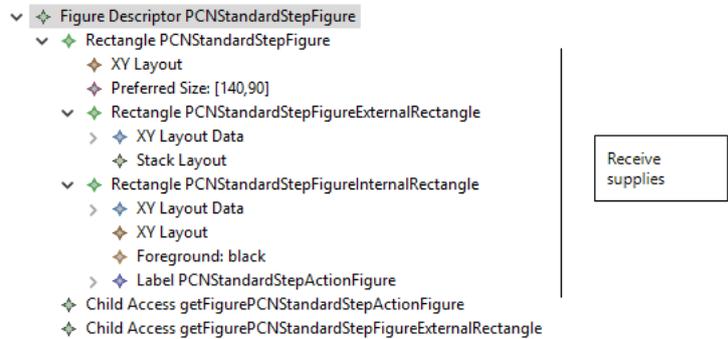


Figura 3-14. Definición gráfica de acciones PCN con GMF.

Una vez definida la sintaxis concreta del editor, el siguiente paso consiste en la construcción de la paleta de herramientas de este. Para ello, se ha de crear el modelo *.gmftool* que recoge las secciones de la paleta, los elementos y los iconos asociados a éstos. A modo de ejemplo, la Figura 3-15 muestra la definición de la paleta de herramientas del editor PCN junto al resultado final.

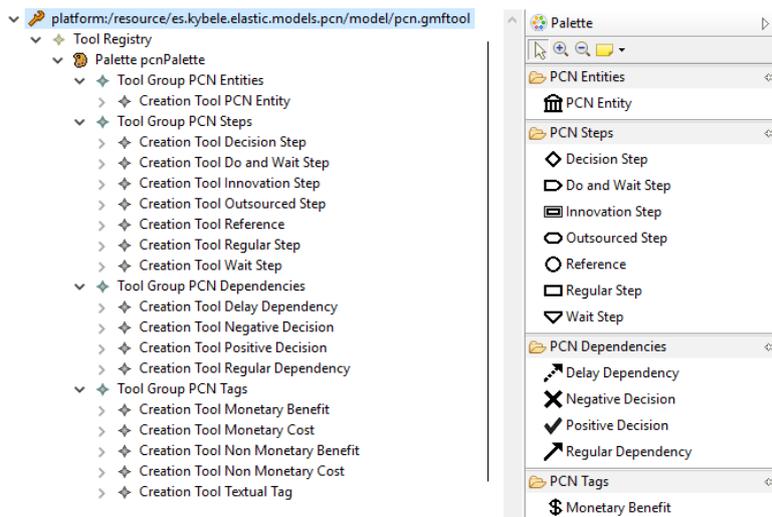


Figura 3-15. Definición de la paleta de herramientas del editor PCN con GMF.

Con los modelos anteriores ya creados, la siguiente tarea consiste en la construcción del modelo *.gmfmap*, que contiene una serie de asociaciones para definir qué elemento del modelo *.ecore* corresponde a qué figura del modelo *.gmfgraph* y a qué herramienta del modelo *.gmftool*. A modo de ejemplo, la Figura 3-16 muestra los principales elementos del modelo *.gmfmap* para el editor PCN.

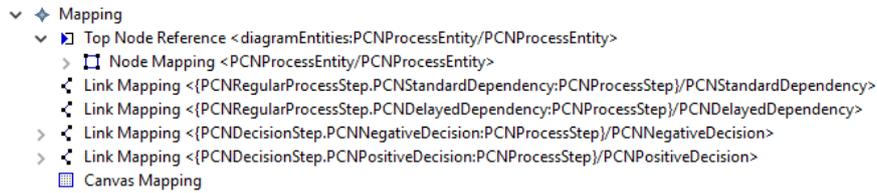


Figura 3-16. Definición del modelo .gmfmap para el editor PCN con GMF.

El último paso a llevar a cabo con el *framework* GMF para obtener el editor gráfico, consiste en la generación del modelo .gmfgen, lo cual se realiza de manera automática a partir de los modelos definidos anteriormente. Este último modelo servirá de enlace entre el código java resultante y los componentes de cada uno de los modelos. La Figura 3-17 muestra, a modo de ejemplo, el editor gráfico obtenido mediante GMF para la notación PCN.

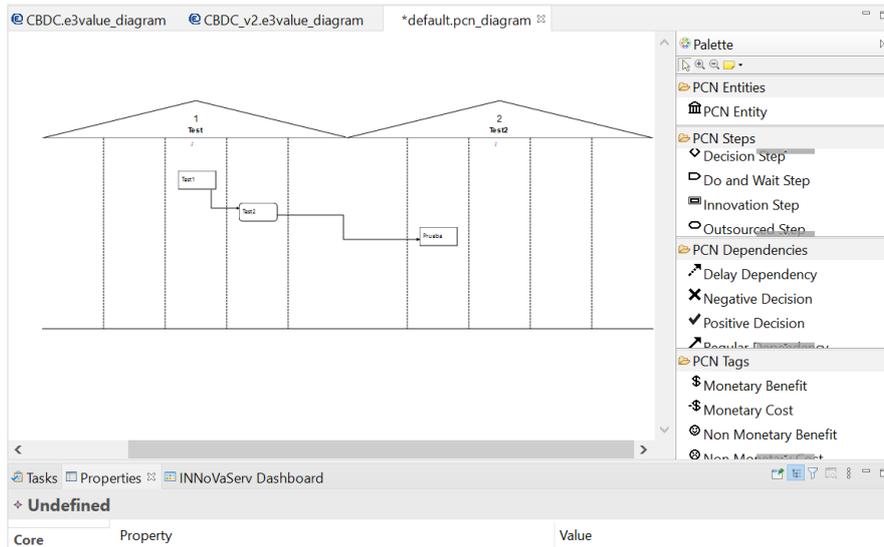


Figura 3-17. Diagramador GMF para la notación PCN.

Generado el editor gráfico, se habrán obtenido diferentes proyectos y clases Java correspondientes a cada uno de los elementos del DSL. Resulta posible, por tanto, modificar dicho código para implementar características no proveídas por GMF. De este modo, una de las modificaciones más habituales durante la construcción de los editores gráficos de esta tesis doctoral, ha sido la relacionada con la redimensión de elementos en el lienzo, ya que por defecto, cuando el usuario modifica el tamaño de una figura, GMF no implementa las acciones

necesarias para redimensionar, en la misma proporción, los componentes que pudiera tener en su interior y que forman parte de la propia figura.

3.3.1.2 Editores Sirius

El desarrollo de editores gráficos mediante Sirius sigue un proceso similar al de Eclipse GMF en lo que se refiere a los pasos a seguir: implementación del metamodelo correspondiente, diseño de la sintaxis concreta, definición de la paleta de herramientas y establecimiento de correspondencias entre los elementos anteriores. Sin embargo, a diferencia de GMF, todos estos componentes se definen en un único modelo con la extensión *.odesign*, a excepción de la sintaxis abstracta, que permanece en un metamodelo Ecore. A modo de ejemplo, la Figura 3-18 muestra un extracto del metamodelo BPMN elaborado mediante Sirius.

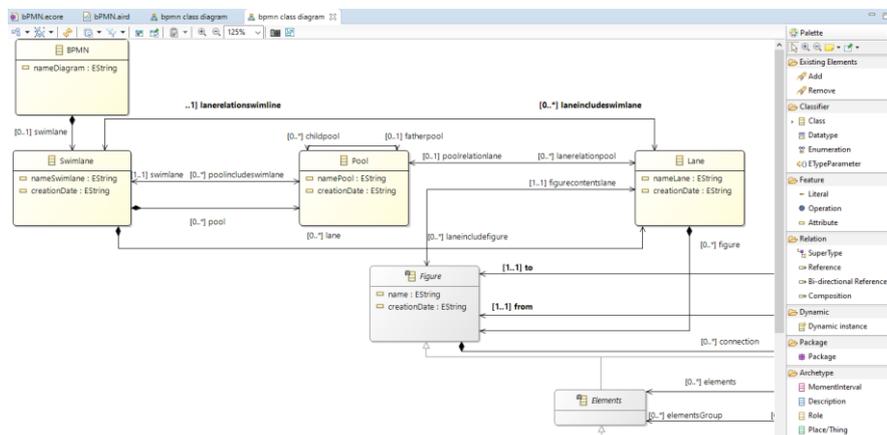


Figura 3-18. Extracto del metamodelo Ecore para BPMN elaborado con Sirius.

Una ventaja de Sirius respecto a GMF, es que permite probar en tiempo real los cambios efectuados sobre el modelo *.odesign*. Para ello, la construcción del editor ha de llevarse a cabo en una segunda instancia de Eclipse que utiliza el metamodelo anterior como un *plug-in* simulado. La siguiente actividad, por tanto, consiste en la definición de la sintaxis concreta del DSL. La Figura 3-19 muestra la definición gráfica de una tarea BPMN, que consta de un rectángulo en cuyo interior se sitúa el icono correspondiente al tipo de tarea. Como puede observarse, otra de las ventajas de Sirius, consiste en que permite definir estilos condicionales para la representación gráfica.

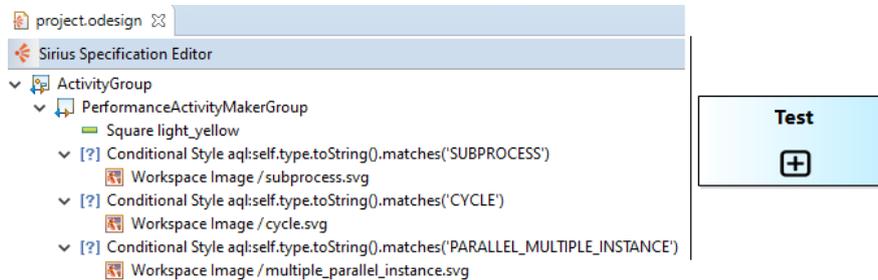


Figura 3-19. Definición gráfica de acciones BPMN con Sirius.

Respecto a la paleta de herramientas, de nuevo se define de manera similar a GMF, con la particularidad de que Sirius permite asociar a cada herramienta el elemento del metamodelo y figura gráfica correspondientes. De este modo, se evita el proceso de crear un modelo específico para la definición de correspondencias. La Figura 3-20 muestra, a modo de ejemplo, la asociación entre la herramienta para crear tareas BPMN y su correspondiente clase del metamodelo y representación gráfica.

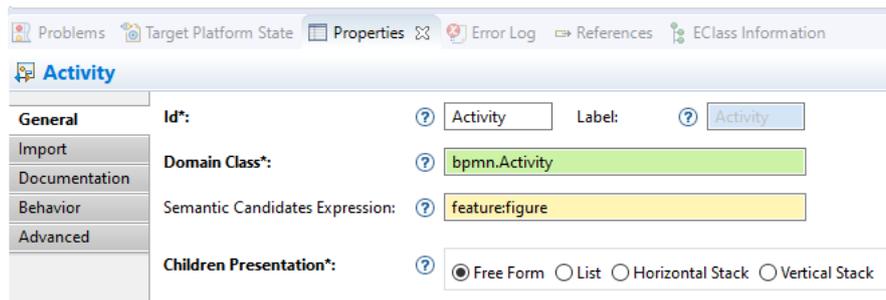


Figura 3-20. Creación de la paleta de herramientas en Sirius.

Una vez definida la sintaxis abstracta, paleta de herramientas y correspondencias en el modelo *.odesign*, Sirius genera automáticamente el editor gráfico para el DSL correspondiente, además del código Java que lo implementa. Por lo tanto, al igual que con GMF, es posible realizar modificaciones sobre este código para implementar nuevas funcionalidades o modificar las ya existentes. A título ilustrativo, la Figura 3-21 muestra el resultado del diagramador Sirius para la notación BPMN.

Por último, cabe mencionar que durante la definición de la paleta de herramientas y sintaxis abstracta, Sirius también permite implementar validaciones y restricciones para los distintos elementos mediante el lenguaje Aceleo, así como definir diferentes modelos *.odesign* para un mismo metamodelo, de tal

forma que se obtienen diferentes sintaxis concretas para una misma sintaxis abstracta.

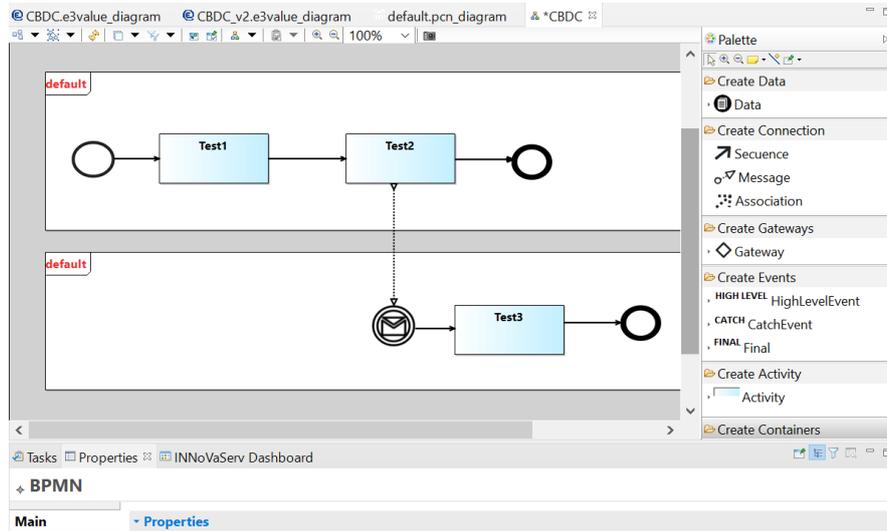


Figura 3-21. Diagramador Sirius para la notación BPMN.

3.3.2 Puentes tecnológicos

Siguiendo la arquitectura conceptual de la Figura 3-2, una vez implementados los editores gráficos para cada uno de los DSLs soportados por INNoVaServ, el siguiente paso consiste en el desarrollo de puentes tecnológicos que permitan establecer correspondencias entre las distintas notaciones. En esta línea, las siguientes subsecciones muestran los detalles más relevantes respecto a la implementación de tales correspondencias y de un panel de control para la gestión de modelos.

3.3.2.1 Transformaciones de modelos

Dado que inicialmente INNoVaServ soporta cinco notaciones diferentes, habría lugar a realizar veinte transformaciones que las conecten (combinaciones dos a dos). 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 cinco notaciones contempladas, con la menor pérdida de información posible. Así, las transformaciones implementadas son las que aparecen en la Figura 3-22. Además, este subconjunto de transformaciones, sin excluir otras posibilidades, permite

diseñar un servicio partiendo de una visión global (áreas de negocio), de la que pueden especificarse los detalles de implementación a través de distintos niveles de abstracción, gracias a las diferentes notaciones de modelado de proceso.

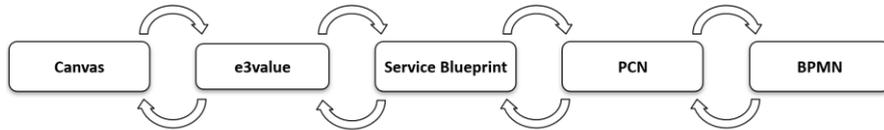


Figura 3-22. Transformaciones directas implementadas en INNOVaServ.

Por otro lado, para recoger las relaciones entre los modelos involucrados en cada transformación, se ha optado por la utilización del metamodelo de trazas (o relaciones) genérico mostrado en la Figura 3-23. De este modo, cuando se ejecute cualquiera de las posibles transformaciones, además del modelo de destino, se generará el modelo de trazas correspondiente.

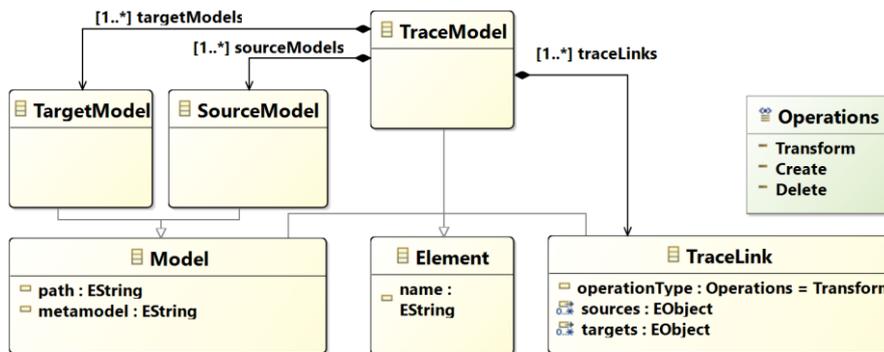


Figura 3-23. Metamodelo de trazas genérico.

Por su parte, ETL (sección 2.4.5) 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, para lo que se ha tomado como base el análisis de correspondencias de la sección 3.2.3. 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. La Figura 3-24 muestra un extracto de la implementación de transformaciones entre las notaciones Canvas y e³value. Como puede observarse, a partir de un modelo Canvas, se genera el modelo e³value y el correspondiente modelo de trazas.

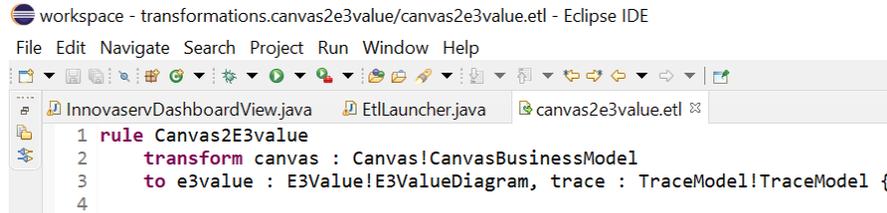


Figura 3-24. Detalle de implementación ETL.

Por último, para visualizar y gestionar los modelos involucrados en cada transformación, se ha integrado en el conjunto de INNoVaServ la herramienta Modelink (sección 2.4.5), un editor multipanel que, como muestra la Figura 3-25, permite visualizar los modelos origen y destino junto al de trazas, además de permitir editar las relaciones existentes.

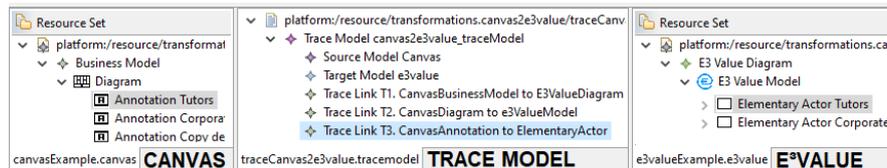


Figura 3-25. Uso de Modelink para la gestión de modelos.

3.3.2.2 Panel para la gestión de modelos

Una vez implementadas las distintas transformaciones entre las notaciones soportadas, lo cierto es que su uso no resulta especialmente intuitivo ya que deben lanzarse manualmente utilizando las opciones de configuración de ejecución que proporciona Eclipse. Dado el perfil de los potenciales usuarios del entorno, y aunque se trate de un prototipo para tareas de investigación, resulta necesario hacer un esfuerzo por mejorar la usabilidad de la herramienta. En esta misma línea, conviene además que los usuarios dispongan de una forma intuitiva y ágil de gestionar los diferentes modelos utilizados en el contexto de un proyecto de diseño de servicios.

Para dar respuesta a las dos necesidades apuntadas en el párrafo anterior, se ha incorporado en INNoVaServ un panel de control o *dashboard* que permite gestionar los diferentes modelos de un proyecto de diseño de servicios. Tal y como ilustra la Figura 3-26, el desarrollo del panel ha sido inspirado por el proporcionado por Eclipse GMF para trabajar con los modelos implicados en el desarrollo de un editor gráfico (ver Figura 2-18). El panel permite la selección y borrado de modelos en distintas notaciones, así como la generación de modelos en una determinada notación a partir de modelos construidos con otra notación.

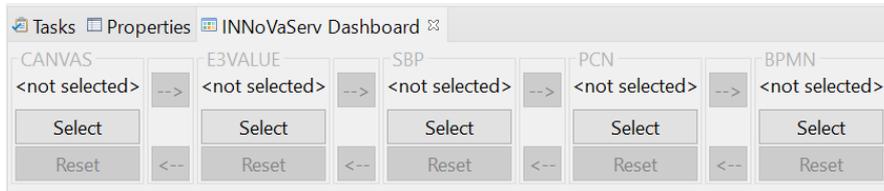


Figura 3-26. Panel de control de modelos de INNoVaServ.

Para la construcción del panel se ha desarrollado un nuevo *plug-in* a partir de la plantilla *View contribution using 4.x API*, lo que ha permitido incorporar una nueva vista al entorno utilizando SWT. El *plug-in* resultante se ha incorporado en el conjunto del entorno de INNoVaServ, si bien es posible su descarga e instalación en cualquier otra distribución de Eclipse.

Como se observa en la Figura 3-26, el panel de control está dividido en pequeños bloques correspondientes a cada una de las notaciones soportadas. El sistema permite seleccionar modelos existentes en un proyecto para cada una de dichas notaciones. Una vez cargado un modelo, se activan los controles que permiten, tomándolo como entrada, lanzar la transformación correspondiente para generar otro modelo en una notación distinta. De este modo, la usabilidad de las transformaciones se ve sustancialmente mejorada respecto a su funcionamiento original.

Por último, cabe mencionar que el panel de control se ha desarrollado siguiendo unos principios mínimos de modularidad, de tal forma que resulta relativamente sencillo añadir bloques y controles adicionales para nuevas notaciones soportadas por INNoVaServ.

3.3.3 Validación de modelos

Además de las validaciones y correcciones automáticas de modelos ya mencionadas, implementadas con el lenguaje ETL de Epsilon y Acceleo en el caso del editor Sirius, se ha trabajado en la validación formal de algunas de las notaciones soportadas por INNoVaServ junto con otros grupos de investigación de distintas universidades.

De esta forma, junto a Elena Gómez-Martínez y Juan de Lara, ambos pertenecientes al grupo de investigación MISO²³ de la Universidad Autónoma de Madrid, se ha llevado a cabo un trabajo para la validación formal de modelos PCN [158]. Dicho estudio toma como punto de partida la falta de soporte tecnológico

²³ <http://miso.es/index.html>

para la validación formal de la notación PCN. Dado que otras notaciones más formales, como son las redes de Petri, poseen una amplia variedad de herramientas para su validación, el trabajo propone la transformación de modelos PCN a redes de Petri estocásticas para facilitar la validación de la primera notación.

Como hechos reseñables de este trabajo en lo que respecta a la notación PCN, a su metamodelo (Figura 3-6) y al listado de elementos gráficos mostrado en la Tabla 2-4, se incorporaron algunos elementos formales referentes a tiempos y probabilidades de los procesos representados. Los elementos añadidos, que no están recogidos en la especificación original de Sampson [24], pueden consultarse en la Tabla 3-3.

Tabla 3-3. Elementos gráficos adicionales de la notación PCN.

<i>Elemento</i>	<i>Representación</i>	<i>Descripción</i>
Dependencia positiva / negativa		Dependencias para establecer formalmente las dependencias positivas y afirmativas en un paso de decisión.
Etiqueta de probabilidad		Etiqueta para establecer formalmente las probabilidades de cada dependencia en un paso de decisión.
Etiqueta de tiempo		Etiqueta para establecer formalmente el tiempo de espera en un paso de espera.
Etiqueta de sincronización		Etiqueta que, situada junto a cualquier acción, indica que se tienen que completar todas las dependencias entrantes antes de su ejecución.

Por otro lado, en colaboración con Montserrat Estañol, Xavier Oriol y Ernest Teniente, todos del grupo de investigación IMP²⁴ de la Universitat Politècnica de Catalunya, se trabajó en la validación formal de modelos Service Blueprint por medio de técnicas de simulación [157]. A grandes rasgos, en este trabajo se propone dotar a los modelos Service Blueprint de una capa de información formal, utilizando UML para representar el modelo de datos subyacente, para posteriormente generar un conjunto de reglas semánticas simulables por medio de la librería OpExec²⁵, que es capaz de detectar violaciones de restricciones definidas previamente.

²⁴ <https://imp.upc.edu/es>

²⁵ <https://www.essi.upc.edu/~xoriol/opexec/>

Validación

De acuerdo a las directrices del método de investigación *Design Science* seguido en esta tesis doctoral, una de las actividades fundamentales del ciclo de diseño consiste en la validación del artefacto a construir. En este sentido, el presente capítulo describe la validación de la propuesta tecnológica presentada anteriormente.

Para ello, la sección 4.1 presenta el protocolo de validación llevado a cabo, mientras que la sección 4.2 describe un caso de estudio con el objetivo de ilustrar cómo puede llevarse a cabo el modelado de un nuevo servicio por medio de INNoVaServ. Por su parte, la sección 4.3 presenta un estudio empírico realizado con el fin de evaluar la calidad del entorno desarrollado y, de este modo, obtener la información necesaria para tratar de dar respuesta a la hipótesis de partida de esta tesis doctoral. Por último, la sección 4.4 resume las principales conclusiones obtenidas a partir del conjunto del proceso de validación.

4.1 Protocolo de validación

Siguiendo la metodología *Design Science* vista en la sección 1.4.1, y de acuerdo a la definición de los distintos ciclos propuestos por Hevner [53] y mostrados en la Figura 1-5, el ciclo de diseño debe iterar entre la construcción del artefacto en cuestión y la evaluación del mismo, volviendo, en caso necesario, a las tareas de diseño y construcción.

En la línea anterior, resulta por tanto necesario diseñar experimentos que permitan validar las teorías propuestas, identificar los escenarios en los que funcionan correctamente, conocer sus limitaciones y descubrir posibles puntos de mejora [58]. Tres de los métodos empíricos más utilizados para lograr estos objetivos, consisten en los casos de estudio, las encuestas y los experimentos.

De este modo, para la validación del artefacto propuesto en esta tesis doctoral, se han combinado los tres métodos anteriores de tal forma que se ha utilizado un caso de estudio y un experimento, que incluye una serie de encuestas con las que extraer la información necesaria para tratar de verificar o refutar la hipótesis de partida de esta tesis doctoral.

Respecto a los casos de estudio, estos consisten en un método de investigación empleado en distintos ámbitos de la ciencia con el objetivo de ampliar la experiencia o reforzar lo que ya se conoce partiendo de investigaciones previas [163]. Como se menciona en la sección 1.4.3, el caso de estudio llevado a

cabo en esta tesis doctoral ha seguido las directrices propuestas en [61] y [65], siendo las fases más destacadas las del diseño del caso de estudio (definición de objetivos, preguntas de investigación y planificación general), la recolección de datos, el análisis e interpretación de éstos, y la presentación de informes.

Por su parte, como se vio en la sección 1.4.4, los estudios empíricos o experimentos son un método de investigación basado en la observación de evidencias con el objetivo de validar o refutar una hipótesis determinada [164]. Además de un caso de estudio, la presente tesis doctoral hace uso de un estudio empírico como parte de la validación del artefacto propuesto. Para ello se han seguido las guías descritas en [66], además de emplear una serie de encuestas, que son investigaciones empíricas en las que los sujetos de estudio constituyen una muestra representativa de la población a la que pertenecen [165].

Con todo lo anterior, las siguientes secciones describen el caso de estudio empleado para la validación del artefacto propuesto en esta tesis doctoral, para continuar con el estudio empírico o experimento con el que reforzar tal validación mediante el uso de encuestas y, por último, presentar algunas de las principales conclusiones extraídas a partir de este proceso de validación.

4.2 Caso de estudio

Dado que según Wieringa [52] *Desing Science* consiste en el diseño e investigación de artefactos en un contexto determinado, resulta razonable utilizar el método de casos de estudio para la validación de tales artefactos, ya que éstos permiten el análisis del comportamiento de una propuesta dentro de su contexto real [61]. Por lo tanto, en esta tesis doctoral se emplea un caso de estudio de laboratorio para la validación de la solución tecnológica desarrollada.

En concreto, el caso de estudio llevado a cabo en esta tesis doctoral, se ha elaborado siguiendo las directrices propuestas en [62] y [63] y descritas en la sección 1.4.3. Además, la Figura 1-8 muestra un resumen gráfico de las etapas seguidas para la validación mediante casos de estudio del artefacto construido, que pueden resumirse en la definición del caso de estudio, definición del protocolo de recogida de datos, la ejecución del caso de estudio, y la elaboración de informes donde recoger las principales conclusiones. De este modo, las siguientes subsecciones realizan un recorrido por cada una de las etapas mencionadas.

4.2.1 Definición del caso de estudio

En el caso de estudio de laboratorio diseñado para la validación del artefacto construido en esta tesis doctoral, se propone el diseño de un servicio que dé respuesta a una problemática determinada en una organización, que puede enfocarse desde la metodología del diseño de servicios.

En concreto, el problema para el que se pretende diseñar una solución viene provocado por el elevado incremento de nuevos estudiantes en una determinada universidad pública en los últimos seis años. Este aumento de alumnos ha generado a su vez un incremento en el número de Trabajos Fin de Grado (TFGs), lo que deriva en una dificultad para gestionar el alto número de estos trabajos, ya que entre otras razones, son varias las partes que se ven involucradas y han de coordinarse en esta tarea (tutores, personal de administración, Unidad de TFGs, etc.). Para facilitar tal gestión, la Unidad de TFGs ha propuesto el uso de una aplicación informática que centralice la acción de todos los participantes en los procesos involucrados.

De este modo, se propone hacer uso del entorno de modelado INNoVaServ para el diseño del servicio correspondiente, y la consecuente construcción de los modelos involucrados en cada una de las etapas del modelo del doble diamante del diseño de servicios presentado en la sección 2.1.1.

4.2.2 Protocolo de recogida de datos

En el contexto de una validación mediante casos de estudio, el protocolo de recogida de datos consiste fundamentalmente en la definición de una serie de cuestiones que han de ser respondidas tras la ejecución del caso de estudio, haciendo uso para ello de la información recopilada durante el mismo. Por lo tanto, en esta etapa también se deben definir las fuentes de datos que se utilizarán para dar respuesta a tales preguntas.

En el caso de esta tesis doctoral, la validación tiene el objetivo de comprobar el correcto funcionamiento de las diferentes características integradas en el entorno de modelado INNoVaServ que permiten, de manera conjunta, diseñar servicios mediante la elaboración de modelos de negocio y proceso característicos de las distintas etapas del modelo del doble diamante para el diseño de servicios.

En la línea anterior, a continuación se enumeran las preguntas del caso de estudio (PCE) a las que se dará respuesta junto al análisis del mismo en la sección 4.2.4:

- **PCE-1:** ¿Ha sido posible trabajar de manera sencilla e intuitiva con todas las notaciones soportadas por INNoVaServ?
- **PCE-2:** ¿En qué medida reducen el esfuerzo necesario para el diseño de un servicio las transformaciones soportadas por INNoVaServ?
- **PCE-3:** ¿En qué medida se aproximan a la realidad los modelos resultantes de cada transformación soportada por INNoVaServ?
- **PCE-4:** ¿Resulta transparente la integración de editores gráficos elaborados mediante distintas herramientas (Eclipse GMF, Sirius)?
- **PCE-5:** ¿Ha sido posible poner en práctica con INNoVaServ el modelo del doble diamante característico del diseño de servicios?

Respecto a las fuentes de datos que se utilizarán para recopilar las evidencias con las que responder a las preguntas anteriores, se ha decidido utilizar las siguientes:

- Observación directa de la ejecución del caso de estudio.
- Modelos gráficos utilizados durante el desarrollo del servicio.
- Modelos textuales subyacentes a los modelos gráficos.
- Modelos de trazas generados en las transformaciones.

4.2.3 Ejecución del caso de estudio

Siguiendo el modelo del doble diamante para el diseño de servicios (Figura 2-1) durante la ejecución del caso de estudio, en primer lugar se llevó a cabo la fase de descubrimiento del problema con el objetivo de visualizar y producir una comprensión común del contexto del mismo. Para ello, se elaboró primero el modelo de negocio Canvas que puede visualizarse en la Figura 4-1. En él se muestran las diferentes propuestas, actividades, participantes y recursos relacionados con el actor principal de este escenario: la Unidad de TFGs, que se encarga de la correcta gestión de los trabajos de fin de grado de la universidad. A modo de ejemplo, nótese que los socios clave de esta unidad son los tutores y dos organizaciones que proporcionan sistemas software con los que gestionar los distintos trabajos.

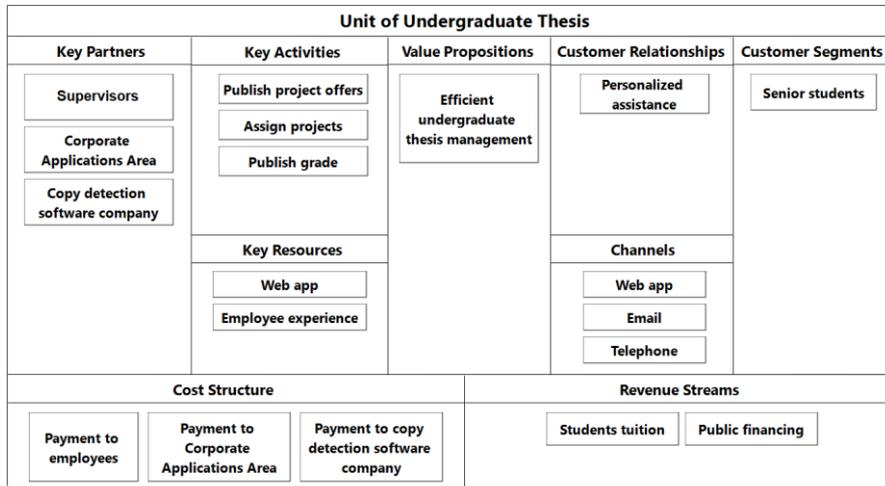


Figura 4-1. Modelo Canvas del caso de estudio Unidad TFG elaborado con INNoVaServ.

Posteriormente, el modelo Canvas se utilizó para generar, mediante las transformaciones semiautomáticas de INNoVaServ, una vista parcial del correspondiente modelo e³value que tuvo que ser completado manualmente (Figura 4-2). Este modelo aporta información adicional sobre la propuesta de valor, como los intercambios de valor que tienen lugar entre los diferentes actores. Por lo tanto, los modelos de la Figura 4-1 y Figura 4-2 aportan dos perspectivas complementarias del contexto del problema: mientras el modelo de negocio Canvas proporciona una visión global de la organización (en este caso la Unidad de TFGs), el modelo e³value permite especificar las cadenas de valor y las operaciones de servicio que tienen lugar.

Además del modelo e³value, la transformación ejecutada anteriormente también genera un modelo de trazas (Figura 4-3) que recoge las correspondencias entre los elementos de los modelos de origen y destino (Canvas y e³value). Obsérvese, por ejemplo, que los socios clave del modelo Canvas se transforman en actores del modelo e³value. Asimismo, muchos de los objetos de valor que aparecen en el modelo e³value, se generan a partir de anotaciones del modelo Canvas. Por ejemplo, a cambio de un pago al Área de Aplicaciones Corporativas (uno de los elementos de la estructura de costes del modelo Canvas), esta proporciona una aplicación web a la Unidad de TFGs. Por su parte, esta aplicación corresponde a uno de los elementos identificados como recurso clave en el modelo Canvas.

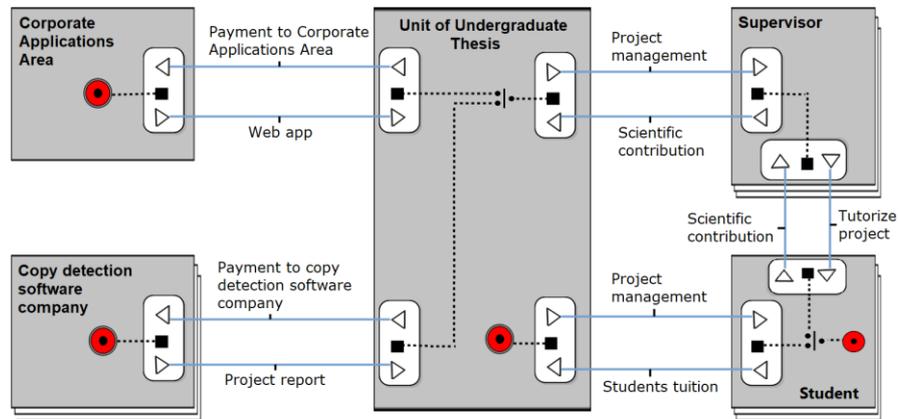


Figura 4-2. Modelo e³value del caso de estudio Unidad TFG elaborado con INNoVaServ.

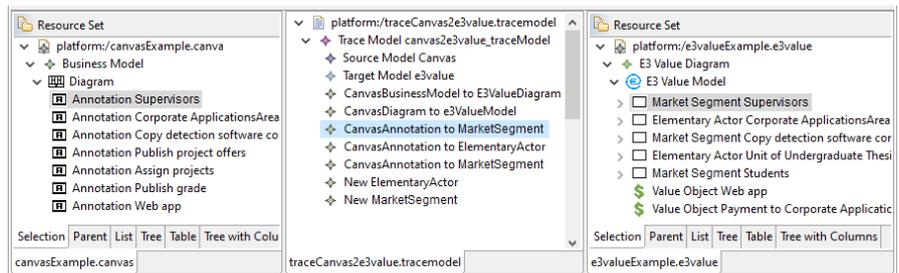


Figura 4-3. Modelo de trazas generado por INNoVaServ en la conversión de Canvas a e³value.

Tras explorar el espacio del problema, el siguiente paso consiste en el diseño de la solución. Para ello se han utilizado las tres notaciones de modelado de procesos de negocio soportadas por INNoVaServ, con las que es posible especificar las operaciones de servicio implicadas en los caminos de dependencia contenidos en el modelo e³value. En esta línea, los modelos definidos con estas notaciones desempeñan un papel clave durante las fases de reflexión e implementación del proceso de diseño de servicios, ya que ayudan a prototipar, desarrollar y probar soluciones.

De este modo, el siguiente modelo que se ha desarrollado con INNoVaServ es el correspondiente a la notación Service Blueprint (Figura 4-4). Dicho modelo se definió durante la fase de reflexión del caso de estudio con el objetivo de diseñar un servicio que ayudara en la gestión del ciclo de vida de cada TFG, el cual hace uso de una aplicación de apoyo. Nótese que dicho modelo recoge las interacciones entre el estudiante (consumidor del servicio) y su tutor (proveedor del servicio), así como los distintos procesos internos y de apoyo.

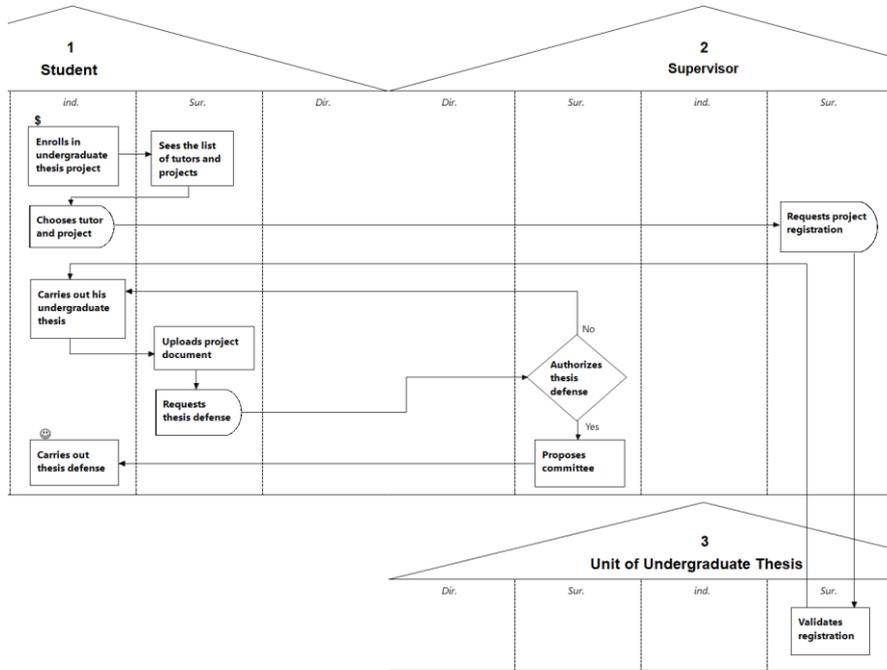


Figura 4-5. Modelo PCN del caso de estudio Unidad TFG elaborado con INNOVaServ.

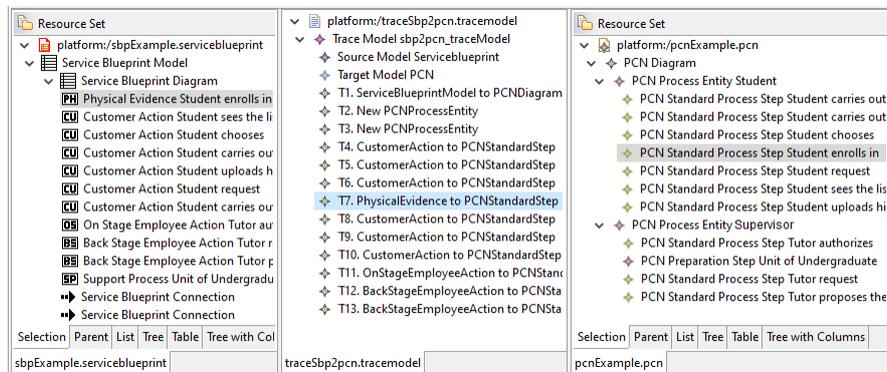


Figura 4-6. Modelo de trazas generado por INNOVaServ en la conversión de Service Blueprint a PCN.

Por último, en un intento por acercar el diseño de la solución a un nivel más cercano a su implementación, se generó la vista parcial del modelo BPMN a partir del PCN, debiendo, una vez más, completar manualmente dicho modelo tomando las decisiones de diseño correspondientes. El resultado final puede observarse en la Figura 4-7.

Adicionalmente, la Figura 4-8 muestra el modelo de trazas correspondiente a la generación del modelo BPMN. Como puede observarse, todas las entidades PCN son convertidas en *pools* BPMN, que contienen todas las actividades y compuertas correspondientes. Del mismo modo, todas las secuencias contenidas en el modelo PCN se mantienen en el modelo BPMN generado.

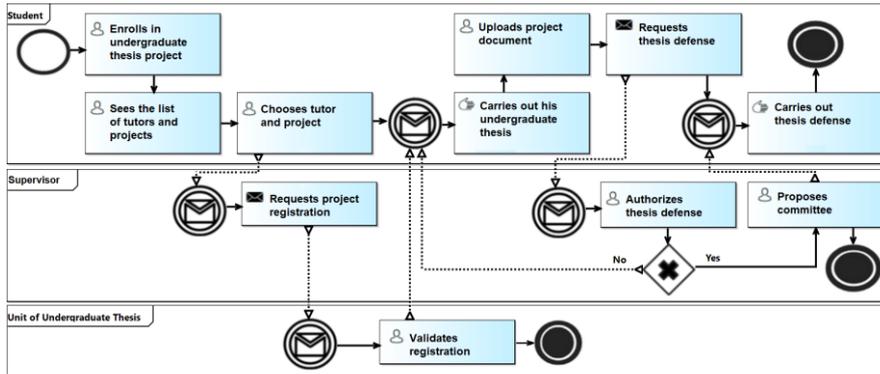


Figura 4-7. Modelo BPMN del caso de estudio Unidad TFG elaborado con INNoVaServ.

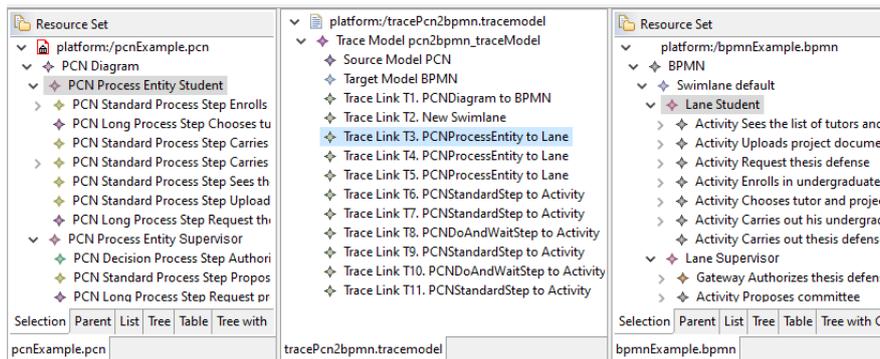


Figura 4-8. Modelo de trazas generado por INNoVaServ en la conversión de PCN a BPMN.

4.2.4 Análisis del caso de estudio

Una vez ejecutado el caso de estudio, a continuación se presentan las respuestas a cada una de las preguntas planteadas en la definición del protocolo de recogida de datos (sección 4.2.2):

- **PCE-1:** ¿Ha sido posible trabajar de manera sencilla e intuitiva con todas las notaciones soportadas por INNoVaServ?

Sí. INNoVaServ permite construir modelos por medio de editores gráficos en los que el usuario básicamente debe seleccionar elementos de la paleta de

herramientas y la posición del lienzo en la que desea posicionarlos, lo que resulta sencillo e intuitivo por ser el modo de trabajo habitual en este tipo de soluciones de modelado.

Sin embargo, la complejidad de cada editor está en gran medida relacionada con la propia complejidad de la notación a soportar. De esta forma, los editores para notaciones como Canvas o Service Blueprint (que no poseen una gran cantidad de elementos anidados), ofrecen una mayor usabilidad que los correspondientes a notaciones más complejas (PCN o BPMN), en los que el usuario debe ser mucho más preciso a la hora de posicionar elementos anidados (p.ej. etiquetas de acciones PCN).

Por último, valorando la usabilidad de la herramienta en general, en esta fase se observó que la ejecución de las transformaciones semiautomáticas no resultaba del todo intuitiva, debiendo lanzarse desde diferentes paneles de control de Eclipse, por lo que se decidió implementar la funcionalidad correspondiente en el panel de control de modelos, la cual se describe en la sección 3.3.2.2.

- **PCE-2:** ¿En qué medida reducen el esfuerzo necesario para el diseño de un servicio las transformaciones soportadas por INNoVaServ?

La respuesta a esta pregunta se ofrece en términos del tiempo empleado para la elaboración de cada modelo del caso de estudio. De este modo, se ha contrastado el tiempo utilizado para la construcción manual de cada modelo, respecto a la elaboración del mismo modelo mediante la generación semiautomática de vistas parciales (de acuerdo con las transformaciones directas de la Figura 3-22), incluyendo también en este caso el tiempo utilizado para las modificaciones posteriores de la vista parcial. Para evitar contabilizar el tiempo empleado en la toma de decisiones de diseño, este análisis se ha llevado a cabo una vez conocido cuál debe ser el resultado final de cada modelo. De este modo, los resultados obtenidos (medidos en segundos) se muestran en la Tabla 4-1, representando su última fila el ahorro de tiempo que supone crear un modelo a partir de la generación de vistas parciales, respecto a un proceso completamente manual. Nótese asimismo que en este análisis no se han considerado las transformaciones entre modelos de negocio y proceso, ya que como se vio en la sección 3.2.3.3, en estos casos tan solo se produce una estimación del número de modelos de proceso necesarios para representar una propuesta de valor.

Con todo lo anterior, y en base a los resultados mostrados en la Tabla 4-1, es posible afirmar que las transformaciones de modelos soportadas por INNoVaServ, efectivamente suponen una reducción del esfuerzo empleado por el usuario, ya que en el peor de los casos, la generación de vistas parciales permite

un ahorro de aproximadamente el 10% del tiempo dedicado a la construcción de modelos. No obstante, durante este análisis se han identificado una serie de puntos de mejora que permitirían reducir aún más el esfuerzo empleado por el usuario, como por ejemplo la disposición de los elementos gráficos de los diagramas generados por las transformaciones.

Por último, a pesar de que este análisis no considera el tiempo empleado en cuestiones de diseño por tratarse de una variable no controlable, lo cierto es que un valor añadido de la generación de vistas parciales, es la ayuda en la toma de este tipo de decisiones, ya que algunas de las transformaciones interactúan con el usuario a modo de asistente para la construcción del modelo de destino.

Tabla 4-1. Tiempos empleados (en segundos) para la elaboración de los modelos del caso de estudio.

	Canvas	e ³ value	SBP	PCN	BPMN
Manual	329	661	342	505	805
Desde Canvas	-	490	-	-	-
Desde e ³ value	297	-	-	-	-
Desde SBP	-	-	-	478	-
Desde PCN	-	-	296	-	512
Desde BPMN	-	-	-	430	-
% Ahorro	9,73%	25,87%	13,45%	10,10% ²⁶	36,40%

- **PCE-3:** ¿En qué medida se aproximan a la realidad los modelos resultantes de cada transformación soportada por INNoVaServ?

Para dar respuesta a esta pregunta, se ha llevado a cabo un análisis diferencial en el que se ha contrastado el número de elementos (a nivel de metamodelo) que posee cada modelo generado en una transformación (sin modificaciones posteriores), respecto al mismo modelo, pero elaborado manualmente, considerando este último como completo o más cercano a la realidad que representa. Esta comparación se ha llevado a cabo para las transformaciones directas de la Figura 3-22, a excepción de las que involucran modelos de negocio y proceso, ya que como se vio en la sección 3.2.3.3, en estos casos tan solo se estima el número de modelos de proceso necesarios para

²⁶ Valor medio para las dos transformaciones. Los valores parciales son 5,35% desde SBP y 14,85% desde BPMN.

representar una propuesta de valor. Adicionalmente, además del caso de estudio presentado en la sección actual, en este análisis se han utilizado cuatro casos de laboratorio más con el objetivo de obtener unos porcentajes de completitud medios. Dichos casos son los correspondientes a un servicio de reparto de comida a domicilio, una compañía de alquiler de vehículos, una empresa de alquiler de bicicletas y un instituto de neuropsicología.

Con todo lo anterior, la Tabla 4-2 muestra el porcentaje de elementos generados por la herramienta para cada uno de los casos de estudio. A la luz de estos datos, es posible concluir que, en general, las vistas parciales generadas por la herramienta se aproximan bastante a la realidad: de seis posibles escenarios, en cuatro de ellos el porcentaje medio de elementos generados supera el 67%, mientras que los otros dos están entre el 36% y el 45%. Finalmente, cabe mencionar que una de las limitaciones de este estudio consiste en que el análisis se ve condicionado por la cantidad de información recogida en los modelos de origen, por lo que, si éstos fuesen incompletos, las transformaciones serían menos efectivas.

Tabla 4-2. Porcentaje de elementos generados por INNoVaServ en cada transformación.

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

- **PCE-4:** ¿Resulta transparente la integración de editores gráficos elaborados mediante distintas herramientas (Eclipse GMF, Sirius)?

Parcialmente. De cara al usuario, si bien no se aprecia diferencia alguna durante las etapas de modelado entre los editores GMF y el construido con Sirius, lo cierto es que existen pequeñas diferencias en las acciones a llevar a cabo a la hora de crear nuevos modelos o generar la representación gráfica de éstos.

Por otro lado, de cara al sistema interno de INNoVaServ, esta integración resulta completamente transparente, ya que las tareas de gestión de modelos se

llevan a cabo únicamente con los modelos EMF subyacentes, siendo algo completamente independiente del editor gráfico utilizado.

- **PCE-5:** ¿Ha sido posible poner en práctica con INNoVaServ el modelo del doble diamante característico del diseño de servicios?

Parcialmente. Como se ha podido comprobar durante la ejecución del caso de estudio, de manera general, es posible el diseño y modelado de un servicio con INNoVaServ, realizando un recorrido a través de los modelos involucrados en las distintas etapas del modelo del doble diamante tal y como viene definido en la sección 2.1.1. Sin embargo, durante la transición entre las etapas de exploración del problema y diseño de la solución, si bien la herramienta puede generar el número de modelos de proceso necesarios para representar la propuesta de valor, lo cierto es que sería deseable una mayor conexión entre ambas etapas, por ejemplo, creando enlaces concretos entre los diferentes elementos de los modelos de proceso y negocio.

4.3 Estudio empírico

En el capítulo 3 se describieron las principales características del entorno tecnológico construido a lo largo de esta tesis doctoral, que tiene como propósito el diseño y modelado de servicios en el contexto del diseño de servicios, por medio del soporte a distintas notaciones de negocio y proceso y su integración mediante diferentes técnicas. Sin embargo, tal y como se menciona en el estado del arte de esta tesis doctoral (sección 2.2), existen muchas otras soluciones que dan soporte aislado, en mayor o menor medida, a algunas de las notaciones integradas en INNoVaServ.

La intuición inicial hace pensar que el proceso de modelado de un servicio será más eficiente en un entorno que permite trabajar de manera integrada con distintas técnicas, en lugar de tener que recurrir a diferentes herramientas para cada notación. Con el objetivo de comprobar la veracidad de estas impresiones, se ha llevado a cabo un estudio empírico con el que obtener datos para contrastar el proceso de desarrollo de un servicio mediante INNoVaServ respecto a otras alternativas.

Dado que el entorno construido en esta tesis doctoral consiste en un prototipo funcional con determinadas limitaciones de usabilidad, el mencionado estudio empírico se llevó a cabo en el contexto de un *workshop* asistido en el que participaron diferentes investigadores y profesionales relacionados con el ámbito del diseño de servicios. Asimismo, para su ejecución se siguieron unas directrices

que marcan la necesidad de describir el enfoque seguido, el material y método empleado, y el análisis de los resultados obtenidos [166].

4.3.1 Planificación

Durante el estudio empírico descrito en esta sección, se han seguido las guías propuestas en [167]. En esta línea, a continuación se enumeran los tres componentes principales del estudio, que serán descritos con mayor detalle en las siguientes subsecciones:

- Los sujetos que participaron en el estudio empírico eran nueve personas repartidas entre profesionales e investigadores relacionados con el ámbito del diseño de servicios, existiendo diferentes niveles de experiencia previa de modelado.
- El material utilizado consistió en la definición de una especificación de servicio a modelar, una serie de manuales sobre cada notación soportada por INNoVaServ, así como una guía de uso de la herramienta y la asistencia personal proporcionada durante la ejecución. Asimismo, se proveyó a los sujetos de una versión de INNoVaServ sobre la que llevar a cabo el modelado del servicio propuesto, y dos cuestionarios: uno sobre conocimientos previos, y otro con el que evaluar el proceso de modelado llevado a cabo mediante INNoVaServ.
- Por último, se ejecutó un análisis con el objetivo de identificar conocimientos previos de los sujetos, y otro con el que valorar el proceso de desarrollo de servicios con INNoVaServ, para posteriormente procesar las respuestas de los distintos cuestionarios.

4.3.2 Sujetos

Como ya se ha mencionado, este estudio empírico contó con la participación de nueve profesionales e investigadores relacionados con el ámbito del diseño de servicios, impartiendo algunos de ellos diferentes materias en el Grado en Ciencia, Gestión e Ingeniería de Servicios de la Universidad Rey Juan Carlos. Dado que el artefacto a analizar consiste en un prototipo para tareas de investigación centrado en la funcionalidad y con determinadas deficiencias de usabilidad, no se ha considerado su ejecución con un número mayor de participantes.

Con el objetivo de lograr una mayor representatividad de los resultados y evitar sesgos como, por ejemplo, que un participante esté acostumbrado a trabajar

con una determinada herramienta, dentro del grupo de sujetos del estudio se han seleccionado a investigadores con diferentes niveles de conocimientos relacionados, así como de experiencia previa en el uso de soluciones tecnológicas de modelado.

Por otro lado, el principal requisito de los participantes ha consistido en poseer unos conocimientos mínimos de diseño de servicios y modelado. Cabe mencionar que no se han considerado otras cuestiones, como por ejemplo tener conocimientos sobre el IDE de Eclipse, ya que, por tratarse de un prototipo de investigación, los participantes del experimento han contado con la asistencia del doctorando con los objetivos de resolver las dudas de instalación e identificar errores o problemas de uso a mejorar.

4.3.3 Material

Debido a la situación sanitaria causada por la pandemia de la COVID-19, el estudio empírico se llevó a cabo de manera remota a través de la herramienta Microsoft Teams. Por este motivo, cada participante contó con sus propios medios informáticos en los que se instalaron sendas distribuciones de INNoVaServ basadas en Eclipse IDE 2020-12. Para su correcto funcionamiento, previamente se configuró en cada equipo la versión 11.0.9 de *Java Development Kit* (JDK).

El primer material dado a los participantes, consistió en un cuestionario sobre conocimientos previos de modelado de negocio y proceso, experiencia de uso con herramientas relacionadas, y preferencias personales en este contexto. En el Apéndice A puede consultarse este cuestionario.

Una vez completado el cuestionario anterior, para su consulta en caso de duda, se distribuyó entre los participantes un manual sobre cada una de las notaciones soportadas por INNoVaServ similar a lo descrito en la sección 2.1.2. Junto a este manual, se entregó otro documento con la descripción del caso de estudio a realizar, que puede consultarse en el Apéndice A. En este enunciado, básicamente se pide a los participantes el rediseño de un servicio por medio de las distintas notaciones soportadas por INNoVaServ, empleando para ello este entorno integrado de modelado. Cabe destacar que en este experimento no se esperaba ninguna solución específica del caso de estudio, sino que simplemente se perseguía que los participantes tuviesen contacto directo con INNoVaServ, para así poder analizar diferentes factores de la herramienta y poder establecer una comparativa con las alternativas previamente conocidas. Por último, en esta fase también se entregó a los participantes una guía de uso de las principales características de INNoVaServ, que puede consultarse en el Apéndice B.

Una vez que los participantes del experimento hubieron trabajado en el caso de estudio planteado, completaron un cuestionario final en el que expresaron su opinión sobre diferentes aspectos de INNoVaServ en lo que se refiere a valoraciones globales de la herramienta, características concretas de la misma, o problemas encontrados durante su utilización.

4.3.4 Análisis y resultados

Una vez ejecutado el experimento empírico, se llevó a cabo el análisis de los datos obtenidos en los dos cuestionarios completados por los participantes. De este modo, en primer lugar se analizaron las respuestas dadas al cuestionario inicial (Apéndice A), que tenía por objetivo conocer los conocimientos previos de los sujetos, herramientas de modelado de negocio y/o proceso utilizadas, y preferencias personales sobre este tipo de soluciones tecnológicas.

La Tabla 4-3 muestra un resumen de las respuestas dadas al cuestionario inicial. Como puede observarse, los datos de la primera pregunta demuestran que las notaciones más populares entre los participantes del experimento, son el modelo de negocio Canvas y el modelo de proceso BPMN, seguidos de cerca por la notación Service Blueprint. Menos conocidas son las notaciones e³value y PCN, lo cual es compatible con las conclusiones extraídas en la SLR llevada a cabo en esta tesis doctoral (2.2.8). Cabe mencionar que otra de las notaciones indicadas por algunos de los participantes, es el modelo de negocio *Lean Canvas*.

En la misma línea es posible encontrar los resultados correspondientes a la segunda pregunta, que muestran que la notación más utilizada entre los participantes es BPMN, seguida de Canvas y Service Blueprint. Analizando los resultados dados a esta pregunta y la anterior, es posible afirmar que el conjunto de notaciones soportado por INNoVaServ, tiene una popularidad aceptable entre los profesionales del diseño de servicios, si bien podría considerarse añadir nuevas notaciones como *Lean Canvas*.

En la tercera pregunta, se pidió a los participantes que seleccionasen las herramientas conocidas entre las alternativas a INNoVaServ analizadas en la SLR de esta tesis doctoral (Tabla 2-11), si bien se permitía añadir otras no consideradas en dicho análisis. Cabe mencionar que en el resumen de la Tabla 4-3, por limitaciones de espacio, solo se encuentran aquellas herramientas utilizadas por cuatro o más participantes. De este modo, es posible comprobar que *Draw.io* y *MS Visio* son las herramientas más utilizadas, posiblemente debido a su naturaleza genérica o el acceso web y gratuito en el caso de *Draw.io*. Utilizadas, aunque en menor medida, son también *Bizagi* (3), *BonitaSoft* (3), *Miro* (2) o incluso *e³value*

editor (3), probablemente debido a que era la única herramienta disponible para esta notación hasta la construcción del editor integrado en INNoVaServ.

En la cuarta pregunta se pedía valorar del 1 al 5 la utilidad para el diseño de servicios de las herramientas utilizadas, teniendo en cuenta aspectos como la usabilidad, capacidades de exportación, etc. Los resultados mostrados en la Tabla 4-3 para esta pregunta, consisten en la valoración media dada por los participantes que utilizaron la alternativa correspondiente, por lo que una limitación en este sentido es que las valoraciones serán más precisas en aquellos casos que tienen más usuarios. Como puede observarse, las dos herramientas con mayor valoración son *Bpmn.io* y *Canvanizer*, siendo ambas dos soluciones online centradas en una única notación con una interfaz muy cuidada.

En la última pregunta del cuestionario previo, se pedía seleccionar aquellas características que se considerasen necesarias para un entorno de modelado en el contexto del diseño de servicios. Como puede observarse, todos los participantes expresaron que tanto la facilidad de uso, como el soporte al trabajo colaborativo, son características deseables en este tipo de entornos. Por otro lado, cabe destacar que la integración de distintas notaciones en una misma herramienta tuvo una aceptación considerable con seis puntos sobre nueve. Además, entre otras características que mencionaron los participantes, destacan las de ser una solución online (sin instalación y accesible desde cualquier parte), la asistencia durante el proceso de diseño, o la capacidad de exportación en múltiples formatos.

Tabla 4-3. Resumen de resultados del cuestionario previo.

Pregunta 1	Canvas	e ³ value	SBP	PCN	BPMN	Otras
<i>Notaciones de modelado de negocio y proceso conocidas</i>	9	4	8	2	9	4
Pregunta 2	Canvas	e ³ value	SBP	PCN	BPMN	Otras
<i>Notaciones de modelado de negocio y proceso utilizadas</i>	8	3	7	1	9	3
Pregunta 3	Canvanizer	Draw.io	Lucidchart	MS Visio	Bpmn.io	
<i>Herramientas de modelado utilizadas</i>	6	8	6	7	6	
Pregunta 4	Canvanizer	Draw.io	Lucidchart	MS Visio	Bpmn.io	
<i>Valoración media (1 a 5) de herramientas utilizadas</i>	4	3,12	3,5	3,57	4,16	
Pregunta 5	Facilidad de uso	No conocimientos previos	Integración de notaciones	Trabajo colaborativo		
<i>Características consideradas más importantes</i>	9	5	6	9		

Tras trabajar con INNoVaServ en el caso de estudio propuesto, los participantes completaron un cuestionario final (Apéndice A) en el que debían valorar diferentes aspectos del entorno de modelado. De manera general, dicha encuesta contenía dos bloques de preguntas: uno sobre características concretas de la herramienta, y otro sobre cuestiones globales de la misma.

La Tabla 4-4 muestra los resultados medios para el primer bloque de preguntas del cuestionario. Como puede observarse, en primer lugar se preguntaba acerca de la facilidad de uso de los diferentes diagramadores de INNoVaServ, destacando por la puntuación otorgada los correspondientes a las notaciones Canvas y Service Blueprint. Este hecho corrobora la respuesta dada a la primera pregunta del caso de estudio (sección 4.2.4), en la que se proponía que la complejidad de uso de cada editor, se ve en gran medida afectada por la propia complejidad de la notación a soportar. Por su parte, los editores para las notaciones PCN y BPMN obtienen puntuaciones más bajas, posiblemente debido a la gran cantidad de elementos de estas técnicas de modelado, o a las diferencias en las acciones a llevar a cabo para crear nuevos modelos en el caso de BPMN.

En la segunda pregunta se pedía valorar la utilidad de las distintas transformaciones soportadas por INNoVaServ. Resulta llamativo que, si bien no existen diferencias demasiado considerables, obtienen mejores puntuaciones aquellas transformaciones que permiten crear un servicio comenzando por una visión global del mismo mediante un modelo Canvas, para especificar después de varias etapas los detalles de implementación con BPMN (ver Figura 3-22). Así mismo, destaca la baja puntuación otorgada a las transformaciones entre modelos de negocio (e³value) y modelos de proceso (Service Blueprint) y viceversa, muy probablemente debido a que el enfoque seguido en esta tesis doctoral, si bien es capaz de estimar el número de modelos resultantes de la transformación, no permite automatizar la generación de elementos específicos como acciones o dependencias.

Por su parte, la tercera pregunta consistía en valorar la cantidad de esfuerzo que las transformaciones soportadas pueden ahorrar al diseñador de servicios. En este caso, si bien las respuestas de los participantes ofrecen resultados aceptables en la mayoría de las transformaciones, vuelven a destacar por su baja puntuación aquellas que involucran al modelo de negocio e³value y al modelo de proceso Service Blueprint.

Respecto a la cuarta pregunta, en la que se pedía valorar la utilidad del panel de control de modelos para el diseño de servicios incluido en INNoVaServ, se obtuvo una valoración media generalmente buena de 3,89 puntos sobre 5.

Finalmente, la quinta pregunta de este bloque pedía valorar la utilidad de las validaciones automáticas de modelos, obteniendo en este caso una valoración media de 2,78 sobre 5, probablemente debido a que dichas validaciones se limitan a descubrir pequeños olvidos como la no asignación de nombres a entidades, o acciones sin conectar. Además, como pudo comprobarse durante el experimento, en algunos casos las soluciones rápidas ofrecidas por las validaciones demoran un tiempo mayor que la reparación manual.

Tabla 4-4. Resultados medios del cuestionario final a preguntas sobre características específicas.

Pregunta 1	Canvas	e ³ value	SBP	PCN	BPMN			
<i>Facilidad de uso de los diagramadores (1 a 5)</i>	4,89	3,78	4,33	3	2,44			
Pregunta 2	Canvas desde e ³ value	e ³ value desde Canvas	e ³ value desde SBP	SBP desde e ³ value	SBP desde PCN	PCN desde SBP	PCN desde BPMN	BPMN desde PCN
<i>Utilidad de las transformaciones (1 a 5)</i>	3,89	4,22	2	2,11	3,78	4,11	3,67	4,67
Pregunta 3	Canvas desde e ³ value	e ³ value desde Canvas	e ³ value desde SBP	SBP desde e ³ value	SBP desde PCN	PCN desde SBP	PCN desde BPMN	BPMN desde PCN
<i>Ahorro de esfuerzo por transformaciones (1 a 5)</i>	4	3,22	1,11	1,55	3,44	3,33	3,33	4
Pregunta 4								3,89
<i>Utilidad del panel de control o dashboard en el proceso de diseño de un servicio (1 a 5)</i>								
Pregunta 5								2,78
<i>Utilidad de las validaciones de modelos en el proceso de diseño de un servicio (1 a 5)</i>								

Respecto a las cuestiones sobre aspectos globales de INNoVaServ (Tabla 4-5), las respuestas dadas a la sexta pregunta permiten concluir que la curva de aprendizaje de la herramienta es demasiado pronunciada (2,22 puntos sobre 5), probablemente debido a la complejidad inherente de utilizar un entorno basado en Eclipse por parte de usuarios no familiarizados con este IDE. Sin embargo, la valoración media de los participantes a la séptima pregunta (3,11 puntos sobre 5), permite considerar la usabilidad de la herramienta como aceptable.

En la octava pregunta se pidió a los participantes que valorasen en qué medida los distintos modelos del caso de estudio se encontraban integrados, esto es, que existiesen claras relaciones entre los modelos diseñados y generados previamente mediante las distintas transformaciones. En este caso se obtuvo una valoración media de 3,78 puntos sobre 5. Si bien es un buen resultado, si tenemos en cuenta las puntuaciones de las preguntas 2 y 3, es posible concluir que esta valoración se ve reducida por la baja recepción entre los participantes de las transformaciones entre e³value y Service Blueprint.

Por otro lado, en la novena pregunta se pedía valorar la eficiencia de INNoVaServ en el diseño de servicios respecto a la alternativa de utilizar diferentes herramientas específicas para cada notación. En este caso se obtuvo un buen resultado de 4,44 sobre 5, ya que, como demuestra la valoración otorgada a la pregunta 11, la característica más destacable de INNoVaServ por parte de los participantes, es precisamente la integración de soporte para distintas notaciones.

Respecto a la décima pregunta, INNoVaServ obtuvo una valoración media sobre su utilidad global en el proceso de diseño de servicios de 3,78 puntos sobre 5. Estableciendo una comparativa entre este valor y los datos obtenidos en la pregunta 4 del cuestionario previo (Tabla 4-3), INNoVaServ tan solo se encuentra por debajo de herramientas como *Canvanizer* y *Bpmn.io*.

Tabla 4-5. Resultados del cuestionario final a preguntas sobre características globales.

Pregunta 6	
Valoración sobre la curva de aprendizaje de INNoVaServ (1 –muy pronunciada- a 5 –nada pronunciada-)	2,22
Pregunta 7	
Valoración sobre la usabilidad global de INNoVaServ (1 –muy usable- a 5 – nada usable-)	3,11
Pregunta 8	
Valoración sobre la integración de modelos en un proyecto de diseño de servicios (1 –nada integrados- a 5 –muy integrados-)	3,78
Pregunta 9	
Eficiencia de INNoVaServ en el diseño de un servicio, respecto al uso de alternativas específicas para cada notación (1 –nada eficiente- a 5 –muy eficiente-)	4,44
Pregunta 10	
Utilidad de INNoVaServ en el proceso de diseño de un servicio (1 –nada útil- a 5 –muy útil)	3,78

Pregunta 11	Facilidad de uso	No conocimientos previos	Integración de notaciones	Trabajo colaborativo
<i>Características destacables de INNoVaServ</i>	3	1	9	0

Por otro lado, con el objetivo de demostrar que las valoraciones dadas por los participantes a las preguntas anteriores son fiables desde una perspectiva estadística, se ha calculado el Coeficiente de Correlación Intraclase (ICC, *Intraclass Correlation Coefficient*) [168] para las preguntas cuantitativas 1 a 11 del cuestionario final.

De manera general, el ICC puede calcularse mediante la fórmula mostrada a continuación, donde $var\beta$ representa la variabilidad debida a diferencias entre preguntas del cuestionario, $var\alpha$ la variabilidad debida a diferencias entre evaluadores, y $var\epsilon$ la variabilidad residual asociada al error inherente de cualquier medición. A su vez, los resultados de este cálculo pueden oscilar entre 0 (ausencia total de consistencia) y 1 (consistencia absoluta), siendo que los valores inferiores a 0,4 representan una baja fiabilidad, aquellos entre 0,4 y 0,75 una buena fiabilidad, y los superiores a 0,75 una excelente fiabilidad [169].

$$ICC = \frac{var\beta}{var\beta + var\epsilon + var\alpha}$$

Tras aplicar la fórmula anterior al conjunto de datos obtenido en el cuestionario final, se ha obtenido el valor de 0,7213246, lo cual indica una buena fiabilidad, si bien existe cierta variabilidad en las respuestas ofrecidas por los participantes, probablemente debido a los diferentes grados de experiencia en el uso de herramientas de modelado de negocio y/o proceso.

Por último, cabe mencionar que algunos comentarios de los participantes en este cuestionario, han permitido identificar puntos de mejora en la herramienta, como por ejemplo el hecho de que, en algunos casos, resulta necesario comprender el análisis de relaciones en el que se basan las transformaciones para poder ejecutarlas con garantías de éxito, pequeñas modificaciones que permitirían acelerar determinadas transformaciones, o la necesidad de mejorar la disposición gráfica de los elementos contenidos en los modelos generados por las mismas.

4.4 Conclusiones de la validación

A lo largo de este capítulo, se ha descrito el procedimiento llevado a cabo para la validación del artefacto construido en el contexto de la presente tesis

doctoral. Para ello se ha empleado un caso de estudio con el que comprobar el correcto funcionamiento de la herramienta y dar respuesta a una serie de preguntas de investigación, y un experimento empírico con el que identificar la mejora que supone el artefacto respecto a alternativas similares.

Sin embargo, durante el proceso de validación, también ha sido posible identificar una serie de limitaciones o puntos de mejora que se describen a continuación:

- Dado que el artefacto validado consiste en un prototipo para tareas de investigación, ha quedado demostrado que su usabilidad es uno de los puntos a mejorar. Durante el estudio empírico, se observó que parte de la dificultad de su uso es causada por la dificultad inherente al IDE de Eclipse. Por ello, actualmente se está trabajando en la posibilidad de trasladar el artefacto a un entorno web, lo que además de reducir la complejidad de uso, permitiría su utilización desde cualquier tipo de dispositivo y posibilitaría introducir fácilmente la funcionalidad de trabajo colaborativo, siendo esta una característica que los participantes en el estudio empírico demandaron como esencial en el contexto del diseño de servicios, y que actualmente INNoVaServ no soporta.
- Si bien se ha demostrado la efectividad que tienen en el proceso de diseño de servicios la mayoría de las transformaciones soportadas por la herramienta, lo cierto es que no se ha logrado un alto grado de integración entre las notaciones de negocio y de proceso. El enfoque seguido en esta tesis doctoral, consiste en estimar el número de modelos de proceso necesario para representar una propuesta de valor a partir de un modelo e³value. Sin embargo, como posible solución a los problemas encontrados, actualmente se está colaborando en un trabajo para dar soporte a la generación de modelos BPMN a partir de modelos e³value y viceversa, siguiendo un enfoque diferente [160], [161].
- Durante el estudio empírico, algunos participantes manifestaron su preocupación con el hecho de que en ciertos casos es necesario entender el análisis de relaciones subyacente para poder completar correctamente determinadas transformaciones. En este sentido, se está valorando la posibilidad de incluir comentarios explicativos o guías visuales en los cuadros de diálogo que aparecen durante estas tareas.
- Otros de los puntos de mejora que se identificaron durante el proceso de validación y que están en fase de desarrollo son: optimización de la disposición gráfica de los distintos elementos de los modelos generados en

una transformación; introducción de pequeñas modificaciones que permitirían acelerar los procesos de generación de vistas parciales; estandarización de los procesos para crear nuevos modelos y diagramas, independientemente de la herramienta utilizada para su desarrollo.

Por último, una vez completados los puntos anteriores, resultaría necesario llevar a cabo una nueva validación, preferiblemente con un número mayor de participantes, con el objetivo de valorar la mejora introducida en la correspondiente iteración respecto a la anterior.

Conclusiones

Como conclusión de esta tesis doctoral, el presente capítulo realiza, en la sección 5.1, un repaso de los diferentes objetivos fijados al comienzo de la tesis, para valorar su grado de cumplimiento. Del mismo modo, en la sección 5.2 se presentan las principales contribuciones científicas, resultados obtenidos, y las publicaciones que sirven para contrastarlos en la sección 5.3. Por último, en la sección 5.4 se identifican y describen algunas líneas de investigación futuras, que servirán para continuar trabajando en la mejora de la propuesta presentada y explorar otras vías de trabajo relacionadas con esta tesis doctoral.

5.1 Análisis de consecución de objetivos

En la sección 1.2 se enumeraron los diferentes objetivos parciales necesarios para alcanzar el principal objetivo de esta tesis doctoral: el diseño, construcción y evaluación de un entorno de modelado que permita utilizar, de forma integrada, varias notaciones para el modelado de negocio y proceso en el contexto del diseño de servicios.

Todos estos objetivos parciales han sido alcanzados de manera satisfactoria. Es posible concluir, por tanto, que la hipótesis de partida de esta tesis doctoral ha quedado probada. No obstante, a continuación se analiza el grado de cumplimiento de esos objetivos parciales:

O1. Estudio de trabajos previos:

Con el objetivo de identificar el cuerpo de conocimiento de esta tesis doctoral, en la sección 2.2 se llevó a cabo una revisión sistemática de la literatura sobre herramientas tecnológicas para el modelado de negocio y proceso. Para la realización de dicha SLR, se siguieron las guías propuestas por Biolchini [56] y Bárbara Kitchenham [57].

Con los resultados de esta revisión, se identificó un conjunto de herramientas para el modelado de negocio y proceso mediante diferentes técnicas. Tras el análisis de estos entornos y diferentes trabajos relacionados, se concluyó que, en el momento de realizar la revisión, existía soporte tecnológico, en mayor o menor medida, para la mayoría de las técnicas tenidas en cuenta en esta tesis doctoral, con la única excepción de la notación de modelado de procesos PCN, para la que no se encontró ninguna herramienta que la soportara.

Además, según se pudo comprobar, la mayoría de soluciones encontradas proveen soporte específico para una determinada notación, no encontrando ninguna herramienta que permita trabajar con varias técnicas para el modelado de negocio y proceso. Sí se encontraron, sin embargo, herramientas de diagramación 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.

La principal conclusión de esta fase de estudio de la literatura fue que, hasta la fecha, no existía un entorno que permitiese trabajar con diferentes técnicas para el modelado de negocio y proceso, lo que facilitaría el uso de modelos de negocio y proceso como elementos de primer orden en el diseño de servicios. Como consecuencia, se fijó como objetivo de la presente tesis doctoral el desarrollo de un entorno que reuniese estas características.

Dado que la construcción del entorno propuesto implicaba el uso intensivo de técnicas y principios de la MDE, se revisaron las bases de este paradigma (sección 2.3), así como algunas herramientas específicas para su puesta en marcha (sección 2.4). Sin embargo, antes de proceder a la construcción del entorno, se hacía necesaria la selección de un conjunto de notaciones para el diseño de servicios que pudieran ser integradas en la herramienta, lo que formaba parte del siguiente objetivo parcial.

O2. Estudio y selección de las notaciones a soportar.

Uno de los objetivos parciales de esta tesis doctoral, consistía en el estudio y selección de un conjunto de notaciones para el modelado de negocio y proceso que fuese apropiado en el contexto del diseño de servicios, con el fin de integrarlas en el entorno de modelado propuesto. Para ello, se analizó en primer lugar el popular modelo del doble diamante del *UK Design Council*²⁷ para el diseño de servicios (sección 2.1.1).

Dado que, de manera general, dicho modelo divide el proceso de diseño de un servicio en las etapas de descubrimiento del problema y diseño de la solución, se consideró apropiado utilizar para la primera fase notaciones de modelado de negocio, que aportan una visión global de una organización, y para la segunda fase notaciones para el modelado de proceso, que permiten especificar los detalles de implementación de una solución. En concreto, las notaciones seleccionadas fueron los modelos de negocio Canvas y e³value, y los modelos de proceso Service

²⁷ <https://www.designcouncil.org.uk/>

Blueprint, PCN y BPMN. Cabe mencionar que este conjunto de notaciones no está cerrado, ya que la arquitectura del prototipo desarrollado en esta tesis doctoral, soporta una fácil integración de nuevas notaciones.

Una vez seleccionado el conjunto de notaciones a soportar, se llevó a cabo un estudio detallado de la especificación de cada una, que puede consultarse en la sección 2.1.2. En este estudio pudo observarse que las notaciones seleccionadas presentan similitudes y diferencias entre ellas, pero en todos los casos la información que proporcionan es complementaria.

Estudiadas las notaciones a soportar, el siguiente paso consistía en el diseño y construcción del entorno de modelado, lo que se llevó a cabo como parte del siguiente objetivo parcial.

O3. Construcción de un entorno de modelado de negocio y proceso para el diseño de servicios.

El siguiente objetivo parcial de esta tesis doctoral, pasaba por la construcción de un entorno de modelado de negocio y proceso para el diseño de servicios, que soportase las notaciones previamente seleccionadas y analizadas. Dicho entorno ha sido denominado INNoVaServ, y el proceso de su diseño y construcción se detalla en el capítulo 3.

Para alcanzar este objetivo se construyó un conjunto de DSLs gráficos, uno por cada notación a soportar. Para ello, en primer lugar se definió la sintaxis abstracta de cada DSL en forma de metamodelo. Los detalles de estos metamodelos pueden consultarse en la sección 3.2.2.

A continuación se definió la sintaxis concreta de cada DSL, construyendo un diagramador gráfico para cada uno. Dado que se optó por utilizar EMF [162] como marco de referencia, para su construcción podía emplearse cualquier herramienta que generase modelos compatibles. En esta tesis doctoral se han empleado las soluciones de Eclipse GMF [170] y Sirius [147]. Los detalles de la construcción de estos DSLs gráficos pueden consultarse en la sección 3.3.1.

Posteriormente, se añadió a estos DSLs capacidades de validación y corrección automática de modelos. Además, tal y como se describe en la sección 3.3.3, en algunos casos se desarrollaron mecanismos de validación formal y simulación.

Por último, los DSLs gráficos fueron integrados en un entorno único basado en el IDE de Eclipse, si bien están disponibles en forma de *plug-ins*²⁸ que pueden instalarse de forma independiente en cualquier otra versión de Eclipse, siempre que se satisfagan las correspondientes dependencias.

A partir de esta versión inicial del entorno de modelado, el siguiente objetivo parcial consistía en la construcción de puentes tecnológicos que permitiesen conectar las notaciones soportadas.

O4. Construcción de puentes tecnológicos que permitan conectar las distintas notaciones seleccionadas

Una vez integrados en un único entorno de modelado los distintos DSLs gráficos, el siguiente objetivo parcial pasaba por la construcción de puentes tecnológicos que permitiesen conectar las diferentes notaciones.

Para alcanzar este objetivo, en primer lugar se llevó a cabo un análisis de correspondencias entre las notaciones soportadas, que puede consultarse en la sección 3.2.3. Dicho estudio fue dividido en tres fases en función de la naturaleza de las notaciones involucradas: correspondencias entre notaciones para el modelado de negocio; correspondencias entre notaciones para el modelado de proceso; correspondencias entre notaciones para el modelado de negocio y proceso.

Si bien en algunos casos las correspondencias identificadas permiten establecer relaciones directas entre los elementos de distintos modelos, lo cierto es que en la mayoría de ocasiones requieren de la toma de ciertas decisiones de diseño para pasar de un modelo a otro. Por este motivo, se determinó que las transformaciones entre modelos a implementar, lejos de ser completamente automáticas, debían incluir, en gran medida, interacción con el usuario para la toma de estas decisiones de diseño.

Como se describe en la sección 3.3.2.1, para evitar una complejidad excesiva, en esta fase fue necesario seleccionar un subconjunto de transformaciones (Figura 3-22) que permitiese generar un modelo en una determinada notación, a partir de cualquier otro, ejecutando, en caso necesario, una cadena de transformaciones. Este subconjunto de transformaciones se seleccionó teniendo presente el modelo del doble diamante mencionado, de tal modo que permitiese el diseño de un servicio comenzando por una visión global de éste (ofrecida por los modelos de negocio Canvas y e³value), y continuando

²⁸ <http://kybele.es/innovaserv/>

con los detalles de su implementación a través de los diferentes modelos de proceso soportados (Service Blueprint, PCN y BPMN).

Para completar el objetivo de conectar las diferentes notaciones, se especificó en la sección 3.3.2 un metamodelo de trazas para la generación de modelos de relaciones entre los modelos involucrados en cada transformación implementada. De esta forma, además del modelo de destino, cada transformación genera el correspondiente modelo de trazas.

Para facilitar la visualización de la información proporcionada por estas trazas, se incluyó en INNoVaServ el visor de modelos Modelink de Epsilon [149], que permite la visualización simultánea de los modelos de origen, destino, y el correspondiente modelo de trazas, resaltando las relaciones existentes entre todos ellos.

Finalmente, con el objetivo de facilitar la gestión de todos los modelos involucrados en un proyecto de diseño de servicios y proporcionar una visión general de éste, tal y como recoge la sección 3.3.2.2, se construyó e integró en INNoVaServ un panel de control que permite, de un modo sencillo, la selección de modelos y la invocación de las diferentes transformaciones.

Finalizada la construcción del entorno de modelado propuesto en esta tesis doctoral, el siguiente objetivo parcial pasaba por su validación y evaluación.

O5. Evaluación de la propuesta.

El último de los objetivos parciales de esta tesis doctoral, era la validación y evaluación del entorno desarrollado. Para ello, en primer lugar se ha llevado a cabo un caso de estudio con el que comprobar el correcto funcionamiento del entorno y evaluar algunas de sus características, como por ejemplo la facilidad de uso de los distintos editores gráficos, la utilidad de las transformaciones implementadas, la calidad de los modelos generados por estas transformaciones, o la puesta en práctica con INNoVaServ del modelo del doble diamante. Este caso de estudio se ha realizado en base a las guías propuestas en [62] y [63] y puede consultarse en la sección 4.2.

Posteriormente se realizó un estudio empírico con el que fue posible evaluar, de manera cuantitativa, tanto características específicas de la herramienta construida, como aspectos más generales de la propuesta, para de este modo poder establecer una comparativa con otras alternativas existentes. Además, este análisis ha permitido identificar algunas ventajas de INNoVaServ respecto a otras soluciones, así como determinados puntos de mejora. Este estudio empírico se ha

desarrollado de acuerdo a las guías descritas en [66] y puede consultarse en la sección 4.3

5.2 Principales contribuciones

En esta sección se enumeran las principales contribuciones de la presente tesis doctoral, si bien la mayoría de ellas se alinean con los objetivos descritos en el punto anterior.

- **Estudio y análisis de herramientas para el modelado de negocio y proceso.**

En las fases iniciales de esta tesis doctoral, se llevó a cabo un estudio sobre herramientas tecnológicas para el modelado de negocio y proceso. De este modo, se realizó una revisión sistemática de la literatura en la que se establecieron una serie de características a analizar, con el fin de establecer una comparativa entre las herramientas existentes, así como proporcionar una visión general del contexto actual en lo que se refiere al modelado de negocio y proceso.

Como principal resultado de este estudio, fue posible concluir que, en el momento de su realización, no existía ningún entorno tecnológico que soportase el modelado y la gestión integrada de modelos de negocio y proceso con diferentes técnicas, en el contexto del diseño de servicios.

- **Selección de un conjunto de notaciones para el modelado de negocio y proceso en el contexto del diseño de servicios.**

A la luz de los resultados del estudio anterior, esta tesis definió como objetivo la construcción de un entorno de modelado para el diseño de servicios. Para ello, fue necesaria la selección de un conjunto inicial de notaciones a soportar por la herramienta.

Tomando como referencia las diferentes etapas del modelo del doble diamante para el diseño de servicios, se seleccionaron distintas notaciones de modelado de negocio (Canvas, e³value) y de proceso (Service Blueprint, PCN y BPMN). En cualquier caso, esta elección de notaciones es solo una propuesta inicial para el desarrollo del prototipo presentado. Por ello, el diseño e implementación del entorno, se realizó de forma que la integración de nuevas notaciones sea no sólo factible, sino casi inmediata.

- **Estudio de correspondencias entre notaciones para el modelado de negocio y proceso en el contexto del diseño de servicios.**

La construcción del entorno de modelado propuesto en esta tesis doctoral, requería del establecimiento de puentes tecnológicos entre las notaciones soportadas, que permitiesen generar modelos parciales en una determinada notación, a partir de modelos elaborados con notaciones diferentes.

Para ello, se llevó a cabo un análisis de correspondencias entre las notaciones seleccionadas: se estudiaron las relaciones a nivel de metamodelo entre las notaciones para el modelado de negocio, las notaciones para el modelado de proceso, y entre las notaciones para el modelado de negocio y proceso.

Como resultado adicional de este análisis, se concluyó que muchas de las correspondencias encontradas no permiten establecer relaciones directas entre los elementos de distintas notaciones, sino que requieren de la toma de ciertas decisiones de diseño. Por lo tanto, no sería posible una automatización completa de las transformaciones implementadas, sino que éstas requerirían de la asistencia del diseñador de servicios para tomar esas decisiones a la hora de pasar de un modelo a otro.

- **Construcción de un entorno unificado de modelado de negocio y proceso para el diseño de servicios.**

Una de las contribuciones más importantes de esta tesis doctoral, es la construcción de un entorno tecnológico llamado INNoVaServ, que soporta el modelado y la gestión integrada de modelos de negocio y proceso en el contexto del diseño de servicios.

INNoVaServ integra para ello un conjunto de DSLs gráficos soportan el modelado de negocio mediante las notaciones Canvas y e³value, y el modelado de proceso mediante Service Blueprint, PCN y BPMN. Además, el entorno se construyó utilizando técnicas y herramientas de la ingeniería dirigida por modelos. Así, lejos de ser un simple diagramador, la herramienta permite procesar la información recogida en los modelos representados por cada diagrama.

Para facilitar la gestión de los modelos involucrados en un determinado proyecto de diseño de servicios, INNoVaServ incluye un panel de control o *dashboard* que permite seleccionar modelos e invocar las distintas transformaciones automáticas.

Asimismo, cabe mencionar que, de acuerdo con los resultados de la revisión sistemática sobre herramientas de modelado de negocio y proceso llevada

a cabo en esta tesis doctoral, INNoVaServ es la primera solución tecnológica en dar soporte a la notación PCN.

- **Validación y evaluación del entorno construido mediante un caso de estudio y un estudio empírico.**

Para validar la propuesta presentada en esta tesis doctoral, se llevó a cabo un caso de estudio de laboratorio que permitió comprobar el correcto funcionamiento de la herramienta, y evaluar algunos aspectos de la misma, como el ahorro en términos de esfuerzo que supone la generación semiautomática de vistas parciales, o la calidad de los modelos generados.

Posteriormente, se realizó un estudio empírico en el que los participantes, entre los que se encontraban investigadores y profesionales del ámbito de los servicios, expresaron su opinión sobre la herramienta en general, y algunas de sus características en particular. Este análisis permitió identificar las ventajas de INNoVaServ respecto a otras alternativas, y algunos aspectos de mejora del entorno.

- **Desarrollo de mecanismos para la validación formal de modelos PCN y Service Blueprint.**

Además de las contribuciones anteriores, en el contexto de esta tesis doctoral también se ha trabajado, en colaboración con distintos grupos de investigación, en el desarrollo de mecanismos para la validación formal de determinadas notaciones.

De esta forma, junto al grupo MISO²⁹ de la Universidad Autónoma de Madrid, se ha llevado a cabo un trabajo para soportar la validación formal de modelos PCN por medio de transformaciones a redes de Petri [158].

Del mismo modo, junto al grupo IMP³⁰ de la Universitat Politècnica de Catalunya, se trabajó en la validación formal de modelos Service Blueprint mediante la generación de diagramas UML asociados al modelo inicial, y la definición de una serie de reglas semánticas, que permiten detectar violaciones de restricciones definidas previamente [157].

²⁹ <http://miso.es/index.html>

³⁰ <https://imp.upc.edu/es>

- **Primeros pasos hacia la integración en el entorno construido de un DSL para la representación de contratos inteligentes, y su generación a partir de modelos e³value.**

Finalmente, vale la pena mencionar como otra de las contribuciones adicionales de esta tesis doctoral, que se ha comenzado a trabajar en la integración en INNoVaServ de un DSL para la representación de *Smart Contracts*, y su generación automática a partir de modelos e³value. Algunos resultados preliminares de este trabajo, pueden consultarse en [48].

5.3 Resultados científicos

Algunos de los resultados de esta tesis doctoral han sido publicados en diferentes foros, tanto nacionales como internacionales. A continuación se enumeran estas publicaciones, agrupadas por el tipo de comunicación:

- **Artículos en conferencias internacionales:**

- Francisco J. Pérez-Blanco, Juan Manuel Vara, Cristian Gómez, Valeria De Castro, Esperanza Marcos (2022). *Leveraging Service Design by bridging business and process modeling*. Proceedings of the 55th Hawaii International Conference on System Sciences (**HICSS 55**), pp. 7476-7485. DOI: 10.24251/HICSS.2022.897. Disponible en: <https://bit.ly/3sxKikg>

GGs Class 2021: 2. GGS Rating 2021: A

- Cristian Gómez, Francisco J. Pérez-Blanco, Juan Manuel Vara, Valeria De Castro, Esperanza Marcos (2021). *Design and development of Smart Contracts for E-government through Value and Business Process Modeling*. Proceedings of the 54th Hawaii International Conference on System Sciences (**HICSS 54**), pp. 2069-2078. DOI: 10.24251/HICSS.2021.254. Disponible en: <https://bit.ly/3tqhVDR>

GGs Class 2018: 1. GGS Rating 2018: A+

- Francisco J. Pérez-Blanco, Juan M. Vara, Cristian Gómez, Valeria De Castro, Esperanza Marcos (2020). *Model-based tool support for Service Design*. 23rd International Conference on Fundamental Approaches to Software (FASE 2020). Lecture Notes in Computer Science, Vol. 12076, pp. 266-272. Springer, Cham. DOI: 10.1007/978-3-030-45234-6_13. Disponible en: <https://bit.ly/35rBaVr>

CORE 2018: B. GGS Class 2018: 2. GGS Rating 2018: A-

- Elena Gómez-Martínez, Francisco J. Pérez-Blanco, Juan de Lara, Juan Manuel Vara, Esperanza Marcos (2019). *Formal support of process chain networks using model-driven engineering and petri nets*. Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing (**SAC 2019**), pp. 98-100. DOI: 10.1145/3297280.3299752. Disponible en: <https://bit.ly/35NaH4M>

CORE 2018: B

- Montserrat Estañol, Esperanza Marcos, Xavier Oriol, Francisco J. Pérez-Blanco, Ernest Teniente, Juan M. Vara (2017). *Validation of Service Blueprint Models by Means of Formal Simulation Techniques*. 15th International Conference on Service Oriented Computing (**ICSOC 2017**). Lecture Notes in Computer Science, Vol. 10601, pp. 80-95. Springer, Cham. DOI: 10.1007/978-3-319-69035-3_6. Disponible en: <https://bit.ly/3pxxb0p>

CORE 2014: A. GGS Class 2015: 2. GGS Rating 2015: A-

- David Granada, Juan M. Vara, Francisco J. Pérez-Blanco, Esperanza Marcos (2017). *Model-based Tool Support for the Development of Visual Editors. A Systematic Mapping Study*. Proceedings of the 12th International Conference on Software Technologies (**ICSOFT 2017**), pp. 330-337. DOI: 10.5220/0006430503300337. Disponible en: <https://bit.ly/3vBu4IH>

CORE 2014: B

- María Luz Martín-Peña, Eloísa Díaz-Garrido, Juan Manuel Vara, Francisco J. Pérez-Blanco (2017). *Servitization and Service design: Model-based tool support for Product-Service Systems Business Modeling*. 6th International Conference on Business Servitization (**ICBS 2017**). Book of Abstracts, pp. 59-63. DOI: 10.3926/serv2017. Disponible en: <https://bit.ly/3puI9no>

• **Artículos en conferencias nacionales:**

- Francisco J. Pérez-Blanco, Juan M. Vara, Cristian Gómez, Valeria De Castro, Esperanza Marcos (2021). *Facilitando la gestión de modelos para el diseño de servicios*. Actas de las XXV Jornadas de Ingeniería del Software y Bases Datos (**JISBD 2021**). Disponible en: <https://bit.ly/3MlqAzW>
- Cristian Gómez, Juan M. Vara, Francisco J. Pérez-Blanco, Esperanza Marcos (2021). *SmaC: Soportando el Modelado de Contratos Inteligentes*.

Actas de las XXV Jornadas de Ingeniería del Software y Bases Datos (**JISBD 2021**). Disponible en: <https://bit.ly/3toQcDq>

- Francisco J. Pérez-Blanco, Juan M. Vara, Cristian Gómez, Valeria De Castro, David Granada, Esperanza Marcos (2019). *Acercando modelos de negocio y de proceso para el diseño de servicios*. Actas de las XXIV Jornadas de Ingeniería del Software y Bases Datos (**JISBD 2019**). Disponible en: <https://bit.ly/3vwhWJ3>
- Cristian Gómez, Juan M. Vara, Francisco J. Pérez-Blanco, Esperanza Marcos (2019). *Una propuesta para soportar la especificación a alto nivel de contratos inteligentes*. Actas de las XXIV Jornadas de Ingeniería del Software y Bases Datos (**JISBD 2019**). Disponible en: <https://bit.ly/3HIpxM>
- Francisco J. Pérez-Blanco, Juan M. Vara, Valeria De Castro, David Granada, Esperanza Marcos (2018). *Hacia la integración de lenguajes de modelado para el diseño de servicios*. Actas de las XXIII Jornadas de Ingeniería del Software y Bases Datos (**JISBD 2018**). Disponible en: <https://bit.ly/34cOzQM>
- Francisco J. Pérez-Blanco, Juan M. Vara, Valeria De Castro, David Granada, Esperanza Marcos (2017). *Soportando el modelado de procesos de negocio con Process Chain Network*. Actas de las XXII Jornadas de Ingeniería del Software y Bases Datos (**JISBD 2017**). Disponible en: <https://bit.ly/3IAmVMq>

- **Capítulos de libro:**

- Juan M. Vara, Francisco J. Pérez-Blanco, David Granada, Esperanza Marcos (2021). *Facilitando el proceso de aprendizaje en primeros cursos de programación: uso de Codeboard en Aula Virtual*. En Estrategias de adaptación metodológica ante la pandemia del COVID-19 en la universidad (pp. 277-297). Dykinson. Disponible en: <https://bit.ly/3vwxHjf>

- **Trabajos enviados:**

- Fadime Kaya, Glenda Amaral, Francisco J. Pérez-Blanco, Marc X. Makkes, Jaap Gordijn, Tina Van der Linden (2022). *An Ontological Exploration of Central Bank Digital Currency*. 35th Bled eConference: Digital Restructuring and Human (Re)Action (**Bled eConference 2022**).

- Fadime Kaya, Francisco J. Pérez-Blanco, Joris Dekker, Jaap Gordijn, Juan M. Vara (2022). *Design Fundamentals for Decentralized Governance in Fractional Reserve Banking*. 56th Hawaii International Conference on System Sciences (**HICSS 56**).

GGs Class 2021: 2. GGS Rating 2021: A

- Isaac da Silva, Francisco J. Pérez-Blanco, Jaap Gordijn, Juan M. Vara (2022). *On deriving business models from process models: An empirical study*. 41st International Conference on Conceptual Modeling (**ER 2022**).

CORE 2021: A. GGS Class 2021: 3. GGS Rating 2021: B

- Francisco J. Pérez-Blanco, Juan Manuel Vara, Cristian Gómez, Valeria De Castro, Esperanza Marcos (2022). *INNoVaServ: a modelling environment for Service Design*. Information and Software Technology.

JCR Factor de Impacto 2020: 2,730. Posición 31/108 (2^{do} Cuartil)

5.4 Trabajos futuros

A partir de las contribuciones realizadas por esta tesis doctoral, se han identificado varias líneas de investigación futuras, que se resumen en las siguientes subsecciones.

5.4.1 Construcción de una versión web de INNoVaServ

A corto plazo, uno de los objetivos inmediatos pasa por el desarrollo de una versión web del entorno de modelado.

Por un lado, esta actualización evitaría los inconvenientes asociados al uso de una distribución de escritorio y offline del IDE de Eclipse, como la descarga y configuración del entorno, instalación de dependencias, disponibilidad, portabilidad, etc. De este modo, INNoVaServ sería accesible en cualquier dispositivo por medio de un navegador web. Además, el desarrollo de esta nueva versión permitirá abordar algunas de las deficiencias identificadas durante la validación del prototipo inicial llevada a cabo en esta tesis doctoral.

Por otro lado, una versión de INNoVaServ con las características descritas en el párrafo anterior, abre la puerta a soportar el trabajo colaborativo durante el proceso de diseño y modelado de un servicio. Esta es una característica que, como se comprobó durante la validación, fue muy valorada por los participantes del estudio empírico.

5.4.2 *Mejorar la integración entre notaciones de negocio y proceso*

Durante el desarrollo de esta tesis doctoral, se llevó a cabo un análisis de correspondencias entre las notaciones seleccionadas con el objetivo de establecer puentes tecnológicos que las conectasen. Como se vio en la sección 3.2.3, el enfoque seguido en esta tesis doctoral para conectar las notaciones de modelado de negocio con aquellas de modelado de proceso, consiste en estimar el número de modelos de proceso que se necesitan para representar una determinada propuesta de valor, a partir de un modelo e³value.

Fruto de la estancia predoctoral y colaboración llevada a cabo con el grupo de investigación DISE³¹ de la Vrije Universiteit de Ámsterdam, ha sido posible conocer otros enfoques que complementan al seguido en esta tesis doctoral [160], [161]. De este modo, otro de los objetivos futuros pasa por la integración en INNoVaServ del soporte tecnológico necesario para lograr la semi-automatización de las guías de derivación propuestas en dichos enfoques.

Adicionalmente, a partir de los resultados de la validación realizada en esta tesis doctoral, se pretende abordar mejoras identificadas respecto a la generación semiautomática de modelos parciales.

5.4.3 *Conexión de modelos con datos reales*

A medio y largo plazo, otro de los objetivos que se plantean, consiste en conectar los modelos elaborados mediante INNoVaServ con datos generados por la operativa diaria de las organizaciones, así como la generación de nuevos modelos a partir de dichos datos mediante la aplicación de técnicas de minería de procesos [171]. De este modo, enriqueciendo los modelos con la información de alto nivel generada a partir de esos datos, podría facilitarse la identificación de fortalezas y debilidades de un servicio, y guiar o facilitar la toma de decisiones estratégicas de la organización.

5.4.4 *Integración de un DSL para Smart Contracts*

Finalmente, como se ha mencionado en la sección 5.2, se ha comenzado a trabajar en la integración de un DSL para la representación de *Smart Contracts* en el entorno de INNoVaServ [48]. Otro de los trabajos futuros consiste, por tanto, en continuar en esta línea de investigación, con el objetivo de lograr una semi-automatización de la generación de los contratos inteligentes necesarios para la

³¹ <https://dise-lab.nl/>

implementación de un determinado servicio, a partir de los modelos e³value utilizados para su diseño.

*Apéndice A: Recursos del
Estudio Empírico*

Este apéndice recoge los cuestionarios y el enunciado del caso de estudio utilizado durante el experimento llevado a cabo en la fase de evaluación de la propuesta construida en esta tesis doctoral, cuyos detalles se describen en la sección 4.3.

A.1 Cuestionario inicial

1. De las siguientes notaciones para el modelado de negocio y proceso, marque aquellas que **conozca** (es suficiente con que haya oído hablar de ellas):
 - a. Business Model Canvas.
 - b. e³value.
 - c. Service Blueprint.
 - d. Process Chain Network (PCN).
 - e. BPMN.
 - f. Otras

2. De las siguientes notaciones para el modelado de negocio y de proceso, marque las que haya **utilizado** en algún momento:
 - a. Business Model Canvas.
 - b. e³value.
 - c. Service Blueprint.
 - d. Process Chain Network (PCN).
 - e. BPMN.
 - f. Otras:

3. De las siguientes herramientas de modelado, indique las que haya utilizado en algún momento:
 - a. Canvanizer.
 - b. Miro.com.
 - c. e³value editor.

- d. Draw.io.
 - e. Lucidchart.com.
 - f. Gliffy.com.
 - g. MS Visio.
 - h. Miro.
 - i. Signavio.
 - j. Strategyzer.
 - k. Bpmn.io.
 - l. BonitaSoft.
 - m. Bizagi.
 - n. Otras:
4. En caso de haber utilizado alguna herramienta de modelado anteriormente, valore del 1 al 5 su utilidad en el proceso de diseño de servicios (usabilidad, capacidades de exportación, posibilidad de visualizar diferentes puntos de vista del servicio, etc.). Indique la nota junto a cada herramienta utilizada de la pregunta anterior.
5. De las siguientes características, seleccione las que considere más importantes en una herramienta de modelado orientada al diseño de servicios (puede seleccionar más de una):
- a. Facilidad de uso.
 - b. No necesidad de conocimientos previos sobre las notaciones soportadas.
 - c. Soporte para un amplio número de notaciones de manera integrada.
 - d. Soporte para trabajo colaborativo.
 - e. Otras:

A.2 Cuestionario final

1. Valore del 1 al 5 la facilidad de uso de cada diagramador de INNoVaServ, siendo 5 muy sencillo y 1 demasiado complicado de utilizar:

Canvas	e ³ value	SBP	PCN	BPMN

2. Valore del 1 al 5 la utilidad que presenta cada transformación incluida en INNoVaServ durante un proceso de diseño de servicios, siendo 5 muy útil y 1 nada útil:

Canvas	e ³ value	e ³ value	SBP	SBP	PCN	PCN	BPMN
>>	>>	>>	>>	>>	>>	>>	>>
e ³ value	Canvas	SBP	e ³ value	PCN	SBP	BPMN	PCN

3. Valore del 1 al 5 la cantidad de esfuerzo que ahorra cada transformación incluida en INNoVaServ durante un proceso de diseño de servicios, siendo 5 mucho y 1 nada:

Canvas	e ³ value	e ³ value	SBP	SBP	PCN	PCN	BPMN
>>	>>	>>	>>	>>	>>	>>	>>
e ³ value	Canvas	SBP	e ³ value	PCN	SBP	BPMN	PCN

4. Valore del 1 al 5 la utilidad del panel de control incluido en INNoVaServ a la hora de diseñar un servicio, siendo 5 muy útil y 1 nada útil:

- Utilidad del panel de control:

5. Valore del 1 al 5 la utilidad de las validaciones de modelos incluidas en INNoVaServ a la hora de diseñar un servicio, siendo 5 muy útil y 1 nada útil:

- Utilidad de las validaciones:

6. Valore del 1 al 5 la curva de aprendizaje global de INNoVaServ, siendo 5 nada pronunciada y 1 muy pronunciada:

- Curva de aprendizaje:
7. Valore del 1 al 5 la usabilidad global de INNoVaServ, siendo 5 muy usable y 1 nada usable:
 - Usabilidad:
 8. Valore del 1 al 5 si considera que los distintos modelos elaborados con INNoVaServ para el caso de estudio se encuentran relacionados entre sí, siendo 5 muy relacionados y 1 nada relacionados:
 - Integración de modelos:
 9. Valore del 1 al 5 la eficiencia de INNoVaServ para el desarrollo de un servicio respecto a la alternativa de tener que recurrir a diferentes soluciones específicas, siendo 5 muy eficiente y 1 nada eficiente:
 - Eficiencia respecto a alternativas específicas:
 10. Valore del 1 al 5 la utilidad de INNoVaServ a la hora de diseñar un servicio determinado, siendo 5 muy útil y 1 muy poco útil (usabilidad, capacidades de exportación, posibilidad de visualizar diferentes puntos de vista del servicio, etc.):
 - Utilidad global de INNoVaServ:
 11. Seleccione, si corresponde, alguna característica destacable de INNoVaServ:
 - a. Facilidad de uso.
 - b. No necesidad de conocimientos previos sobre las notaciones soportadas.
 - c. Soporte para un amplio número de notaciones de manera integrada.
 - d. Soporte para trabajo colaborativo.
 - e. Otras:
 12. Otros comentarios y/o aspectos de mejora.

A.3 Enunciado del caso de estudio

A continuación se presenta el enunciado del caso de estudio entregado a los participantes del experimento llevado a cabo en la fase de evaluación del artefacto construido en esta tesis doctoral.

A.3.1 Contextualización

A. Pádel es una pequeña empresa gestionada por dos socios, especializada en la enseñanza y práctica de pádel en los niveles bajo y medio. La gestión de sus dueños ha permitido que el negocio sobreviva durante años, pero aprovechando el parón provocado por la pandemia de la COVID-19, se disponen a redefinir el servicio ofrecido para dar respuesta a determinadas peticiones de sus clientes.

A.3.2 Situación actual

A pesar de ser una pequeña empresa, A. Pádel dispone de seis pistas de pádel en propiedad. Por su parte, los entrenadores encargados de las clases son colaboradores externos a los que se les paga por clase impartida. El material utilizado en estas clases (pelotas, recoge-pelotas, conos, etc.), es cedido a la empresa por parte de determinados anunciantes relacionados con el deporte en cuestión, quienes además aportan una determinada suma económica.

La comunicación con los clientes la llevan a cabo los propios responsables de A. Pádel a través del teléfono, email y redes sociales. Además, disponen de una pequeña página web que es mantenida por una empresa de marketing.

Los principales ingresos de A. Pádel proceden de las cuotas trimestrales pagadas por los alumnos y las aportaciones de los anunciantes, mientras que los principales costes son debidos al mantenimiento de las instalaciones y página web, pago a entrenadores, y pago de recibos e impuestos.

Cuando un nuevo cliente acude a A. Pádel, si existen plazas libres, uno de los responsables se hace cargo personalmente de realizarle una pequeña prueba de nivel para asignarle un grupo. A partir de ese momento, el cliente paga una determinada cuota trimestral hasta que decide abandonar la escuela.

A.3.3 Problema

A. Pádel ha identificado dos principales quejas de sus clientes. La primera de ellas fue expresada por un alumno en particular, quien sufrió un problema cardíaco durante una de sus clases. Para prevenir futuros problemas relacionados,

los dueños de A. Pádel han pensado en plantear un acuerdo de colaboración a una cercana clínica especializada en reconocimientos médicos.

Por otro lado, son varios los clientes que han expresado su malestar con las diferencias de nivel que se producen entre alumnos de un mismo grupo al cabo de unos meses de entrenamiento. Para abordar este problema, los dueños de A. Pádel han pensado en varias posibilidades como, por ejemplo, realizar nuevas pruebas de nivel al inicio de cada trimestre similares a la inicial o en forma torneos, si bien están abiertos a otras ideas.

Con todo lo anterior, se pide rediseñar con INNoVaServ el servicio ofrecido por A. Pádel. Para ello, ha de construirse al menos un modelo Canvas y un modelo e³value, así como los modelos Service Blueprint, PCN y BPMN que representan el proceso que ocurre desde que un cliente llega a A. Pádel, hasta que decide darse de baja. Para el diseño de estos modelos, es necesario tener en cuenta las sugerencias de mejora propuestas por los dueños de A. Pádel, así como cualquier otra idea que permita mejorar el servicio ofrecido y generar mayores ingresos y/o menores costes. Durante este proceso, pueden utilizarse las características de INNoVaServ descritas en el manual del Apéndice B.

***Apéndice B: Manual de
Usuario de INNoVaServ***

En este apéndice se presenta un manual de usuario de INNoVaServ: el entorno de modelado para el diseño de servicios construido en esta tesis doctoral, cuyos detalles se describen a lo largo del capítulo 3.

B.1 Descarga e instalación

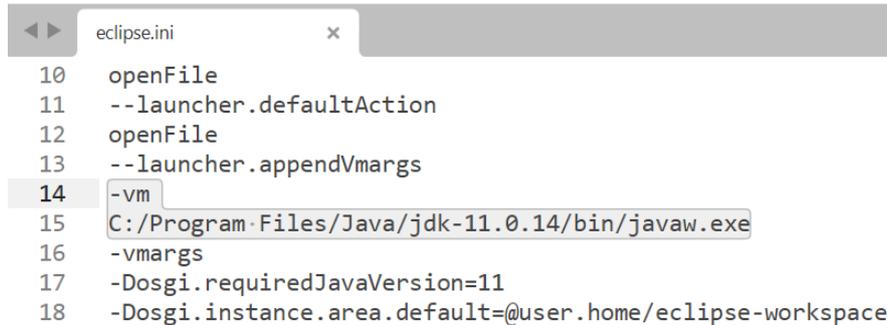
El entorno de modelado INNoVaServ consiste en una distribución del IDE de Eclipse con una serie de *plug-ins* preinstalados, por lo que su puesta en marcha tan solo requiere de su descarga y ejecución. Además, dado que la última versión de INNoVaServ está basada en Eclipse IDE 2020-12, se debe disponer de la versión 11 del JDK o superior.

Si bien resulta posible la descarga e instalación individual en cualquier distribución de Eclipse de los *plug-ins* que componen INNoVaServ, este manual se centra en el entorno completo. La versión más reciente de INNoVaServ solo está disponible para el sistema operativo Windows, y puede descargarse desde el sitio web del entorno³².

Una vez descargada la herramienta, el siguiente paso consiste en descomprimirla. Para ello, basta con hacer clic derecho sobre el archivo descargado, seleccionar la opción “*Extraer todo...*”, y elegir la carpeta en la que ubicar los elementos extraídos.

Para lanzar INNoVaServ, se debe acceder a la carpeta extraída y hacer doble clic sobre el archivo *INNoVaServ.exe*. Si el sistema mostrase un error relacionado con la versión utilizada de Java y/o el JDK, debe abrirse el archivo *eclipse.ini* con un editor de texto plano, y escribirse, bajo el parámetro *-vm*, la ruta completa del archivo *javaw.exe* en el equipo, tal y como muestra el ejemplo de la Figura C-1. Cuando INNoVaServ se ejecute correctamente, debería aparecer una pantalla de carga como la mostrada en la Figura 3-1. A continuación, es posible seleccionar el *workspace* por defecto (que contiene un proyecto de modelado de ejemplo), o crear uno nuevo.

³² <http://kybele.es/innovaserv/>



```

10  openFile
11  --launcher.defaultAction
12  openFile
13  --launcher.appendVmargs
14  -vm
15  C:/Program Files/Java/jdk-11.0.14/bin/javaw.exe
16  -vmargs
17  -Dosgi.requiredJavaVersion=11
18  -Dosgi.instance.area.default=@user.home/eclipse-workspace

```

Figura C-1. Configuración del JDK en el archivo *eclipse.ini*.

Si se ha iniciado INNoVaServ con el *workspace* por defecto, se cargará el proyecto de ejemplo *finalThesis_examples*, que contiene los modelos realizados en el caso de estudio de la sección 4.2 de esta tesis doctoral. La Figura C-2 muestra la estructura del proyecto. Los archivos con la terminación *_diagram* corresponden a los diagramas de cada notación. Haciendo doble clic sobre cualquiera de ellos, se abre el diagrama para su edición junto a la correspondiente paleta de herramientas. Los archivos sin la terminación *_diagram*, corresponden a los modelos que recogen la información subyacente a cada diagrama.

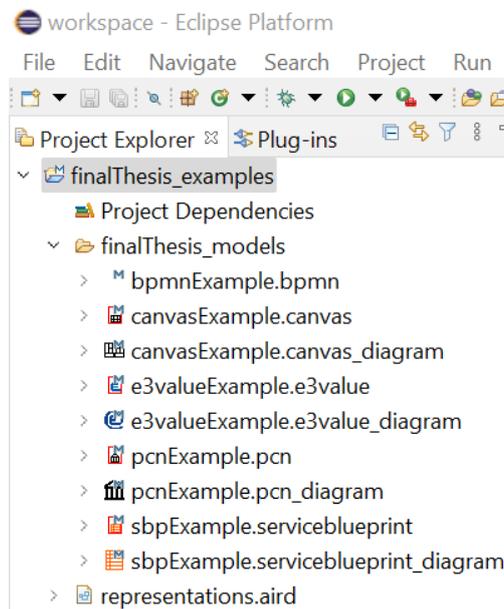


Figura C-2. Estructura del proyecto *finalThesis_examples*.

B.2 Crear un nuevo proyecto

Para crear un nuevo proyecto de diseño de servicios en INNoVaServ, se debe seleccionar la opción *File* en la barra de opciones superior de Eclipse. A continuación, ha de pulsarse sobre *New* y *Other...* En el panel que aparece, debe elegirse la opción *Modeling Project* de la carpeta *Sirius*, tal y como muestra la Figura C-3.

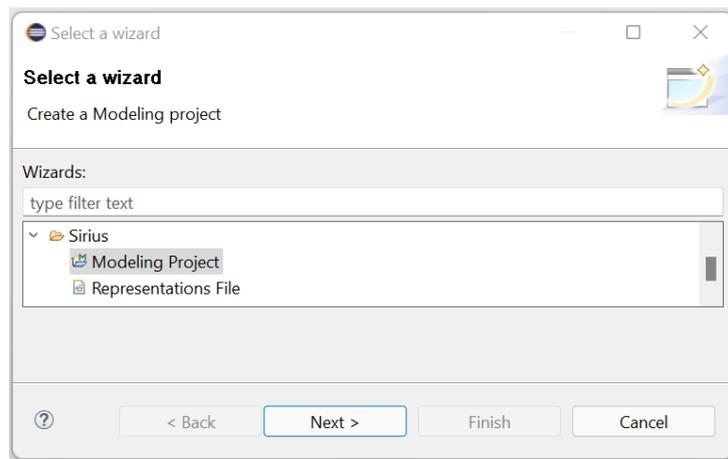


Figura C-3. Creación de un nuevo proyecto en INNoVaServ.

Tras pulsar en *Next*, debe asignarse un nombre al proyecto y seleccionar *Finish*. Hecho esto, el proyecto debería aparecer en el explorador de proyectos de Eclipse, y estaría listo para albergar nuevos modelos.

B.3 Trabajar con modelos Canvas, e³value, SBP y PCN

Para crear modelos Canvas, e³value, Service Blueprint y PCN, debe hacerse clic derecho sobre el proyecto en el que se ubicarán, y seleccionar la opción *New* y *Other...* A continuación, aparecerá un panel similar al de la Figura C-3, en el que podremos buscar por el modelo deseado, o desplegar la carpeta *INNoVaServ Toolkit*. En cualquier caso, resulta importante seleccionar el elemento cuyo nombre termina en *Diagram*, ya que de este modo, el asistente creará tanto el modelo como el diagrama correspondiente.

Tras pulsar en *Next*, podrá modificarse el nombre por defecto del diagrama. A continuación, debe volver a pulsarse en *Next* para asignar un nombre al modelo. Es importante que el modelo y el diagrama compartan el mismo nombre, si bien la extensión será diferente.

Al pulsar en *Finish*, se crearán dos archivos en el proyecto correspondiente: uno para el diagrama y otro para el modelo. Además, se abrirá automáticamente el diagrama en el editor de Eclipse.

La Figura C-4 muestra el entorno de trabajo en el caso de la edición de un diagrama Service Blueprint, si bien los elementos principales (explorador de proyectos, lienzo y paleta de herramientas) serán los mismos en el caso de trabajar con otras notaciones.

Para **añadir elementos** al diagrama, es necesario pulsar sobre la herramienta correspondiente de la paleta y a continuación, pulsar sobre el lugar del diagrama en el que quiera posicionarse. Para **eliminar elementos** del diagrama, debe seleccionarse el elemento a eliminar y pulsar la tecla *Supr* del teclado. Para **mover elementos** dentro del diagrama, es suficiente con seleccionarlos y arrastrarlos al lugar deseado. Por último, para **añadir dependencias** en los casos de las notaciones que las soportan, se debe pulsar en el elemento correspondiente de la paleta, pulsar sobre el nodo de origen en el diagrama, y arrastrar, sin soltar el botón izquierdo del ratón, hasta el nodo de destino.

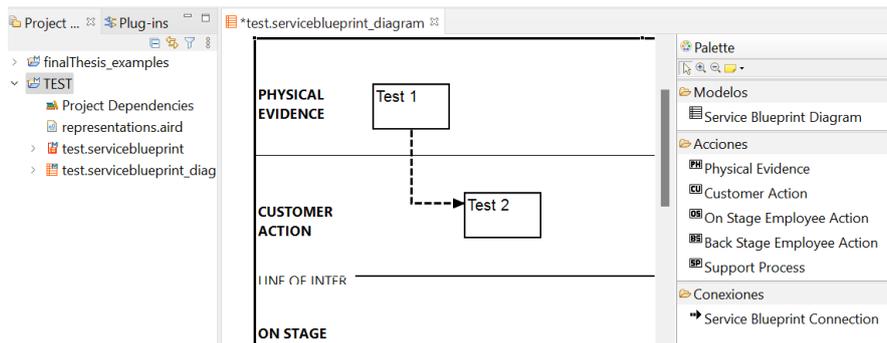


Figura C-4. Edición en INNoVaServ de un diagrama Service Blueprint.

B.4 Trabajar con modelos BPMN

Para crear un modelo BPMN, debe hacerse clic derecho sobre el proyecto en el que se ubicará y seleccionar *New* y *Other...*. En el panel que se abre, debe seleccionarse la opción *BPMN Model*, dentro de la carpeta *INNoVaServ Toolkit*. Al pulsar en *Next*, es posible modificar el nombre por defecto del modelo. A continuación, debe pulsarse de nuevo en *Next* y en la siguiente pantalla, seleccionar el elemento BPMN como *Model Object*, tal y como muestra la Figura C-5.

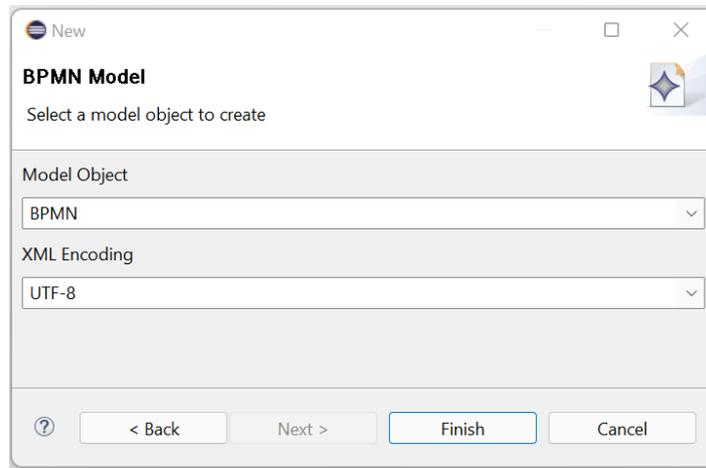


Figura C-5. Creación de modelos BPMN en INNoVaServ.

Al pulsar en *Finish*, se creará el modelo BPMN en el proyecto, pero no el diagrama. Para crearlo, debe hacerse clic derecho sobre el proyecto y seleccionar *Viewpoints Selection*. En el panel que se abrirá, debe marcarse el elemento BPMN y pulsar en *OK*. A continuación, tal y como muestra la Figura C-6, debe expandirse el modelo BPMN en el explorador de proyectos, hacer clic derecho sobre el elemento BPMN, y seleccionar *New representation* y *New BPMNDiagram*.

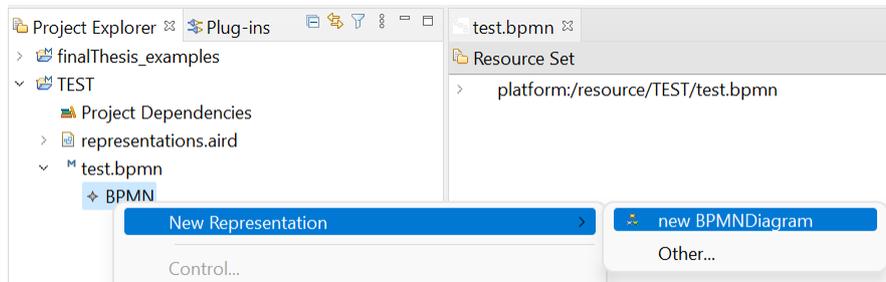


Figura C-6. Creación de diagramas BPMN en INNoVaServ.

En el panel que se abrirá, es posible asignar un nombre al diagrama. Finalmente, tras pulsar en *OK*, se creará dicho diagrama y se abrirá automáticamente para su edición, la cual puede llevarse a cabo de manera similar al resto de diagramadores de las distintas notaciones.

B.5 Validar modelos

INNoVaServ incorpora mecanismos para la validación de modelos. Esta funcionalidad puede ejecutarse abriendo un diagrama y pulsando sobre la opción *Edit* y *Validate* en la barra de opciones superior de Eclipse. Si existiesen errores en el modelo, aparecería una marca indicativa junto a los elementos correspondientes. Posicionando el cursor sobre estas marcas, el sistema indica información sobre la posible causa de los errores. Adicionalmente, como muestra la Figura C-7, estos errores se muestran también en la vista de problemas de eclipse.

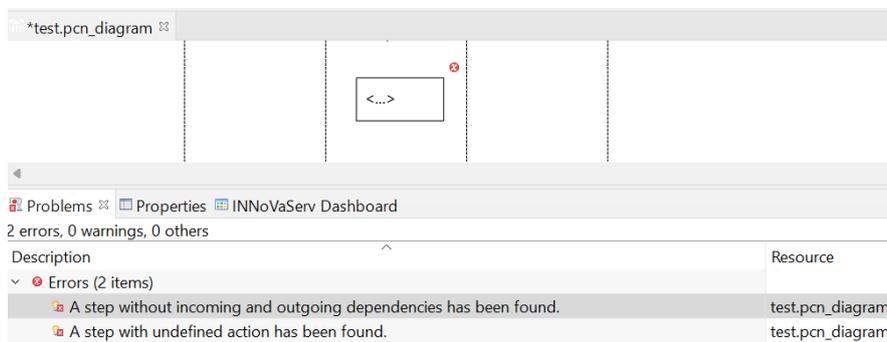


Figura C-7. Validación de modelos con INNoVaServ.

Finalmente, haciendo clic derecho sobre los errores de la vista de problemas, es posible seleccionar la opción *Quick fix* para ver un listado de soluciones propuestas, y ejecutarlas automáticamente.

B.6 Generar vistas parciales de modelos

Otra de las funcionalidades de INNoVaServ, consiste en la posibilidad de generar modelos de manera semiautomática, a partir de modelos expresados con una notación diferente. Estas transformaciones deben ejecutarse desde el panel de control de modelos de INNoVaServ (*INNoVaServ Dashboard*).

Si el panel de control no estuviese abierto en la parte inferior de Eclipse, puede abrirse pulsando en la opción *Window* del menú superior, y seleccionando *Show view, Other... e INNoVaServ Dashboard*.

Para lanzar una transformación, en primer lugar ha de cargarse el modelo de origen en el panel de control. Para ello, basta con pulsar el botón *Select* de la notación correspondiente en el panel, y elegir un modelo válido. Como muestra la Figura C-8, ha de seleccionarse el archivo que contiene el modelo, y no el diagrama.

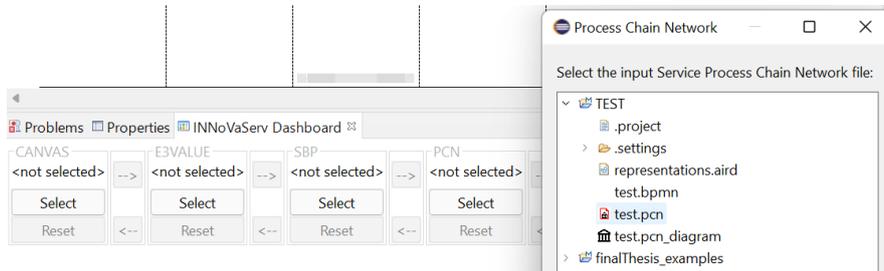


Figura C-8. Carga de modelos en el panel de control de INNoVaServ.

Una vez cargado el modelo, se habilitarán los botones del panel que nos permiten, tomándolo como entrada, generar modelos en otras notaciones. Cabe mencionar que la mayoría de transformaciones requieren la asistencia del usuario a través de un proceso guiado mediante cuadros de diálogo.

Cada vez que se ejecuta una transformación, se generan en el proyecto correspondiente dos archivos: el modelo de destino, y un modelo con las relaciones entre el modelo de origen y el de destino. Para generar el diagrama correspondiente al modelo de destino, debe hacerse clic derecho sobre el archivo en el explorador de proyectos, y seleccionar la opción *Initialize diagram file*. En caso de que la notación del modelo de destino fuese BPMN, para generar el correspondiente diagrama, han de seguirse los pasos descritos en la sección B.4.

Un error habitual tras la ejecución de transformaciones, consiste en la no actualización automática del explorador de proyectos, por lo que los archivos creados automáticamente no aparecen en este panel. Para solucionarlo, basta con seleccionar las opciones *Window, Preferences, General, Workspace* y marcar *Refresh using native hooks and polling checkbox*.

B.7 Visualización de modelos de trazas

INNoVaServ incluye la herramienta *Modelink* para la visualización de los modelos involucrados en una transformación, incluido el correspondiente modelo de trazas. Para utilizarla, debe hacerse clic derecho sobre un proyecto y seleccionar las opciones *New, Other...* y *EMF Modelink*. A continuación, debe asignarse un nombre al archivo, pulsar en *Next* y marcar la opción *Three panels*.

Volviendo a pulsar en *Next*, se debe seleccionar el modelo para el panel izquierdo, es decir, el modelo de origen de la transformación. A continuación, se debe seleccionar el modelo de trazas para posicionarlo en el panel central. Por último, se selecciona el modelo de destino para el panel derecho. Al finalizar este

proceso, INNoVaServ mostrará una vista similar a la de la Figura 3-25, en la que, al seleccionar un elemento de cualquier modelo, se resaltan los elementos del resto de modelos con los que se encuentra relacionado.

Apéndice C: Conclusions

Conclusions of this PhD Thesis are discussed in this chapter. Section C.1 reviews the different objectives set at the beginning of the thesis to assess their degree of fulfilment. Section C.2 presents the main scientific contributions and obtained results, while Section C.3 lists the publications that serve to verify them. Finally, Section □ - identifies and describes some future research lines, which will serve to continue working on the improvement of the presented proposal and to explore other lines related to this PhD Thesis.

C.1 Analysis of achievements

Section 1.2 listed the set of partial objectives that were necessary to achieve the main goal of this PhD Thesis: the design, development and evaluation of a modelling environment that allows to use in an integrated way several notations for business and process modelling, in the context of service design.

All these partial objectives have been satisfactorily attained. Therefore, it is possible to conclude that the proposed hypothesis of this thesis has been proved. The degree of fulfilment of these partial objectives is analysed below:

O1. Study of previous concepts.

To identify the body of knowledge of this PhD Thesis, in Section 2.2 a systematic literature review about technological tools for business and process modelling was carried out. The guidelines proposed by Biolchini [56] and Barbara Kitchenham [57] were followed to conduct this SLR.

As a result of this review, a set of tools for business and process modelling using different techniques was identified. After the analysis of these environments and different related works, it was concluded that, at the time of the review, there was technological support for most of the techniques considered in this PhD Thesis, with the only exception of the PCN process modelling notation, for which no tool was found to support it.

In addition, it was found that most of the solutions provide specific support for a certain notation, while there was not found any tool that allows working with several techniques for business and process modelling. However, we found generic diagramming tools such as MS Visio or Lucidchart which although they may be good options for quick sketching, they were not designed to facilitate the processing of the information gathered in the models.

The main conclusion of this literature review was that at the time of its execution, there was not any environment that allows working with different techniques for business and process modelling, which would facilitate the use of business and process models as first order elements in service design. Consequently, it was set as an objective of this PhD Thesis the development of an environment with these features.

The construction of the proposed environment involves intensive use of MDE techniques and principles. Therefore, the foundations of this paradigm were reviewed in Section 2.3, while some specific tools for its implementation were studied in Section 2.4. However, before proceeding with the construction of the environment, it was necessary to select a set of notations for service design that could be integrated into the tool, which was part of the next partial objective.

O2. Study and selection of notations to support.

One of the partial objectives of this PhD Thesis was to study and select a set of notations for business and process modelling that would be appropriate in the context of service design, in order to be integrated into the proposed modelling environment. To this end, the well-known double diamond model for service design of the UK Design Council³³ was analysed in Section 2.1.1.

Given that, in general terms, this model divides the process of designing a service into the stages of *problem discovery* and *solution design*, it was considered appropriate to use business modelling notations (which provide a global view of an organization) for the first phase, and process modelling notations (which allow to specify the details of the solution implementation) for the second stage. Specifically, the selected notations were the Canvas Business Model and e³value business models, and the Service Blueprint, PCN and BPMN process models. It is worth mentioning that this set of notations is not closed, since the architecture of the prototype developed in this thesis supports an easy integration of new notations.

Once the set of notations to be supported had been selected, a detailed study of the specification of each one was carried out, which can be consulted in Section 2.1.2. During this study it was observed that the selected notations present similarities and differences between them, but in all cases the information they provide is complementary.

³³ <https://www.designcouncil.org.uk/>

Having studied the notations to be supported, the next step was the design and development of the modelling environment, which was carried out as part of the next partial objective.

O3. Building a business and process modelling environment for service design.

The next partial objective of this thesis was the construction of a business and process modelling environment for service design that would support the previously selected and analysed notations. This environment has been named INNoVaServ. The process of its design and construction is detailed in Chapter 3.

To achieve this goal, a set of graphical DSLs was constructed, one for each notation to be supported. To this end, the abstract syntax of each DSL was first defined in the form of a metamodel. Details of these metamodels can be found in Section 3.2.2.

The concrete syntax of each DSL was then defined, so a graphical diagrammer was built for each one. Since EMF [162] was chosen as the reference framework, any tool that generates compatible models could be used for its construction. In this thesis, the solutions Eclipse GMF [170] and Sirius [147] have been used. Details of the construction of these graphical DSLs can be found in Section 3.3.1.

Subsequently, automatic model validation and correction capabilities were added to these DSLs. In addition, as described in Section 3.3.3, formal validation and simulation mechanisms were developed for some notations.

Finally, the graphical DSLs were integrated into a single environment based on the Eclipse IDE, although they are available as plug-ins³⁴ that can be independently installed in any other version of Eclipse, provided that the corresponding dependencies are met.

From this initial version of the modelling environment, the next partial objective was to build technological bridges to connect the supported notations.

O4. Building technological bridges to connect the selected notations.

Once the different graphical DSLs had been integrated into a single modelling environment, the next partial objective was to build technological bridges to connect the different notations.

³⁴ <http://kybele.es/innovaserv/>

To achieve this goal, a correspondence analysis between the supported notations was first carried out, which can be found in Section 3.2.3. This study was divided into three phases according to the nature of the involved notations: correspondences between notations for business modelling; correspondences between notations for process modelling; correspondences between notations for business and process modelling.

In some cases, the identified correspondences allow to establish direct relationships between the elements of different models. However, in most cases it results necessary to take certain design decisions to move from one model to another one. For this reason, it was determined that the transformations to be implemented between models could not be completely automatic, while they should include, to a large extent, interaction with the user to take these design decisions.

As described in Section 3.3.2.1, to avoid an excessive complexity, in this stage it was necessary to select a subset of transformations (Figure 3-22) that would allow the generation of a model in a given notation, starting from any other model, running, if necessary, a chain of transformations. This subset of transformations was selected bearing in mind the double diamond model, so it could be possible the design of a service starting with a global vision (offered by the Canvas Business Model and e³value model) and continuing with the details of its implementation through the different process models supported (Service Blueprint, PCN and BPMN).

To complete the objective of connecting the different notations, it was specified in Section 3.3.2 a trace metamodel for the generation of relationship models between the models involved in each implemented transformation. In this way, in addition to the target model, each transformation generates the corresponding trace model.

To facilitate the visualization of the information provided by these traces, the Epsilon Modelink model viewer [149] was included into INNoVaServ, which allows the simultaneous visualization of the source, target and corresponding trace models, highlighting the relationships between all of them.

Finally, in order to facilitate the management of all the models involved in a service design project, and to provide an overview of that project, as described in Section 3.3.2.2, a dashboard that allows, in a simple way, the selection of models and the invocation of the different transformations, was built and integrated into INNoVaServ.

Once the construction of the modelling environment proposed in this PhD Thesis was completed, the next partial objective was its validation and evaluation.

O5. Evaluation of the proposed modelling environment.

The last partial objective of this PhD Thesis was the validation and evaluation of the developed modelling environment. To this end, a case study has been first carried out to check the correct functioning of the environment and to evaluate some of its characteristics, such as the ease of use of the different graphical editors, the usefulness of the implemented transformations, the quality of the generated models by these transformations, or the implementation of the double diamond model with INNoVaServ. This case study has been carried out based on the guidelines proposed in [62] and [63] and can be consulted in Section 4.2.

Finally, to establish a comparison with other existing alternatives, an empirical study was carried out to evaluate, in a quantitative manner, specific characteristics of the tool, as well as other general aspects of the proposal. In addition, this analysis has made it possible to identify some advantages of INNoVaServ with respect to other solutions, as well as certain points for improvement. This empirical study has been developed according to the guidelines described in [66] and its details can be consulted in Section 4.3.

C.2 Main contributions

This section provides a description of the main contributions of this PhD Thesis, which are directly aligned with the objectives described in the previous section.

- **Study and analysis of tools for business and process modelling.**

At the initial stages of this PhD Thesis, a study about technological tools for business and process modelling was carried out. In this way, a systematic literature review was executed, and a set of characteristics to be analysed was established to make a comparison between the existing tools, as well as to provide an overview of the current context about business and process modelling.

As result of this study, it was possible to conclude that, at the time of its execution, there was not any technological environment that supported the

integrated modelling and management of business and process models with different techniques in the context of service design.

- **Selection of a set of notations for business and process modelling suitable for service design.**

Considering the results of the previous study, this thesis defined as an objective the construction of a modelling environment for service design. To that end, it was required the selection of an initial set of notations to be supported by the tool.

Having as reference the different stages of the double diamond model for service design, different business modelling notations (Canvas Business Model and e³value) and process modelling notations (Service Blueprint, PCN and BPMN) were selected. In any case, this choice of notations is only an initial proposal for the development of the presented prototype. Therefore, the design and implementation of the environment was carried out in a way that the integration of new notations is not only feasible, but almost immediate.

- **Correspondences analysis between notations for business and process modelling in the context of service design.**

The construction of the modelling environment proposed in this PhD Thesis required the establishment of technological bridges between the supported notations, which would allow the generation of partial models in a given notation from models elaborated with different techniques.

To that end, a correspondence analysis was carried out between the selected notations, so the relationships at metamodel level between the notations for business modelling, the notations for process modelling, and between the notations for business and process modelling, were studied.

As an additional result of this analysis, it was concluded that many of the correspondences found do not allow to establish direct relationships between elements of different notations, while they require to take certain design decisions. Therefore, a full automation of the implemented transformations would not be possible but would require the assistance of the service designer to make these decisions when moving from one model to another one.

- **Building a unified business and process modelling environment for service design.**

One of the main contributions of this PhD Thesis is the construction of a technological environment, called INNoVaServ, which supports the modelling and integrated management of business and process models in the context of service design.

This way, INNoVaServ integrates a set of graphical DSLs supporting business modelling using Canvas Business Model and e³value notations, and process modelling using Service Blueprint, PCN and BPMN. In addition, the environment was built using model-driven engineering techniques and tools. Thus, far from being a simple diagrammer, the tool allows processing the information collected in the models represented by each diagram.

To ease the management of the models involved in a given service design project, INNoVaServ includes a dashboard that allows users to select models and launch the different transformations.

It is also worth mentioning that, according to the results of the systematic literature review about business and process modelling tools carried out in this thesis, INNoVaServ is the first technological solution to support the PCN notation.

- **Validation and evaluation of the proposed environment through a case study and an empirical study.**

To validate the proposal presented in this PhD Thesis, a laboratory case study was carried out to check the correct functioning of the tool, and to evaluate some aspects of it, such as the savings in terms of effort of the semi-automatic generation of partial models, or the quality of the generated models.

Subsequently, an empirical study was carried out in which participants, including researchers and service professionals, expressed their opinion about the tool in general terms, and about some of its specific features. This analysis made it possible to identify the advantages of INNoVaServ with respect to other alternatives, and some aspects of improvement of the environment.

- **Development of mechanisms for formal validation of PCN and Service Blueprint models.**

In addition to the previous contributions, in the context of this PhD Thesis, work has also been done in collaboration with different research groups to develop mechanisms for the formal validation of certain notations.

This way, together with the MISO³⁵ research group of the Universidad Autónoma de Madrid, a work has been carried out to support the formal validation of PCN models by means of its transformation to Petri nets [158].

Same way, together with the IMP³⁶ research group of the Universitat Politècnica de Catalunya, a work was carried out on the formal validation of Service Blueprint models, by generating UML diagrams associated with the initial model and defining a series of semantic rules that allow the detection of violations of the previously defined restrictions [157].

- **First steps towards the integration into the proposed environment of a DSL for the representation of smart contracts, and their generation from e³value models.**

Finally, it is worth mentioning as another additional contribution of this PhD Thesis, that work has begun on the integration into INNoVaServ of a DSL for the representation of Smart Contracts, and their automatic generation from e³value models. Some preliminary results of this work can be found in [48].

C.3 Scientific Results

Some of the results obtained in the context of this PhD Thesis have been published in different national and international forums. These publications are listed below, grouped by type of communication:

- **Articles in international conferences:**
 - Francisco J. Pérez-Blanco, Juan Manuel Vara, Cristian Gómez, Valeria De Castro, Esperanza Marcos (2022). *Leveraging Service Design by bridging business and process modeling*. Proceedings of the 55th Hawaii International Conference on System Sciences (**HICSS 55**), pp. 7476-7485. DOI: 10.24251/HICSS.2022.897. Available at: <https://bit.ly/3sxKikg>
GGs Class 2021: 2. GGS Rating 2021: A
 - Cristian Gómez, Francisco J. Pérez-Blanco, Juan Manuel Vara, Valeria De Castro, Esperanza Marcos (2021). *Design and development of Smart Contracts for E-government through Value and Business Process*

³⁵ <http://miso.es/index.html>

³⁶ <https://imp.upc.edu/es>

Modeling. Proceedings of the 54th Hawaii International Conference on System Sciences (**HICSS 54**), pp. 2069-2078. DOI: 10.24251/HICSS.2021.254. Available at: <https://bit.ly/3tqhVDR>

GGs Class 2018: 1. GGS Rating 2018: A+

- Francisco J. Pérez-Blanco, Juan M. Vara, Cristian Gómez, Valeria De Castro, Esperanza Marcos (2020). *Model-based tool support for Service Design*. 23rd International Conference on Fundamental Approaches to Software (**FASE 2020**). Lecture Notes in Computer Science, Vol. 12076, pp. 266-272. Springer, Cham. DOI: 10.1007/978-3-030-45234-6_13. Available at: <https://bit.ly/35rBaVr>

CORE 2018: B. GGS Class 2018: 2. GGS Rating 2018: A-

- Elena Gómez-Martínez, Francisco J. Pérez-Blanco, Juan de Lara, Juan Manuel Vara, Esperanza Marcos (2019). *Formal support of process chain networks using model-driven engineering and petri nets*. Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing (**SAC 2019**), pp. 98-100. DOI: 10.1145/3297280.3299752. Available at: <https://bit.ly/35NaH4M>

CORE 2018: B

- Montserrat Estañol, Esperanza Marcos, Xavier Oriol, Francisco J. Pérez-Blanco, Ernest Teniente, Juan M. Vara (2017). *Validation of Service Blueprint Models by Means of Formal Simulation Techniques*. 15th International Conference on Service Oriented Computing (**ICSOC 2017**). Lecture Notes in Computer Science, Vol. 10601, pp. 80-95. Springer, Cham. DOI: 10.1007/978-3-319-69035-3_6. Available at: <https://bit.ly/3pxxb0p>

CORE 2014: A. GGS Class 2015: 2. GGS Rating 2015: A-

- David Granada, Juan M. Vara, Francisco J. Pérez-Blanco, Esperanza Marcos (2017). *Model-based Tool Support for the Development of Visual Editors. A Systematic Mapping Study*. Proceedings of the 12th International Conference on Software Technologies (**ICSOFT 2017**), pp. 330-337. DOI: 10.5220/0006430503300337. Available at: <https://bit.ly/3vBu4IH>

CORE 2014: B

- María Luz Martín-Peña, Eloísa Díaz-Garrido, Juan Manuel Vara, Francisco J. Pérez-Blanco (2017). *Servitization and Service design: Model-based tool support for Product-Service Systems Business Modeling*. 6th International

Conference on Business Servitization (**ICBS 2017**). Book of Abstracts, pp. 59-63. DOI: 10.3926/serv2017. Available at: <https://bit.ly/3puI9no>

• **Articles in national conferences:**

- Francisco J. Pérez-Blanco, Juan M. Vara, Cristian Gómez, Valeria De Castro, Esperanza Marcos (2021). *Facilitando la gestión de modelos para el diseño de servicios*. Proceedings of the XXV Jornadas de Ingeniería del Software y Bases Datos (**JISBD 2021**). Available at: <https://bit.ly/3MlqAzW>
- Cristian Gómez, Juan M. Vara, Francisco J. Pérez-Blanco, Esperanza Marcos (2021). *SmaC: Soportando el Modelado de Contratos Inteligentes*. Proceedings of the XXV Jornadas de Ingeniería del Software y Bases Datos (**JISBD 2021**). Available at: <https://bit.ly/3toQcDq>
- Francisco J. Pérez-Blanco, Juan M. Vara, Cristian Gómez, Valeria De Castro, David Granada, Esperanza Marcos (2019). *Acercando modelos de negocio y de proceso para el diseño de servicios*. Proceedings of the XXIV Jornadas de Ingeniería del Software y Bases Datos (**JISBD 2019**). Available at: <https://bit.ly/3vwhWJ3>
- Cristian Gómez, Juan M. Vara, Francisco J. Pérez-Blanco, Esperanza Marcos (2019). *Una propuesta para soportar la especificación a alto nivel de contratos inteligentes*. Proceedings of the XXIV Jornadas de Ingeniería del Software y Bases Datos (**JISBD 2019**). Available at: <https://bit.ly/3HIfpXM>
- Francisco J. Pérez-Blanco, Juan M. Vara, Valeria De Castro, David Granada, Esperanza Marcos (2018). *Hacia la integración de lenguajes de modelado para el diseño de servicios*. Proceedings of the XXIII Jornadas de Ingeniería del Software y Bases Datos (**JISBD 2018**). Available at: <https://bit.ly/34cOzQM>
- Francisco J. Pérez-Blanco, Juan M. Vara, Valeria De Castro, David Granada, Esperanza Marcos (2017). *Soportando el modelado de procesos de negocio con Process Chain Network*. Proceedings of the XXII Jornadas de Ingeniería del Software y Bases Datos (**JISBD 2017**). Available at: <https://bit.ly/3IAmVMq>

- **Book chapters:**

- Juan M. Vara, Francisco J. Pérez-Blanco, David Granada, Esperanza Marcos (2021). *Facilitando el proceso de aprendizaje en primeros cursos de programación: uso de Codeboard en Aula Virtual*. En Estrategias de adaptación metodológica ante la pandemia del COVID-19 en la universidad (pp. 277-297). Dykinson. Available at: <https://bit.ly/3vwxHjf>

- **Submitted works:**

- Fadime Kaya, Glenda Amaral, Francisco J. Pérez-Blanco, Marc X. Makkes, Jaap Gordijn, Tina Van der Linden (2022). *An Ontological Exploration of Central Bank Digital Currency*. 35th Bled eConference: Digital Restructuring and Human (Re)Action (**Bled eConference 2022**).
- Fadime Kaya, Francisco J. Pérez-Blanco, Joris Dekker, Jaap Gordijn, Juan M. Vara (2022). *Design Fundamentals for Decentralized Governance in Fractional Reserve Banking*. 56th Hawaii International Conference on System Sciences (**HICSS 56**).

GGs Class 2021: 2. GGS Rating 2021: A

- Isaac da Silva, Francisco J. Pérez-Blanco, Jaap Gordijn, Juan M. Vara (2022). *On deriving business models from process models: An empirical study*. 41st International Conference on Conceptual Modeling (**ER 2022**).

CORE 2021: A. GGS Class 2021: 3. GGS Rating 2021: B

- Francisco J. Pérez-Blanco, Juan Manuel Vara, Cristian Gómez, Valeria De Castro, Esperanza Marcos (2022). *INNoVaServ: a modelling environment for Service Design*. Information and Software Technology.

JCR Impact Factor2020: 2,730. Position 31/108 (2Q)

C.4 Future work

Based on the contributions of this PhD Thesis, several lines for future research have been identified. These lines are summarized in the following subsections.

C.4.1 Building a web version of INNoVaServ

In the short term, one of the immediate objectives is the development of a web version of the modelling environment.

On the one hand, this update would avoid the inconveniences associated with the use of a desktop and offline distribution of the Eclipse IDE, such as downloading and configuring the environment, installing dependencies, availability, portability, etc. In this way, INNoVaServ would be accessible on any device via web browser. Furthermore, the development of this new version will address some of the shortcomings identified during the validation and evaluation of the initial prototype carried out in this PhD Thesis.

On the other hand, a version of INNoVaServ with the features described in the previous paragraph, opens the door to support collaborative work during the design and modelling process of a service. This is actually a feature that was highly valued by the participants of the empirical study, as it was verified during the evaluation of the tool.

C.4.2 Improving integration between business and process notations

During the development of this PhD Thesis, a correspondence analysis was carried out between the selected notations with the aim of establishing technological bridges connecting them. As seen in Section 3.2.3, the approach followed in this work to connect the business modelling notations with those of process modelling, consists of estimating the number of process models needed to represent a given value proposition, starting from an e³value model.

As a result of the pre-doctoral research stay and collaboration carried out with the DISE³⁷ research group at the Vrije Universiteit of Amsterdam, it has been possible to learn about other approaches that complement the one followed in this thesis [160], [161]. Thus, another future goal is the integration into INNoVaServ of the necessary technological support to achieve the semi-automation of the referral guidelines proposed in these approaches.

Additionally, based on the results of the validation carried out in this thesis, the aim is to address the improvements identified with respect to the semi-automatic generation of partial models.

C.4.3 Connecting models with real data

In the medium and long term, another of the proposed objectives is to connect the models developed with INNoVaServ, with the data generated by daily

³⁷ <https://dise-lab.nl/>

operations of organizations, as well as the generation of new models based on this data by applying process mining techniques [171]. In this way, by enriching the models with the high-level information generated from these data, it could facilitate the identification of strengths and weaknesses of a service, and guide or facilitate the organization's strategic decision-making.

C.4.4 Integration of a DSL for Smart Contracts

Finally, as mentioned in Section 5.2, work has started on the integration of a DSL for the representation of Smart Contracts into the INNoVaServ environment [48]. Therefore, another future work consists in continuing this line of research, with the aim of achieving a semi-automation of the generation of the necessary Smart Contracts for the implementation of a given service, based on the e³value models used for its design.

Referencias

- [1] C. M. DaSilva and P. Trkman, "Business model: What it is and what it is not," *Long Range Plann.*, vol. 47, no. 6, pp. 379–389, 2014.
- [2] R. Amit and C. Zott, "Value creation in e-business," *Strateg. Manag. J.*, vol. 22, no. 6–7, pp. 493–520, Jun. 2001, doi: 10.1002/SMJ.187.
- [3] R. Casadesus-Masanell and J. E. Ricart, "From Strategy to Business Models and onto Tactics," *Long Range Plann.*, vol. 43, no. 2–3, pp. 195–215, Apr. 2010, doi: 10.1016/J.LRP.2010.01.004.
- [4] H. Chesbrough and R. S. Rosenbloom, "The role of the business model in capturing value from innovation: evidence from Xerox Corporation's technology spin-off companies," *Ind. Corp. Chang.*, vol. 11, no. 3, pp. 529–555, Jun. 2002, doi: 10.1093/ICC/11.3.529.
- [5] M. Johnson, C. Christensen, H. K.-H. business review, and undefined 2008, "Reinventing your business model," *syv.pt*, 2000, Accessed: Sep. 10, 2021. [Online]. Available: [http://syv.pt/login/upload/userfiles/file/Reinventing Your business model HBR.pdf#page=57](http://syv.pt/login/upload/userfiles/file/Reinventing>Your%20business%20model%20HBR.pdf#page=57).
- [6] J. Magretta, "Why business models matter." Harvard Business School Boston, MA, 2002.
- [7] M. Morris, M. Schindehutte, J. A.-J. of business research, and undefined 2005, "The entrepreneur's business model: toward a unified perspective," *Elsevier*, Accessed: Sep. 10, 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S014829630300242X>.
- [8] C. Zott and R. Amit, "Business Model Design: An Activity System Perspective," *Long Range Plann.*, vol. 43, no. 2–3, pp. 216–226, Apr. 2010, doi: 10.1016/J.LRP.2009.07.004.
- [9] D. J. Teece, "Business models, business strategy and innovation," *Long Range Plann.*, vol. 43, no. 2–3, pp. 172–194, 2010.
- [10] A. Osterwalder and Y. Pigneur, *Business model generation: a handbook for visionaries, game changers, and challengers*. John Wiley & Sons, 2010.
- [11] M. Lewis, *The new new thing: A Silicon Valley story*. WW Norton & Company, 2000.
- [12] B. W. Wirtz, A. Pistoia, S. Ullrich, and V. Göttel, "Business models: Origin, development and future research perspectives," *Long Range Plann.*, vol. 49, no. 1, pp. 36–54, 2016.
- [13] D. M. Bridgeland and R. Zahavi, *Business modeling: a practical guide to realizing business value*. Morgan Kaufmann, 2008.
- [14] F. J. Pérez-Blanco, J. M. Vara, C. Gómez, V. De Castro, D. Granada, and E. Marcos, "Acercando modelos de negocio y de proceso para el diseño de servicios," *Actas las 24th Jornadas Ing. del Softw. y Bases Datos, JISBD 2019*, 2019, Accessed: Sep. 13, 2021. [Online]. Available: <https://biblioteca.sistedes.es/submissions/descargas/2019/JISBD/2019-JISBD-034.pdf>.
- [15] R. Saco, A. G.-D. management review, and undefined 2008, "Service design: An appraisal," *ida.liu.se*, vol. 19, no. 1, 2008, Accessed: Sep. 13, 2021. [Online]. Available: [https://www.ida.liu.se/divisions/hcs/ixs/material/servicedesignGbg10/2 Additional readings/Goncalves_service design an appraisal.pdf](https://www.ida.liu.se/divisions/hcs/ixs/material/servicedesignGbg10/2%20Additional%20readings/Goncalves_service%20design%20an%20appraisal.pdf).
- [16] L. S. Cook, D. E. Bowen, R. B. Chase, S. Dasu, D. M. Stewart, and D. A. Tansik,

- “Human issues in service design,” *J. Oper. Manag.*, vol. 20, no. 2, pp. 159–174, 2002.
- [17] A. Meroni and D. Sangiorgi, *Design for services*. Routledge, 2016.
- [18] K. Wetter-Edman, D. Sangiorgi, B. Edvardsson, S. Holmlid, C. Grönroos, and T. Mattelmäki, “Design for value co-creation: Exploring synergies between design for service and service logic,” *Serv. Sci.*, vol. 6, no. 2, pp. 106–121, 2014.
- [19] L. Shostack, “Desing Services that delivery,” *Harv. Bus. Rev.*, vol. 84115, 1984.
- [20] S. Cavalieri and G. Pezzotta, “Product--Service Systems Engineering: State of the art and research challenges,” *Comput. Ind.*, vol. 63, no. 4, pp. 278–288, 2012.
- [21] S. Holmlid and S. Evenson, “Bringing service design to service sciences, management and engineering,” in *Service science, management and engineering education for the 21st century*, Springer, 2008, pp. 341–345.
- [22] J. Gordijn, “E-business value modelling using the e3-value ontology,” in *Value creation from e-business models*, Elsevier, 2004, pp. 98–127.
- [23] M. Chinosi and A. Trombetta, “BPMN: An introduction to the standard,” *Comput. Stand. Interfaces*, vol. 34, no. 1, pp. 124–134, 2012.
- [24] S. E. Sampson, “Visualizing service operations,” *J. Serv. Res.*, vol. 15, no. 2, pp. 182–198, 2012.
- [25] M. Kerremans, “Gartner Market Guide for Process Mining, Research Note G00353970 (2018).” .
- [26] M. Kerremans, “Market guide for process mining,” *Gart. Inc*, 2018.
- [27] J. M. Vara and E. Marcos, “A framework for model-driven development of information systems: Technical decisions and lessons learned,” *J. Syst. Softw.*, vol. 85, no. 10, pp. 2368–2384, 2012.
- [28] L. Yin, J. Liu, Z. D.-2011 F. I. C. on, and undefined 2011, “Modeling and prototyping business processes in AutoPA,” *ieeexplore.ieee.org*, Accessed: Sep. 13, 2021. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/6042075/>.
- [29] Á. J. Rielo, “M2DAT-DB Desarollo de Editores Gráficos para el Modelado de Bases de Datos Objeto-Relacionales: SQL2003 y Oracle 10g,” 2010, Accessed: Sep. 13, 2021. [Online]. Available: <https://burjcdigital.urjc.es/handle/10115/4119>.
- [30] M. Vieira, S. C.-P. of the 23rd B. Symposium, and undefined 2017, “Model-driven Engineering in the Development of Ubiquitous Applications: Technologies, Tools and Languages,” *dl.acm.org*, pp. 29–32, Oct. 2017, doi: 10.1145/3126858.3131633.
- [31] J. Whittle, J. Hutchinson, and M. Rouncefield, “The state of practice in model-driven engineering,” *IEEE Softw.*, vol. 31, no. 3, pp. 79–85, 2013.
- [32] V. de Castro, E. Marcos, and P. Cáceres, “A user service oriented method to model web information systems,” in *International Conference on Web Information Systems Engineering*, 2004, pp. 41–52.
- [33] V. De Castro, E. Marcos, and M. Lopez Sanz, “A model driven method for service composition modelling: a case study,” *Int. J. Web Eng. Technol.*, vol. 2, no. 4, pp. 335–353, 2006.
- [34] M. López-Sanz, C. J. Acuña, C. E. Cuesta, and E. Marcos, “Modelling of service-oriented architectures with UML,” *Electron. Notes Theor. Comput. Sci.*, vol. 194, no. 4, pp. 23–37, 2008.
- [35] J. M. Vara, B. Vela, J. M. Cavero, and E. Marcos, “Model transformation for object-relational database development,” in *Proceedings of the 2007 ACM symposium on Applied computing*, 2007, pp. 1012–1019.
- [36] B. Vela, E. Fernández-Medina, E. Marcos, and M. Piattini, “Model driven development of secure XML databases,” *ACM Sigmod Rec.*, vol. 35, no. 3, pp. 22–

- 27, 2006.
- [37] M. V. de Castro, "Aproximación MDA para el desarrollo orientado a servicios de sistemas de información WEB," PhD thesis, Universidad Rey Juan Carlos, 2007.
- [38] M. Belaunde, C. Casanave, D. DSouza, and K. Duddy, "MDA Guide Version 1.0. 1," 2003, Accessed: Sep. 15, 2021. [Online]. Available: <https://it-dev.mpiwg-berlin.mpg.de/tracs/JET/export/864/trunk/doc/latex/Diplomarbeit/websources/OMG/03-06-01.pdf>.
- [39] C. J. Acuña, "PISA-Arquitectura de integración de portales Web: un enfoque dirigido por modelos y basado en Servicios Web semánticos," PhD thesis, Universidad Rey Juan Carlos, 2007.
- [40] J. M. Vara Mesa, "M2DAT: a technical solution for model-driven development of Web information systems," PhD thesis, Universidad Rey Juan Carlos, 2009.
- [41] V. A. Bollati, "MeTAGeM: Entorno de Desarrollo de Transformaciones de Modelos Dirigido por Modelos," PhD thesis, Universidad Rey Juan Carlos, 2010.
- [42] M. López Sanz, "ArchiMeDeS: A Service-Oriented Framework for Model-Driven Development of Software Architectures," PhD thesis, Universidad Rey Juan Carlos, 2011.
- [43] Á. Jiménez Rielo, "Incorporando la Gestión de la Trazabilidad en un entorno de Desarrollo de Transformaciones de Modelos Dirigido por Modelos."
- [44] I. Santiago Viñambres, "iTrace: un framework para la gestión y el análisis de información de trazabilidad en el contexto de la Ingeniería Dirigida por Modelos," PhD thesis, Universidad Rey Juan Carlos, 2015.
- [45] D. Granada, "Desarrollo dirigido por modelos de editores gráficos cognitivamente eficaces para Lenguajes Específicos de Dominio," PhD thesis, Universidad Rey Juan Carlos, 2016.
- [46] M. Salgado, E. Marcos, J. M. Vara, V. De Castro, and M. L. Martín-Peña, "Service Design Methodologies. A Systematic Literature Review," in *The 7th Naples Forum on Service. Book of Abstract*, 2021, p. 72.
- [47] M. F. Papa, M. S. Quezada, M. V. de Castro, L. Olsina, and E. M. Martínez, "Calidad Interna de Servicio: Alternativa de Mejora aun cuando la Satisfacción del Cliente es Positiva," in *2020 IEEE Congreso Bienal de Argentina (ARGENCON)*, 2020, pp. 1–8.
- [48] C. Gómez, F. J. Pérez Blanco, J. M. Vara, V. De Castro, and E. Marcos, "Design and development of Smart Contracts for E-government through Value and Business Process Modeling," in *Proceedings of the 54th Hawaii International Conference on System Sciences*, 2021, p. 2069.
- [49] F. Kaya and J. Gordijn, "DECENT: An Ontology for Decentralized Governance in the Renewable Energy Sector," in *2021 IEEE 23rd Conference on Business Informatics (CBI)*, 2021, vol. 1, pp. 11–20.
- [50] A. Hevner and S. Chatterjee, "Introduction to design science research," in *Design Research in Information Systems*, Springer, 2010, pp. 1–8.
- [51] G. Tebes, D. Peppino, M. B. Rivera, P. Becker, F. Papa, and L. Olsina, "Especificación del Proceso de Design Science Research: Caso Aplicado a una Ontología de Testing de Software," 2019.
- [52] R. J. Wieringa, *Design science methodology for information systems and software engineering*. Springer, 2014.
- [53] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design science in information systems research," *MIS Q.*, pp. 75–105, 2004.
- [54] A. R. Hevner, "A three cycle view of design science research," *Scand. J. Inf. Syst.*, vol. 19, no. 2, p. 4, 2007.
- [55] H. Simon, *The Sciences of Artificial*, 3rd Editio. Cambridge, MA, 1996.

- [56] J. Biolchini, P. Gomes Mian, A. Candida Cruz Natali, and G. Horta Travassos, "Systematic Review in Software Engineering," 2005.
- [57] B. K. and S. Charters, "Guidelines for performing Systematic Literature Reviews in Software Engineering," 2007.
- [58] L. Kish, *Some statistical problems in research design*. American Sociological Review, 1959.
- [59] M. Serrano, M. Piattini, C. Calero, M. Genero, D. Miranda, and G. Alarcos, "Un método para la definición de métricas de software," in *1er Workshop en Métodos de Investigación y Fundamentos filosóficos en Ingeniería del Software y Sistemas de Información (MIFISIS'2002)*, 2002, pp. 65–74.
- [60] M. V. Zelkowitz and D. R. Wallace, "Experimental models for validating technology," *Computer (Long. Beach. Calif.)*, vol. 31, no. 5, pp. 23–31, 1998.
- [61] R. K. Yin, "Case study research: design and methods. Vol. 5 California." Thousand Oaks: Sage Publications, 1994.
- [62] P. Baxter and S. Jack, "Qualitative case study methodology: Study design and implementation for novice researchers." *The Qualitative Report*, vol. 13 (4), pp. 544–559, 2008.
- [63] K. M. Eisenhardt, "Building Theories from Case Study Research," *Acad. Manag. Rev.*, vol. 14, no. 4, pp. 532–550, 1989, [Online]. Available: <http://www.jstor.org/stable/258557>.
- [64] R. E. Stake, *The art of case study research*. sage, 1995.
- [65] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, *Experimentation in software engineering*. Springer Science & Business Media, 2012.
- [66] A. Jedlitschka and D. Pfahl, "Reporting guidelines for controlled experiments in software engineering," in *2005 International Symposium on Empirical Software Engineering, 2005.*, 2005, pp. 10--pp.
- [67] A. Maurya, *Running lean: iterate from plan A to a plan that works*. " O'Reilly Media, Inc.," 2012.
- [68] R. Pichler, "The Product Canvas." Obtenido de [http://www. romanpichler. com/tools/product-canvas](http://www.romanpichler.com/tools/product-canvas), 2013.
- [69] G. Geerts and W. E. McCarthy, "An accounting object infrastructure for knowledge-based enterprise models," *IEEE Intell. Syst. Their Appl.*, vol. 14, no. 4, pp. 89–94, 1999.
- [70] OMG, "UML 2.1.1 Formal Specification." OMG Document - formal/07-02-03, 2007.
- [71] J. L. Peterson, "Petri nets," *ACM Comput. Surv.*, vol. 9, no. 3, pp. 223–252, 1977.
- [72] S. E. Castillo-López, S. K. Flores Guirao, E. Roldán-Reyes, A. A. Maldonado Macias, and D. M. Duque, "A service design process based on the business model CANVAS and the CK theory," in *Managing innovation in highly restrictive environments*, Springer, 2019, pp. 279–301.
- [73] J. Gordijn, E. Yu, and B. Van Der Raadt, "E-service design using i* and e/sup 3/value modeling," *IEEE Softw.*, vol. 23, no. 3, pp. 26–33, 2006.
- [74] P.-T. Chuang, "Combining service blueprint and FMEA for service design," *Serv. Ind. J.*, vol. 27, no. 2, pp. 91–104, 2007.
- [75] Y. Kazemzadeh, S. K. Milton, and L. W. Johnson, "Process Chain Network (PCN) and Business Process Modeling Notation (BPMN): A Comparison of Concepts," *J. Manag. Strateg.*, vol. 6, no. 1, 2015, doi: 10.5430/jms.v6n1p88.
- [76] K. Tschimmel, "Design Thinking as an effective Toolkit for Innovation," in *ISPIM Conference Proceedings*, 2012, p. 1.
- [77] J. Gordijn and H. Akkermans, "e3-value : Design and Evaluation of e-Business

- Models,” *IEEE Intell. Syst.*, vol. 16, no. 4, pp. 11–17, 2001.
- [78] S. E. Sampson and M. Passey, “Introduction to Process-Chain-Network Analysis,” *Brigham Young Univ. Provo, Utah*, 2011.
- [79] I. Compagnucci, F. Corradini, F. Fornari, and B. Re, “Trends on the Usage of BPMN 2.0 from Publicly Available Repositories,” in *International Conference on Business Informatics Research*, 2021, pp. 84–99.
- [80] OMG, “Business Process Model and Notation (BPMN).” 2013, [Online]. Available: <https://www.omg.org/spec/BPMN>.
- [81] M. M. Al-Debi, R. El-Haddadeh, and D. Avison, “Defining the business model in the new world of digital business,” *AMCIS 2008 Proc.*, p. 300, 2008.
- [82] M. Kajanus, A. Iire, T. Eskelinen, M. Heinonen, and E. Hansen, “Business model design: new tools for business systems innovation,” *Scand. J. For. Res.*, vol. 29, no. 6, pp. 603–614, 2014.
- [83] T.-P. Liang, Y.-W. Wang, and P.-J. Wu, “A system for service blueprint design,” in *2013 Fifth International Conference on Service Science and Innovation*, 2013, pp. 252–253.
- [84] H. Weigand, “The e3value ontology for value networks: Current state and future directions,” *J. Inf. Syst.*, vol. 30, no. 2, pp. 113–133, 2016.
- [85] S. Boßelmann and T. Margaria, “Domain-specific business modeling with the business model developer,” in *International Symposium On Leveraging Applications of Formal Methods, Verification and Validation*, 2014, pp. 545–560.
- [86] Y. Rhazali, Y. Hadi, and A. Mouloudi, “Model Transformation with ATL into MDA from CIM to PIM Structured through MVC,” *Procedia Comput. Sci.*, vol. 83, pp. 1096–1101, 2016.
- [87] O. Gjoni, “Bizagi Process Management Suite as an Application of the Model Driven Architecture Approach for Developing Information Systems,” *Acad. J. Interdiscip. Stud.*, vol. 3, no. 6, p. 475, 2014.
- [88] N. Chabanoles and P. Ozil, “Bonita BPM: An open-source BPM-based application development platform to build adaptable business applications,” 2015.
- [89] S. Ivanov, A. A. Kalenkova, and W. M. P. van der Aalst, “BPMNDiffViz: A Tool for BPMN Models Comparison..” in *BPM (Demos)*, 2015, pp. 35–39.
- [90] K. Clauberg and T. William, “BPM and Simulation,” *Signavio Inc, Spain*, 2013.
- [91] B. Özenç and E. Solak, “Visual modelling of morphology.”
- [92] M. Dirix, X. Le Pallec, and A. Muller, “Software support requirements for awareness in collaborative modeling,” in *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, 2014, pp. 382–399.
- [93] A. Faulkner and Contributor, “Lucidchart for easy workflow mapping,” *Ser. Rev.*, vol. 44, no. 2, pp. 157–162, 2018.
- [94] D. J. Parker, *Microsoft visio 2013 business process diagramming and validation*. Packt Publishing Ltd, 2013.
- [95] D. Szopinski, T. Schoormann, T. John, R. Knackstedt, and D. Kundisch, “How Software Can Support Innovating Business Models: A Taxonomy of Functions of Business Model Development Tools,” 2017.
- [96] L. J. Lee, “Tools: MIRO Real-time board, visual collaborations and tools, easy screen sharing and presentation,” 2019.
- [97] I. M.-M. de Oca, M. Snoeck, H. A. Reijers, and A. Rodríguez-Morffi, “A systematic literature review of studies on business process modeling quality,” *Inf. Softw. Technol.*, vol. 58, pp. 187–205, 2015.
- [98] G. Polančič and B. Cegnar, “Complexity metrics for process models--A systematic literature review,” *Comput. Stand. Interfaces*, vol. 51, pp. 104–117, 2017.
- [99] M. Estanol, M.-R. Sancho, and E. Teniente, “Verification and validation of UML

- artifact-centric business process models,” in *International Conference on Advanced Information Systems Engineering*, 2015, pp. 434–449.
- [100] J. Solís-Martínez, J. P. Espada, B. C. P. G-Bustelo, and J. M. C. Lovelle, “BPMN MUSIM: Approach to improve the domain expert’s efficiency in business processes modeling for the generation of specific software applications,” *Expert Syst. Appl.*, vol. 41, no. 4, pp. 1864–1874, 2014.
- [101] E. Umuhoza, M. Brambilla, D. Ripamonti, and J. Cabot, “An empirical study on simplification of business process modeling languages,” in *Proceedings of the 2015 ACM SIGPLAN International Conference on Software Language Engineering*, 2015, pp. 13–24.
- [102] J. Horkoff *et al.*, “Strategic business modeling: representation and reasoning,” *Softw. Syst. Model.*, vol. 13, no. 3, pp. 1015–1041, 2014.
- [103] D. C. Schmidt, “Model-driven engineering,” *IEEE Comput.*, vol. 39, no. 2, pp. 25–31, 2006.
- [104] M. Völter, “From Programming to Modeling-and Back Again,” *IEEE Softw.*, vol. 28, no. 6, pp. 20–25, 2011.
- [105] J. G. Molina, F. O. G. Rubio, V. Pelechano, A. Vallecillo, J. M. Vara, and C. Vicente-Chicote, *Desarrollo de software dirigido por modelos: conceptos, métodos y herramientas*. Ra-Ma, 2013.
- [106] B. Selic, “The Pragmatics of Model-Driven Development,” *IEEE Comput.*, vol. 20, no. 5, pp. 19–25, 2003.
- [107] M. Fowler, *Code Generation for Dummies*. METHODS & TOOLS, Spring, 2009.
- [108] I. Abdelhalim, J. Sharp, S. Schneider, and H. Treharne, “Formal Verification of Tokeneer Behaviours Modelled in fUML Using CSP,” *Form. Methods Softw. Eng. Springer Berlin / Heidelb.*, vol. 6447, pp. 371–387, 2010.
- [109] M. Dausend and F. Raiser, “Model Transformation using Constraint Handling Rules as a basis for Model Interpretation,” in *Proceedings of the Eighth International Workshop on Constraint Handling Rules*, 2011, pp. 64–78.
- [110] G. Edwards, C. Seo, and N. Medvidovic, “Model Interpreter Frameworks: A foundation for the analysis of Domain-specific software architectures,” *J. Univers. Comput. Sci.*, vol. 14, no. 8, pp. 1182–1206, 2008.
- [111] S. Li, S. Liu, X. Wang, and Z. Geng, “A Research and Implementation of Model Execution Method Based on MOF,” in *Proceedings of the Third International Symposium on Computer Science and Computational Technology(ISCSCCT '10)*, 2010, pp. 91–93.
- [112] J. M. Favre, “Towards a Basic Theory to Model – Model Driven Engineering,” 2004.
- [113] F. Flore, “MDA: The proof is in automating transformations between models,” *OptimalJ White Paper*. pp. 1–4, 2003.
- [114] D. S. Frankel, “Model Driven Architecture: Applying MDA to Enterprise Computing.” New York, USA: John Wiley & Sons, 2002.
- [115] M. Guttman and J. Parodi, “Real-life MDA: solving business problems with model driven architecture.” Morgan Kaufmann, 2007.
- [116] A. G. Kleppe, J. Warmer, and W. Bast, “MDA explained: the model driven architecture: practice and promise.” Addison-Wesley, 2003.
- [117] S. J. Mellor, S. Kendall, A. Uhl, and D. Weise, “MDA distilled.” Addison Wesley, 2004.
- [118] B. Selic, “MDA manifestations,” *Eur. J. Informatics Prof.*, vol. IX, no. 2, pp. 12–16, 2008.
- [119] A. Watson, “A brief history of MDA,” *Eur. J. Informatics Prof.*, vol. 9, no. 2, pp. 7–11, 2008.

- [120] OMG, "The Meta Object Facility (MOF) Core Specification." version 2.0, OMG Document - formal/06-01-01, 2006.
- [121] OMG, "Object Constraint Language Specification (OCL)." version 2.0. OMG Document - formal/2006-05-01, 2006.
- [122] OMG, "XML Metadata Interchange (XMI) specification." version 2.1.1. OMG Document - formal/2007-12-01, 2007.
- [123] I. Kurtev, "Adaptability of model transformations," University of Twente, Twente, 2005.
- [124] D. M. Sánchez, J. M. Caverio, and E. Marcos, "The concepts of model in information systems engineering: a proposal for an ontology of models," *Knowl. Eng. Rev.*, vol. 24, no. 1, pp. 5–21, 2009.
- [125] G. Booch, J. Rumbaugh, and I. Jacobson, *The unified modeling language user guide*. Addison-wesley, 1999.
- [126] J. Teixeira, L. Patricio, N. J. Nunes, L. Nóbrega, R. P. Fisk, and L. Constantine, "Customer experience modeling: from customer experience to service design," *J. Serv. Manag.*, 2012.
- [127] R. S. Pressman, *Ingeniería del Software, un enfoque práctico*. Madrid: McGraw-Hill, Interamericana de España, 2002.
- [128] I. Sommerville, *Ingeniería del Software*. Madrid: Pearson Educación, 2005.
- [129] C. Atkinson and T. Kuhne, "Model-Driven Development: A Metamodeling Foundation," *IEEE Softw.*, vol. 20, no. 5, pp. 36–41, 2003.
- [130] T. Clark, P. Sammut, and J. Willans, *Applied metamodelling: a foundation for language driven development*. 2008.
- [131] E. Seidewitz, "What models mean," *IEEE Softw.*, vol. 20, no. 5, pp. 26–32, 2003.
- [132] S. Sendall and W. Kozaczynski, "Model transformation: the heart and soul of model-driven software development," *IEEE Softw.*, vol. 20, no. 5, pp. 42–45, 2003.
- [133] N. Aizenbud-Reshef, B. T. Nolan, J. Rubin, and Y. Shaham-Gafni, "Model traceability," *IBM Syst. J.*, vol. 45, no. 3, pp. 515–526, 2006.
- [134] B. Ramesh, C. Stubbs, T. Powers, and M. Edwards, "Requirements traceability: Theory and practice," *Ann. Softw. Eng.*, vol. 3, pp. 397–415, 1997.
- [135] O. C. Z. Gotel and C. W. Finkelstein, "An analysis of the requirements traceability problem," in *Proceedings of the First International Conference on Requirements Engineering*, 1994, pp. 94–101.
- [136] H. Asuncion, "Towards practical software traceability," in *Companion of the 30th international conference on Software engineering*, 2008, pp. 1023–1026.
- [137] R. Brcina and M. Riebisch, "Defining a traceability link semantics for design decision support," in *Proceedings of the 4th European Conference on Model Driven Architecture - Traceability Workshop (ECMDA-TW'08)*, 2008, pp. 39–48.
- [138] S. Melby, "Traceability in Model Driven Engineering," University of Oslo, Norway, 2007.
- [139] S. Winkler and J. von Pilgrim, "A survey of traceability in requirements engineering and model-driven development," *Softw. Syst. Model.*, vol. 9, no. 4, pp. 529–565, 2010.
- [140] D. Kolovos, R. Paige, and F. Polack, "On-demand merging of traceability links with models," 2006.
- [141] G. K. Olsen and J. Oldevik, "Scenarios of traceability in model to text transformations," in *Proceedings of the 3rd European Conference on Model Driven Architecture - Foundations and Applications (ECMDA-FA'07)*, 2007, pp. 144–156.
- [142] M. Völter, "MD*/DSL Best Practices (Version 2.0)." 2011, [Online]. Available:

- <http://voelter.de/data/pub/DSLBestPractices-2011Update.pdf>.
- [143] D. Granada, J. M. Vara, F. P. Blanco, and E. Marcos, "Model-based Tool Support for the Development of Visual Editors-A Systematic Mapping Study.," in *ICSOFT*, 2017, pp. 330–337.
 - [144] Á. J. Rielo, "M2DAT - DB Desarrollo de Editores Gráficos para el Modelado de Bases de Datos Objeto-Relacionales: SQL2003 y Oracle 10g," Universidad Rey Juan Carlos, 2010.
 - [145] V. Vujović, M. Maksimović, and B. Perišić, "Comparative analysis of DSM graphical editor frameworks: Graphiti vs. Sirius," in *23rd International Electrotechnical and Computer Science Conference ERK*, 2014, pp. 7–10.
 - [146] V. Vuyović, M. Maksimović, and B. Perišić, "Sirius: A rapid development of DSM graphical editor," in *IEEE 18th International Conference on Intelligent Engineering Systems INES 2014*, 2014, pp. 233–238, doi: 10.1109/INES.2014.6909375.
 - [147] F. Madiot, M. P.-P. MoDELS, and undefined 2015, "Eclipse Sirius Demonstration.," *ceur-ws.org*, Accessed: Sep. 14, 2021. [Online]. Available: http://ceur-ws.org/Vol-1554/PD_MoDELS_2015_paper_4.pdf.
 - [148] D. S. Kolovos, R. F. Paige, L. M. Rose, and J. Williams, "Integrated model management with Epsilon," in *European Conference on Modelling Foundations and Applications*, 2011, pp. 391–392.
 - [149] D. Kolovos, L. Rose, R. Paige, and A. García-Domínguez, "The epsilon book," *Structure*, vol. 178, pp. 1–10, 2010.
 - [150] D. S. Kolovos, R. F. Paige, and F. A. C. Polack, "Eclipse development tools for epsilon," in *Eclipse Summit Europe, Eclipse Modeling Symposium*, 2006, vol. 20062, p. 200.
 - [151] D. S. Kolovos, R. F. Paige, and F. A. C. Polack, "The epsilon object language (EOL)," in *European Conference on Model Driven Architecture-Foundations and Applications*, 2006, pp. 128–142.
 - [152] D. S. Kolovos, R. F. Paige, and F. A. C. Polack, "The epsilon transformation language," in *International Conference on Theory and Practice of Model Transformations*, 2008, pp. 46–60.
 - [153] D. S. Kolovos, R. F. Paige, F. A. C. Polack, and L. M. Rose, "Update transformations in the small with the epsilon wizard language," *J. Object Technol.*, vol. 6, no. 9, 2003.
 - [154] D. S. Kolovos, "Establishing correspondences between models with the epsilon comparison language," in *European conference on model driven architecture-foundations and applications*, 2009, pp. 146–157.
 - [155] L. M. Rose, R. F. Paige, D. S. Kolovos, and F. A. C. Polack, "The epsilon generation language," in *European Conference on Model Driven Architecture-Foundations and Applications*, 2008, pp. 1–16.
 - [156] D. S. Kolovos, R. F. Paige, and F. A. C. Polack, "Merging models with the epsilon merging language (eml)," in *International Conference on Model Driven Engineering Languages and Systems*, 2006, pp. 215–229.
 - [157] M. Estañol, E. Marcos, X. Oriol, F. J. Pérez, E. Teniente, and J. M. Vara, "Validation of service blueprint models by means of formal simulation techniques," in *International Conference on Service-Oriented Computing*, 2017, pp. 80–95.
 - [158] E. Gómez-Martínez, F. Pérez-Blanco, J. de Lara, J. M. Vara, and E. Marcos, "Formal support of process chain networks using model-driven engineering and petri nets," in *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, 2019, pp. 98–100.

- [159] D. L. Parnas, "On the criteria to be used in decomposing systems into modules," in *Pioneers and Their Contributions to Software Engineering*, Springer, 1972, pp. 479–498.
- [160] I. da Silva Torres, J. Gordijn, M. Fantinato, and J. F. da Fountoura Vieira, "Designing an Ecosystem Value Model Based on a Process Model--An Empirical Approach," in *IFIP Working Conference on The Practice of Enterprise Modeling*, 2020, pp. 293–303.
- [161] F. Hotie and J. Gordijn, "Value-based process model design," *Bus. Inf. Syst. Eng.*, vol. 61, no. 2, pp. 163–180, 2019.
- [162] D. Steinberg, F. Budinsky, E. Merks, and M. Paternostro, "EMF: eclipse modeling framework," 2008, Accessed: Sep. 14, 2021. [Online]. Available: https://books.google.es/books?hl=es&lr=&id=sA0zOZuDXhgC&oi=fnd&pg=PT23&dq=eclipse+modeling+framework&ots=2KKEY0_kFs&sig=qgHskBXML3Co5mFkoIBZQzuFYeU.
- [163] A. K. Soy, "The Case Study as a Research Method. University of Texas." 2005.
- [164] V. R. Basili, F. Shull, and F. Lanubile, "Building knowledge through families of experiments," *IEEE Trans. Softw. Eng.*, vol. 25, no. 4, pp. 456–473, 1999.
- [165] S. L. Pfleeger and B. A. Kitchenham, "Principles of survey research: part 1: turning lemons into lemonade," *ACM SIGSOFT Softw. Eng. Notes*, vol. 26, no. 6, pp. 16–18, 2001.
- [166] F. J. Shull, J. C. Carver, S. Vegas, and N. Juristo, "The role of replications in empirical software engineering," *Empir. Softw. Eng.*, vol. 13, no. 2, pp. 211–218, 2008.
- [167] B. Mora, F. García, F. Ruiz, and M. Piattini, "Graphical versus textual software measurement modelling: an empirical study," *Softw. Qual. J.*, vol. 19, no. 1, pp. 201–233, 2011.
- [168] R. A. Fisher, "On the 'probable error' of a coefficient of correlation deduced from a small sample," *Metron*, vol. 1, pp. 1–32, 1921.
- [169] J. L. Fleiss, *Design and analysis of clinical experiments*. John Wiley & Sons, 2011.
- [170] F. Seehusen, ... K. S.-C. on T. and P. of M., and undefined 2011, "An evaluation of the graphical modeling framework (gmf) based on the development of the coras tool," *Springer*, vol. 6707 LNCS, pp. 152–166, 2011, doi: 10.1007/978-3-642-21732-6_11.
- [171] W. Abohamad, A. Ramy, and A. Arisha, "A hybrid process-mining approach for simulation modeling," in *2017 Winter Simulation Conference (WSC)*, 2017, pp. 1527–1538.

Tabla de Acrónimos

Acrónimo	Descripción
ADM	<i>Architecture-Driven Modernization</i>
BPMN	<i>Business Process Model and Notation</i>
CIM	<i>Computation Independent Model</i>
DSL	<i>Domain Specific Language</i>
DSM	<i>Domain Specific Modeling</i>
DSR	<i>Design Science Research</i>
ECL	<i>Epsilon Comparison Language</i>
EGL	<i>Epsilon Generation Language</i>
EML	<i>Epsilon Merging Language</i>
EMF	<i>Eclipse Modeling Framework</i>
EOL	<i>Epsilon Object Language</i>
ETL	<i>Epsilon Transformation Language</i>
EVL	<i>Epsilon Validation Language</i>
EWL	<i>Epsilon Wizard Language</i>
GEF	<i>Graphical Editing Framework</i>
GMF	<i>Graphical Modeling Framework</i>
GPL	<i>General-purpose language</i>
ICC	<i>Intraclass Correlation Coefficient</i>
IDE	<i>Integrated Development Environment</i>
JDK	<i>Java Development Kit</i>
MDA	<i>Model-Driven Architecture</i>
MDD	<i>Model-Driven Development</i>

Acrónimo	Descripción
MDE	<i>Model-Driven Engineering</i>
MDR	<i>Model-Driven Reengineering</i>
MOF	<i>Meta-Object Facility</i>
OCL	<i>Object Constraint Language</i>
OMG	<i>Object Management Group</i>
PCN	<i>Process Chain Network</i>
PIM	<i>Platform Independent Model</i>
PSM	<i>Platform Specific Model</i>
SBP	<i>Service Blueprint</i>
SI	<i>Sistema de Información</i>
SLR	<i>Systematic Literature Review</i>
TFG	<i>Trabajo Fin de Grado</i>
UML	<i>Unified Modeling Language</i>
XMI	<i>XML Metadata Interchange</i>
XML	<i>Extensible Markup Language</i>