



Universidad de Valladolid

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA

DEPARTAMENTO DE INFORMÁTICA

TESIS DOCTORAL

**LeadFlow4LD: Un Método para Representar
Computacionalmente el Flujo de Actividades y el Flujo de
Datos en Guiones de Situaciones de Aprendizaje
Colaborativo**

Presentada por LUIS PALOMINO RAMÍREZ para optar al grado de
doctor por la Universidad de Valladolid

Dirigida por:

Miguel Luis Bote Lorenzo
Doctor por la Universidad de Valladolid

Juan Ignacio Asensio Pérez
Doctor por la Universidad de Valladolid

Agradecimientos

Resumen

La aproximación de Diseño de Aprendizaje basada en flujo de trabajo (*workflow*) aplicado al dominio educativo (LD - *Learning Design*), busca la representación computacional de procesos de enseñanza-aprendizaje. Estas representaciones computacionales (también denominadas "diseños de aprendizaje") tienen las mismas ventajas y objetivos que los procesos de flujo de trabajo: automatización del proceso, reutilización de la definición del proceso, e interoperabilidad con sistemas y herramientas. Con la automatización del proceso, es posible utilizar sistemas gestores del proceso de aprendizaje que se encargarán de la secuencia de actividades a realizar, así como de la secuencia de invocaciones de herramientas y los artefactos requeridos por las herramientas. Con la reutilización de un diseño de aprendizaje, diseñadores instruccionales definen el diseño de aprendizaje una vez, y luego lo adaptan y lo reutilizan una y otra vez en distintas situaciones de aprendizaje y contextos. Por último, con la especificación de diseños de aprendizaje a través de un lenguaje estándar (como los denominados Lenguajes de Modelado Educativo o EML - *Educational Modeling Language*), educadores pueden intercambiar la lógica del diseño de sus cursos entre sistemas y herramientas.

Un diseño de aprendizaje consiste en describir la secuencia de actividades de aprendizaje y recursos educativos (llamado el flujo de aprendizaje), así como la secuencia de invocaciones de herramientas y la relación entre datos y herramientas (llamado el flujo de datos). En un diseño de aprendizaje se puede describir el flujo de datos a nivel de actividades, pero no a nivel de herramientas. Como resultado, los participantes (no el sistema) son los responsables de gestionar manualmente el flujo de datos, impidiendo alcanzar el objetivo de automatización del flujo de datos. Además, en escenarios de aprendizaje colaborativo se requiere describir también la secuencia de invocaciones de instancias de herramientas, así como la relación entre instancias de datos e instancias de herramientas. Sin embargo, tal descripción del flujo de datos a nivel de instancias no permite alcanzar el objetivo de reutilización del diseño.

En esta tesis se presenta LeadFlow4LD, un método para describir diseños de aprendizaje de una forma interoperable con estándares actuales. LeadFlow4LD está basado en la composición del flujo de aprendizaje descrito en el lenguaje estándar de facto de LD (IMS LD), y el flujo de datos a nivel de herramientas descrito en un lenguaje estándar de flujo de trabajo (XPDL). El problema de reutilización del diseño de aprendizaje en escenarios de aprendizaje colaborativo, se resuelve separando la definición de la estructura del proceso de la definición de su instanciación. LeadFlow4LD ha sido evaluado a través de la especificación y puesta en marcha de distintas situaciones de aprendizaje colaborativo significativas. Los resultados proporcionan evidencias sobre la adecuación de la expresividad del método propuesto para automatizar el proceso, reutilizar el diseño de aprendizaje, y alcanzar distintos niveles de interoperabilidad con sistemas y herramientas de ambas

comunidades LD y flujo de trabajo. Además, con el trabajo propuesto se fomentan las prácticas y tecnologías dentro de la comunidad LD.

Abstract

The workflow-based Learning Design Approach applied in the educative domain (LD - *Learning Design*), is used to achieve such a computer representation of teaching-learning processes as *learning designs*. Such a computer representation (also named *learning designs*) has the same advantages and goals as workflow processes: process automation, process definition reuse and interoperability between systems and tools. On the one hand, by process automation, the different actors of the teaching-learning process would not have to be concerned about the sequence of learning activities nor the resources supporting an activity because the learning management system would be responsible to deliver them automatically. On the other hand, by the *learning design* reuse, the educator specifies the *learning design* just once, then suit and reuse the *learning design* again and again for different contexts and learning situations. Finally, by specifying a *learning design* using a standard EML (Educational Modeling Language), educators around the world could interchange the rationale of their course designs between each other and between systems and tools.

A *learning design* consists in modelling, not only the sequence of learning activities and educative resources (called the learning flow), but also the sequence of invocations of tools needed to carry out the learning activities as well as the relationship between data and tools (called the data flow). In a *learning design* is feasible modeling the data flow at the level of activities but not at the level of tools. Then, the participants (not the system) are the own responsible to handle the data flow, impeding to achieve the goal of data flow automation. Moreover, in collaborative learning scenarios, is necessary to model also the sequence of tool instances as well as the relationship between data and tool instances (called an instance of the data flow). Nevertheless, such instance-level data flow model, impede to achieve the goal of *learning design* reuse.

In this thesis, we present LeadFlow4LD, a method used to achieve such a computational representation of collaborative learning processes as *learning designs*, in an interoperable and standard way. LeadFlow4LD comprises of modeling the learning flow, specified by the standard de facto language in LD (IMS-LD) and modeling the tool-level data flow, specified by a standard workflow language (XPDL). The learning design reuse issue in collaborative learning scenarios is addressed by separating the process structure definition from the process instantiation definition. The proposed method has been assessed through the specification and enactment of different relevant collaborative learning situations. Results provide evidence about the adequacy and the expressiveness of the proposed method in order to achieve the process automation (learning flow and data flow automation), the learning design reuse, as well as different levels of interoperability between systems and tools in both LD and workflow communities. Moreover, with this proposal the practices and technologies within the LD community are fostered.

Índice general

1	Introducción	2
1.1	Objetivos y contribuciones esperadas en esta Tesis Doctoral	8
1.2	Metodología de investigación	10
1.3	Estructura del documento.....	11
2	Contexto de investigación y estado del arte	14
2.1	Introducción	14
2.2	Situaciones y estructuras de aprendizaje colaborativo	15
2.2.1	Definición.....	15
2.2.2	Patrones de procesos de aprendizaje colaborativo	16
2.3	Guiones colaborativos	17
2.3.2	Caracterización de guiones colaborativos	19
2.3.3	Flujo de aprendizaje	19
2.3.4	Flujo de datos	20
2.3.5	Representación de guiones colaborativos	20
2.3.6	Lenguajes de Modelado Educativo	21
2.4	Diseño de aprendizaje	23
2.4.1	Evolución del campo.....	23
2.4.2	Interpretaciones	25
2.4.3	Herramientas y sistemas del campo de Diseño de Aprendizaje.....	28
2.4.4	Diseño de aprendizaje en el contexto de esta Tesis Doctoral.....	32
2.5	IMS LD	34
2.5.2	Requisitos de especificaciones de interoperabilidad y estándares	35
2.5.3	Características de IMS LD	35
2.5.4	Definición y objetivos de IMS LD.....	36
2.5.6	Modelo conceptual de IMS LD.....	37
2.5.7	El ciclo de vida de diseños de aprendizaje especificados con IMS LD	38
2.6	Expresividad en LD para describir guiones colaborativos	38
2.6.1	Expresividad en LD para describir el flujo de aprendizaje en guiones colaborativos	40

2.6.2	Expresividad en LD para describir el flujo de datos en guiones colaborativos	41
2.7	Análisis del problema del flujo de datos a nivel de herramientas	45
2.7.1	Análisis de un ejemplo ilustrativo: revisión entre pares.....	45
2.7.2	Análisis de un ejemplo ilustrativo: convergencia del conocimiento	50
2.7.3	Análisis de un caso de estudio: el caso MOSAIC	51
2.7.4	Planteamiento del problema.....	59
2.8	Flujo de trabajo	60
2.8.1	Definición.....	60
2.8.2	Ciclo de vida de procesos de flujo de trabajo.....	60
2.8.3	Ventajas y desventajas de la aproximación de flujo de trabajo.....	63
2.8.4	Caracterización de los procesos de flujo de trabajo	64
2.8.5	Complementariedad de LD y flujo de trabajo	65
2.8.6	Relación entre LD y Flujo de Trabajo.....	66
2.9	Trabajo relacionado.....	68
2.9.1	Adecuación de IMS LD	68
2.9.2	Extensión de IMS LD.....	69
2.9.3	Descripción del flujo de datos en el campo de flujo de trabajo	70
2.9.4	Contribuciones de flujo de trabajo a escenarios de aprendizaje.....	74
2.9.5	Contribuciones de flujo de trabajo en LD	76
2.9.6	Discusión.....	79
2.10	Conclusiones	80
3	LeadFlow4LD: Definición y Puesta en Marcha del Método Propuesto	84
3.1	Introducción	84
3.2	Selección de una aproximación para la solución del problema.....	85
3.3	Selección del lenguaje de flujo de trabajo para describir el flujo de datos	86
3.3.1	Lenguajes de Flujo de Trabajo	87
3.3.2	Evaluación de lenguajes de flujo de trabajo.....	88
3.3.3	XPDL	93
3.4	Especificación de un diseño de aprendizaje con LeadFlow4LD.....	95
3.4.1	Vista general de LeadFlow4LD	95
3.4.2	Especificación de la estructura de aprendizaje colaborativo.....	96

3.4.3	El documento de coordinación de flujos	105
3.4.4	Instanciación de la estructura de aprendizaje colaborativo	107
3.4.5	Documento de instanciación de grupos.....	110
3.4.6	Documento de población de grupos	110
3.4.7	Distribución de diseños de aprendizaje conformes a LeadFlow4LD: UoLeadFlow4LD	112
3.5	Aproximaciones de implementación del sistema de puesta en marcha de una UoLeadFlow4LD	114
3.6	Sistema de puesta en marcha de una UoLeadFlow4LD.....	116
3.6.1	Arquitectura del sistema propuesto	118
3.6.2	Funcionalidad del Componente Analizador LeadFlow4LD	120
3.6.3	Procedimiento a seguir por el componente analizador LeadFlow4LD	123
3.6.4	Arquitectura del sistema prototipo de puesta en marcha.....	130
3.6.5	Selección de una infraestructura tecnológica para un prototipo de puesta en marcha	131
3.6.6	Discusión.....	132
3.7	Conclusiones	134
4	Evaluación de LeadFlow4LD	136
4.1	Introducción	136
4.2	Metodología de evaluación	137
4.3	Selección y justificación de situaciones de aprendizaje.....	142
4.4	Ejemplo ilustrativo: revisión entre pares.....	143
4.4.1	Especificación de la estructura de aprendizaje colaborativo.....	144
4.4.2	Especificación de la instanciación de la estructura	145
4.4.3	Puesta en marcha de las situaciones de aprendizaje.....	146
4.4.4	Discusión.....	149
4.4.5	Comparación con otras soluciones.....	150
4.5	Ejemplo ilustrativo: convergencia del conocimiento	150
4.5.1	Especificación de la estructura de aprendizaje colaborativo.....	151
4.5.2	Especificación de la instanciación de la estructura	154
4.5.3	Puesta en marcha de la situación de aprendizaje.....	158
4.5.4	Discusión.....	163

4.5.5	Comparación con otras soluciones	164
4.6	Ejemplo ilustrativo: caso MOSAIC	165
4.6.1	Especificación de la estructura de aprendizaje colaborativo	166
4.6.2	Especificación de la instanciación de la estructura	171
4.6.3	Puesta en marcha de la situación de aprendizaje.....	175
4.6.4	Discusión.....	179
4.6.5	Comparación con otras soluciones.....	181
4.7	Discusión global.....	182
4.7.1	Efectos colaterales.....	185
4.8	Conclusiones	186
5	Conclusiones y trabajo futuro	190
5.1	Conclusiones	190
5.2	Trabajo futuro.....	196
Apéndice A	Caso de estudio MOSAIC	200
A.1	Resultados	200
Apéndice B	Esquemas XML Propuestos	206
B.1	Esquema del documento de coordinación entre flujos.....	206
B.2	Esquema del documento de población de grupos	207
Apéndice C	Especificación de situaciones de aprendizaje de los ejemplos.....	210
C.1	Instancia de la Estructura del Flujo de Datos del Caso MOSAIC.....	210
Apéndice D	Puesta en marcha de una UoLeadFlow4LD.....	226
D.1	Documentos de estructuras y situaciones.....	226
	Debido a la gran cantidad de documentos que describen todas las estructuras y situaciones de aprendizaje colaborativo, y que fueron usados para validar el método propuesto, todos estos documentos son incluidos en el disco anexo en la ruta: \ejemplos. Un resumen de tales documentos es mostrado en la <i>Tabla D.1</i>	226
D.2	Listado de software	228
D.3	Procedimiento de preparación de la puesta en marcha de una UoLeadFlow4LD	228
D.4	Despliegue del motor de ejecución del flujo de aprendizaje.....	248
D.5	Despliegue del motor de ejecución de flujo de trabajo	249
D.6	Adaptador del motor de flujo de aprendizaje	252
D.7	Adaptador del motor de flujo de trabajo	255

D.8	Adaptador de herramientas Web	260
D.9	Publicación e instanciación del documento del flujo de aprendizaje sincronizado	261
Apéndice E	Contenido del disco anexo	264
E.1	Estructura del contenido.....	264
Referencias.....		266

1 Introducción

La aplicación de las tecnologías de la información y la comunicación (TIC) en el ámbito educativo ha provocado el surgimiento de nuevos paradigmas de enseñanza y procesos de aprendizaje (Sife, Edda & Camilius, 2007). Las distintas formas de concebir el uso de la tecnología para dar soporte y mejorar el aprendizaje han provocado el surgimiento del llamado Aprendizaje Mejorado por Tecnología (TEL – *Technology Enhanced Learning*) (Chan, Roschelle, Hsi, Kinshuk, Sharples, Brown & Hoppe, 2006). En este contexto, el paradigma de las tecnologías de aprendizaje en el que se enmarca esta Tesis Doctoral es el Aprendizaje Colaborativo Soportado por Ordenador (CSCL – *Computer Supported Collaborative Learning*) (Koschmann, 1996). El CSCL es un campo de investigación multidisciplinario que busca usar la tecnología para apoyar la adquisición de conocimiento y habilidades cuando se producen interacciones entre los participantes del proceso de aprendizaje colaborativo (Stahl, Koschmann & Suthers, 2006).

El aprendizaje colaborativo es un proceso por el cual los estudiantes pueden adquirir conocimiento, habilidades, actitudes y valores a través de la realización de actividades colaborativas e individuales con el propósito de alcanzar un objetivo común (Cabrera, Donoso & Marín, 1993). El caso de los patrones de procesos o técnicas de aprendizaje colaborativo son un ejemplo de buenas prácticas de aprendizaje colaborativo. Estos patrones o técnicas son procesos de aprendizaje guiados y adaptados por educadores para cumplir sus necesidades instruccionales (Barkley, Cross & Major, 2005). Algunos ejemplos de estos patrones o técnicas de aprendizaje colaborativo son: revisión entre pares, lluvia de ideas, mesa redonda, *jigsaw*, etc. (Barkley, Cross & Major, 2005), los cuales pueden ser soportados o no mediante el uso de tecnología.

Una forma de aplicar la tecnología para mejorar el aprendizaje colaborativo puede ser a través de la representación computacional de procesos de aprendizaje colaborativo. Tal representación computacional tiene ventajas, por ejemplo, el uso de sistemas basados en ordenador para guiar **automáticamente** a los participantes de un curso (electrónico o cara a cara) en la secuenciación de las actividades de aprendizaje, así como en la gestión del flujo de información que soporta el proceso de aprendizaje (Koper & Miao, 2008). O bien, para que la descripción del proceso de aprendizaje pueda ser fácilmente editada y adaptada por otros educadores para su **reutilización** efectiva en distintas situaciones y contextos (Donald, Blake, Girault, Datt, & Ramsay, 2009). Adicionalmente, si la representación computacional de tales procesos se realiza de una forma **interoperable** y estándar, entonces se tiene la ventaja adicional de que los procesos de aprendizaje pueden ser compartidos entre educadores e intercambiados entre sistemas y herramientas (Koper & Miao, 2008).

Los Lenguajes de Modelado Educativo (EML –*Educative Modeling Languages*) han surgido con el propósito de que diseñadores instruccionales y educadores puedan describir computacionalmente procesos de aprendizaje, incluyendo el aprendizaje colaborativo

1. Introducción

(Rawlings, van Rosmalen, Koper, Rodríguez-Atacho & Lefrere, 2002). El primer paso hacia la representación computacional de procesos de aprendizaje colaborativo usando un EML, puede ser su descripción como guiones colaborativos (*collaboration scripts*). Un guion colaborativo (Kollar, Fischer & Slotta, 2007) describe (computacionalmente o no) la forma en que los estudiantes colaboran: fases de trabajo, roles, distribución de tareas o actividades, entregables, etc. Este contrato de colaboración puede realizarse a través de instrucciones iniciales dadas por el profesor –sin el uso de tecnología–, o guiado a lo largo de todo el proceso de aprendizaje –con el soporte de un entorno tecnológico– (Dillenbourg & Jermann, 2007).

Distintos EML pueden usarse para representar computacionalmente guiones colaborativos (Martel, Vignollet, Ferraris, & Durand, 2006) (Caeiro-Rodríguez & Llamas-Nistal, 2006) (Miao, Hoeksema, Hoppe & Harrer, 2005), pero es la especificación IMS LD (IMS LD, 2003) el EML que más atención ha atraído de la comunidad investigadora (Derntl, Neuman, & Oberhuemer, 2011) (König & Paramythis, 2010). IMS LD tiene su origen en el EML de la Universidad Abierta de Holanda (OUNL) (Koper & Tattersall, 2005) y se usa como formato de interoperabilidad por un gran número de herramientas de autoría: Web Collage (Villasclaras-Fernández, Hernández-Leo, Asensio-Pérez, & Dimitriadis, 2013); OpenGLM (Derntl, Neumann, & Oberhuemer, 2011); CADMOS (Katsamani & Symeon, 2011); SceneEdit (Emin, Pernin, & Aguirre, 2010); o GLM (Neumann, Oberhuemer, & Derntl, 2009B).

En la actualidad, IMS LD continúa siendo tomado en cuenta en investigación y desarrollo de tecnologías en *e-learning*. Por ejemplo, en el desarrollo de nuevas aproximaciones de autoría y entrega de cursos flexibles en línea (Hermans, Janssen & Koper, 2014). O el caso de CoLearn (Stylianakis, Moumoutzis, Arapi, Mylonakis & Christodoulakis, 2014), que es una plataforma que se puede integrar por encima de infraestructuras existentes de aprendizaje electrónico (VLE –*Virtual Learning Environments*, Entornos de Aprendizaje Virtual), con el propósito de ofrecer funcionalidad del flujo de aprendizaje basado en IMS LD. O como el caso de Liferay (Vogten & Koper, 2014), que es un portal que integra la funcionalidad de distintos servicios *portlets*, y que ha sido extendido con *portlets* educativos con el propósito de convertir Liferay en un VLE. Uno de tales *portlets*, es el *portlet* de secuenciación, el cual permite la construcción de diseños de aprendizaje en el nivel A de IMS LD.

1. Introducción

IMS LD es considerado el principal impulsor del campo de Diseño de Aprendizaje (*Learning Design, LD*) (MacLean & Scot, 2007) (Dalziel, 2012). Además, debido a que se usa como formato de interoperabilidad por un gran número de herramientas de autoría, es considerado por distintos investigadores el estándar de facto del campo LD (Derntl, Neuman & Oberhuemer, 2011) (König & Paramythis, 2010). El término diseño de aprendizaje tiene distintos significados en la literatura (Mor, Craft & Hernández-Leo, 2013) (Dalziel, 2012) (Berggren, Burgos, Fontana, Hinkelman, Hung, Tielemans, 2005) como, por ejemplo, se usa para denotar un campo, una aproximación, un proceso y un producto.

El campo Diseño de Aprendizaje (campo LD) comprende la integración de teorías de diseño instruccional, teorías pedagógicas, las mejores prácticas, y tiene como objetivo soportar a diseñadores instruccionales y educadores en la práctica educativa (Koper & Tattersall, 2005). Pueden existir varias formas de implementar los objetivos del campo LD, una de las cuales es la aproximación de Diseño de Aprendizaje basada en flujo de trabajo aplicado al dominio educativo (LD). LD es similar a la coordinación de actividades que caracteriza a los procesos de flujo de trabajo (Baggetun, Barros, Fesakis, Girardin, Hoeksema, Hämäläinen, Miao & Vantroys, 2004). El término flujo de trabajo (*workflow*), se usa (dentro del campo de Flujo de Trabajo) para enfatizar la coordinación de tareas realizadas por personas o sistemas para lograr un objetivo (van der Aalst & van Hee, 2004). Tres ventajas de la definición de procesos de flujo de trabajo son la **automatización** del proceso, la **reutilización** de la definición del proceso, y la **interoperabilidad** (van der Aalst & van Hee, 2002), las cuales están alineadas con las ventajas mencionadas en la representación computacional de procesos de aprendizaje, así como con los requisitos que debe cumplir la especificación IMS LD (IMS LD, 2003).

El proceso de diseño que sigue la aproximación LD (diseño de aprendizaje como proceso) consiste en describir la secuencia de actividades de aprendizaje que los estudiantes emprenden para cumplir los objetivos de aprendizaje, incluyendo los recursos y mecanismos de soporte requeridos para ayudar a los estudiantes a completar las actividades (Koper & Miao, 2008). Finalmente, el término diseño de aprendizaje también se usa para describir el resultado del proceso de diseño en LD (diseño de aprendizaje como producto). Un diseño de aprendizaje, que describe un proceso de aprendizaje según la aproximación LD, se empaqueta junto con todos los recursos educativos en una unidad de aprendizaje (Koper & Olivier, 2004). Un diseño de aprendizaje que describe una situación de aprendizaje colaborativo, equivale a un guion colaborativo CSCL (*CSCL script*).

El marco de investigación de esta Tesis Doctoral dentro del campo LD, es la aproximación de Diseño de Aprendizaje basada en flujo de trabajo (aproximación LD, en adelante LD), y que comprende un EML estándar de LD (IMS LD), un modelo conceptual (el meta-modelo de IMS LD), así como un número considerable de herramientas y sistemas que son usados para soportar a una comunidad de práctica (la comunidad LD). La ventaja de LD, es que pone especial atención a las actividades de aprendizaje realizadas por los participantes mientras interactúan con recursos a lo largo del proceso de aprendizaje, en lugar de sólo

1. Introducción

enfocarse a la entrega de contenidos educativos, como ocurre con otras aproximaciones de tecnologías de aprendizaje en *e-learning* como, por ejemplo, la aproximación de Objetos de Aprendizaje (*Learning Objects*) (Kopper, 2001).

La representación computacional de procesos de aprendizaje colaborativo se describe entonces en LD como un diseño de aprendizaje. Un diseño de aprendizaje describe un flujo de aprendizaje y un flujo de datos, ambos relacionados entre sí. El **flujo de aprendizaje** describe la secuencia de actividades de aprendizaje, así como los recursos (contenidos educativos, herramientas) necesarios para soportarlas (Koper & Oliver, 2004). El **flujo de datos** describe la secuencia de invocaciones de herramientas, así como la relación entre los artefactos creados por las herramientas y las propias herramientas (Palomino-Ramírez, Martínez-Monés, Bote-Lorenzo, Asensio-Pérez, & Dimitriadis, 2007). Sin embargo, ambos flujos no son independientes entre sí: a cada una de las actividades de aprendizaje (del flujo de aprendizaje) le corresponde (o no) una secuencia de invocaciones de herramientas (del flujo de datos), que es usada para soportar la propia actividad de aprendizaje. Cuando una secuencia de invocaciones de herramientas de una cierta actividad de aprendizaje termina, entonces se continúa con la siguiente actividad de aprendizaje, y con su respectiva secuencia de invocaciones de herramientas. El flujo de aprendizaje y el flujo de datos se alternan repetidas veces a lo largo del proceso de aprendizaje.

Aunque LD ha sido usado con cierto éxito para describir el flujo de aprendizaje en escenarios de aprendizaje colaborativo (Hernández-Leo, Asensio-Pérez & Dimitriadis, 2005), sin embargo, este mismo éxito no se ha alcanzado para describir el flujo de datos. Hay dos aspectos importantes del flujo de datos de un diseño de aprendizaje (colaborativo o no) que no se pueden describir en LD: la **relación entre datos y herramientas** (en LD sólo se describe la relación entre datos y actividades) (Bordies, Villasclaras-Fernández, Dimitriadis & Ruiz-Calleja, 2012) (König & Paramythis, 2010) (Caeiro-Rodríguez, Anido-Rifón, & LLamas-Nistal, 2010) (Neumann, Klebl, Griffiths, Hernández-Leo, la Fuente-Valentín, Hummel & Oberhuemer, 2009) (Miao, Hoeksema, Hoppe & Harrer, 2005) (Peter & Vantroys, 2005) (Wilson, 2005), y la **secuenciación de herramientas** en una misma actividad de aprendizaje (Palomino-Ramírez, Bote-Lorenzo, Asensio-Pérez & Dimitriadis, 2008b).

LD permite describir en un diseño de aprendizaje el flujo de datos a nivel de actividades, pero no a nivel de herramientas. Como consecuencia, los participantes del proceso de aprendizaje son los responsables de gestionar el flujo de datos; por ejemplo, tomando la salida generada por una herramienta y usándola después como entrada de otras herramientas en la misma o en distintas actividades de aprendizaje. Estas limitaciones en LD son especialmente relevantes dentro del contexto CSCL, debido a que muchos procesos de aprendizaje colaborativo incluyen complejos flujos de datos que son difíciles de manejar por los participantes, dificultando por tanto la realización de las actividades de aprendizaje. En este sentido, varios hallazgos reportados en la literatura en (Palomino-Ramírez, Bote-Lorenzo, Asensio-Pérez, Dimitriadis & de la Fuente-Valentín, 2008) sugieren que la gestión

1. Introducción

manual del flujo de datos en procesos de aprendizaje colaborativo incrementa innecesariamente la carga cognitiva de los participantes conduciendo a situaciones propensas a error. Esta limitación de LD para describir el flujo de datos a nivel de herramientas impide alcanzar el objetivo mencionado de automatización de procesos basados en flujo de trabajo, por lo menos en lo que respecta al flujo de datos.

Un diseño de aprendizaje debe describirse de tal forma que otros educadores puedan entenderlo, compartirlo, adaptarlo y usarlo en su propio contexto (Agostino, 2011) (Donald, Blake, Girault, Datt, & Ramsay, 2009). Sin embargo, en aprendizaje colaborativo esto requiere distinguir entre estructuras y situaciones de aprendizaje (tanto para el flujo de aprendizaje como para el flujo de datos). Por un lado, una situación de aprendizaje colaborativo es un escenario de aprendizaje concreto creado con la intención de que los estudiantes construyan conocimiento a través de la realización de una serie de actividades de aprendizaje colaborativo e individual (Osuna-Gómez, 1999). Por otro lado, una estructura de aprendizaje colaborativo es una generalización de situaciones de aprendizaje que comparten una estructura común y que son fácilmente adaptables a las necesidades de múltiples entornos, disciplinas y contenidos (Bote-Lorenzo, 2005) (Hernández-Leo, Asencio-Peréz & Dimitriadis, 2004). Con LD es posible describir la estructura del flujo de aprendizaje, a partir de la cual se puedan derivar o instanciar distintas situaciones de aprendizaje. Sin embargo, esto mismo no se cumple para el flujo de datos. Debido a que no es posible describir en LD la relación entre artefactos y herramientas, entonces no sólo no es posible describir la estructura del flujo de datos a nivel de herramientas, sino que tampoco es posible describir instancias de dicha estructura. Esto es, describir las relaciones que ocurren entre instancias de datos e instancias de herramientas correspondientes a una situación de aprendizaje particular. Esta limitación de LD para describir el flujo de datos a nivel de herramientas tanto en estructuras como en situaciones de aprendizaje colaborativo es relevante, ya que impide alcanzar el objetivo mencionado de reutilización de procesos basados en flujo de trabajo.

Algunos autores en la literatura de LD han propuesto formas indirectas de especificar instancias del flujo de datos pero sólo a nivel de actividades (no a nivel de herramientas), y lo consiguen a través de hacer adecuaciones a la estructura del flujo de aprendizaje. Por ejemplo, en (Koper & Miao, 2008) (Miao, Hoeksema, Hoppe & Harrer, 2005) proponen incluir en el diseño de aprendizaje información de los artefactos producidos por los participantes (esto es, instancias de datos) en la forma de actividades asociadas a roles (se definen tantas actividades y roles como instancias de datos). Sin embargo, con esta aproximación, además de que no se consigue la automatización del flujo de datos, los diseños de aprendizaje resultantes no representan estructuras de aprendizaje, sino solamente situaciones de aprendizaje que no pueden ser reutilizadas en distintas situaciones y contextos.

Existen otras aproximaciones distintas a LD, que también buscan la automatización en *e-learning*. Por ejemplo, en (Rius, Conesa, García-Barriocanal & Sicilia, 2014), se propone un marco ontológico para la automatización de la implementación de procesos educativos. El problema de esta aproximación, es que se centra en la automatización de procesos dentro del

1. Introducción

contexto de objetos de aprendizaje y servicios educativos en *e-learning*, más que en automatizar la secuenciación de actividades de aprendizaje (procesos de flujo de trabajo basados en actividad). Además, para la especificación de procesos, el método requiere definir una ontología, la cual debe ser suficientemente general y aceptada por la comunidad para ser reutilizada en diferentes situaciones y contextos. Otro ejemplo de automatización en CSCL, particularmente en diseño instruccional, es (Challco, Gerosa, Bittencourt & Isotani, 2014). Ellos proponen definir un escenario CSCL como una tarea del diseñador instruccional y un conjunto de recursos del dominio educativo. La aproximación se basa en un método de planificación HTN (*Hierarchical Task Network*, Red Jerárquica de Tareas). En este método, se genera una secuencia de acciones (llamadas planes) que conduce al cumplimiento de ciertos objetivos. Los planes se describen a través de una red jerárquica de tareas, las cuales se descomponen a su vez en tareas no primitivas o sub-tareas, hasta alcanzar un nivel de tareas primitivas que puedan resolverse mediante operadores. El problema de esta aproximación, es que se centra solamente en modelar el flujo de actividades de aprendizaje colaborativo (representado por tareas), pero no dice cómo modelar el flujo de datos. Otra desventaja de esta aproximación, es que no es considerado un estándar en el campo LD.

Tomando en cuenta la relación intrínseca que existe entre LD y flujo de trabajo (Vignollet, Bote-Lorenzo & Asensio-Pérez, 2010) (Mariño, Casallas, Villalobos, Correal & Contamines, 2007) (Baggetun, y otros, 2004), distintos autores han realizado contribuciones del campo de Flujo de Trabajo hacia LD. Por ejemplo, para tomar ventaja de motores existentes de flujo de trabajo y poner en marcha procesos educativos, Vantroys & Peter (2003) proponen la transformación de diseños de aprendizaje especificados en el contexto educativo con IMS LD, a procesos de flujo de trabajo especificados (para su ejecución) en una extensión del lenguaje de flujo de trabajo XPDL (*XML Process Definition Language*, Lenguaje Definición de Procesos basados en XML) (XPDL, 2008). Sin embargo, esta solución no podría aplicarse (como es) para resolver el problema de especificar el flujo de datos, ya que el flujo de datos es parte del contexto educativo y por tanto, de acuerdo a la propuesta de Vantroys debería especificarse en IMS LD, cosa que precisamente IMS LD no puede hacer. Similarmente, con el objetivo de resolver distintos problemas identificados en LD para especificar procesos de aprendizaje colaborativo, Miao et al. proponen un nuevo EML, llamado lenguaje de guiones CSCL (Miao, Hoeksema, Hoppe & Harrer, 2005). Si bien esta solución podría utilizarse para especificar la estructura del flujo de datos, no se indica cómo distinguir entre especificar estructuras y situaciones de aprendizaje colaborativo. Además, el diseño de aprendizaje resultante no sería interoperable con herramientas desarrolladas dentro de la comunidad LD.

Partiendo de que el flujo de aprendizaje puede ser especificado en el lenguaje estándar de LD, y que el flujo de datos puede ser especificado en un lenguaje estándar de flujo de trabajo (Russell, ter Hofstede, Edmond & van der Aalst, 2005), entonces parece razonable que una solución al problema de especificar el flujo de datos a nivel de herramientas en LD puede ser una combinación de lo mejor de los dos mundos. Más aún, puesto que ambos mundos tienen en común el paradigma de flujo de trabajo, es razonable pensar que dicha

solución puede alcanzar los objetivos no alcanzados hasta ahora: automatización del proceso, reutilización de la definición del proceso, e interoperabilidad con sistemas y herramientas.

De acuerdo a lo anterior, este documento presenta una Tesis Doctoral que busca resolver las limitaciones de expresividad del flujo de datos a nivel de herramientas en LD, tanto en estructuras como en situaciones de aprendizaje colaborativo, al mismo tiempo que se facilite la reutilización del diseño de aprendizaje, y se mantenga la interoperabilidad con sistemas y herramientas.

A partir de estas limitaciones identificadas y a la posibilidad de encontrar, en el campo de Flujo de Trabajo, una solución al problema de especificar el flujo de datos a nivel de herramientas, tanto en estructuras como en situaciones de aprendizaje colaborativo, se definen los siguientes objetivos que se pretende alcanzar en la Tesis, así como la metodología de investigación que se seguirá para ello.

1.1 Objetivos y contribuciones esperadas en esta Tesis Doctoral

En esta tesis se persiguen los siguientes objetivos:

Objetivo 1. Conseguir la automatización del flujo de datos en estructuras y situaciones de aprendizaje colaborativo de un diseño de aprendizaje.

Para alcanzar este primer objetivo se buscará representar computacionalmente el flujo de datos, tanto para estructuras como para situaciones de aprendizaje colaborativo, con un lenguaje estándar del campo de Flujo de Trabajo.

Además de este primer objetivo, es posible plantear dos objetivos adicionales. Dichos objetivos, así como la principal contribución original esperada de esta tesis derivada de los mismos, los cuales aparecen recogidos de manera esquemática en la *Figura 1.1*, se describen a continuación:

Objetivo 2. Conseguir la reutilización del diseño de aprendizaje para distintas situaciones y contextos.

Para alcanzar este objetivo se buscará separar en un diseño de aprendizaje (tanto para el flujo de aprendizaje como para el flujo de datos) aspectos reutilizables de aquellos que no lo son.

Objetivo 3. Mantener interoperabilidad del diseño de aprendizaje con sistemas y herramientas que sean conformes a estándares actuales.

Para alcanzar este objetivo se buscará especificar el flujo de datos en algún lenguaje estándar de la comunidad de Flujo de Trabajo, al mismo tiempo que se mantiene la especificación del flujo de aprendizaje en el lenguaje estándar de LD, esto es, IMS LD.

1. Introducción

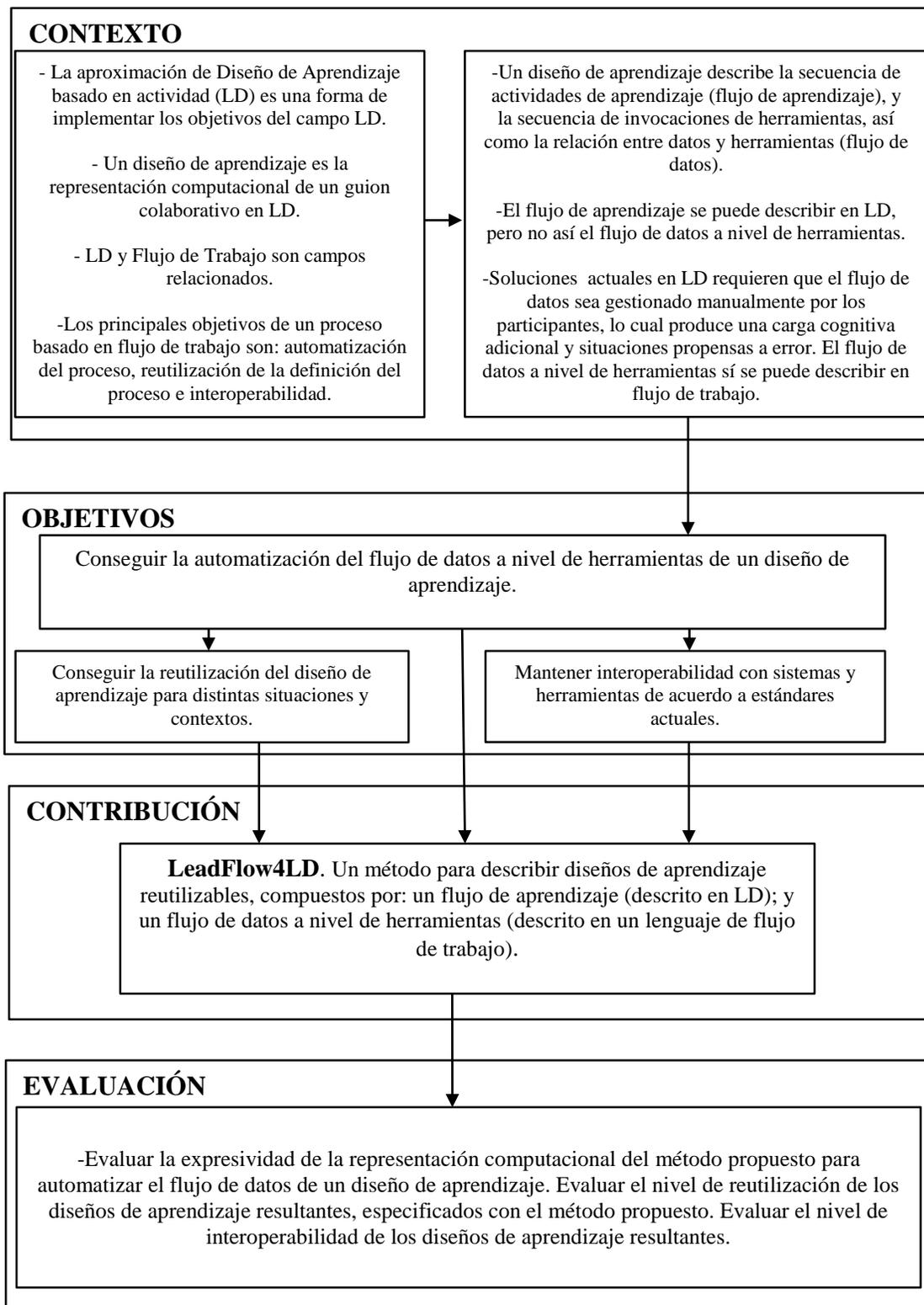


Figura 1.1: Esquema general de la tesis.

1. Introducción

Es importante hacer notar que, de acuerdo con el método de ingeniería, esta metodología sería aplicada de manera iterativa con el objetivo de poder refinar progresivamente las soluciones que se presenten en esta tesis. Así, primero se llevaría a cabo un estudio del problema, se plantearía una primera aproximación para su solución; se regresaría a la definición del problema y así sucesivamente hasta finalmente alcanzar la solución definitiva propuesta.

La consecución de estos tres objetivos, nos llevaría a alcanzar la principal contribución esperada de esta tesis: **LeadFlow4LD: un método para conseguir la representación computacional de estructuras y situaciones de aprendizaje colaborativo reutilizables, basado en la composición del flujo de aprendizaje (especificado en el estándar de LD), y el flujo de datos (especificado en un lenguaje estándar de Flujo de Trabajo).**

Por último, la evaluación de los objetivos de esta tesis se llevará a cabo a través de la evaluación de LeadFlow4LD, la cual debería abarcar, como se muestra en la *Figura 1.1*, tanto la evaluación de la expresividad de la representación computacional del método propuesto, que permita alcanzar la automatización del flujo de datos; evaluar el nivel de reusabilidad de los diseños de aprendizaje resultantes, especificados con el método propuesto; y así también, evaluar el nivel de interoperabilidad de los diseños de aprendizaje resultantes, especificados de igual manera con el método propuesto.

1.2 Metodología de investigación

Para la consecución de los objetivos perseguidos por esta tesis se seguirá la metodología de investigación definida por el denominado método de ingeniería (Adrion, 1993) (Glass, 1995). Dicho método establece que el proceso (iterativo) de investigación debe seguir un total de cuatro etapas que se describen a continuación:

- **Estudiar las soluciones existentes.** Aquí se identifican los problemas que se van a abordar en el trabajo de investigación, así como las soluciones existentes reportadas en la literatura. Por este motivo se llevará a cabo un estudio del problema. Primero, se revisará la literatura de LD buscando referencias al problema de especificar el flujo de datos en LD. Después, se estudiará el modelo conceptual de LD con el objetivo de comprender la relación entre datos y herramientas en el modelo. Además, el problema será analizado también estudiando un caso de estudio real y ejemplos ilustrativos. Por último, se revisará la literatura de flujo de trabajo con el objetivo de identificar alguna relación con el problema.
- **Proponer una mejor solución.** Aquí se debe hacer una contribución original que supere las limitaciones detectadas en las soluciones existentes. Las primeras aproximaciones de la solución pueden surgir directamente de la literatura de LD, según la forma en que otros autores han resuelto el problema. Después, otras soluciones pueden surgir de la relación encontrada entre los campos de LD y Flujo de Trabajo, ya que se sabe que los lenguajes de flujo de trabajo pueden ser usados para

especificar el flujo de datos. Entonces, a partir de un espacio de posibles soluciones y una lista de requisitos que se desea cumpla la solución, se seleccionará la mejor de ellas.

- **Desarrollar la nueva solución.** Aquí se describe e implementa la solución propuesta en la etapa anterior. En este caso, se describe el método que permitirá especificar diseños de aprendizaje reutilizables e interoperables con estándares actuales, tanto para estructuras como para situaciones de aprendizaje colaborativo.
- **Evaluar la nueva solución.** Finalmente, es necesario mostrar que la solución propuesta supera los problemas que fueron detectados en la primera etapa de esta metodología. Para ello se evaluará la expresividad del método propuesto usando una metodología de evaluación basada en ejemplos no triviales, la cual es reconocida como buenas prácticas de evaluación en el campo de la ingeniería de software (Shaw, 2002), la cual es contrastada también con ideas de (Kitchenham, Linkman & Law, 1997) sobre la evaluación de metodologías y herramientas en ingeniería de software.

1.3 Estructura del documento

Después de esta introducción, el capítulo 2 se centra en establecer el marco teórico que sustenta el trabajo de investigación de esta Tesis Doctoral, y que corresponde a la etapa de “estudiar las soluciones existentes” descrita en la metodología de investigación. Para ello, se introduce el CSCL como el contexto de esta investigación y se revisan las ideas más importantes del aprendizaje colaborativo incluyendo los conceptos de situaciones y estructuras de aprendizaje colaborativo, así como el concepto de patrones de procesos o técnicas de aprendizaje colaborativo. Después, se introduce el concepto de guiones colaborativos y su caracterización en un flujo de aprendizaje y un flujo de datos. A continuación, para conseguir la representación computacional de dichos guiones colaborativos se introducen los Lenguajes de Modelado Educativo (EML) y el concepto de Diseño de Aprendizaje. Primero, se hace una revisión de la evolución del campo de Diseño de Aprendizaje, incluyendo las distintas interpretaciones que se dan del término; las principales herramientas y sistemas que existentes en el campo y finalmente se define Diseño de Aprendizaje en el contexto de esta tesis: aproximación de Diseño de Aprendizaje basado en flujo de trabajo (LD). Después, se introduce la especificación IMS LD, como el lenguaje estándar de facto usado por la comunidad LD para especificar y compartir diseños de aprendizaje, así como para mantener interoperabilidad entre herramientas. Se estudia la expresividad en LD para describir el flujo de aprendizaje y el flujo de datos en guiones colaborativos y se estudia cómo se describe el flujo de datos en LD. Se hace un análisis del problema de describir el flujo de datos, primero a través de un par de ejemplos simples pero ilustrativos: el patrón de revisión entre pares y el guion de convergencia del conocimiento, y después a través del análisis de un caso de estudio auténtico y complejo: el caso MOSAIC. Después de este análisis, se lleva a cabo el planteamiento del problema que será tratado en esta Tesis Doctoral. Se estudia la relación de LD con el campo de Flujo de Trabajo, incluyendo definición, ciclo de vida y objetivos del campo. Se caracterizan los procesos de

1. Introducción

flujo de trabajo y a partir de aquí se establece una relación de complementariedad entre LD y Flujo de Trabajo. Por último, se revisa el trabajo relacionado con el problema de describir el flujo de datos en LD: adecuación de IMS LD; extensión de IMS LD; transformación de IMS LD a XPDL; y combinación de IMS LD a XPDL, de donde surge la necesidad de proponer una nueva solución al problema.

En el capítulo 3 se propone un método para resolver las limitaciones identificadas en el capítulo anterior, y que corresponde a la etapa de “proponer y desarrollar una mejor solución” descrita en la metodología de investigación. Primero, se identifican los requisitos que debería cumplir una solución que resuelva los problemas identificados: automatización del proceso, reutilización de la definición del proceso, e interoperabilidad entre sistemas y herramientas. A continuación, tomando en cuenta que el flujo de datos se puede modelar en el campo de Flujo de Trabajo, y a partir de las distintas contribuciones de este campo en LD, se propone un espacio de soluciones del problema, de donde se justifica la selección de una solución por composición de LD y flujo de trabajo. Aquí, se propone utilizar el estándar de LD (IMS LD) para especificar el flujo de aprendizaje de un diseño de aprendizaje, y utilizar un lenguaje estándar de flujo de trabajo para especificar el flujo de datos a nivel de herramientas. Después, se justifica la selección de XPDL como el lenguaje estándar de flujo de trabajo usado para especificar el flujo de datos. Entonces, tomando en cuenta la aproximación por composición propuesta, se describe primero una vista general del método propuesto (llamado LeadFlow4LD) y después se describen cada uno de los documentos que conforman un diseño de aprendizaje especificado conforme a LeadFlow4LD. A continuación, se incluye una sección que describe un sistema prototipo y un procedimiento que será usado en el siguiente capítulo para poner en marcha y validar diseños de aprendizaje especificados conforme al método propuesto.

En el capítulo 4 se lleva a cabo la evaluación de LeadFlow4LD, y que corresponde a la etapa de “evaluar la nueva solución” descrita en la metodología de investigación. Esto se consigue aplicando una metodología de evaluación basada en ejemplos no triviales, que es reconocida como buena práctica en el campo de la ingeniería de software (Shaw, 2002), metodología que es contrastada con ideas de (Kitchenham, Linkman & Law, 1997) sobre la evaluación de metodologías y herramientas en ingeniería de software. Primero, se seleccionan distintas estructuras y situaciones de aprendizaje colaborativo que responden a las necesidades del problema de especificar la automatización del flujo de datos y la reutilización del diseño de aprendizaje resultante, tanto para estructuras como para situaciones de aprendizaje colaborativo. Después, se utiliza el método propuesto para especificar cada una de las situaciones de aprendizaje. A continuación, se ilustran las observaciones obtenidas durante la puesta en marcha de cada una de las situaciones de aprendizaje colaborativo, y se realiza una discusión sobre si se consiguen o no los tres objetivos de evaluación: automatización del proceso (flujo de aprendizaje y flujo de datos), reutilización de la definición del proceso (diseño de aprendizaje) e interoperabilidad con sistemas y herramientas.

1. Introducción

Finalmente, el capítulo 5 presenta las principales contribuciones y conclusiones extraídas de la tesis, junto con una discusión de las posibles líneas de trabajo futuro.

2 Contexto de investigación y estado del arte

El objetivo de este capítulo es doble, por un lado establecer el contexto de investigación de esta Tesis Doctoral incluyendo la identificación, análisis y planteamiento del problema, y por otro lado, revisar el trabajo relacionado con el problema para que en el siguiente capítulo se proponga una mejor solución. Primero se introduce el CSCL como el campo dedicado a la investigación del uso de la tecnología para el soporte del aprendizaje colaborativo. Después, se revisan conceptos del aprendizaje colaborativo, primero desde el punto de vista pedagógico y después desde el punto de vista de su representación computacional. Se introduce el concepto de guion colaborativo como una forma de estructurar el aprendizaje colaborativo y se revisan los EML como medio para conseguir la representación computacional de dichos guiones. A continuación se introduce el campo de Diseño de Aprendizaje, en donde se define la aproximación de Diseño de Aprendizaje basada en flujo de trabajo (LD) como una forma de aproximar el campo de Diseño de Aprendizaje. En este contexto, se introduce el concepto de diseño de aprendizaje como producto, como la descripción de guiones colaborativos siguiendo la aproximación LD. Entonces, se caracteriza un diseño de aprendizaje como la descripción de un flujo de actividades de aprendizaje y un flujo de datos. El flujo de datos es tema central de investigación en esta tesis y tiene que ver con la descripción de artefactos y herramientas, y las relaciones entre ellos. Más tarde, se introduce IMS LD, el cual es considerado el lenguaje estándar de facto de LD. Entonces, se hace una revisión de la expresividad en LD para describir el flujo de aprendizaje y el flujo de datos de diseños de aprendizaje en escenarios de aprendizaje colaborativo. Como resultado, se identifican limitaciones en LD para describir el flujo de datos. El problema de describir el flujo de datos en LD es analizado a distintos niveles: primero a nivel del modelo conceptual de LD, después a nivel de ejemplos ilustrativos, y por último a nivel de un caso de estudio auténtico. Por último, debido a la relación inherente entre LD y el campo de Flujo de Trabajo, se lleva a cabo una revisión de este campo y su relación con LD. Entonces, se revisan las soluciones que algunos autores proponen para resolver el problema.

2.1 Introducción

En teoría educativa se distinguen tres formas básicas de aprendizaje (Johnson, Johnson, 1988): aprendizaje individual, competitivo y colaborativo. En el aprendizaje individual los estudiantes trabajan independientemente para alcanzar su propio objetivo, sin ayudar a otros estudiantes a conseguir los suyos. En el aprendizaje competitivo los estudiantes compiten entre sí, pero la competencia por ganar abre la posibilidad de que el objetivo no sea alcanzado por todos los estudiantes. En cambio, en el aprendizaje colaborativo los estudiantes comparten un objetivo común que puede ser alcanzado por todos los estudiantes a través del trabajo en grupo (Dillenbourg, 2002).

Existen en la literatura distintas definiciones del aprendizaje colaborativo. Dillenbourg (1999) lo define como una situación en la que dos o más personas aprenden o intentan aprender algo juntas. Para Dodl (1990) el aprendizaje colaborativo es el conjunto de estrategias de enseñanza que dependen de la interacción de un grupo de estudiantes. Schmuck

y Schmuck (1988) lo definen como la interacción entre grupos de estudiantes con algún grado de influencia recíproca. Para Roschelle et al. (1998) el aprendizaje colaborativo es una forma de resolver coordinadamente un problema dado. Por último, para Slavin (1987) el aprendizaje colaborativo es una forma de organización e interacción entre grupos de estudiantes para el aprendizaje de contenidos.

El foco de investigación de esta Tesis Doctoral es el aprendizaje colaborativo, en particular el que es soportado por ordenador. Por tanto, el campo dentro de TEL en el que se enmarca esta Tesis Doctoral es el Aprendizaje Colaborativo Apoyado por Ordenador (CSCL). El CSCL es un campo de estudio dedicado a la investigación de la aplicación de las TIC en la educación (Stahl, Koschmann & Suthers, 2006). Se trata de un campo multidisciplinario que involucra a otros campos: educación, psicología, comunicación, ciencias computacionales y ciencias sociales (Stahl, Spada, Miyake & Law, 2011); además de la práctica educativa, cuestiones psicológicas del aprendizaje colaborativo y las facilidades que ofrecen las TIC (Osuna-Gómez, 1999). El CSCL emergió como resultado de la comunidad investigadora del Trabajo Cooperativo Soportado por Ordenador (CSCW, *Computer Supported Cooperative Work*), el cual estudia el soporte de las TIC en los procesos de colaboración (Ellis, Gibs & Rein, 1991). El principal distintivo del CSCL en TEL es que investiga el soporte de las interacciones sociales entre los estudiantes, con el educador como facilitador (Stahl, Koschmann & Suthers, 2006).

2.2 Situaciones y estructuras de aprendizaje colaborativo

Dos conceptos fundamentales en aprendizaje colaborativo en general, y en esta tesis en particular, es la distinción entre situaciones y estructuras de aprendizaje colaborativo, las cuales se definen a continuación.

2.2.1 Definición

Osuna-Gómez (1999) define una situación de aprendizaje colaborativo como un escenario de aprendizaje concreto creado con la intención de que los estudiantes construyan conocimiento a través de realizar una serie de actividades de aprendizaje colaborativo e individual. En esta definición, la idea de que una situación de aprendizaje se compone de una serie de actividades de aprendizaje, implica primero que una actividad de aprendizaje es de grano más fino respecto de una situación de aprendizaje, y segundo que el objetivo general de la situación de aprendizaje se puede alcanzar a través de la consecución del conjunto de objetivos particulares de las actividades de aprendizaje. Para ello, se debe asegurar la conexión adecuada entre las actividades individuales y colaborativas, de forma que el objetivo general de la situación de aprendizaje se pueda alcanzar a través de la consecución del conjunto de objetivos particulares de todas las actividades de aprendizaje. Más aún, una situación de aprendizaje debe incluir elementos que faciliten la interacción entre los participantes (Osuna-Gómez, 1999).

Diseñar situaciones de aprendizaje colaborativo es difícil si se parte desde cero (Bote-Lorenzo, 2005). Una posibilidad de diseñar situaciones de aprendizaje colaborativo es aplicar una estructura de aprendizaje colaborativo. Las estructuras son situaciones de aprendizaje colaborativo genéricas que son fácilmente adaptables a las necesidades de múltiples entornos, disciplinas y contenidos (Bote-Lorenzo, 2005). Ejemplos de estructuras de aprendizaje colaborativo son los patrones de procesos de aprendizaje colaborativo (Barkley, Cross & Major, 2005), los cuales se describen a continuación.

2.2.2 Patrones de procesos de aprendizaje colaborativo

Los patrones de procesos de aprendizaje colaborativo, también llamados técnicas o estrategias de aprendizaje colaborativo, son una colección de recetas o guías, que son utilizados como puntos de partida por los educadores y a las que se pueden añadir, sustituir o combinar elementos con el fin de adaptarlos para cumplir sus necesidades instruccionales (Barkley, Cross & Major, 2005). A continuación se describen algunos ejemplos de estos patrones de procesos de aprendizaje colaborativo:

- **Revisión entre pares** (*Peer Reviewing*). En esta técnica o patrón, pares de estudiantes llevan a cabo una revisión crítica y retroalimentación entre sí sobre trabajos, ensayos, informes, argumentos, etc. La revisión entre pares ayuda a los estudiantes a identificar las características buenas y malas del trabajo de otros, por tanto les permite desarrollar habilidades de evaluación crítica que ellos pueden aplicar a sus propios trabajos. Además, la retroalimentación recibida proporciona a los autores una crítica constructiva que pueden utilizar para mejorar sus propios trabajos antes de someterlos a consideración (Barkley, Cross & Major, 2005).
- **Rompecabezas** (*Jigsaw*). En esta técnica o patrón, estudiantes trabajan en pequeños grupos para desarrollar conocimiento sobre un tópico dado y para formular formas efectivas de enseñarlo a otros. A continuación, estos grupos de estudiantes “expertos” se rompe y se mueven a nuevos grupos de *jigsaw*, donde cada grupo está compuesto por estudiantes que han desarrollado pericia en diferentes sub-tópicos. Esta técnica es útil para motivar a los estudiantes a aceptar la responsabilidad de aprender algo lo suficientemente bien para enseñarlo a sus pares, así como para conducir la discusión de grupo (Barkley, Cross & Major, 2005).
- **Pirámide o bola de nieve** (*Snowball*). En esta técnica o patrón, al nivel más bajo de la pirámide, cada estudiante elabora individualmente una lista de ideas ante un problema planteado por el profesor. Después, en el siguiente nivel de la pirámide, los estudiantes se juntan en grupos para elaborar una lista con las mejores ideas. A continuación, en el siguiente nivel de la pirámide, grupos más grandes de estudiantes se juntan para mejorar todavía más la lista de ideas. El proceso continúa de esta manera hasta formar un único grupo, en el nivel más alto de la pirámide, en el que los estudiantes ponen sus ideas en común (Hernández-Leo, 2007).
- **Mesa Redonda** (*Round Table*). En esta técnica o patrón, un grupo de estudiantes generan ideas (lluvia de ideas) que inicialmente no son elaboradas, cuestionadas ni

2. Contexto de investigación y estado del arte

explicadas. A continuación, los participantes responden en turnos a preguntas concisas. El *round robin* asegura la participación equitativa de todos los estudiantes. Esta técnica es efectiva para generar muchas ideas porque requiere la participación de todos los estudiantes y porque desalienta comentarios que interrumpen o inhiban el flujo de ideas. Las ideas generadas por los estudiantes como resultado del *round robin* son compiladas y usadas como punto de partida para la siguiente tarea (Barkley, Cross & Major, 2005).

Importante es aclarar que no todas las estructuras de aprendizaje colaborativo son patrones de procesos de aprendizaje colaborativo, pero que, sin embargo, todos estos patrones son estructuras a partir de la cual se pueden derivar distintas situaciones de aprendizaje colaborativo. En (National Institute for Science Education, 1997) (Osuna-Gómez, 1999) (Hernández-Leo, 2007) (Barkley, Cross & Major, 2005) es posible encontrar la descripción de numerosos patrones de procesos de aprendizaje colaborativo.

2.3 Guiones colaborativos

Un aspecto fundamental en aprendizaje colaborativo son las interacciones. Una interacción en un contexto educativo es un evento que se produce entre un estudiante y su entorno (Wagner, 1994). De acuerdo con Moore (1989) las interacciones durante el aprendizaje colaborativo ocurren entre personas y entre personas y el contenido u artefacto de estudio. Interacciones entre personas pueden ser, por ejemplo, interacciones entre estudiantes individuales, entre grupos de estudiantes y entre estudiantes y educadores. La interacción entre estudiantes y el contenido es una característica definitiva en educación. De hecho, según Moore, sin ésta no puede haber educación, puesto que el proceso de interactuar intelectualmente con el contenido resulta en cambios en la comprensión del estudiante, la perspectiva del estudiante, o la estructura cognitiva de la mente del estudiante (Moore, 1989).

Sin embargo, las interacciones debidas a la colaboración libre no producen sistemáticamente el aprendizaje (Dillenbourg, 2002). Una forma de asistir de forma efectiva el aprendizaje colaborativo es estructurar las interacciones enganchando a los estudiantes a través de guiones bien definidos llamados guiones colaborativos (*collaboration scripts*). Existen distintas definiciones de guiones colaborativos. O'Donnell y Dansereau (1992) los definen como un conjunto de instrucciones que prescriben cómo los estudiantes deben formar grupos, y cómo deben colaborar e interactuar para resolver problemas. Dillenbourg y Jermann (2007) los definen como un contrato didáctico de la forma en que los estudiantes colaboran. Kobbe et al. (2007) definen los guiones colaborativos como un medio de facilitar el proceso cognitivo y social del aprendizaje. Un guion colaborativo prescribe una secuencia de actividades de aprendizaje, junto con los roles apropiados, y la forma en que los estudiantes interactúan unos con otros (Kobbe, Weinberger, Dillenbourg, Harrer, Hämäläinen, Häkkinen, Fischer, 2007). Weinberger et al. (2009) afirman que los guiones colaborativos son una aproximación para preparar y facilitar el aprendizaje colaborativo efectivo y pueden definirse como un tipo específico de soporte instruccional con enfoque en

2. Contexto de investigación y estado del arte

los procesos de colaboración entre dos o más miembros del grupo (tanto para grupos cara a cara, como los mediados por ordenador).

Los guiones colaborativos han sido estudiados desde distintas perspectivas como la cognitiva, la educativa o la computacional (Fisher, Kollar, Mandl & Haake, 2007). Dillenbourg (2012) caracterizó un guion colaborativo como una secuencia lineal o no-lineal de fases, donde cada fase se caracteriza por 5 atributos que describen cómo los estudiantes deben de colaborar para resolver un problema: la descripción de la tarea, la composición del grupo, la distribución de la tarea dentro y entre los grupos, el modo de la interacción y el tiempo de la fase. Dillenbourg y Jermann (2007) clasifican los guiones colaborativos en dos tipos generales: *micro* y *macro-scripts*. Los *micro-scripts* son usados para describir modelos de diálogo, por ejemplo, modelar una argumentación y contra-argumentación entre dos estudiantes. Los *macro-scripts* son usados para describir modelos didácticos que faciliten la generación de interacciones educativas entre estudiantes. Los *macro-scripts* describen una secuencia de actividades involucrando aspectos organizacionales, tales como con quién deben colaborar los estudiantes, en qué roles deben participar, así como la distribución de tareas en grupo. Los *macro-scripts* a menudo estructuran la colaboración gestionando recursos y entregables y definiendo roles y fases con el fin de producir interacciones específicas que conduzcan a situaciones de aprendizaje efectivo (Dillenbourg & Jermann, 2007). Dillenbourg & Jermann (2007) extendieron el concepto de guion colaborativo para incluir los **macro-scripts CSCL integrados**, es decir guiones que integran dentro de un flujo de trabajo simple actividades individuales y colaborativas tanto de grupos pequeños como de toda la clase, y que están distribuidas a través de múltiples lugares y múltiples planos sociales (individual, colaborativo, colectivo) (Dillenbourg & Hong, 2008). El aspecto integrativo se vuelve más evidente en el contexto CSCL, en el que los *macro-scripts* CSCL integran actividades virtuales con actividades físicas **y gestionan el flujo de datos** entre ellos (Kobbe, Weinberger, Dillenbourg, Harrer, Hämäläinen, Häkkinen & Fischer, 2007). Kobbe et al. (2007) caracterizan los guiones colaborativos a través de distinguir componentes y mecanismos. Por un lado, los componentes incluyen los participantes en el guion, las actividades en las que participan, los roles que asumen, los recursos de que hacen uso y los grupos que forman. Por otro lado, los mecanismos son usados para describir la naturaleza distributiva del guion; es decir, cómo las actividades, roles y recursos son distribuidos a través de los participantes (distribución de tareas), cómo los participantes son distribuidos a través de los grupos (formación de grupos), y cómo ambos componentes y grupos están distribuidos en el tiempo (secuenciación).

A continuación se describe la caracterización de los guiones colaborativos que será usada en esta Tesis Doctoral para describir estructuras y situaciones de aprendizaje colaborativo.

2.3.2 Caracterización de guiones colaborativos

Debido al amplio espectro de actividades que puede abarcar un guion colaborativo, esta Tesis Doctoral se centra en los *macro-scripts* CSCL integrados, que en adelante llamaremos simplemente guiones colaborativos. De acuerdo con Dillenbourg y Jermann (2007) un **macro-script CSCL integrado**, integra dentro de un flujo de trabajo: actividades individuales y colaborativas, actividades en grupos pequeños y de toda la clase, y actividades virtuales y/o físicas. Además, integra también la gestión del flujo de datos (Dillenbourg & Jermann, 2007). Esta integración de la gestión del flujo de datos al guion colaborativo, es consistente con el modelo de interacción en aprendizaje colaborativo propuesto por Anderson (2003). Anderson argumenta que debido al incremento del grado de interacción en las redes de comunicación, el modelo de interacción entre personas propuesto por Moore (1989) debe incluir además otra perspectiva de interacción: la interacción entre personas y sistemas automáticos, como por ejemplo la interacción de estudiantes con programas de inteligencia artificial (aplicaciones o herramientas), o la interacción entre estudiantes y la transferencia de datos en los sistemas de soporte en línea (Anderson, 2003).

Por tanto, un guion colaborativo en esta Tesis Doctoral se caracteriza por la integración de un flujo de actividades de aprendizaje llamado en adelante el **flujo de aprendizaje** y la gestión del flujo de datos a nivel de herramientas (sistemas automáticos) llamado en adelante el **flujo de datos**. A continuación se describen ambos conceptos, los cuales son centrales en esta Tesis Doctoral.

2.3.3 Flujo de aprendizaje

El flujo de aprendizaje se refiere a la caracterización que distintos autores hacen de un guion colaborativo, cuando se refieren a la secuenciación (lineal o no lineal) de las actividades de aprendizaje, a los roles de los participantes en las actividades de aprendizaje, y a los recursos que usarán en las actividades. Además, se refiere también a la distribución de los participantes en las actividades de aprendizaje, a la distribución de las tareas, y a los grupos que se forman. Sin embargo, la práctica educativa efectiva requiere la adaptación de guiones colaborativos para su compartición y reutilización en distintos contextos (Agostino, 2011) (Donald, Blake, Girault, Datt & Ramsay, 2009). Con este propósito, conviene distinguir el flujo de aprendizaje en estructuras y en situaciones de aprendizaje colaborativo. En esta Tesis Doctoral se define el flujo de aprendizaje de una estructura de aprendizaje colaborativo, como la secuenciación (lineal o no-lineal) de las actividades de aprendizaje, los roles de los participantes abstractos en las actividades de aprendizaje, y los recursos abstractos que son usados para soportarlas. La caracterización que distintos autores hacen de guiones colaborativos cuando se refieren a la distribución de los participantes concretos en las actividades de aprendizaje, a los roles que asumen los participantes concretos, los recursos concretos de que hacen uso y a los grupos concretos que se forman, define el flujo de aprendizaje de una situación de aprendizaje concreta.

2. Contexto de investigación y estado del arte

El objetivo del flujo de datos es apoyar el flujo de aprendizaje, el cual se define a continuación.

2.3.4 Flujo de datos

Son muchos los investigadores que refieren la importancia del flujo de datos en procesos de aprendizaje colaborativo. Dillenbourg y Jermann (2007) refieren explícitamente al flujo de datos cuando caracterizan guiones colaborativos. Peter y Vantroys (2005) mencionan la necesidad de describir la gestión del flujo de datos en proceso de aprendizaje colaborativo. Neumann et al. (2009) se refieren más explícitamente a la gestión del flujo de datos a nivel de herramientas. Wilson (2005) señala la necesidad de describir que los resultados de un servicio sean exportados hacia algún otro servicio. Dalziel (2006) apunta la necesidad de describir herramientas y servicios, descripciones de herramientas, así como descripciones de preparación e instanciación de herramientas, así como de mecanismos que permitan pasar información entre herramientas.

En este contexto, herramientas usadas para soportar actividades de aprendizaje individuales o colaborativas pueden requerir artefactos (datos de entrada) o producir artefactos (datos de salida) individuales o compartidos. Tales artefactos pueden ser, por ejemplo, un documento con las conclusiones de una discusión, las respuestas de un test, el fichero de especificación de una simulación, etc. Puesto que estas herramientas pueden pertenecer a diferentes actividades o a una misma actividad, entonces es posible definir **datos** y **herramientas** y relacionar con las actividades de aprendizaje una **secuencia de herramientas**, así como una **relación entre datos y herramientas**; es decir, los artefactos que son producidos y consumidos por herramientas durante la realización de actividades y que es definido como el **flujo de datos**.

Sin embargo, como ocurrió para el flujo de aprendizaje, la práctica educativa efectiva requiere la adaptación de guiones colaborativos para su compartición y reutilización en distintos contextos (Agostino, 2011) (Donald, Blake, Girault, Datt & Ramsay, 2009). Con este propósito, conviene distinguir el flujo de datos en estructuras y situaciones de aprendizaje colaborativo. En esta Tesis Doctoral se define el flujo de datos de una estructura de aprendizaje colaborativo, como la secuenciación (lineal o no-lineal) de herramientas abstractas y la relación entre datos y herramientas abstractas (los artefactos de entrada y salida de las herramientas) que dan soporte a las actividades de aprendizaje. Similarmente, se define el flujo de datos de una situación de aprendizaje colaborativo, como la secuenciación (lineal o no-lineal) de instancias de herramientas y la relación entre instancias de datos e instancias de herramientas que dan soporte a las actividades de aprendizaje.

2.3.5 Representación de guiones colaborativos

La representación de la práctica de enseñanza-aprendizaje documentada en algún formato notacional (entendible por ordenador o no) puede servir como modelo o *template* adaptable por un profesor para adecuar su contexto (Agostino, 2011). Donald et al. (2009) afirman

también que un proceso de aprendizaje que documenta y describe actividades de aprendizaje permite que otros educadores puedan entenderlo, compartirlo, adaptarlo y usarlo en su propio contexto. Para Agostino (2011), la representación de la práctica educativa tiene dos tradiciones de investigación (Agostino, 2011): una enfocada en las representaciones entendibles por humanos para compartir conocimiento de diseño entre educadores, y la otra enfocada en las representaciones entendibles por ordenador en entornos de aprendizaje digitales. Por un lado, la representación de la práctica de enseñanza-aprendizaje documentada en algún formato notacional (entendible por humanos) puede permitir que otros educadores puedan entenderlo, compartirlo, adaptarlo y usarlo en su propio contexto (Agostino, 2011) (Donald, Blake, Girault, Datt & Ramsay, 2009). Por otro lado, un proceso de enseñanza-aprendizaje solo puede ser compartido y reutilizado de manera efectiva por otros educadores a través una representación entendible por ordenador (Cameron, 2009), además de que permitirá su integración a través de sistemas (Kopper, 2001).

Por tanto, la práctica educativa efectiva representada a través de guiones colaborativos tiene que ver con dos aspectos: la adaptación de guiones colaborativos para su compartición y reutilización en distintos contextos, y la representación computacional de guiones colaborativos para una efectiva compartición y reutilización. Por una parte, para abordar la reutilización de guiones colaborativos, se puede aprovechar de la distinción realizada entre estructuras y situaciones de aprendizaje colaborativo. Por otro lado, para abordar la representación computacional de guiones colaborativos, se introduce a continuación el concepto de lenguajes de modelado educativo.

2.3.6 Lenguajes de Modelado Educativo

Con la llegada de las TIC para asistir el aprendizaje colaborativo y la posibilidad de representar computacionalmente guiones colaborativos para alcanzar su ejecución automática y reutilización, surgen los llamados Lenguajes de Modelado de Educativo (EML). El concepto de EML fue acuñado por primera vez en la Universidad Abierta de Holanda (OUNL, por sus siglas en inglés) y fue usado para denominar un lenguaje concreto: el lenguaje EML de dicha Universidad (Koper & Manderveld, 2004). Después, la denominación de EML fue adoptada por la comunidad investigadora para referirse a lenguajes con propósitos similares, como es el caso de la revisión sobre EMLs realizada en (Rawlings, van Rosmalen, Koper, Rodríguez-Atacho & Lefrere, 2002) donde se estudiaron y compararon lenguajes similares al EML de la OUNL, y de donde se obtuvo la siguiente definición de EML:

Un lenguaje de modelado educativo es un modelo de información semántica y ligado, usado para describir tanto el contenido y el proceso de una unidad de aprendizaje desde una perspectiva pedagógica, con el objetivo de permitir la reutilización y la interoperabilidad (Rawlings, van Rosmalen, Koper, Rodríguez-Atacho & Lefrere, 2002).

2. Contexto de investigación y estado del arte

En esta definición de EML, el modelo de información semántica y ligado se refiere al modelo conceptual de un EML y su representación XML, el cual es distinto para cada lenguaje, y describe los elementos que componen el lenguaje y las relaciones que pueden establecerse entre éstos. Así también, una unidad de aprendizaje se refiere, por ejemplo, a un guion colaborativo representado en el lenguaje junto con todos los recursos de contenido necesarios para alcanzar los objetivos de aprendizaje. A continuación se presenta una vista general de los principales EML:

- **PALO:** Lenguaje para especificar unidades de aprendizaje, basándose en la organización de contenidos educativos como módulos estructurados (Rodríguez-Artacho & Maillo, 2004).
- **MOT** (*Modélisation par Objects Typés*- Modelado por Tipos de Objetos): Lenguaje visual para describir unidades de aprendizaje basándose en MISA (*Methode d'Ingénierie d'un Système d'Apprentissage*) (Paquette, Aubin & Crevier, 1999), la cual es una metodología de diseño de unidades de aprendizaje fundamentada en el concepto pedagógico llamado "Ingeniería Instruccional" (Paquette, 2004).
- **COWL** (*Cooperative Open Workflow Language*- Lenguaje de Flujo de Trabajo Abierto y Cooperativo): Lenguaje para poner en marcha unidades de aprendizaje especificadas en IMS LD basándose en el lenguaje de flujo de trabajo XPDL (Vantroys & Peter, 2003).
- **XEDU:** Lenguaje usado para estudiar el desarrollo de sistemas software que permitan la puesta en marcha de unidades de aprendizaje especificadas en este lenguaje (Buendía, 2003).
- **LAMS** (*Learning Activity Management System*, Sistema de Gestión de Actividades de Aprendizaje): Sistema que incluye herramienta de autoría para el diseño y puesta en marcha de unidades de aprendizaje (Dalziel, 2003).
- **E²ML:** Lenguaje visual para la autoría de unidades de aprendizaje, basado en metodologías de diseño didáctico (Botturi, 2003).
- **IMS LD** (*IMS Learning Design*, Diseño de Aprendizaje): Lenguaje estándar de facto de la comunidad LD, independiente de pedagogía, usado para especificar unidades de aprendizaje.
- **Lenguaje de Guiones CSCL:** Lenguaje para describir unidades de aprendizaje como guiones colaborativos (Miao, Hoeksema, Hoppe & Harrer, 2005).
- **CPM** (*Cooperative Problem based learning Metamodel*, Meta-modelo de aprendizaje Cooperativo basado en Problemas): Lenguaje para describir unidades de aprendizaje orientado al diseño de aprendizaje cooperativo basado en problemas (Nodenot & Laforcade, 2006).
- **PoEML:** Lenguaje para describir unidades de aprendizaje que es independiente de pedagogía, basado en el concepto de separación de asuntos (*separation of concerns*) (Caeiro-Rodríguez & Llamas-Nistal, 2006).
- **MOT + LD:** Lenguaje visual para especificar unidades de aprendizaje en IMS LD a partir de elementos definidos en MISA (Paquette, 2006).

2. Contexto de investigación y estado del arte

- **LDL** (*Learning Design Language*, Lenguaje de Diseño de Aprendizaje): Lenguaje para describir unidades de aprendizaje, basado en la comunicación y cooperación entre participantes, pero sin tomar en cuenta la coordinación de actividades (Martel & Vignollet, 2008).

Aunque distintos EML pueden ser usados para representar computacionalmente guiones colaborativos, es un hecho que la especificación IMS LD es el EML que más ha atraído la atención de la comunidad investigadora (König & Paramythis, 2010) (Derntl, Neumann, Griffiths & Oberhuemer, 2012) y es considerado por distintos investigadores el estándar de facto (Miao, Burgos, Griffiths & Koper, 2008) (Caeiro-Rodríguez, 2007) (Koper, 2007). IMS LD tiene su origen en el lenguaje EML de la OUNL (Koper & Tattersall, 2005), y es considerado el principal impulsor del campo de Diseño de Aprendizaje (McLean & Scot, 2007) (Dalziel, 2012), el cual es tema central en esta tesis y se describe a continuación.

2.4 Diseño de aprendizaje

Diseño de aprendizaje es un concepto que tiene distintas interpretaciones en la literatura. En esta sección se revisa la evolución del campo, las definiciones e interpretaciones que distintos autores dan del término, las herramientas y sistemas presentes en el campo, y se finaliza con la interpretación del término que se tendrá en este documento de tesis.

2.4.1 Evolución del campo

El concepto de diseño de aprendizaje aparece por primera vez en la literatura de psicología y educación (años 60 y 70), donde se usa para describir un método de instrucción de una sesión de aprendizaje. Más tarde, un diseño de aprendizaje es expresado como un formato para estructurar el proceso de aprendizaje a través de un marco de pasos ordenados para adquirir conocimiento, actitudes o habilidades, y este es el contexto de diseño de aprendizaje convencional que continúa hasta finales de los años 90 y 2000 (MacLean & Scot, 2007).

A mediados de los años 90 el término diseño de aprendizaje comenzó a asociarse con la promesa de TEL y fue hasta finales de los 90 que el término se asoció con la creación de contenidos educativos en *e-learning* y con experiencias de aprendizaje en línea (MacLean & Scot, 2007) (Mor & Brock, 2012). A partir de entonces, el término comienza a ser usado, a menudo de forma intercambiable, con el término diseño instruccional en contextos relacionados con *e-learning* y pierde su asociación con aproximaciones tradicionales de aprendizaje (MacLean & Scot, 2007).

Cuando en el año 2000 surge la primera colección de estándares y especificaciones para la entrega de contenidos educativos en *e-learning*: SCORM (*Sharable Content Object Reference Model*, Modelo de Referencia de Objetos de Contenido Compartido) (Bohl, Scheuhase, Sengler, & Winand, 2002), la cual tiene sus orígenes en la práctica del diseño instruccional (MacLean & Scot, 2007), se creía que el aprendizaje era lo mismo que la transferencia del conocimiento, es decir, la entrega de contenidos educativos a los estudiantes

2. Contexto de investigación y estado del arte

de acuerdo a alguna estructura pedagógica. Sin embargo, proporcionar el conocimiento no es suficiente ya que éste tiene que ser aprendido, y esta es la labor del proceso de enseñanza-aprendizaje (Kopper, 2001). En este contexto, algunos autores sugieren que el término diseño de aprendizaje debería sustituirse por el de diseño para el aprendizaje, o mejor aún, diseñadores para el aprendizaje (en referencia al rol de los educadores), ya que se parte de la convicción de que la educación no es solamente la elaboración y entrega de contenido educativo, sino de crear las condiciones para que los estudiantes aprendan, proporcionándoles la motivación para el aprendizaje, las actividades por las cuales aprenderán y los recursos y herramientas que usarán en las actividades (Beetham & Sharpe, 2013) (Smith & Ragan, 2005) (MacLean & Scot, 2007).

En esta dirección de describir procesos de aprendizaje en *e-learning* y no solamente contenidos educativos, en el año 2000 la Universidad Abierta de Holanda (OUNL) propuso el primer EML. Después, en el año 2001, el Consorcio Global IMS dedicado a la generación de estándares de interoperabilidad en *e-learning*, reconoció la necesidad de una especificación que resolviera la descripción de procesos de aprendizaje y puso en marcha el Grupo de Trabajo de Diseño de Aprendizaje (LDWG, *Learning Design Working Group*). Entonces, casi por el mismo tiempo (2001), el Grupo de Trabajo de Tecnologías de Aprendizaje (LTWG, *Learning Technologies Working Group*) realizó un estudio donde se compararon EMLs con características similares al EML de la OUNL (Rawlings, van Rosmalen, Koper, Rodríguez-Atacho & and Lefrere, 2002).

En este mismo año (2001), el LDGW aceptó el EML de la OUNL como la base desde la cual desarrollar la especificación de Diseño de Aprendizaje IMS LD, especificación que fue aprobada en Febrero de 2003. De esta forma, unos de los principales impulsores del aumento en el uso del término diseño de aprendizaje en un contexto de describir procesos de aprendizaje en *e-learning* fue el desarrollo de la especificación IMS LD (McLean & Scot, 2007), la cual surge por la necesidad de expandir y remediar el déficit percibido en las primeras especificaciones de *e-learning* como, por ejemplo, SCORM. Similarmente, Dalziel (2012) asegura también que el primer proyecto y fundador del campo de Diseño de Aprendizaje fue el predecesor de IMS LD: el EML de la OUNL. Desde entonces, un amplio rango de proyectos relacionados (LDG, METIS, JISC), conferencias y actividades de investigación muestran el interés y esfuerzo de la comunidad investigadora así como el progreso del campo (Perisco, Pozzi, Anastopoulou, Conole, Craft, Dimitriadis & Walmsley 2013).

Más recientemente, en la búsqueda de un marco conceptual más amplio, algunos investigadores tratan de desvincular el campo de Diseño de Aprendizaje de IMS LD. De esta forma, algunos autores se refieren al campo de Diseño de Aprendizaje, como un campo emergente, que está en la búsqueda de un marco contextual y un lenguaje común de diseño de aprendizaje. Por ejemplo, Mor y Brock (2012) hablan de la necesidad de un lenguaje común de diseño de aprendizaje y una plataforma que lo soporte. Para Dalziel (2012), el diseño de aprendizaje es un nuevo campo que busca desarrollar un marco descriptivo para actividades

2. Contexto de investigación y estado del arte

de enseñanza-aprendizaje y explorar cómo este marco puede asistir a educadores a compartir y adoptar grandes ideas de enseñanza. Dalziel afirma que actualmente las ideas fundamentales y definiciones del campo de Diseño de Aprendizaje no son suficientemente claras, y que existe la necesidad de crear unas bases fundamentales con el fin de alentar la siguiente generación de investigación y desarrollo. Dalziel asegura que muchos educadores usan el término diseño de aprendizaje en un sentido mucho más general que un marco abstracto para describir actividades de aprendizaje. En este contexto, Dalziel propone ideas de lo que podría ser el fundamento para el futuro campo de Diseño de Aprendizaje: un mapa conceptual de diseño de aprendizaje que proporciona la relación entre el concepto central del marco de diseño de aprendizaje y la práctica del día a día.

Para Cameron (2009), el campo de diseño de aprendizaje mantiene la promesa de proporcionar a los educadores un marco que les permitirá diseñar experiencias de aprendizaje innovadoras, efectivas, y de alta calidad para sus estudiantes; la posibilidad de construir sus estrategias de enseñanza; añadir la reflexión de su propia práctica; documentar y soportar actividades de aprendizaje innovadoras; y compartir y reutilizar prácticas de expertos. Para Mor y Brock (2012) uno de los retos en el campo de Diseño de Aprendizaje es la estandarización de una infraestructura representacional comprensiva: formas gráficas (quizás dinámicas), textuales, entendibles y representables por humanos. Para Vignollet et al. (2008) las herramientas de diseño de aprendizaje tienen que ser simples, y no ligadas a tecnologías o a conceptos de meta-lenguajes complejos como IMS LD o LDL (Martel, Vignollet, Ferraris & Durand, 2006). Para conseguirlo, la tendencia es el desarrollo de lenguajes más abstractos específicos de dominio (Vignollet, Ferraris, Martel & Burgos, 2008).

2.4.2 Interpretaciones

Al inicio del campo de Diseño de Aprendizaje existía un debate de si la especificación IMS LD era “el diseño de aprendizaje” o solo un ejemplo de estos conceptos (Dalziel, 2012). Para resolver esta dificultad, uno de los primeros intentos, era usar el término Diseño de Aprendizaje (con la primera letra de cada palabra en mayúsculas) para referirse a la especificación IMS LD, y el término diseño de aprendizaje (en minúsculas), para referirse a un concepto más amplio (Britain, 2004). Sin embargo, actualmente muchos investigadores prefieren usar el término Diseño de Aprendizaje para referirse al campo y usar IMS LD para referirse solamente a la especificación (Dalziel, 2012).

Mor, Craft y Hernández-Leo (2013) distinguen dos interpretaciones del término diseño de aprendizaje. La primera es la construcción de sistemas de cómputo usados para orquestrar la entrega de recursos y actividades de aprendizaje soportadas por ordenador. Y la segunda, es la necesidad de encontrar formas efectivas de compartir innovación en la práctica de TEL, proporcionando una ayuda para el desarrollo profesional y eficiente de los educadores. Cameron (2009) distingue, por otra parte, tres interpretaciones del concepto de diseño de aprendizaje: diseño de aprendizaje como implementación; diseño de aprendizaje como concepto general (proceso); y diseño de aprendizaje como producto. En la primera

2. Contexto de investigación y estado del arte

interpretación, Diseño de Aprendizaje (con la primera letra de cada palabra en mayúscula) se refiere a la implementación del proceso con la especificación IMS LD; en la segunda interpretación, diseño de aprendizaje (en minúsculas) se refiere a un concepto más general: el proceso de diseñar; y en la tercera, un diseño de aprendizaje (anteponiendo el artículo) se refiere al resultado del proceso de diseño, o sea el producto. Para Dobozy (2011) existen tres interpretaciones: diseño de aprendizaje como concepto; diseño de aprendizaje como proceso; y diseño de aprendizaje como producto. En la primera interpretación, Diseño de Aprendizaje se refiere al concepto o idea más general; en la segunda, diseño de aprendizaje se refiere al proceso de diseño; y en la tercera, un diseño de aprendizaje se refiere al resultado del proceso de diseño, esto es, el producto.

Tomando como base la interpretación de Dobozy (2011), a continuación se busca clarificar el término diseño de aprendizaje desde las tres distintas perspectivas.

Diseño de Aprendizaje como Concepto

Las primeras ideas sobre el concepto de diseño de aprendizaje están relacionadas al origen en el 2003 de la especificación IMS LD y a su naturaleza de flujo de trabajo. Para Bagettun et al. (2004) el modelo de IMS LD y el modelo de flujo de trabajo son relativamente cercanos, ya que ambos están orientados a proceso y se enfocan en la coordinación de actividades. Para Britain (2004), un diseño de aprendizaje es un marco descriptivo que permite a los profesores desempacar el proceso de aprendizaje separando el contenido de la pedagogía. Esto motiva a los profesores a reflexionar de una forma más creativa y profunda y ver cómo ellos diseñan y estructuran actividades para los estudiantes. La primera idea general detrás del concepto de diseño de aprendizaje es que los estudiantes aprenden mejor cuando se involucran en hacer algo (por ejemplo, enganchados en una actividad de aprendizaje). La segunda idea es que las actividades de aprendizaje pueden ser secuenciadas o estructuradas cuidadosamente y entregadas en un flujo de trabajo de aprendizaje para promover un aprendizaje más efectivo. La tercera idea es que sería de utilidad ser capaz de registrar el resultado del proceso de diseño para compartir y reutilizarlo en el futuro (Britain, 2004).

Para Koper y Miao (2008), el concepto de diseño de aprendizaje tiene que ver con la secuencia de actividades de aprendizaje que los estudiantes emprenden para cumplir los objetivos de aprendizaje, incluyendo los recursos y mecanismos de soporte requeridos para ayudar a los estudiantes a completar las actividades. La idea central detrás de diseño de aprendizaje es que las actividades que los estudiantes emprenden son centrales en cualquier proceso de aprendizaje. El educador define las tareas, proporciona los contextos y recursos para realizar la tarea, soportar el aprendizaje durante la realización de las tareas y proporcionar retroalimentación a los resultados. Las actividades de aprendizaje que se requieren para obtener algunos objetivos de aprendizaje son cuidadosamente secuenciadas de acuerdo a principios pedagógicos. Para Knight et al. (2006), el concepto de diseño de aprendizaje puede predeterminar el orden en que el contenido será presentado a los

2. Contexto de investigación y estado del arte

estudiantes, cómo será integrado en servicios de soporte de aprendizaje, cómo será secuenciado y cómo será asignado a los estudiantes en una lección.

También existe la relación entre los conceptos de diseño de aprendizaje y diseño instruccional. Para Koper y Tattersall (2005), diseño de aprendizaje es un concepto más general que diseño instruccional, ya que abarca la aplicación del conocimiento de teorías de aprendizaje, teorías de diseño instruccional, las mejores prácticas y la experiencia práctica para crear oportunidades de aprendizaje. Por el contrario, el diseño instruccional es un término altamente didáctico y es un subconjunto de diseño de aprendizaje (MacLean & Scot, 2007). Mientras el diseño instruccional surge desde una perspectiva fuertemente influenciada por objetivos de comportamiento, el diseño de aprendizaje surge desde una perspectiva constructivista del proceso de aprendizaje, situado dentro de una diversidad de herramientas tecnológicas que lo soportan (Mor & Brock, 2012).

Otros autores se refieren a diseño de aprendizaje como un concepto más general, en donde se enfatiza las aproximaciones pedagógicas y la práctica educativa. Para MacLean y Scot (2007), diseño de aprendizaje se refiere al proceso de diseño de experiencias de aprendizaje efectivas para una variedad de contextos. La práctica de diseño involucra un amplio conjunto de conocimientos, habilidades y competencias, incluyendo teorías de aprendizaje y sus aplicaciones, principios de diseño de cursos y procedimientos, uso de multimedia, uso de diferentes tecnologías, etc. Conole (2008) afirma que diseño de aprendizaje es una metodología que permite a los diseñadores/educadores tomar decisiones más informadas de cómo diseñar actividades de aprendizaje y hacer efectivo el uso apropiado de recursos y tecnologías. Esto incluye el diseño de recursos y actividades de aprendizaje individuales hasta el diseño a nivel curricular. Un principio básico es ayudar a hacer el proceso de diseño más explícito y compartible. Diseño de aprendizaje como área de investigación y desarrollo incluye tanto la obtención de evidencia hasta entender el proceso de diseño, así como el desarrollo de un rango de recursos, herramientas y actividades. Es crucial que cualquier definición de diseño de aprendizaje incluya una forma de describir actividades de aprendizaje que puedan ser compartidas y reutilizadas (Conole, 2008).

Para Cameron (2009), diseño de aprendizaje es un término que establece un puente entre la teoría y la práctica y que abarca tanto una aproximación sistemática con reglas basadas en evidencia y un conjunto de prácticas contextualizadas que son constantemente adaptadas a circunstancias. Para Perisco et al. (2013), la meta de diseño de aprendizaje es idear aproximaciones y herramientas para asistir a diseñadores-educadores en planear eventos educativos de varios tipos, basados en criterios pedagógicos para la identificación de objetivos de aprendizaje, estrategias de aprendizaje apropiadas, criterios de evaluación, herramientas digitales y multimedia.

Diseño de Aprendizaje como Proceso de Diseño

El proceso de diseño es determinar qué se requiere para aprender y cómo debe ser soportado el proceso de aprendizaje para que el estudiante pueda aprender (Laurillard, 2006).

Beetham y Sharpe (2013) proporcionan una visión general del proceso de diseño: investigación, aplicación, modelado e iteración. Britan (2004) proporciona una lista mucho más prescriptiva: definir objetivos de aprendizaje, desarrollar la descripción narrativa del escenario de enseñanza-aprendizaje, y crear un flujo de trabajo de actividades de aprendizaje. Britain afirma que un diseño de aprendizaje debe describirse a suficiente nivel de abstracción tal que pueda ser generalizado más allá del simple contexto de enseñanza-aprendizaje para el cual fue creado, pero no a tal nivel de abstracción que se pierda la riqueza del valor pedagógico (Britain, 2007). En este mismo sentido, la descripción de procesos de aprendizaje descritos como guiones colaborativos deben realizarse de una forma abstracta que sea independiente de contexto (tanto como sea posible), por ejemplo, evitar describir un guion colaborativo en función de herramientas concretas o del tamaño concreto del grupo. La descripción de un guion colaborativo debe distinguir entre componentes y mecanismos. Los componentes incluye a los participantes, las actividades, los roles, los grupos y los recursos; mientras que los mecanismos incluyen la descripción de las tareas, la formación de grupos y la secuenciación (Kobbe, Weinberger, Dillenbourg, Harrer, Hämäläinen, Häkkinen & Fischer, 2007). Dillenbourg (2002) propone una secuencia de fases para describir guiones colaborativos: el tipo de la tarea que se va a realizar; la formación de los grupos; la distribución de las tareas dentro y entre grupos, incluyendo actividades y asignación de roles; los recursos virtuales o físicos; el tipo y modo de la interacción; y el tiempo de la fase. Dillenbourg y Jerman (2006) extendieron esta descripción a los guiones integrados CSCL, los cuales integran actividades físicas con virtuales y gestionan el flujo de datos entre ellas.

Diseño de Aprendizaje como Producto

Algunos autores usan el término diseño de aprendizaje para describir el resultado (producto) del proceso de diseño (Cameron, 2009) (Dobozy, 2011) (Koper & Olivier, 2004). Un diseño de aprendizaje puede ser de cualquier granularidad, desde una actividad individual hasta un curso. El alcance del diseño está determinado por los objetivos de aprendizaje a cumplir (Cameron, 2009). Un diseño de aprendizaje puede hacer referencia a recursos físicos como, por ejemplo, objetos de aprendizaje y servicios de aprendizaje, los cuales se requieren para soportar el proceso de enseñanza-aprendizaje. El diseño de aprendizaje y los recursos físicos pueden ser empaquetados juntos en una unidad de aprendizaje (UoL). Una UoL (curso, lección, actividad, etc.) puede ser compartida entre educadores y herramientas, o reutilizada en distintos escenarios de aprendizaje y contextos (Koper & Olivier, 2004).

De la revisión anterior, es claro que no existe un consenso general acerca del término de diseño de aprendizaje, por lo que es necesario definir el marco conceptual que será usado en esta tesis. Antes, se revisarán las herramientas y sistemas existentes en el campo.

2.4.3 Herramientas y sistemas del campo de Diseño de Aprendizaje

Las herramientas de autoría de diseño de aprendizaje tienen generalmente el objetivo común de facilitar la compartición, adaptación y reutilización de las ideas pedagógicas de los educadores, y también son útiles como herramientas para la reflexión de la práctica educativa

2. Contexto de investigación y estado del arte

(Laurillard, 2012). El estado del arte en el campo de diseño de aprendizaje se caracteriza por la coexistencia de una variedad de métodos y herramientas (Conole, 2013), algunas de las cuales se describen a continuación:

- **Web Collage** (Villasclaras-Fernández, Hernández-Leo, Asensio-Pérez & Dimitriadis, 2013). Herramienta de autoría basada en Web desarrollada para soportar a los educadores en crear escenarios CSCL basado en la metáfora visual de diseño de patrones de flujo de aprendizaje colaborativo (Hernández-Leo, Asensio-Pérez & Dimitriadis, 2005). Web Collage es la versión Web de Collage (Hernández-Leo, Villasclaras-Fernández, Jorrín-Abellán, Asensio-Pérez, Dimitriadis, Ruiz-Requies, Rubia-Avi, 2006). Sin embargo, a diferencia de Collage, Web Collage puede usarse junto con **GLUE!-PS** (Prieto-Santos, Asensio-Pérez, Dimitriadis, Gómez-Sánchez & Muñoz-Cristóbal, 2011) para implementar el diseño de aprendizaje directamente en un VLE, creando automáticamente las actividades requeridas, los grupos, recursos, etc. Web Collage soporta el formato IMS LD.
- **Learning Designer**. (Laurillard, Charlton, Craft, Dimakopoulos, Ljubojevic, Magoulas, Whittlestone, 2013). Herramienta de construcción del conocimiento en comunidad que soporta a educadores en la creación, modificación, localización, compartición y reutilización de diseños de aprendizaje. Su principal objetivo es soportar a la comunidad de enseñanza en colaborar sobre cómo diseñar y expresar sus ideas pedagógicas, especialmente en relación al uso de TEL para mejorar la experiencia del estudiante. **Learning Designer** no soporta por el momento IMS LD.
- **OpenGLM** (Derntl, Neumann & Oberhuemer, 2011). Kit de herramientas de autoría de diseño de aprendizaje para soportar a educadores en la creación de diseños de aprendizaje IMS LD en los niveles A y B.
- **SceneEdit** (Emin, Pernin & Aguirre, 2010). Herramienta de autoría basada en Web desarrollada para estructurar y soportar a educadores y entrenadores en el diseño y reutilización de escenarios pedagógicos mixtos en aprendizaje formal. Combina dos aproximaciones del proceso de diseño: 1. Organizar el escenario facilitando las interacciones del diseñador/educador y representando estrategias de aprendizaje explícitas. 2. Facilita la exploración de componentes reusables en la forma de bibliotecas adaptadas a comunidades específicas de profesores. SceneEdit soporta el formato IMS LD.
- **CADMOS** (Katsamani & Symeon, 2011). Editor gráfico para soportar al educador en crear, compartir y poner en marcha diseños de aprendizaje en un entorno real. Emplea dos modelos: un modelo conceptual usado para describir las actividades de aprendizaje en la que diferentes roles son involucrados, así como los recursos de aprendizaje correspondientes y los servicios; y un modelo de flujo, que describe la orquestación de las actividades de aprendizaje. CADMOS soporta el formato IMS LD.

2. Contexto de investigación y estado del arte

- **GLM** (*Graphical Learning Modeller*) (Neumann, Oberhuemer & Derntl, 2009B). Herramienta de autoría para especificar diseños de aprendizaje que proporciona una interfaz gráfica intuitiva que abstrae de tecnicismos de la especificación IMS LD.
- **CompendiumLD** (Brasher, Conole, Cross, Weller, Clark & White, 2008). Es una herramienta para soportar a los educadores a través del proceso de diseñar actividades de aprendizaje a través de utilizar una interfaz visual flexible. Está siendo desarrollada como una herramienta para soportar a los estudiantes y educadores a articular sus ideas y proyectar el diseño o secuencia de aprendizaje. El proceso de visualizar el diseño hace las ideas de diseño más explícitas y resalta cosas que pueden no notarse de otra forma. Proporciona también formas útiles de representar y compartir sus diseños. CompendiumLD es solamente una herramienta de diseño, no de ejecución. Al día de hoy no soporta IMS LD, pero está planeado en el futuro importar y exportar diseños en los formatos IMS LD y en la secuencia de actividades de LAMS.
- **MoCoLaDe** (Kobbe, Weinberger, Dillenbourg, Harrer, Hämäläinen, Häkkinen & Fischer, 2007). Herramienta de autoría para modelar y diseñar gráficamente guiones colaborativos en el formato IMS LD.
- **ASK-LDT** (*ASK-Learning Designer Toolkit*) (Sampson, Karampiperis & Zervas, 2005). Herramienta de autoría usada para facilitar la creación de estructuras para escenarios de aprendizaje basados en la interconexión de actividades de aprendizaje siguiendo la aproximación IMS LD.
- **Reload** (Collin, Beauvoir & Sharples, 2005). Herramienta de autoría usada para crear diseños de aprendizaje en el formato A y B de IMS LD.

En la *Tabla 2.1* se comparan algunas de las principales características de estas herramientas. La disponibilidad de una amplia elección de herramientas es muestra de la madurez del campo de Diseño de Aprendizaje (Mor & Brock, 2012). En este sentido, es importante hacer notar que la mayoría de las herramientas en la *Tabla 2.1* soportan o exportan hacia el formato IMS LD.

Herramienta de Autoría	Nivel IMS LD Soportado	Edición Gráfica	Uso de CLFPs	Basada en Web
Web Collage		+	+	+
Learning Designer	-	+	-	-
OpenGLM	A, B	+	-	-
SceneEdit	+	+	-	+
CADMOS	A, B	+	-	-
CompendiumLD	-	+	-	+
MoCoLaDe	+	+	-	-
ASK-LDT	+	+	-	-
Reload	A, B, C	-	-	-

Tabla 2.1: Características de las principales herramientas de autoría en el campo de Diseño de Aprendizaje. La mayoría de las herramientas importan o exportan hacia el formato IMS LD.

2. Contexto de investigación y estado del arte

A continuación se describen algunos de los principales sistemas y plataformas que son usados para poner en marcha diseños de aprendizaje:

- **CoLearn** (Stylianakis, Moumoutzis, Arapi, Mylonakis & Christodoulakis, 2014). Es una plataforma que se puede integrar por encima de infraestructuras existentes de aprendizaje electrónico (VLE), con el objetivo de ofrecer funcionalidad del flujo de aprendizaje basado en IMS LD.
- **Lifery** (Vogten & Koper, 2014). Es un portal que integra la funcionalidad de distintos servicios *portlets*, que ha sido extendido con *portlets* educativos con el propósito de convertir Liferay en un VLE. Uno de tales *portlets*, es el *portlet* de secuenciación, el cual permite la construcción de diseños de aprendizaje en el nivel A de IMS LD.
- **LdShake** (Hernández-Leo, Moreno, Chacón & Blat, 2014). Plataforma social en la que educadores comparten y co-editan innovaciones educativas de diseño de aprendizaje usando distintas herramientas de autoría. LdShake está diseñada para integrar herramientas de diseño de aprendizaje basadas en Web (por ejemplo, WebCollage) que adoptan las características de compartición y co-edición asíncronas.
- **GridCole** (Bote-Lorenzo, Gómez-Sánchez, Vega-Gorgojo, Dimitriadis, Asensio-Pérez, Jorrín-Abellán, 2008). Sistema que puede ser fácilmente adaptado por educadores con el objetivo de soportar la realización de situaciones de aprendizaje colaborativo guiados. Para conseguirlo, los educadores proporcionan un guion en formato IMS LD con la secuencia de actividades a ser realizadas por educadores así como las herramientas y documentos requeridos para soportarlas. GridCole puede entonces buscar estas herramientas en un *grid* orientado a servicios con el fin de integrarlas de forma que estén disponibles a los usuarios durante la realización de la situación de aprendizaje.
- **LAMS** (*Learning Activity Management System*) (Ghiglione & Dalziel, 2007). Sistema de Diseño de Aprendizaje usado para diseñar, gestionar y entregar actividades de aprendizaje colaborativo en línea. Integra una herramienta de autoría visual intuitiva para crear secuencias de actividades de aprendizaje. Estas actividades incluyen un rango de tareas individuales, pequeños grupos de trabajo y actividades de toda la clase basadas tanto en contenido y colaboración. LAMS está inspirado en el concepto y principios de la especificación IMS LD. LAMS2 permite exportar a diseños de aprendizaje en el formato nivel C de IMS LD.
- **CopperCore** (Vogten & Martens, 2005). Es la implementación de referencia de un motor de IMS LD que proporciona también un intérprete (*player*) basado en ese motor.
- **SLeD** (*Service-based Learning Design system*) (Escobedo del Cid, de la Fuente Valentín, Gutiérrez, Pardo & Delgado, 2007). Es un intérprete (*player*) de diseños de aprendizaje basado en CopperCore que separa la funcionalidad del intérprete del motor subyacente. Se enfoca en el uso de servicios Web: la comunicación entre el

motor y el intérprete se realiza a través de servicios Web, herramientas de usuarios finales se crearon como servicios Web.

- **CeLS** (*Collaborative e-learning Structures*) (Ronen, Kohen-Vacs, & Raz-Fogel, 2006). Es un entorno basado en Web para crear y conducir actividades colaborativas estructuradas asíncronas en escenarios instruccionales. CeLS está diseñado para crear y reutilizar estructuras de actividades que reflejan varias estrategias instrucciones colaborativas. CeLS soporta el formato IMS LD.
- **COW** (*Cooperative Open Workflow*) (Vantroys & Peter, 2003). Es una plataforma flexible para poner en marcha escenarios de aprendizaje colaborativo. COW toma un diseño de aprendizaje especificado en IMS LD, luego lo transforma en un formato COWL, el cual es una extensión de XPDL con el objetivo de ponerlo en marcha a través de un motor de flujo de trabajo. COW puede integrarse con un Entorno de Aprendizaje Virtual (VLE).
- **Moodle** (Dougiamas & Taylor, 2003). Es un VLE que se usa para estructurar actividades en grupos y para guiar a los estudiantes a través de rutas de aprendizaje. Moodle tiene planeando exportar hacia el formato IMS LD.

Nuevamente, es importante hacer notar que la mayoría de los sistemas y plataformas soportan de una forma u otra la especificación IMS LD.

2.4.4 Diseño de aprendizaje en el contexto de esta Tesis Doctoral

De la revisión anterior, se identifican dos perspectivas del campo de Diseño de Aprendizaje. La primera, busca la representación computacional de procesos de aprendizaje de una forma similar a la coordinación de actividades de flujo de trabajo pero en el ámbito educativo. Y la segunda, es una perspectiva más general del campo, que busca desvincularse de esta primera aproximación. Es decir, que busca no limitarse solamente a la representación computacional de procesos de aprendizaje, o al uso de una única aproximación basada en el paradigma de flujo de trabajo. Por tanto, la primera perspectiva se puede entender como una aproximación para implementar el campo de Diseño de Aprendizaje, pudiendo existir otras aproximaciones que no estén basadas en flujo de trabajo como, por ejemplo una aproximación de Diseño de Aprendizaje basada en un marco ontológico (Rius, Conesa, García-Barriocanal & Sicilia, 2014); o basada en un método de planeación de Red Jerárquica de Tareas (HTN) (Chalco, Gerosa, Bittencourt & Isotani, 2014); o quizás basada en Redes de Petri (Vidal, Lama, Sánchez & Bugarín, 2008), etc. Mientras que la segunda perspectiva se puede entender como el ideal del campo de Diseño de Aprendizaje.

Entonces, mientras la primera perspectiva tiene un marco conceptual bien definido, como lo es el campo de Flujo de Trabajo, la segunda carece de este marco (Mor & Brock, 2012) (Dalziel, 2012) (Cameron, 2009). Sin embargo, una investigación en el campo de Diseño de Aprendizaje debe realizarse en un marco que tenga fundamentos bien definidos, ya que de lo contrario se corre el riesgo de construir castillos sobre pobres cimientos (Goodyear & Dimitriadis, 2013). Por tanto, se pueden identificar dos caminos de investigación dentro

2. Contexto de investigación y estado del arte

del campo LD. El primero, estaría en la dirección de nuevas aproximaciones de implementación del campo LD, mientras que el segundo estaría en la dirección de una aproximación ya existente del campo.

En esta Tesis Doctoral se realizará una investigación dentro del marco conceptual de la aproximación de Diseño de Aprendizaje basada en flujo de trabajo aplicado al dominio educativo. La ventaja de este camino de investigación es que se trabaja en un marco conceptual bien definido, estable y maduro, como lo es el campo de Flujo de Trabajo, que cuenta además con un EML estándar de facto (IMS LD), que soporta una variedad de aproximaciones pedagógicas, incluyendo el aprendizaje colaborativo, así como con una diversidad de sistemas y herramientas que da soporte a toda una comunidad de práctica.

En adelante en este documento de tesis, se entenderá el término diseño de aprendizaje desde las siguientes perspectivas:

1. *Perspectiva del campo.* El campo de Diseño de Aprendizaje (**campo LD**), se refiere al campo más general que el diseño instruccional, y que puede abarcar teorías de diseño instruccional, teorías pedagógicas, estrategias didácticas, TIC y la práctica educativa. El campo de Diseño de Aprendizaje tiene como objetivo soportar a diseñadores instruccionales/educadores en la práctica educativa.
2. *Aproximación de Diseño de Aprendizaje.* La aproximación de Diseño de Aprendizaje (**aproximación LD**) se refiere a una forma de implementar los objetivos del campo LD. Existen varias aproximaciones LD, una de las cuales es la aproximación LD basada en flujo de trabajo aplicado al dominio educativo (en adelante **LD**). LD comprende un modelo conceptual, un EML estándar (IMS LD), así como herramientas y sistemas usados para soportar a diseñadores instruccionales y educadores en la planeación, descripción, despliegue y puesta en marcha de procesos de enseñanza-aprendizaje para una variedad de aproximaciones pedagógicas, incluyendo el aprendizaje colaborativo. LD tiene los mismos importantes objetivos que flujo de trabajo: automatización del proceso, reutilización de la definición del proceso e interoperabilidad. LD es el marco conceptual de la especificación estándar IMS LD. Una diversidad de sistemas y herramientas de autoría conformes a IMS LD soportan la comunidad LD.
3. *Perspectiva del producto.* Un **diseño de aprendizaje** es el resultado del proceso de diseño siguiendo LD. Un diseño de aprendizaje es un guion colaborativo equivalente a un macro-*script* CSCL integrado en el marco CSCL. Un diseño de aprendizaje se empaqueta junto con todos los recursos educativos en una Unidad de Aprendizaje (UoL, *Unit of Learning*).
4. *Perspectiva del proceso.* El proceso de diseño siguiendo LD consiste en describir una secuencia de actividades de aprendizaje y los recursos que las soportan, y que es llamado el **flujo de aprendizaje**, así como una secuencia de herramientas y la relación de entrada y salida de las herramientas, y que es llamado el flujo de datos a nivel de herramientas, o simplemente el **flujo de datos**.

En esta Tesis Doctoral se ha seleccionado la aproximación de Diseño de Aprendizaje basada en flujo de trabajo (LD) para representar computacionalmente guiones colaborativos, por varias razones. Primero, porque LD es un marco maduro (Mor & Brock, 2012); segundo porque es de naturaleza de flujo de trabajo (Baggetun, Barros, Fesakis, Girardin, Hoeksema, Hämäläinen, Vantroys, 2004), el cual es un campo bien conocido, estable y maduro; y tercero, porque LD comprende un EML estándar (IMS LD), un modelo conceptual (el meta-modelo de IMS LD), y porque existen un número considerable de herramientas y sistemas (ver secciones 2.4.3 y 2.4.4) que pueden ser usados para soportar a los educadores en la descripción, despliegue y puesta en marcha de procesos de enseñanza-aprendizaje para una variedad de aproximaciones pedagógicas, incluyendo el aprendizaje colaborativo.

Además, si bien es cierto que IMS LD es solamente una de muchas formas posibles de modelar y poner en marcha diseños de aprendizaje. Sin embargo, la promesa de interoperabilidad y reutilización de diseños de aprendizaje y el amplio rango de herramientas existentes en esta área demandan su particular atención (Britain, 2007). Por ejemplo, Koper y Tattersall (2005) afirman que IMS LD fue creado con el principal objetivo de desarrollar un marco que soportara la diversidad pedagógica e innovación, al mismo tiempo que promoviera el intercambio e interoperabilidad de contenidos en *e-learning*. Caeiro-Rodríguez (2007) menciona que IMS LD se planteó desde su concepción como un lenguaje estándar en la comunidad LD, con el objetivo de facilitar la reutilización de diseños entre educadores y la interoperabilidad entre herramientas. König y Paramythis (2010) consideran a IMS LD como la opción más prometedora para la implementación de guiones colaborativos en *e-learning*. Derntl et al. (2012) afirman que IMS LD es la única especificación de interoperabilidad disponible en el área de TEL que permite la definición y orquestación de complejos flujos de actividades y entornos de recursos en un escenario multi-rol. Miao et al. (2008) consideran a IMS LD como un estándar abierto de *e-learning* usado para modelar procesos de enseñanza-aprendizaje. Koper (2007) define a IMS LD como un estándar abierto que se usa para codificar una variedad de cursos digitales de manera formal, semántica, interoperable y entendible por ordenador. A continuación, se introduce la especificación IMS LD.

2.5 IMS LD

La representación de la práctica educativa es uno de los temas centrales del campo de Diseño de Aprendizaje (Mor, Craft & Hernández-Leo, 2013). En esta sección se describen los requisitos de especificaciones de interoperabilidad y estándares que debe cumplir un lenguaje. Entonces, se describen las características o requisitos que se dice cumple IMS LD. Por último, se enuncian los objetivos de IMS LD en los que se enfoca este documento de tesis.

2.5.2 Requisitos de especificaciones de interoperabilidad y estándares

Para representar computacionalmente diseños de aprendizaje, un lenguaje debe cumplir los siguientes requisitos generales de especificaciones de interoperabilidad y estándares (Koper & Oliver, 2004):

1. *Reutilización.* La especificación debe hacer posible identificar, aislar, descontextualizar e intercambiar objetos de aprendizaje, y **reutilizarlos en otros contextos.**
2. *Formalización.* La especificación debe proporcionar un lenguaje formal para que **diseños de aprendizaje puedan ser procesados automáticamente.**
3. *Reproducibilidad.* La especificación debe permitir a un diseño de aprendizaje ser abstraído de tal forma que sea posible una **ejecución repetida en diferentes escenarios y con diferentes personas.**

2.5.3 Características de IMS LD

Se afirma en (IMS-LD, 2003) que la especificación IMS *Learning Design* (IMS LD) cumple con los siguientes requisitos:

R1. *Compleitud:* IMS LD es capaz de especificar diseños de aprendizaje completos, incluyendo la integración de las actividades de aprendizaje y la integración de los recursos y herramientas o servicios usados durante el aprendizaje.

R2. *Flexibilidad Pedagógica:* IMS LD es capaz de especificar diseños de aprendizaje para una amplia variedad de aproximaciones pedagógicas, incluyendo el aprendizaje colaborativo.

R3. *Personalización:* IMS LD es capaz de especificar diseños de aprendizaje de tal forma que el contenido y las actividades aprendizaje pueden ser **adaptados** a situaciones de aprendizaje.

R4. *Formalización:* IMS LD es capaz de especificar un diseño de aprendizaje de tal forma que se alcance la **automatización** del proceso.

R5. *Reproducibilidad:* IMS LD es capaz de especificar un diseño de aprendizaje abstraído de tal forma que sea posible su ejecución **repetida** para distintos usuarios y situaciones.

R6. *Interoperabilidad:* IMS LD es capaz de soportar **interoperabilidad** de diseños de aprendizaje.

R7. *Compatibilidad:* IMS LD usa estándares disponibles, principalmente IMS *Content Packaging* (IMS-CP, 2004), IMS *Question and Test Interoperability*, (IMS-QTI, 2010), IMS/LOM *Meta-Data* e IMS *Simple Sequencing*.

2. Contexto de investigación y estado del arte

R8. *Reutilización*: IMS LD hace posible especificar diseños de aprendizaje de tal forma que se puedan **reutilizar** artefactos en otros contextos.

Ahora bien, de acuerdo con algunos autores, tales como Koper (2009) y Paquette (2009), las principales características de IMS LD son:

- La descripción estandarizada de procesos de enseñanza-aprendizaje como diseños de aprendizaje interpretables por ordenador de tal forma que se alcance la **automatización** del proceso.
- La descripción de diseños de aprendizaje para una variedad de aproximaciones pedagógicas.
- La descripción de diseños de aprendizaje centrados en actividades de aprendizaje y de soporte y no solamente en los contenidos de aprendizaje.
- La descripción de métodos de enseñanza-aprendizaje como procesos de flujo de trabajo multi-actor, soportando el aprendizaje basado en actividad y basado en roles.
- El soporte para distintos tipos de cursos: presencial puro (cara a cara), a distancia puro, o combinados.
- La integración con una variedad de formatos de contenidos de aprendizaje, tales como *IMS Content Packaging* (IMS-CP, 2004), *IMS Question and Test Interoperability Specification* (IMS-QTI), *Sharable Content Object Reference Model* (SCORM).
- La integración de la descripción de servicios de colaboración/comunicación (por ejemplo, servicios de conferencia de audio/video, foros de discusión, y clases virtuales).
- **Reusabilidad** de los diseños de aprendizaje.

2.5.4 Definición y objetivos de IMS LD

Para Koper y Tattersall (2005), IMS LD es un marco de elementos que pueden ser usados para describir formalmente el diseño de cualquier proceso de enseñanza aprendizaje. De acuerdo con Müller (2010), IMS LD es un lenguaje usado para describir complejos procesos de aprendizaje multi-rol, en donde las actividades de aprendizaje son el centro de atención y por lo tanto va más allá de la mera descripción de los contenidos educativos.

En este documento de tesis, IMS LD es entendido como un lenguaje usado para describir, en un formato entendible por ordenador, la estructura de complejos procesos de aprendizaje multi-rol y multi-pedagogía (llamados diseños de aprendizaje) compuestos por un flujo de aprendizaje y un flujo de datos. El objetivo de IMS LD es alcanzar la **automatización** del proceso (tanto del flujo de aprendizaje como del flujo de datos), facilitar la **reutilización** de los diseños de aprendizaje en distintas situaciones y contextos, y la **interoperabilidad** con herramientas y sistemas desarrollados en la comunidad LD.

2.5.6 Modelo conceptual de IMS LD

IMS LD es un metalenguaje usado para describir diseños de aprendizaje independientemente de la aproximación pedagógica utilizada (es pedagógicamente neutro). IMS LD busca proporcionar un marco de elementos del dominio educativo que pueden ser usados para describir de manera formal una variedad de procesos de enseñanza-aprendizaje (IMS LD, 2003) (Koper & Olivier, 2004).

Como muchos otros lenguajes de modelado de procesos, IMS LD se basa en un modelo conceptual de elementos del dominio educativo que están relacionados entre sí, a partir del cual es posible especificar diseños de aprendizaje. El modelo conceptual de IMS LD se muestra en el diagrama de clases UML de la *Figura 2.1* y se utiliza como medio intuitivo para ayudar a entender las reglas y restricciones entre los elementos que componen el meta-modelo (Hilera & Palomar, 2005).

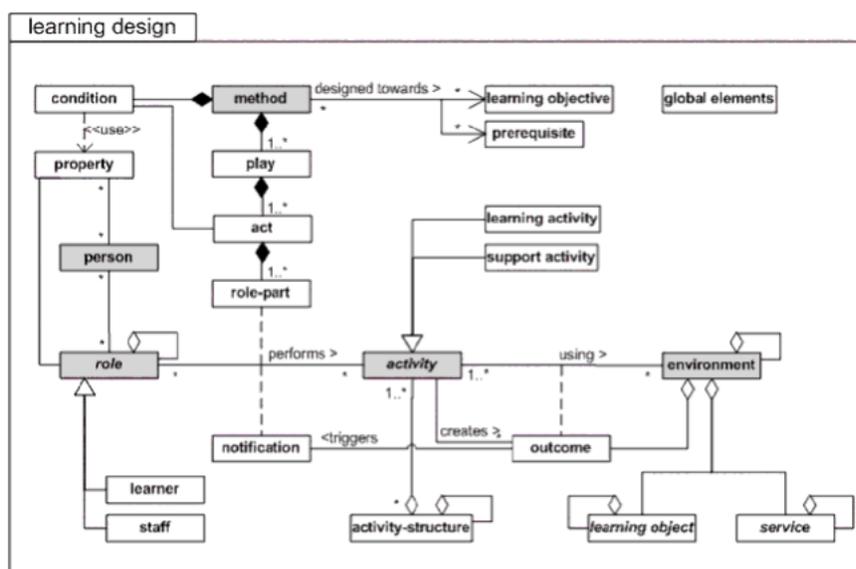


Figura 2.1: Meta-modelo de IMS LD. Fuente:(IMS-LD, 2003).

El meta-modelo de la *Figura 2.1* se puede interpretar como personas (estudiantes, educadores, tutores, etc.) desempeñando diferentes roles (*role*). Las personas en un cierto rol realizan actividades de aprendizaje o de soporte (*activity*) con el objetivo de cumplir los objetivos de aprendizaje. Las actividades son conducidas dentro de un entorno (*environment*) que consiste de objetos de aprendizaje (*learning objects*) y servicios (*services*). El aspecto dinámico del proceso de aprendizaje tiene lugar en el elemento método (*method*). Un método se puede interpretar con la metáfora de una obra de teatro, en la que puede haber varias obras (*play*) que ocurren en paralelo; donde cada obra se compone de actos secuenciales (*act*); y en cada acto, un rol participante puede interpretar un cierto guion del proceso de aprendizaje (*role-part*). Todos estos elementos junto con sus reglas, son después recogidas y plasmadas en un esquema XML (*schema*), conocido también como ligado (*binding*), el cual describe la

2. Contexto de investigación y estado del arte

estructura jerárquica de los elementos que componen el meta-modelo (Hilera & Palomar, 2005).

Con el objetivo de manejar la complejidad en la especificación de diseños de aprendizaje, IMS LD divide su especificación en tres niveles (IMS-LD, 2003):

- **Nivel A.** Contiene todos los elementos básicos que permiten especificar diseños de aprendizaje sencillos, excepto los elementos usados por los otros dos niveles.
- **Nivel B.** Añade al nivel A propiedades y condiciones, que son usados para especificar comportamientos de adaptación y personalización.
- **Nivel C.** Añade al nivel B notificaciones, que son usadas para introducir cambios en la especificación de acuerdo al disparo de eventos.

A continuación se estudia la expresividad de IMS LD para describir procesos de aprendizaje colaborativo.

2.5.7 El ciclo de vida de diseños de aprendizaje especificados con IMS LD

IMS LD es un lenguaje declarativo que describe un diseño de aprendizaje. El diseño de aprendizaje junto con todos los recursos educativos se empaquetan juntos en una unidad de aprendizaje (UoL). Para producir la experiencia de aprendizaje expresada por una UoL, se requiere de un componente software capaz de interpretar la UoL, el cual es llamado motor de flujo de aprendizaje (*learning flow engine*) (Vogten, Tattersall, Koper, van Rosmalen, Sloep, Martens, 2004). La publicación de una UoL es el resultado del pre-procesamiento de una UoL por el motor, y que es usada para validar y preparar su puesta en marcha. Antes de poner en marcha una UoL, los usuarios reales deben ser asignados a ella. IMS LD no define cómo, cuándo y quién debe ser asignado a los roles declarados en el diseño de aprendizaje. Es responsabilidad del motor ligar usuarios reales a roles abstractos (proporcionando interfaces que permiten tal asignación). Una *run* es el mecanismo usado para ligar usuarios reales a una UoL. Una *run* se puede utilizar para poner en marcha situaciones de aprendizaje similares (reutilización de un diseño de aprendizaje), simplemente asignando distintos participantes a cada *run*. Una *run* es por tanto una instancia de una UoL.

De acuerdo a lo anterior, existe un ciclo de vida que va desde la definición de la estructura de un diseño de aprendizaje especificado con IMS LD, hasta la realización de una o más situaciones de aprendizaje. Primero, en tiempo de definición se crea un diseño de aprendizaje y se empaqueta en una UoL. Después, en tiempo de instanciación se publica la UoL y se crean una o más *runs* de la UoL, a través de asignar usuarios concretos a roles abstractos. Por último, en tiempo de ejecución se pone en marcha la situación de aprendizaje.

2.6 Expresividad en LD para describir guiones colaborativos

IMS LD es el EML estándar de facto en LD. Existen, sin embargo, opiniones encontradas acerca de la expresividad de IMS LD en CSCL. Por un lado, Caeiro (2003) es el primero en

criticar IMS LD en aprendizaje colaborativo por su estricto modelo secuencial. Baggetun et al. (2004) afirman que IMS LD es expresivo para describir aprendizaje personalizado y aprendizaje cooperativo asíncrono, sin embargo no proporciona suficiente soporte para modelar actividades de aprendizaje colaborativo síncrono basado en grupos. Hernández et al. (2004) aseguran que IMS LD no permite describir la colaboración de grupos de personas a través de solamente la definición de servicios. Adicionalmente, Baggetun et al. (2004) afirman que IMS LD tampoco soporta la colaboración entre personas a nivel de actividad. Miao et al. (2005) encuentran dificultades para modelar con IMS LD distintos aspectos del aprendizaje colaborativo, tales como el modelado de grupos, artefactos, características dinámicas del proceso de aprendizaje, control de flujo complejo, así como el modelado de una variedad de formas de interacción social. Según Miao et al., con IMS LD es muy difícil describir subgrupos. Por un lado, si se define cada grupo/subgrupo como un rol, se requiere conocer el número de grupos y subgrupos en tiempo de diseño. Por otro lado, si solo se define un rol para todos los subgrupos, entonces todos los participantes tendrán el mismo rol y el sistema no podrá soportar apropiadamente la colaboración entre grupos o dentro del grupo. Dalziel (2006) apunta la necesidad de que IMS LD describa el concepto de grupo y no solo el concepto de rol; la capacidad de pasar roles/grupos e información de herramientas a través de actos; además de conceptos más detallados de secuenciación entre actos, incluyendo sincronización multi-usuario dentro de un acto.

Por otro lado, Koper y Olivier (2004) afirman que IMS LD soporta una variedad de aproximaciones pedagógicas incluyendo el aprendizaje colaborativo. En IMS LD los grupos pueden ser modelados como roles, donde también se puede especificar el número mínimo y máximo de participantes en un grupo. IMS LD permite también representar subroles, cuando se requiere distinguir diferentes comportamientos entre los miembros de un grupo (Koper, 2009). Además, Koper afirma también que IMS LD permite modelar comunicación y colaboración asíncrona y síncrona usando servicios de conferencia, así como también modelar la compartición e intercambio de artefactos entre los miembros de un grupo. Hernández-Leo et al. (2007) muestran la capacidad de IMS LD de expresar *macro-scripts* CSCL, particularmente la representación de la composición de grupos, tales como la jerarquía de grupos, el tamaño de grupo y la dinámica de formación de grupos; la distribución de roles, la rotación de roles y la distribución de recursos; la coordinación del flujo de actividades; la representación de artefactos compartidos, el flujo de datos a nivel de actividades e incluso sofisticados mecanismos de control de piso. Miao et al. (2008) investigaron la expresividad de IMS LD para representar la interacción de grupos a través de mecanismos de coordinación usados en teoría de coordinación. Ellos encontraron que IMS LD soporta la mayoría de los mecanismos básicos de coordinación: dependencias de descomposición de tareas, dependencias de precedencia y dependencias de compartición de recursos.

Una posibilidad de entender esta falta de claridad en la expresividad de IMS LD para soportar el modelado de escenarios de aprendizaje colaborativo es la existencia de distintos modelos/perspectivas de las características del aprendizaje colaborativo. Como afirman

Baggetun et al. (2004): “Un modelo representacional que se ajuste a todas las perspectivas debe ser imposible, o cuando menos difícil, puesto que un modelo solo debe enfocarse en los aspectos importantes y ocultando los detalles que son innecesarios. Por tanto, lo que es un detalle innecesario desde una perspectiva son los aspectos más importantes y esenciales desde otra”. Esta misma razón es compartida por Hernández-Leo et al. (2007), quienes aseguran que no es claro el soporte de IMS LD para implementar guiones CSCL, por lo que propone una serie de características de diseño educativo CSCL que deben cumplir, en concreto los macro-scripts CSCL:

- *Composición de grupo*. Incluye la jerarquía de grupos (grupos de grupos), el tamaño y la cantidad de grupos, las políticas de formación de grupos, la dinámica de formación de grupos y la flexibilidad en la composición de los grupos.
- *Distribución de roles*. Incluye participantes desempeñando uno o más roles al mismo tiempo, rotación de roles, etc.
- *Distribución de recursos*. Incluye los recursos requeridos en las actividades en función del número de grupos, roles y participantes.
- *Coordinación del flujo de actividades aprendizaje*. Incluye la sincronización de grupos y roles durante la realización de actividades (una persona puede requerir esperar a otras personas para iniciar la siguiente actividad)
- *Flujo de datos*. Incluye artefactos (por ejemplo, documentos) usados por un grupo en una actividad y que después son usados por otros grupos en diferentes actividades.
- *Control de piso*. Incluye la guía de las acciones de los estudiantes que participan en una misma actividad de aprendizaje.

Esta Tesis Doctoral está enfocada en la representación computacional de procesos de aprendizaje colaborativos descritos como guiones colaborativos, lo que a su vez son *macro-scripts* CSCL integrados y que se caracterizan por un flujo de aprendizaje y un flujo de datos. A continuación se estudia el soporte en LD para describir el flujo de aprendizaje y el flujo de datos de estos guiones colaborativos.

2.6.1 Expresividad en LD para describir el flujo de aprendizaje en guiones colaborativos

Müller (2010) asegura que IMS LD se usa para describir la secuencia de actividades de aprendizaje, a través de ligar actividades de aprendizaje y de soporte a roles y a un entorno (por ejemplo, objetos de aprendizaje), todo ello coordinado por un método de enseñanza (Müller, 2010), el cual pudiera ser el aprendizaje colaborativo. Hernández-Leo et al. (2007) son más explícitos al referirse a la capacidad de IMS LD para describir las siguientes características del flujo de aprendizaje en *macro-scripts* CSCL:

- *Composición de grupo*. MS LD soporta varias de las características de composición de grupo, tales como la jerarquía de grupos, el tamaño de grupo, y la dinámica de formación de grupos. Esto se puede conseguir con el uso del componente *roles*, sus

elementos y atributos relacionados, junto con el uso de propiedades y condiciones de IMS LD. Sin embargo, IMS LD proporciona soporte limitado para la especificación formal del número de grupos y políticas de formación de grupos.

- *Distribución de roles.* IMS LD soporta la descripción de la distribución de roles a través del componente *roles*, complementado con la coordinación del componente *role-parts*, a través de la cual cada participante puede desempeñar roles diferentes. La rotación de roles puede ser descrita en IMS LD mediante la rotación actividades.
- *Distribución de recursos.* IMS LD soporta la descripción de la distribución de recursos a través de la coordinación de *role-parts*, pero también a través de la posibilidad de referenciar recursos a diferentes elementos de IMS LD, tales como descripciones de actividad o entornos. El uso de propiedades o herramientas de soporte también proporciona otra forma de describir la distribución de recursos.
- *Coordinación del flujo de actividades aprendizaje.* IMS LD soporta la descripción del flujo de aprendizaje a través de usar el componente *method*.
- *Control de piso.* IMS LD soporta sofisticados mecanismos de control de piso los cuales pueden ser descritos en el flujo de aprendizaje.

Algunos autores se refieren al nivel de abstracción con el que se debiera describir un diseño de aprendizaje o guion colaborativo. Con el propósito de generalización y reutilización de un diseño de aprendizaje, Britain (2007) propone que éste debe describirse a suficiente nivel de abstracción tal que pueda ser generalizado. Kobbe et al. (2007) proponen que se debe evitar describir un guion colaborativo en función de herramientas concretas o del tamaño concreto del grupo. Sin embargo, Hernández-Leo et al. (2007) refieren entre otras características el tamaño de grupo como una característica a ser descrita en un guion colaborativo. Sin embargo, cuando ellos enuncian las características no distinguen entre las perspectivas de describir estructuras y situaciones de aprendizaje colaborativo. Por tanto, es razonable concluir de acuerdo al estudio realizado por Hernández-Leo et al. que IMS LD soporta la descripción del flujo de aprendizaje en *macro-scripts* CSCL tanto de estructuras como de situaciones de aprendizaje colaborativo.

2.6.2 Expresividad en LD para describir el flujo de datos en guiones colaborativos

IMS LD soporta la descripción del flujo de datos a nivel de actividades. Con el propósito de comprender mejor el mecanismo usado por IMS LD para describir el flujo de datos, considere la siguiente situación de aprendizaje colaborativo: tres estudiantes (participantes) desempeñando el rol “Editor” (*p1*, *p2*, *p3*), crean cada uno su propio artefacto durante la actividad de edición. Después, los mismos tres estudiantes desempeñando ahora el rol de “Revisor”, critican durante la actividad de revisión el artefacto creado por alguno de sus compañeros durante la actividad anterior (por ejemplo, el participante *p1* revisa el artefacto creado por *p2*, *p2* revisa a *p3*, y *p3* a *p1*). El diagrama de la *Figura 2.2* ilustra la estructura de ésta situación de aprendizaje.

2. Contexto de investigación y estado del arte

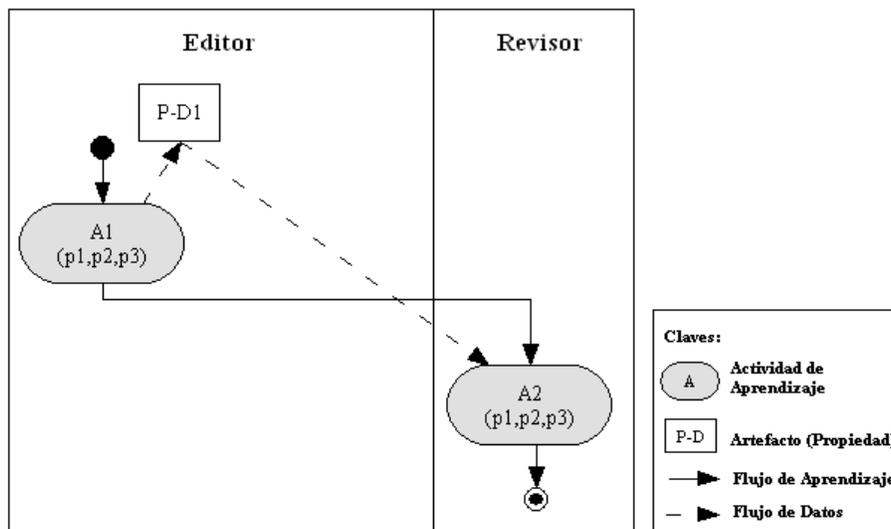


Figura 2.2: Estructura de aprendizaje colaborativo de revisión entre pares en donde el flujo de datos se produce entre actividades.

El flujo de datos a nivel de actividades se describe con IMS LD de la siguiente forma. Primero, el artefacto creado durante la actividad de edición se define con una propiedad local personal de tipo *file* (*P-DI*) (cada participante tendrá su propia copia de la propiedad). Después, se definen dos servicios monitores (ver Figura 2.3), uno para soportar cada actividad de aprendizaje. Un servicio monitor es una facilidad que ofrece IMS LD a los participantes de la actividad con el objetivo de visualizar propiedades (que representan artefactos) ya sea de los propios participantes o de los participantes en un cierto rol (IMS LD, 2003). La idea es que los participantes empleen estos servicios monitores como soporte para guardar o recuperar los artefactos creados por ellos u otros participantes durante la realización de actividades, como se muestra en la Figura 2.5.

```

<imsl:monitor>
<imsl:role-ref ref="role-Editor" />
<imsl:title>monitor-guardar-artefacto</imsl:title>
<imsl:item identifier="item-1" identifierref="resource-1" isvisible="true" />
</imsl:monitor>
...
<imsl:monitor>
<imsl:role-ref ref="role-Revisor" />
<imsl:title>monitor-leer-artefacto</imsl:title>
<imsl:item identifier="item-2" identifierref="resource-2" isvisible="true" />
</imsl:monitor>

```

Figura 2.3: Especificación en IMS-LD de monitores con visibilidad de propiedades propia (*self*) y de propiedades de participantes en el rol revisor (*role-Revisor*).

A continuación, se define el flujo de datos a nivel de actividades definiendo recursos de contenido de tipo *imslcontent* que se asocian a entornos de las actividades de aprendizaje. Esto se consigue definiendo elementos globales *set-property* y *view-property*, como se muestra en la Figura 2.4.

2. Contexto de investigación y estado del arte

```
<ld:set-property property-of="supported-person" ref="P-D1"/>
<ld:view-property property-of="supported-person" ref="P-D1"/>
```

Figura 2.4: Especificación en IMS LD del flujo de datos a nivel de actividades. El elemento global set-property permite asociar una archivo (con el soporte de un monitor) a la propiedad P-D1 (un archivo por cada participante en la actividad), mientras que el elemento global view-property permite visualizar la propiedad P-D1 de todos los participantes asociados al rol revisor.

Desafortunadamente, este mecanismo de gestión de datos usado por IMS LD tiene dos inconvenientes. Primero, la especificación del flujo de datos ocurre entre actividades y no entre herramientas, con lo que no es posible alcanzar la automatización del flujo de datos a nivel de herramientas, es decir, conseguir que los artefactos de salida de una herramienta sean después cargados automáticamente por otras herramientas. Segundo, los servicios monitores hacen visible a los estudiantes de la actividad los artefactos de todos los participantes en un cierto rol y no el artefacto concreto que debe emplear cada estudiante en la actividad. Por ejemplo, los tres estudiantes desempeñando el rol revisor tendrán visibles los artefactos de los tres estudiantes que participan en el mismo rol (el rol revisor). Por tanto, cada estudiante durante la actividad de revisión desconoce el artefacto concreto que debe usar en la actividad, como se ilustra en la *Figura 2.5*.

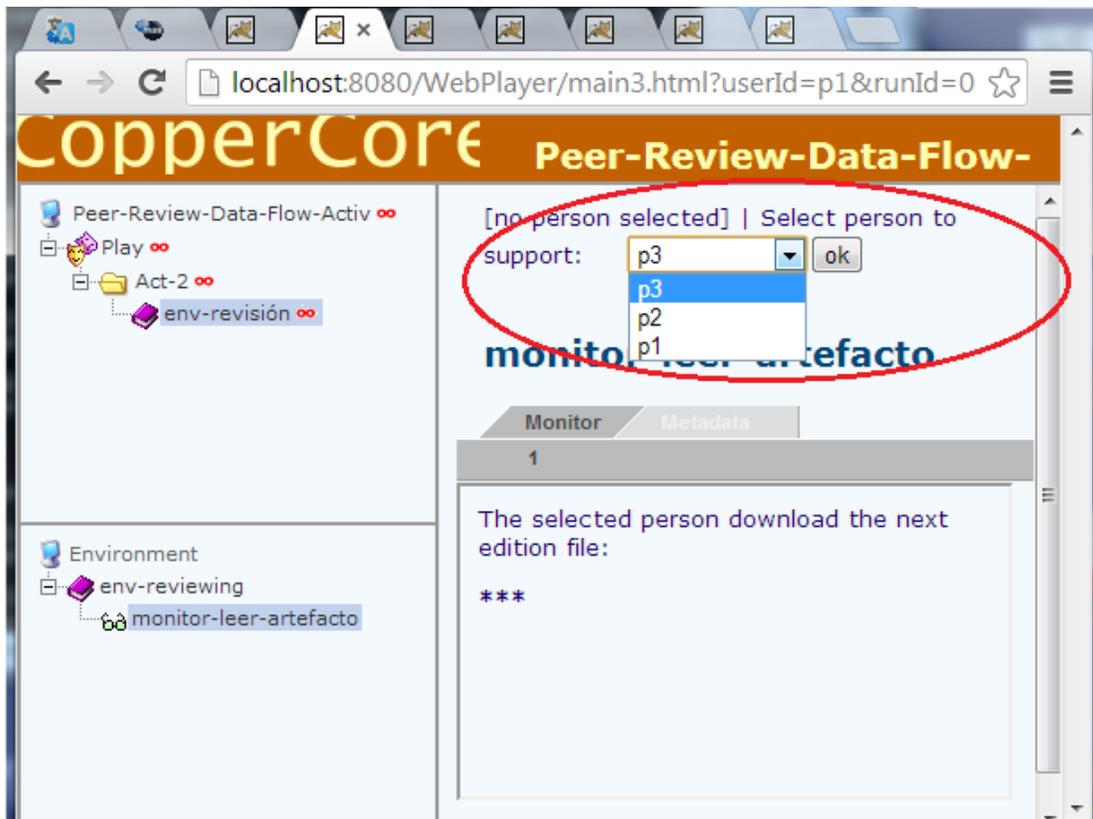


Figura 2.5: Ilustración del mecanismo usado por IMS LD para la gestión del flujo de datos a nivel de actividades. La imagen captura el momento en que un participante realizando la actividad de revisión debe decidir cuál de los artefactos debe revisar: ¿el del participante p1, p2 o p3?

2. Contexto de investigación y estado del arte

El problema es qué no se ha definido qué artefacto debe de revisar cada estudiante participante. Para hacerlo, una forma es describirlo directamente en la descripción de actividad. Con este propósito, primero se debe definir una propiedad local personal (P-estudiante), como se muestra en la *Figura 2.6*, y que represente a los distintos participantes ya que se crea una instancia por cada participante real.

```
<imsld:locpers-property identifier="P-estudiante">
  <imsld:datatype datatype="string" />
</imsld:locpers-property>
```

Figura 2.6: Definición de una propiedad local personal de IMS LD que representa a los estudiantes participantes

Después se define una expresión condicional para cada estudiante en el que se especifica mostrar u ocultar clases de elementos, en función del valor real de la propiedad que representa a los estudiantes. Por ejemplo, la expresión condicional para el participante “*p1*”, especificaría que se muestre los elementos de la clase “*p1*”, pero que se oculten los que no pertenecen a esa clase, como se muestra en la *Figura 2.7*.

```
<imsld:if>
  <imsld:is>
    <imsld:property-ref ref="P-estudiante" />
    <imsld:property-value>p1</imsld:property-value>
  </imsld:is>
</imsld:if>
<imsld:then>
  <imsld:show>
    <imsld:class with-control="false" title="p1" class="p1" />
  </imsld:show>
  <imsld:hide>
    <imsld:class with-control="false" title="p2" class="p2" />
    <imsld:class with-control="false" title="p3" class="p3" />
  </imsld:hide>
</imsld:then>
```

Figura 2.7: Condiciones en el nivel B de IMS LD que definen que se muestre la descripción de actividad del participante p1 y se oculten las descripciones de p2 y p3, si el valor real de la propiedad P-estudiante es p1.

Finalmente, en la descripción de actividad se utilizan elementos *div class* para instruir a cada estudiante sobre el artefacto concreto que debe revisar, como se muestra en la *Figura 2.8*.

```
<head>
  <title>Descargar el fichero creado en la actividad de edición</title>
</head>
<body>
  <P>Revisa el artefacto de tu compañero:</P>
  <div class="p1">Selecciona persona p2</div>
  <div class="p2">Selecciona persona p3</div>
  <div class="p3">Selecciona persona p1</div>
  <ld:view-property property-of="supported-person" ref="P-D1"/>
</body>
```

Figura 2.8: La descripción de actividad mostrada es distinta para cada participante (en función del valor del elemento div class).

Sin embargo, esta solución presenta los siguientes inconvenientes:

- *Carga cognitiva adicional en los participantes.* Los estudiantes son los responsables de gestionar los artefactos en la actividad, a partir de seguir las instrucciones de la actividad, en lugar de que el sistema los entregue y cargue automáticamente en las herramientas.
- *Situaciones propensas a error.* Los guiones instruidos tienen el problema de que pueden ser mal entendidos, mal aplicados o ignorados completamente por los participantes (Dillenbourg, 2002).
- *Problemas de reutilización del diseño.* El diseño resultante requiere definir información de los participantes ($p1$, $p2$, $p3$), sin embargo, definir un diseño de aprendizaje como una situación de aprendizaje, en lugar de una estructura de aprendizaje, impide que la definición del diseño se pueda reutilizar por distintas situaciones de aprendizaje.

Por lo tanto, del análisis anterior se puede concluir que la descripción con IMS LD del flujo de datos a nivel de actividades y no entre herramientas, parece ser un problema relevante:

1. No se alcanzan dos de los tres objetivos de LD: la automatización del proceso (en lo que respecta al flujo de datos) y la reutilización de la definición del proceso.
2. La posibilidad potencial de no conseguir los objetivos aprendizaje debido a un error durante la gestión de artefactos. Por ejemplo, cuando siguiendo equivocadamente las instrucciones de la actividad, un estudiante revisa un artefacto que no le corresponde.

Esta Tesis Doctoral se enfoca en el problema de describir con IMS LD el flujo de datos a nivel de herramientas en diseños de aprendizaje. En la siguiente sección se analiza con detalle este problema.

2.7 Análisis del problema del flujo de datos a nivel de herramientas

En la sección anterior se identificó que LD no soporta la especificación del flujo de datos a nivel de herramientas. Además, se identificaron indicios de que el problema es relevante. En esta sección se analizará el problema desde las perspectivas de describir el flujo de datos en estructuras y situaciones de aprendizaje colaborativo. Con este propósito, primero se estudiará dos ejemplos simples pero ilustrativos y después un caso de estudio auténtico y complejo.

2.7.1 Análisis de un ejemplo ilustrativo: revisión entre pares

Una buena forma de estudiar y entender la necesidad de describir el flujo de datos a nivel de herramientas en estructuras y situaciones de aprendizaje colaborativo, es a través de analizar un ejemplo ilustrativo simple pero representativo. El ejemplo seleccionado para este propósito, es el patrón de aprendizaje colaborativo *peer-review* (revisión entre pares), el cual

2. Contexto de investigación y estado del arte

es ampliamente conocido y difundido en la literatura (Bartels, 2003) (Barkley, Cross & Major, 2005).

La siguiente narrativa describe la estructura de aprendizaje colaborativo de revisión entre pares, misma que se ilustra en el diagrama de la *Figura 2.9*:

“Durante la actividad de edición (A1), un estudiante o grupo de estudiantes editan primero un artefacto (D1), utilizando para ello una herramienta individual o colaborativa (H1). Después, durante la actividad de revisión (A2), otro estudiante o grupo de estudiantes lo revisan, también con el soporte de una herramienta individual o colaborativa (H2)”.

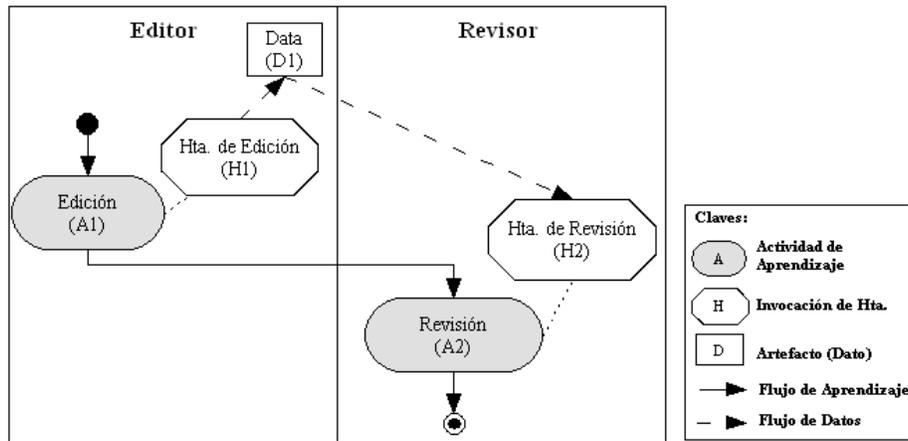
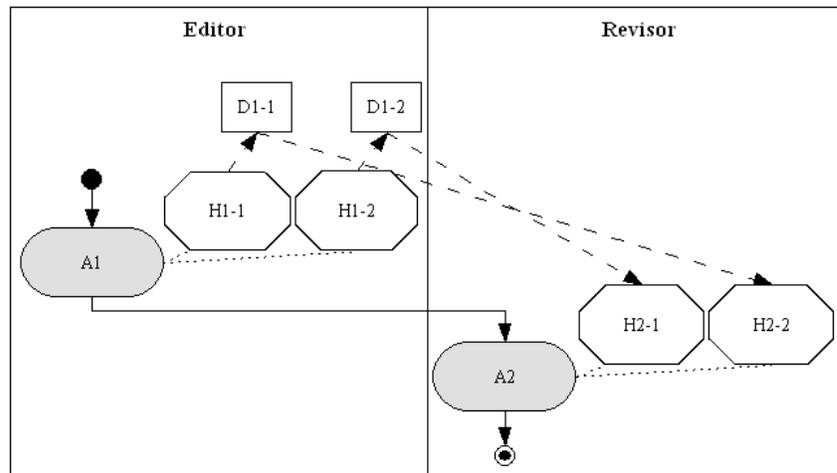


Figura 2.9: Estructura de aprendizaje colaborativo de la revisión entre pares.

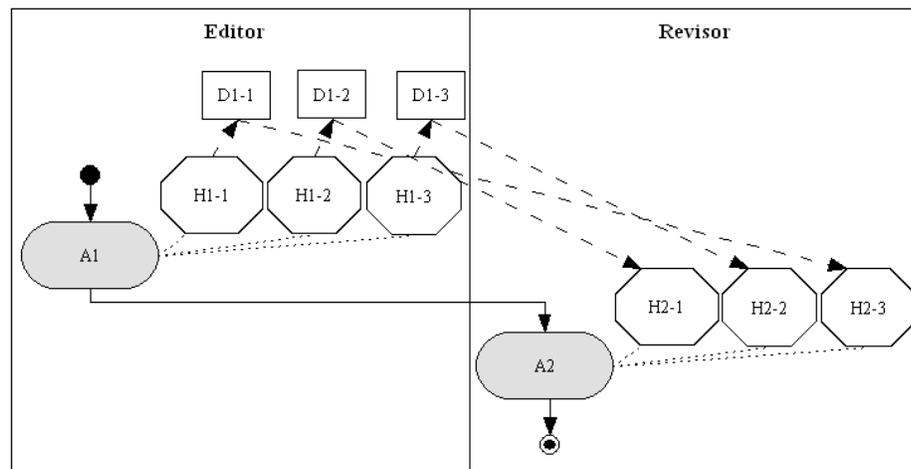
Note que la estructura de aprendizaje colaborativo de la *Figura 2.9* no da información sobre cómo instancias de datos e instancias de herramientas se relacionan entre sí durante situaciones de aprendizaje colaborativo. Por ejemplo, una típica revisión entre dos estudiantes, en la que cada estudiante revisa individualmente, el artefacto creado por su par (ver *Figura 2.10a*); o entre tres estudiantes, en la que, también individualmente, el primer estudiante revisa el artefacto creado por el segundo estudiante, el segundo estudiante revisa el artefacto creado por el tercero y este último revisa el del primero (ver *Figura 2.10b*). Por tanto, muchas otras situaciones de aprendizaje se podrían derivar de la misma estructura de aprendizaje colaborativo de revisión entre pares de la *Figura 2.9*, tan solo con definir el número de instancias de datos e instancias de herramientas, así como de la forma en que éstas se relacionan entre sí.

Por tanto, en aprendizaje colaborativo se requiere describir no solamente la estructura de aprendizaje colaborativo compuesta por el flujo de aprendizaje y el flujo de datos (ver *Figura 2.9*), sino también instancias de la estructura del flujo de datos (ver *Figura 2.10a* y *Figura 2.10b*). Una instancia de la estructura del flujo de datos describe las interacciones colaborativas entre instancias de datos e instancias de herramientas.

2. Contexto de investigación y estado del arte



(a)

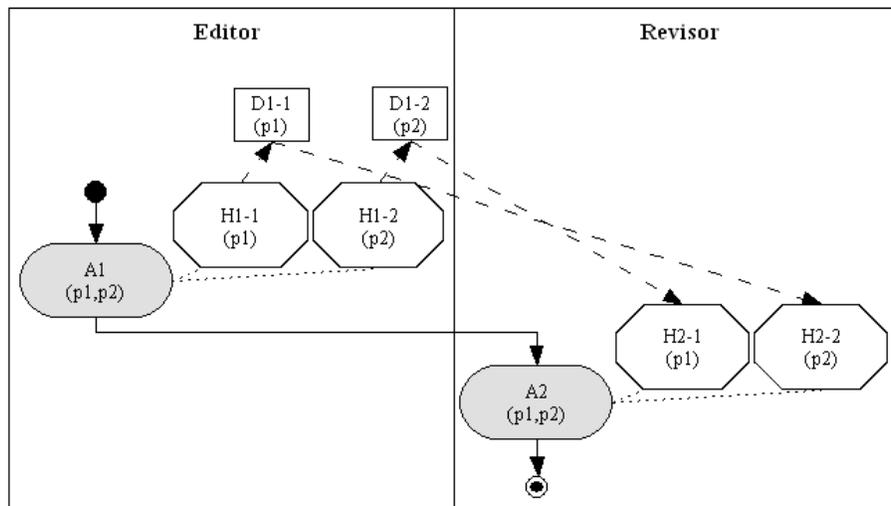


(b)

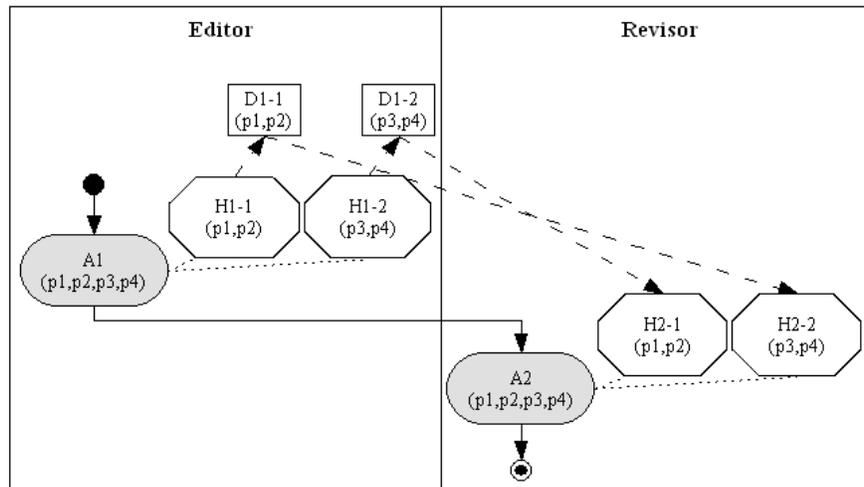
Figura 2.10: Dos instancias derivadas de la estructura del flujo de datos de la Figura 2.9. Interacción colaborativa entre (a) dos instancias de datos y dos instancias de herramientas, e (b) entre tres instancias de datos y tres instancias de herramientas.

Más aún, distintas situaciones de aprendizaje colaborativo se pueden derivar de la misma instancia de la estructura del flujo de datos, en función de cómo se asignen los participantes a las instancias de herramientas. Por ejemplo, la Figura 2.11a y la Figura 2.11b comparten la misma instancia de la estructura del flujo de datos de la Figura 2.10a, sin embargo, representan dos situaciones de aprendizaje colaborativo distintas. Por un lado, en la Figura 2.11a se describe a dos estudiantes que editan individualmente un artefacto y que después se retroalimentan entre sí. Por otro lado, en la Figura 2.11b se describe a dos parejas de estudiantes que trabajan primero colaborativamente para después retroalimentarse entre sí.

2. Contexto de investigación y estado del arte



(a)



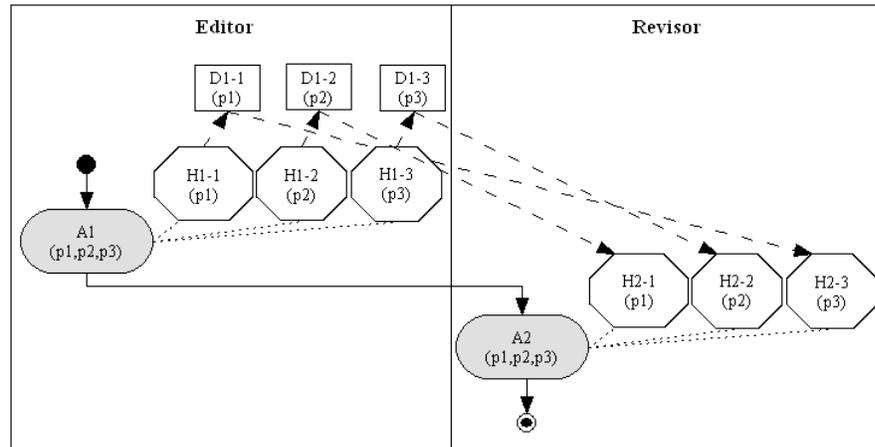
(b)

Figura 2.11: Dos situaciones de aprendizaje colaborativo concretas derivadas de la misma instancia de la estructura del flujo de datos de la Figura 2.10a. Situación de aprendizaje colaborativo en el que participan (a) dos estudiantes trabajando individualmente e (b) cuatro estudiantes trabajando colaborativamente.

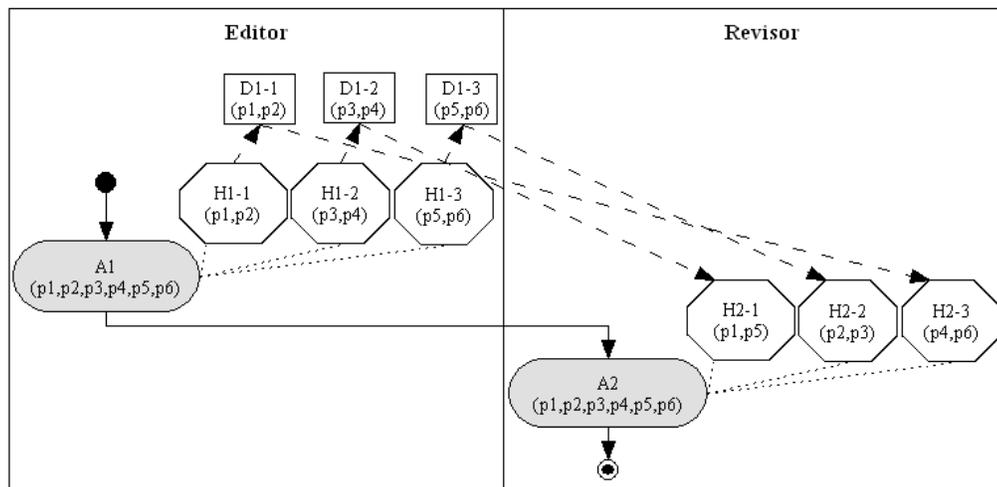
Otro ejemplo se ilustra en la *Figura 2.12a* y en la *Figura 2.12b*. Ambas situaciones de aprendizaje se derivan de la misma instancia de la estructura del flujo de datos de la *Figura 2.10b*, sin embargo, representan dos situaciones de aprendizaje colaborativo distintas. Por un lado, en la *Figura 2.12a* se describe a tres estudiantes editando individualmente su propio artefacto, y después el primer estudiante revisa individualmente el artefacto creado por el segundo estudiante, el segundo estudiante revisa el del tercero y el tercero el del primero. Por otro lado, en la *Figura 2.12b* se describe a tres parejas de estudiantes editando

2. Contexto de investigación y estado del arte

colaborativamente sus propios artefactos y después, trabajando en parejas pero de distintos estudiantes, cada pareja retroalimenta a una tercera.



(a)



(b)

Figura 2.12: Dos situaciones de aprendizaje colaborativo concretas derivadas de la misma instancia de la estructura del flujo de datos de la Figura 2.10b. Situación de aprendizaje colaborativo en el que participan (a) tres estudiantes trabajando individualmente e (b) seis estudiantes trabajando colaborativamente.

En resumen, en escenarios de aprendizaje colaborativo se requiere describir no solo la estructura de aprendizaje colaborativo (el flujo de aprendizaje y el flujo de datos), sino también instancias de la estructura del flujo de datos. Más aún, situaciones de aprendizaje colaborativo concretas requieren describir la asignación de participantes a las actividades de aprendizaje y a las instancias de herramientas. Desafortunadamente, ni la estructura del flujo de datos de la Figura 2.9, ni instancias de la estructura del flujo de datos como las representadas en la Figura 2.10a y en la Figura 2.10b se pueden especificar con IMS LD, ya

que éste no soporta la especificación de la relación entre datos y herramientas. Además, la especificación de la asignación de participantes a las actividades de aprendizaje y a las instancias de herramientas (problema de instanciación en los lenguajes de modelado de procesos) es todavía un problema abierto (Caeiro-Rodríguez, 2008), incluso en el dominio de flujo de trabajo (Decker & Mendling, 2008).

Por último, debido a que IMS LD tampoco soporta especificar la secuencia en la que varias herramientas requieren ser usadas en una misma actividad de aprendizaje (si se requiere más de una), a continuación se analiza este problema. Con este propósito, se utiliza un ejemplo ilustrativo que es una generalización del patrón de revisión entre pares llamado guion de convergencia del conocimiento, en el que se hace explícito este problema.

2.7.2 Análisis de un ejemplo ilustrativo: convergencia del conocimiento

Una forma de estudiar y entender el problema de especificar la secuenciación de herramientas en una misma actividad de aprendizaje, es a través del siguiente ejemplo ilustrativo, el cual está basado en un escenario de aprendizaje colaborativo bien documentado en la literatura (Weinberger, Stegmann & Fischer, 2005) y que es conocido como el guion de convergencia del conocimiento (*knowledge convergence script*).

La siguiente narrativa describe este patrón de aprendizaje colaborativo, cuya estructura de aprendizaje colaborativo se ilustra en el diagrama de la *Figura 2.13*:

“Durante la actividad de edición (A1), un grupo de estudiantes editan reportes individuales (D2) sobre un tema o caso (D1). Después, durante la actividad de revisión (A2), los reportes son revisados y criticados en turnos por el resto de los miembros del grupo. Por último en la actividad de reflexión (A3), los reportes iniciales junto con las críticas realizadas regresan a los autores originales, quienes comparan sus propias ideas junto con las de sus compañeros de grupo para producir así sus reflexiones finales. Para la realización de actividades, los estudiantes utilizan herramientas de edición (H2,H4,H6) y herramientas de visualización de documentos (H1,H3,H5)”.

Distintas situaciones de aprendizaje se pueden derivar de la estructura de aprendizaje colaborativo de la *Figura 2.13*, por ejemplo la *Figura 2.14* muestra una situación de aprendizaje en la que participan tres estudiantes, cada uno de los cuales debiendo editar su propio reporte y revisar y criticar el reporte de sus otros dos compañeros. Esto significa que se debe especificar una secuencia de instancias de herramientas en cada actividad de aprendizaje. Note por ejemplo que para la actividad de revisión en la *Figura 2.14*, cada participante debe revisar primero (a través de una instancia de la herramienta H3) el reporte de uno de sus compañeros y después el reporte del segundo.

2. Contexto de investigación y estado del arte

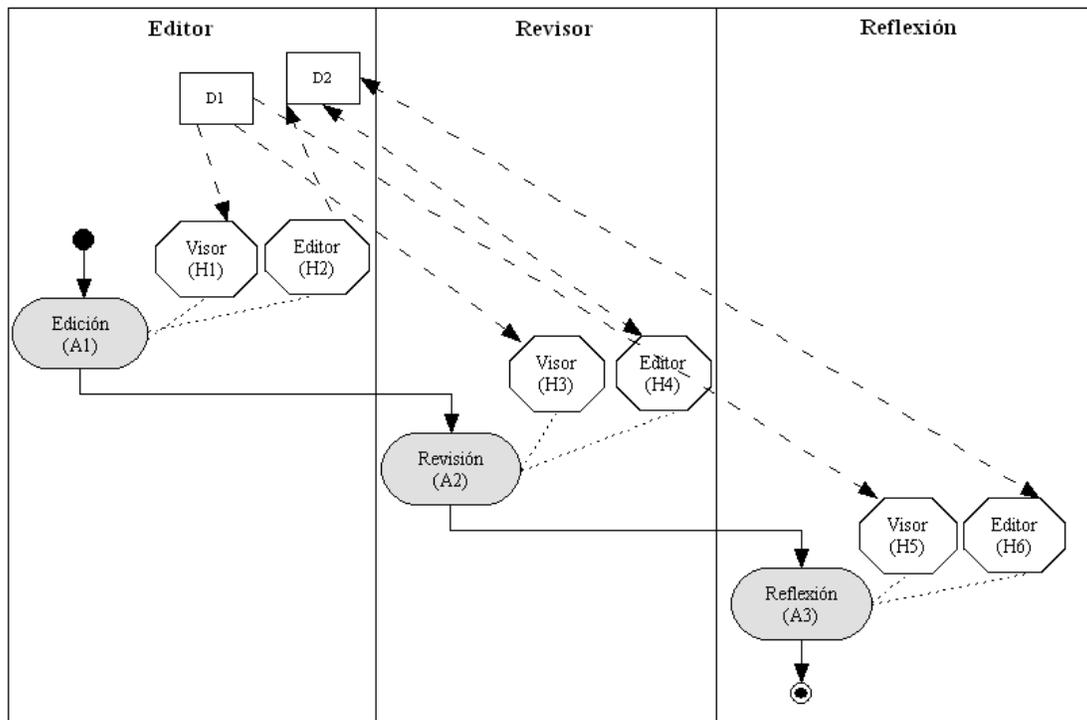


Figura 2.13: Estructura de aprendizaje colaborativo del guion de convergencia del conocimiento.

Desafortunadamente esta secuenciación de herramientas abstractas (y entre instancias de herramientas) de la Figura 2.13 y la Figura 2.14 respectivamente, no se pueden especificar con IMS LD. Por tanto, en procesos de aprendizaje colaborativo se requiere de una forma de especificar no solo la secuenciación de herramientas abstractas sino también de instancias de herramientas en una misma actividad de aprendizaje.

2.7.3 Análisis de un caso de estudio: el caso MOSAIC

Este caso de estudio es una experiencia de aprendizaje auténtica compartida por tres universidades españolas, como parte del principal proyecto nacional de aprendizaje MOSAIC (<http://mosaic.gast.it.uc3m.es>): Universidad Carlos III de Madrid, Universidad Abierta de Cataluña y la Universidad de Valladolid. El objetivo del proyecto MOSAIC es que grupos de investigación de estas tres universidades colaboren para probar herramientas de terceros. Se trata de una situación de aprendizaje colaborativo en la que participaron 12 estudiantes con el objetivo de construir un mapa conceptual sobre el tópico de servicios *Grid* y la computación orientada a servicios. Además de estudiantes, otras personas también participaron en la experiencia desempeñando distintos roles: estudiantes, tutores, diseñadores instruccionales y administradores del curso.

2. Contexto de investigación y estado del arte

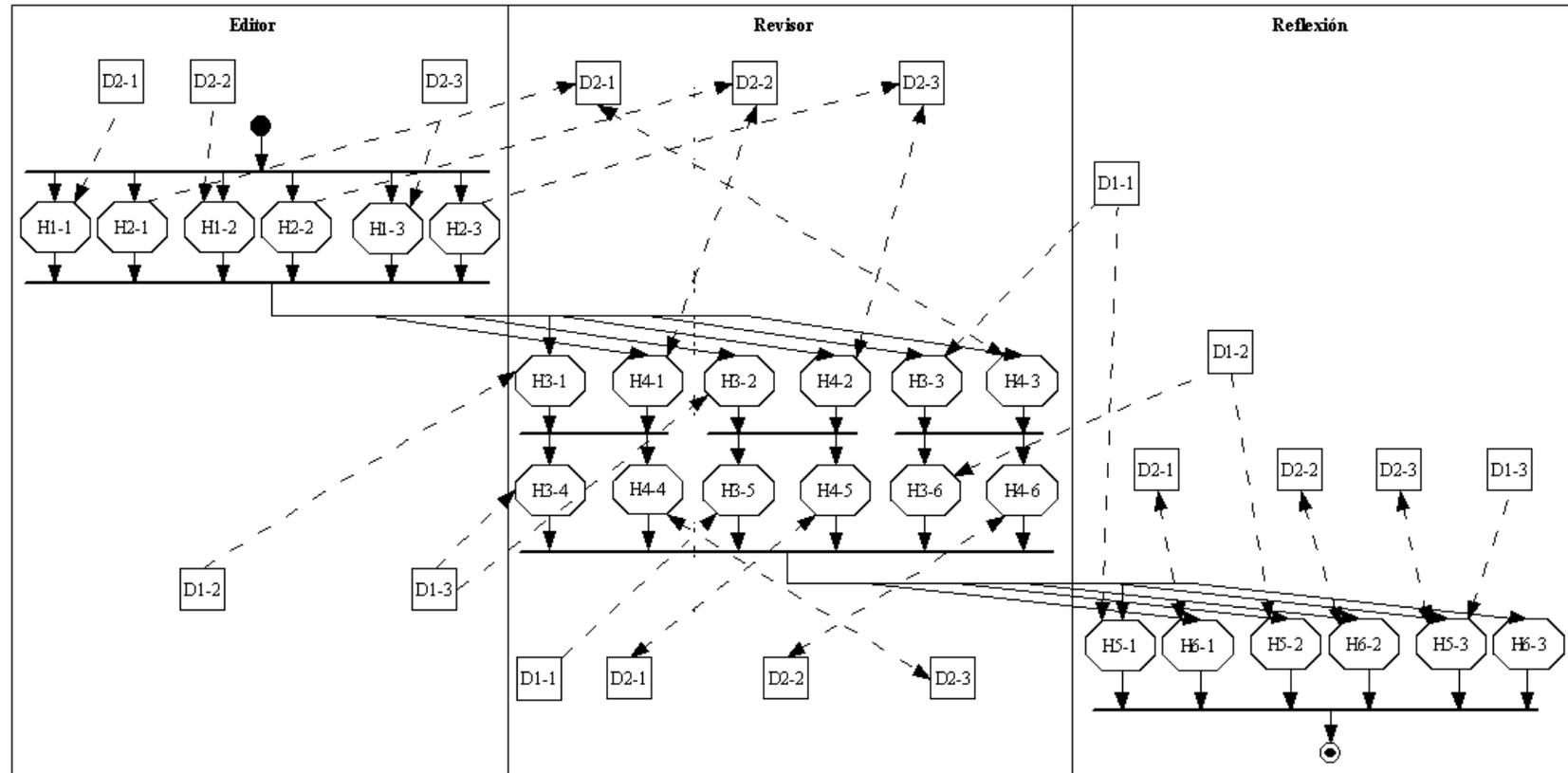


Figura 2.14: Una instancia de la estructura del flujo de datos del guion de convergencia del conocimiento en el que interactúan 3 estudiantes.

2. Contexto de investigación y estado del arte

El diseño de aprendizaje fue especificado por el diseñador instruccional usando IMS LD y siguiendo la aproximación de CLFPs (*Collaborative Learning Flow Patterns*, Patrones de Flujo de Aprendizaje Colaborativo), la cual en este caso combina tres estructuras de aprendizaje colaborativo: pirámide (de 3 niveles), *jigsaw* y revisión entre pares (ver *Figura 2.15*). El primer nivel de la pirámide está compuesto de 3 fases. En la primera fase, de aprendizaje individual, cada estudiante lee uno de tres artículos seleccionados (4 estudiantes leen el mismo artículo) con el fin de construir una primera versión de su mapa conceptual. En la segunda fase, de aprendizaje colaborativo, los cuatro estudiantes que leyeron el mismo artículo trabajan juntos como un grupo de expertos (en total, se formaron tres grupos de expertos, uno por artículo). Cada uno de estos grupos tiene que realizar una revisión entre pares del trabajo previo con el objetivo de construir una versión mejorada y consensuada del mapa conceptual. En la tercera fase, también de aprendizaje colaborativo, se formaron 3 grupos de *jigsaw* de 4 estudiantes expertos cada uno, mientras la revisión entre pares se mantiene nuevamente como la principal actividad. Al final, cada grupo de *jigsaw* construye un nuevo mapa conceptual mejorado y consensuado. En el segundo nivel de la pirámide, se formaron 2 grupos de *jigsaw* (de 6 estudiantes cada uno) con el fin de mejorar una nueva versión consensuada del mapa conceptual. Por último, en el tercer nivel de la pirámide los 12 estudiantes trabajaron juntos para presentar sus conclusiones finales.

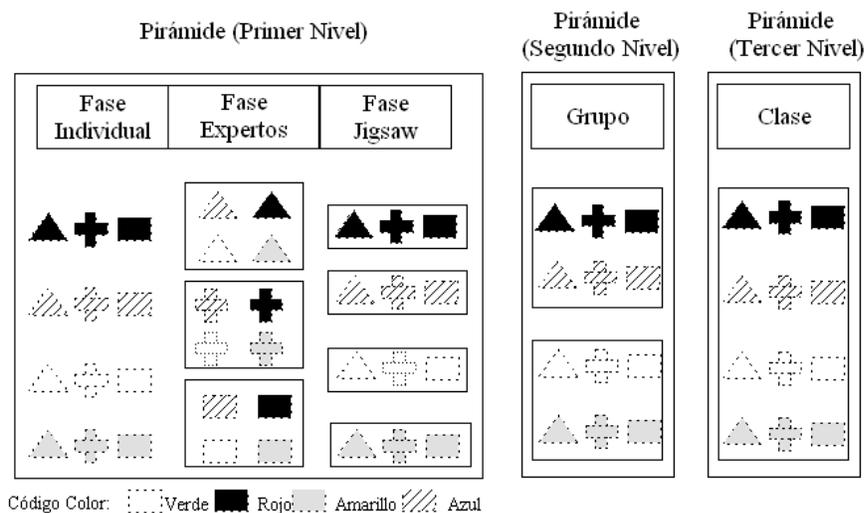


Figura 2.15: Esquema del caso MOSAIC.

Note en el esquema de la *Figura 2.15* el significado de los códigos de forma y “color” de la situación de aprendizaje colaborativo del caso MOSAIC. Los triángulos, cruces y rectángulos corresponden a los miembros de un grupo de expertos, mientras que los “colores” rojo, verde, azul y amarillo denotan el grupo de *jigsaw* al que pertenece cada estudiante.

El diagrama de actividad de la *Figura 2.16* describe la estructura de aprendizaje colaborativo (simplificada) del caso MOSAIC. El flujo de aprendizaje inicia en el rol de

2. Contexto de investigación y estado del arte

expertos con una actividad de estudio individual (A2), que busca hacer expertos a los estudiantes sobre un tema o caso y que, como resultado, producirán un mapa conceptual individual (D1, utilizando el editor de mapas conceptuales H1). La siguiente es una actividad individual de revisión entre pares (A3), cuyo objetivo es retroalimentar (usando la herramienta H2) a sus compañeros expertos sobre sus mapas conceptuales creados por ellos en la actividad anterior. El rol experto termina con una actividad colaborativa de estudio entre expertos (A4), en la que los estudiantes trabajan juntos como grupos de expertos para crear un nuevo mapa conceptual (D2, utilizando la herramienta H2). A continuación, en el rol de *jigsaw*, el flujo de aprendizaje continúa primero con una actividad de revisión entre pares (A5), en la que los estudiantes trabajan juntos para retroalimentar a sus compañeros de *jigsaw* sobre los artefactos creados por ellos en la actividad anterior (usando la herramienta H4); después trabajan juntos en una actividad colaborativa (A6) para crear un nuevo mapa conceptual (D3, usando H5), pero ahora como grupos de *jigsaw*. Después, el flujo de actividades en el nivel 2 de la pirámide continúa de forma similar al rol anterior, primero con una actividad de revisión entre pares (A7) para retroalimentar a sus compañeros sobre los artefactos creados por ellos en la actividad anterior (usando H6), y después con una actividad colaborativa (A8) para crear un nuevo mapa conceptual (D4, usando H7), pero ahora como grupos extendidos de *jigsaw*. Por último, el flujo de aprendizaje termina en el nivel 3 de la pirámide, con una actividad colaborativa (A9) en la que todos los estudiantes crean en consenso un solo mapa conceptual (D5, usando H8), discuten y presentan sus conclusiones finales.

Note que la *Figura 2.16* solo muestra la estructura del flujo de aprendizaje y la estructura del flujo de datos. De hecho, la estructura del flujo de datos real es mucho más compleja que la mostrada en la *Figura 2.16*, ya que en total se definieron 14 artefactos y 32 herramientas individuales o colaborativas. Si bien, el caso MOSAIC es una situación de aprendizaje colaborativo que solamente expertos en técnicas de aprendizaje colaborativo podrían llegar a plantear en la práctica, por otro lado, de acuerdo al número instancias de datos e instancias de herramientas involucradas, podemos afirmar que este caso de estudio es suficientemente significativo dentro del contexto del problema de especificar el flujo de datos en escenarios de aprendizaje colaborativo.

El enfoque para el análisis de este caso de estudio está basado desde dos perspectivas del flujo de datos: la perspectiva de la gestión de datos por parte de los participantes y la perspectiva del diseñador instruccional para especificar el flujo de datos con IMS LD. Con el propósito de soportar los resultados de esta investigación, los datos han sido reunidos a partir de fuentes cuantitativas y cualitativas, y el análisis sigue los principios del método de evaluación mixta propuesto en (Martínez-Monés, Dimitriadis, Rubia-Avi, Gómez-Sánchez & de la Fuente Redondo, 2003). La *Tabla 2.2* muestra un resumen de las diferentes fuentes de datos consideradas en el caso de estudio, las cuales provienen principalmente de cuestionarios, correos electrónicos intercambiados entre los estudiantes participantes, así como del análisis del diseño de aprendizaje usado en el caso de estudio. Al final del caso de estudio, dos tipos de cuestionarios fueron llenados por los estudiantes: un cuestionario de

2. Contexto de investigación y estado del arte

propósito general y otro de propósito particular. El primero enfocado en la experiencia completa y el segundo enfocado en problemas específicamente relacionados con el flujo de datos. Además, tanto el diseñador instruccional como el administrador del curso respondieron a cuestionarios específicos. Todos los cuestionarios y resultados usados en la evaluación de este caso de estudio son referidos al apéndice A.1

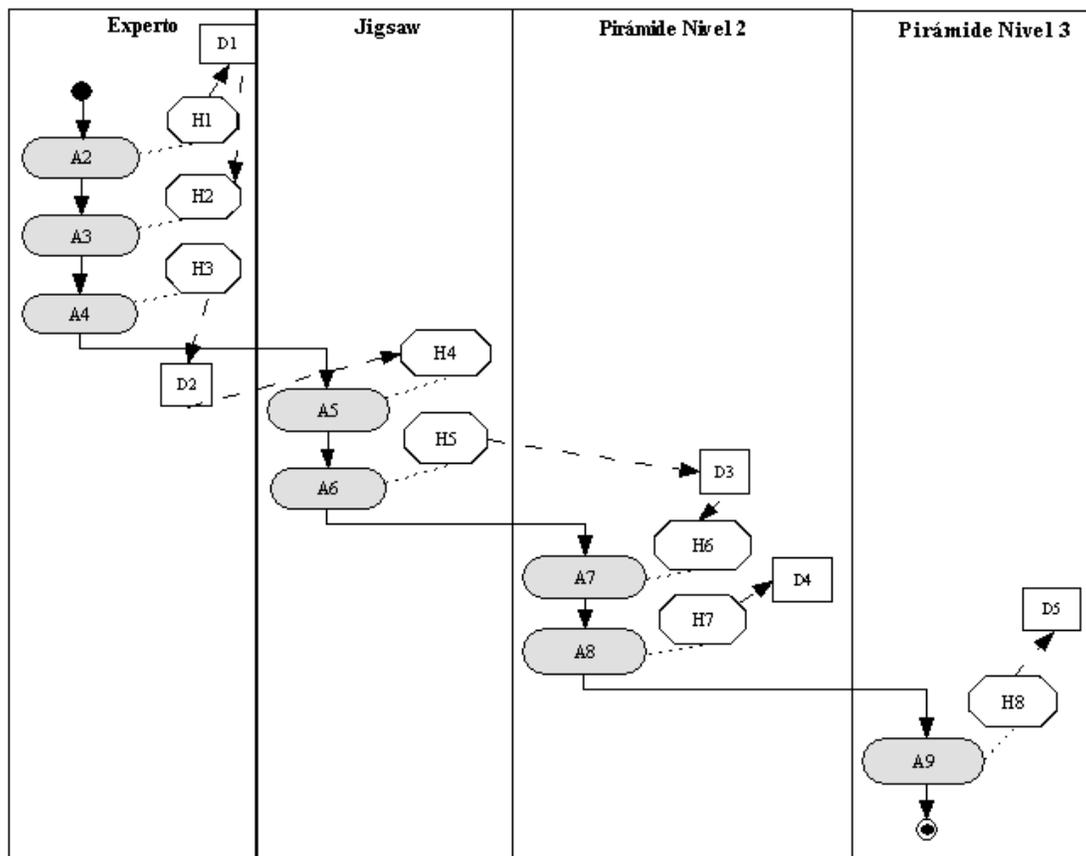


Figura 2.16: Estructura de aprendizaje colaborativo del caso MOSAIC.

Fuente de Datos	Etiquetas
Cuestionario de propósito general aplicado a los estudiantes después de la experiencia	[student-gquest]
Cuestionario de propósito específico aplicado a los estudiantes después de la experiencia	[student-squest]
Cuestionario de propósito específico aplicado a los administradores del curso después de la experiencia	[manager-squest]
Cuestionario de propósito específico aplicado al diseñador del curso después de la experiencia	[designer-squest]
Intercambio de correos electrónicos entre los participantes durante la experiencia (fuente: estudiantes)	[student-email]
Intercambio de correos electrónicos entre los participantes durante la experiencia (fuente: administrador/diseñador)	[designer-email]
Manifiesto de la UoL realizada antes de la experiencia	[designer-uol]

Tabla 2.2: Fuentes de datos usados en la evaluación del caso de estudio MOSAIC y etiquetas usadas en el texto para citarlas.

2. Contexto de investigación y estado del arte

La *Tabla 2.3* resume los principales resultados de investigación que emergieron del análisis de este caso de estudio. El primer resultado de investigación soporta la relevancia de este caso de estudio con respecto a la gestión del flujo de datos en un escenario de aprendizaje colaborativo. Como se puede ver de los datos cuantitativos obtenidos de los estudiantes (en una escala entre 0 y 6, donde 6 es más importante y 0 menos importante), la gestión de ficheros fue muy importante para ellos (4.57, con una desviación estándar de 1.99) y fue usada de manera muy frecuente a lo largo del caso de estudio (4.43, con una desviación de 1.99). Este resultado también es soportado con los datos cualitativos de los estudiantes, lo cual es indicativo de la relevancia del problema.

El segundo resultado de investigación indica que la gestión de datos fue propensa a errores para los participantes, quienes no solo están preocupados en seguir cuidadosamente las instrucciones de la actividad para localizar, cargar, y descargar artefactos hacia o desde un sistema de ficheros, sino que además tienen que compartir sus artefactos con otros participantes en actividades colaborativas.

Los participantes requieren ser instruidos en la descripción de actividad sobre los artefactos que deben usar en un sistema de ficheros compartido. El problema con esta aproximación usada en la gestión del flujo de datos es que las instrucciones pueden ser mal entendidas, mal aplicadas, olvidadas o ignoradas completamente por los participantes (Dillenbourg, 2002). Considere el siguiente comentario de un participante: *“Yo no sé si mi compañero olvidó cargar su fichero o lo subió en otro lugar, o quizás lo guardó con otro nombre como me pasó a mi anteriormente [student-squest]”*. Este resultado de investigación es soportado también por otros datos cualitativos mostrados en la *Tabla 2.3*, así como también por datos cuantitativos de los participantes: en una escala de 0 a 6, donde 6 es más propenso a errores y 0 menos propenso, los estudiantes evaluaron con 3.89 (desviación de 1.27) de que la gestión de datos del caso MOSAIC era propensa a errores; además, el 100% de los participantes afirman haber sufrido algún tipo de situación propensa a error durante la gestión de datos [student-squest].

El tercer resultado de investigación indica que el diseñador del curso enfrenta también situaciones propensas a error durante la descripción del flujo de datos. Considere el siguiente intercambio de correos electrónicos entre un estudiante y su tutor: *“Yo soy el participante rectángulo amarillo, y de acuerdo al resumen de artículos yo tendría que leer el artículo: Grid Architecture, pero no tengo acceso a éste [student-email]”*. A continuación, la respuesta del tutor, quien funge también como el diseñador del curso: *“Lo siento sobre la asignación de artículos. Me perdí con el detalle. He actualizado los recursos, de tal forma que las asignación de artículos ya coincide con la descripción de la actividad [designer-email]”*.

2. Contexto de investigación y estado del arte

Resultados de Investigación	Soporte	Fuente de Datos
1. El caso de estudio es significativo y relevante al problema.	-Opinión de los estudiantes sobre la importancia en la gestión de datos [0,6]: 4.57, desviación: 1.99 -Opinión de los estudiantes sobre la frecuencia del uso de la gestión de datos [0, 6]: 4.43, desviación: 1.99 -“El intercambio de ficheros es esencial en una experiencia como esta” -“La gestión de datos se usó mucho para localizar ficheros de otros compañeros” -“Tenía muchos problemas para localizar rutas y los ficheros correctos”	[student-gquest] [student-gquest] [student-gquest] [student-gquest] [student-gquest]
2. La gestión del flujo de datos es propensa de errores para los participantes.	-Opinión de los estudiantes sobre si la gestión de datos era propensa de errores [0, 6]: 3.89, desviación: 1.27 -“Tenía que poner mucha atención durante la gestión de datos, porque era muy fácil, equivocarse” -“Yo me equivoqué de directorio” -“Los errores pueden ser mayores conforme se compartan mayor número de ficheros” -“Es propenso de errores, en particular al inicio de la experiencia”	[student-squest] [student-squest] [student-squest] [student-squest] [manager-squest]
3. La gestión del flujo de datos es propensa de errores para el diseñador del curso.	-“Sin suficiente tiempo para la descripción detallada del flujo de datos, definitivamente es una situación propensa de errores”	[designer-squest]
4. La gestión del flujo de datos produce un incremento en la carga cognitiva de los participantes.	-Opinión de los estudiantes sobre si la gestión del flujo de datos produce un incremento de su carga cognitiva[0, 6]: 3.67, desviación: 1.73 -“Era tedioso y confuso tener que recordar cada vez dónde y cómo guardar un fichero”	[student-squest] [student-squest]
5. La gestión del flujo de datos produce un incremento en la carga cognitiva del diseñador del curso	-“En IMS LD no hay una forma comúnmente aceptada para especificar el flujo de datos en grupos dinámicos, así que tuvimos que improvisar. Obviamente, esto implica un exceso de carga cognitiva” -“No existe una buena opción para la interacción entre grupos, así que tuve que usar propiedades locales personales, lo cual incrementó mucho la complejidad”	[designer-squest] [designer-squest] [designer-squest]
6. Interacciones colaborativas del flujo de datos requiere especificarse a nivel de instancias	-24 valores de propiedades (información a nivel de instancias), correspondientes a los valores verde, amarillo, azul, rojo, triángulo, rectángulo y cruz están incluidos dentro del manifiesto de la UoL.	[designer-uol]
7. La descripción del flujo de datos a nivel de actividades con IMS LD en aprendizaje colaborativo, requiere instruir a los participantes en la descripción de actividad.	-“Cuando se usa un monitor para ver alguna propiedad, el participante tiene dos opciones: ver las propiedades del propio participante, o ver las propiedades de todos los participantes en el rol especificado. No hay forma de especificar que el estudiante A solo pueda ver el fichero del estudiante B a través del uso de elementos globales y monitores. Entonces, para resolver este problema, los participantes requieren ser instruidos dentro de la descripción de la actividad.	[designer-quest]

Tabla 2.3: Principales resultados de investigación del caso de estudio MOSAIC incluyendo datos de soporte seleccionados y las fuentes de los datos.

2. Contexto de investigación y estado del arte

El cuarto resultado de investigación indica que la gestión de datos por parte de los participantes les produce una carga cognitiva adicional al momento descargar o cargar un artefacto hacia o desde el sistema de ficheros compartido. Considere el siguiente comentario de un participante: *“la gestión de datos no era transparente a los estudiantes, quienes tenían que saber y entender cómo funciona internamente el mecanismo de gestión del flujo de datos. Este es un problema relevante, principalmente para estudiantes que no cuentan con conocimientos técnicos [student-squest]”*. Nuevamente, este resultado de investigación es soportado también por otros datos cualitativos de los participantes que son mostrados en la *Tabla 2.3*, así como también por los siguientes datos cuantitativos de los participantes: en una escala de 0 a 6, donde 6 produce una mayor carga cognitiva, los estudiantes evaluaron con 3.67 (desviación de 1.73) de que la gestión de datos durante el caso MOSAIC producía una carga cognitiva adicional; además, el 100% de los estudiantes afirman haber sufrido algún tipo de carga cognitiva adicional durante la gestión del flujo de datos [student-squest].

El quinto resultado de investigación indica que el diseñador del curso sufre una carga cognitiva adicional en la especificación del flujo de datos. Considere el siguiente comentario del diseñador del curso después de preguntarle sobre el error mencionado durante la asignación de artículos: *“La falla en la asignación de artículos, es solo un error de mi parte, debido a un exceso de carga cognitiva [designer-email]”*. Nuevamente, este resultado es también soportado por otros datos cualitativos del diseñador del curso mostrados en la *Tabla 2.3*.

Finalmente, el sexto y último resultado de investigación indica que un diseño de aprendizaje que describe un proceso de aprendizaje colaborativo (como el caso MOSAIC) requiere especificar información de la situación del flujo de datos. En efecto, debido a las complejas interacciones entre participantes y artefactos que son típicas del aprendizaje colaborativo, la especificación de la estructura del flujo de datos no es suficiente, sino que debe especificarse también información de las interacciones colaborativas entre instancias del flujo de datos, esto es, como se indica por los siguientes datos cuantitativos: se requirieron especificar dentro del diseño de aprendizaje los valores de 24 propiedades correspondientes a los distintos participantes. Sin embargo, incluir información de instancias dentro del diseño de aprendizaje dificultaría su reutilización para distintas situaciones de aprendizaje y contextos. Por tanto, se confirma una vez más que la especificación de un proceso de aprendizaje colaborativo requiere especificar tanto la estructura como una o más instancias de la estructura del flujo de datos. Si bien esta necesidad de describir estructuras y situaciones de aprendizaje colaborativo podría provocar una carga cognitiva adicional en el diseñador instruccional, ésta podría ser reducida o soportada por un sistema computacional que ayudara al diseñador instruccional a gestionar y manejar de forma efectiva dichas instancias y estructuras.

El séptimo y último resultado de investigación indica que en aprendizaje colaborativo, el flujo de datos a nivel de actividades especificado con IMS LD requiere instruir a los participantes directamente en la descripción de actividad (de aquí las 24 propiedades de los

participantes que se requieren especificar). Por tanto, a pesar de que IMS LD soporta la especificación del flujo de datos a nivel de actividades en un escenario de aprendizaje colaborativo real, como es el caso MOSAIC, se debe instruir a los participantes directamente en la descripción de la actividad, con los resultados de investigación encontrados. Entonces, para eliminar o reducir la carga cognitiva adicional y situaciones propensas a error en los participantes se requiere de una forma describir el flujo de datos a nivel de herramientas que permita alcanzar la automatización del flujo de datos. Más aún, separando la descripción de la estructura de aprendizaje colaborativo (flujo de aprendizaje y flujo de datos), de instancias de la estructura del flujo de datos, así como de la asignación de participantes a las actividades de aprendizaje y las instancias de herramientas, se podría alcanzar la reutilización de diseños de aprendizaje de manera efectiva. A continuación se enuncia el planteamiento del problema.

2.7.4 Planteamiento del problema

Del análisis del problema realizado, se obtiene el planteamiento del problema de esta Tesis Doctoral:

1. En LD no es posible especificar el flujo de datos a nivel de herramientas, ya que no es posible especificar en IMS LD la relación entre datos y herramientas abstractas, ni especificar la secuenciación de herramientas abstractas en una misma actividad de aprendizaje.
2. En LD no es posible especificar instancias del flujo de datos, ya que no es posible especificar en IMS LD la relación entre instancias de datos e instancias de herramientas, ni especificar la secuenciación de instancias de herramientas en una misma actividad de aprendizaje.
3. En LD no es posible especificar situaciones de aprendizaje colaborativo concretas, ya que no es posible especificar en IMS LD la asignación de participantes a las actividades de aprendizaje, ni a las instancias de herramientas.

Tomando en cuenta los principales objetivos de LD: automatización del proceso, reutilización de la definición del proceso, e interoperabilidad, todos estos son problemas relevantes, ya que si no se puede especificar el diseño del flujo de datos, no se puede reutilizar el diseño para derivar distintas instancias del flujo de datos (problema de reutilización de la definición del proceso); si no se pueden especificar instancias del flujo de datos, entonces no es posible alcanzar la automatización del proceso, al menos en lo que respecta al flujo de datos (el problema de la automatización del proceso); si no se puede especificar la asignación de usuarios a las actividades de aprendizaje y a las instancias de herramientas, entonces no se pueden describir situaciones de aprendizaje colaborativo concretas (problema de describir tanto estructuras como situaciones de aprendizaje colaborativo).

Distintos autores han relacionado LD con flujo de trabajo, y han realizado contribuciones del campo de Flujo de Trabajo para resolver problemas en LD. Entonces, entendiendo el concepto de flujo de trabajo se puede entender mejor el problema del flujo de

datos a nivel de herramientas en LD. A continuación, se revisa primero el campo de flujo de trabajo para después estudiar su relación con LD y después se estudian las soluciones actuales.

2.8 Flujo de trabajo

El concepto de flujo de trabajo se conoce y se ha aplicado desde hace mucho tiempo en la automatización de oficina (Ellis & Nutt, 1980); en la reingeniería de procesos de negocio (Brouns, 2000); hasta su actual aplicación en la gestión de procesos de negocio en las empresas (van der Aalst & van Hee, 2004). El término de flujo de trabajo se usa para enfatizar la secuencia de tareas realizadas por personas o sistemas para lograr un objetivo (van der Aalst & van Hee, 2004).

2.8.1 Definición

La Asociación para la Gestión de Flujo de Trabajo (WfMC, *Workflow Management Coalition*), quien es la encargada de promover y desarrollar estándares de flujo de trabajo para la interoperabilidad y conectividad de herramientas, lo define como:

“Flujo de trabajo tiene que ver con la automatización de procedimientos donde los documentos, información o tareas son pasados entre los participantes de acuerdo a un conjunto de reglas definidas para alcanzar un objetivo de negocio global” (Hollingsworth & Hampshire, 1994).

“Flujo de trabajo tiene que ver con la automatización de procesos de negocios en una organización que permitan realizar la actividad correcta, en el lugar y en el momento correctos, por las personas correctas y con las herramientas y datos correctos; todo ello con el propósito de satisfacer los objetivos de la compañía” (Hollingsworth & Hampshire, 1994).

La idea subyacente del paradigma de flujo de trabajo es la de separar la lógica del proceso de negocio de la aplicación, de forma análoga a la separación de los datos de las aplicaciones en los sistemas de gestión de bases de datos. De esta forma, almacenando la lógica de negocio como datos estructurados fuera de la aplicación, si la lógica de negocio cambia, la aplicación no lo hace, por lo que ésta es más flexible y más fácil de mantener (van der Aalst & van Hee, 2004).

2.8.2 Ciclo de vida de procesos de flujo de trabajo

El ciclo de vida que caracteriza a los procesos de flujo de trabajo es ilustrado en la *Figura 2.17*. Primero, en tiempo de definición, se diseña, analiza y se especifica el proceso, utilizando para ello herramientas de modelado, análisis y especificación de procesos, respectivamente. Después, en tiempo de ejecución, se pone en marcha el proceso (*enactment*) a partir de instanciarlo (crear instancias del proceso) y monitorear su ejecución, utilizando para ello un

2. Contexto de investigación y estado del arte

entorno de ejecución o sistema gestor de flujo de trabajo (WfMS, *Workflow Management System*). A continuación se describe cada una de las fases del ciclo de vida de procesos de flujo de trabajo.

La definición de un proceso implica la representación computacional de un modelo válido que describa el proceso. Para alcanzar este objetivo, es necesario diseñar el proceso, analizarlo para su validación y especificarlo en un lenguaje de modelado de especificación de procesos. Aunque existen varias metodologías para modelar y definir procesos, una ampliamente aceptada es la aproximación de modelado de flujo de trabajo basada en actividad (Cichocki, helal, Rusinkiewicz & Woelk, 1998).

Con el propósito de que el modelo del proceso sea interpretado computacionalmente, e intercambiado entre herramientas, se requiere de un lenguaje definición de procesos estándar, usualmente basado en un esquema XML (Hilera & Palomar, 2005). Este esquema XML es utilizado entonces para definir e interpretar documentos XML que definen formalmente el proceso y que se dicen conformes a la especificación estándar. En la *Figura 2.18* se muestra el meta-modelo básico de referencia propuesto por la WfMC para la definición de procesos de flujo de trabajo. La idea es que lenguajes de especificación de procesos de flujo de trabajo basados en XML se basen en este modelo conceptual.

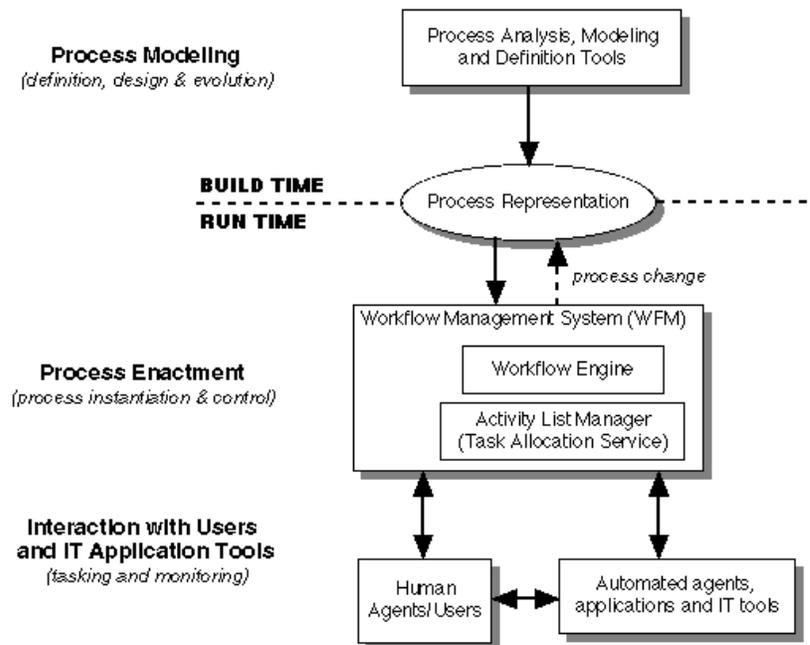


Figura 2.17: Caracterización de los sistemas de flujo de trabajo: definición y ejecución. Fuente: Hollingsworth & Hampshire, 1994).

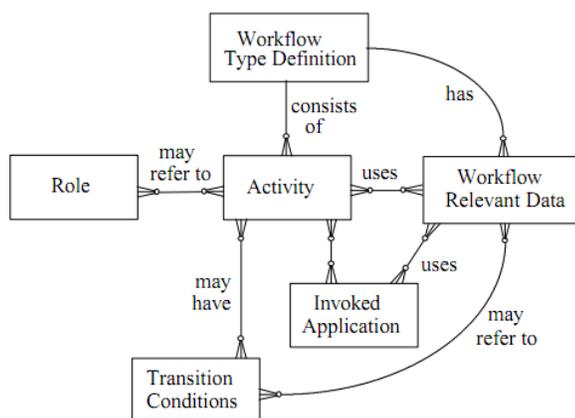


Figura 2.18: Meta-modelo básico de referencia propuesto por la WfMC para la definición de procesos de flujo de trabajo. Fuente: (Hollingsworth & Hampshire, 1994).

En el campo de flujo de trabajo existen varios lenguajes considerados estándares (Hornung, Koschmider & Mendling, 2006). La idea es usar uno de ellos para especificar la definición del proceso en un documento basado en XML. Después, un motor de flujo de trabajo (*workflow engine*) que sea conforme con el lenguaje de flujo de trabajo seleccionado se usa para interpretar y poner en marcha instancias del proceso.

Un caso (*case*) en el contexto de flujo de trabajo, se refiere a una instancia del proceso. El término instancia es sacado del mundo de la orientación a objetos y significa crear una copia del estado declarado en la definición del proceso. Los casos son creados por el motor de flujo de trabajo, a partir de crear instancias de las variables declaradas en la definición del proceso (Baeyens, 2004). De esta forma, casos similares pertenecen al mismo tipo de caso, es decir, son instancias del mismo proceso. Ejemplos de casos son las reclamaciones de seguros de clientes, donde cada solicitud de un cliente corresponde a un caso. Puesto que cada caso tiene un identificador y un estado propios, una de las principales funciones del WfMS es saber en todo momento el estado de un caso. Por ejemplo, saber el estado de la reclamación del seguro de un cliente; responder a un cliente si su solicitud de reclamación ha sido iniciada, aceptada, o rechazada, etc. El WfMS debe proporcionar una interfaz a través de la cual el administrador del sistema instancie y gestione casos.

La definición del proceso de flujo de trabajo se lleva a cabo a través de usar lenguajes declarativos, lo que significa que describen lo que una implementación que es conforme al lenguaje de especificación debe hacer, pero no cómo debe hacerlo (Nutt, 1996). Por tanto, un número de decisiones relativas a la implementación son dejadas al diseñador del motor, resultando en entornos de ejecución heterogéneos y por lo tanto propietarios. Una de estas decisiones, es la responsabilidad del entorno de ejecución de relacionar personas concretas con participantes o roles abstractos definidos en el proceso. Más aún, el WfMS debe proporcionar una interfaz a través de la cual el administrador del sistema defina las políticas para la asignación de personas concretas a las actividades que están listas para ser realizadas (*work items*), correspondientes a cada caso (van der Aalst & van Hee, 2004).

Para poner en marcha un proceso de flujo de trabajo expresado en un documento de definición de proceso, se requiere de un componente software capaz de analizar e interpretar el documento. Este componente es llamado motor de flujo de trabajo, el cual, como se muestra en la *Figura 2.17*, forma parte de un entorno de ejecución mayor: el WfMS, cuyo principal objetivo es su capacidad de instanciar y gestionar casos (van der Aalst & van Hee, 2004).

2.8.3 Ventajas y desventajas de la aproximación de flujo de trabajo

La definición de procesos usando un lenguaje definición de procesos de flujo de trabajo, tiene las siguientes ventajas (van der Aalst & van Hee, 2004):

- **Automatización del proceso.** Las tareas a realizar y los recursos a utilizar son entregados automáticamente a los participantes del proceso.
- **Reutilización del proceso.** La lógica del proceso se define una sola vez y se instancia y reutiliza para distintos casos.
- **Interoperabilidad.** Definiendo el proceso usando un lenguaje estándar definición de procesos de flujo de trabajo, éste se puede intercambiar con distintas herramientas que son desarrolladas conformes al estándar.
- **Aceleración del proceso desarrollo.** Puesto que la lógica de la definición del proceso se especifica en un documento separado de la aplicación, si la lógica del proceso cambia, sólo cambia el documento pero no la aplicación, por tanto se desarrollan sistemas más flexibles y más fáciles de mantener.

Sin embargo, también existen desventajas:

- **Necesidad de nuevos estándares.** Dificultad para poner de acuerdo a toda una comunidad de práctica. Nuevas necesidades en el modelado de procesos, implican nuevos elementos del meta-modelo, por tanto nuevos lenguajes de modelado (o modificación de los existentes) y la necesidad de nuevos estándares.
- **Aplicación a dominios específicos.** Dificultad para que un meta-modelo de flujo de trabajo genérico incluya elementos específicos de dominio, como por ejemplo, el dominio educativo.
- **Rigidez de la definición del proceso.** Dificultad de modificar la lógica del proceso una vez que el proceso ya está en marcha.
- **Definición de la instanciación del proceso.** La semántica de instanciación no está definida en flujo de trabajo (Decker & Mendling, 2008), lo que en la práctica se puede traducir en una dificultad real para intercambiar procesos entre herramientas.

A continuación se introducen algunos de los lenguajes existentes de flujo de trabajo.

2.8.4 Caracterización de los procesos de flujo de trabajo

Georgakopoulos et al. (1995) caracterizan los procesos de flujo de trabajo a largo de un espectro continuo que va desde procesos orientados a personas hasta procesos orientados a sistemas. En un extremo del espectro, están los procesos orientados a personas que involucran a personas realizando tareas, y el sistema gestor de flujo de trabajo solo se utiliza para soportar la coordinación y colaboración de las personas, las personas sin embargo, son las responsables de la consistencia de los datos. En el otro extremo del espectro, los procesos orientados a sistemas involucran a sistemas software automatizados realizando las tareas.

Dayal et al. (2001) caracterizan los procesos de flujo de trabajo en dos dimensiones como se muestra en la *Figura 2.19*: la estructura del proceso y la automatización de la tarea. Por un lado, la dimensión de la estructura del proceso comprende a procesos no estructurados, semi-estructurados y altamente estructurados. En los procesos no estructurados, no existen reglas o secuencias entre tareas y participantes que realizan el trabajo; en los semi-estructurados, parte de las reglas son predefinidas y parte son modificadas o definidas al momento de realizar la tarea; y en los estructurados, las reglas de negocio siguen secuencias de tareas predefinidas. Por otro lado, la dimensión de la automatización de la tarea, va desde su orientación a las personas, cuando las tareas son manuales y requieren la intervención del juicio humano; hasta su orientación a los sistemas, cuando la tarea es ejecutada automáticamente por un sistema software.

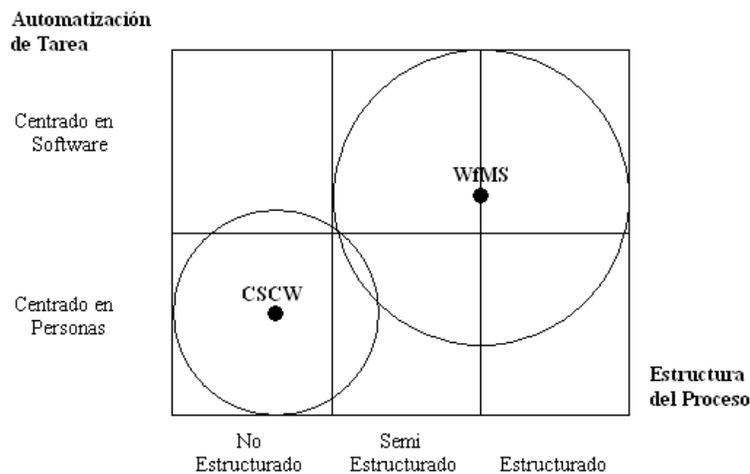


Figura 2.19: Caracterización de flujo de trabajo propuesta por Dayal. Fuente: (Dayal, Hsu & Ladin, 2001).

De esta forma, de acuerdo con Dayal et al. tradicionalmente el centro para el diseño de sistemas de gestión de flujo de trabajo (WfMS) cae en la esquina superior derecha de la *Figura 2.19*, es decir en el diseño de procesos estructurados automatizados; mientras que, tradicionalmente el centro para el diseño del trabajo cooperativo soportado por ordenador

(CSCW) cae en la esquina inferior izquierda, es decir en procesos no estructurados gestionados por personas (Dayal, Hsu, & Ladin, 2001).

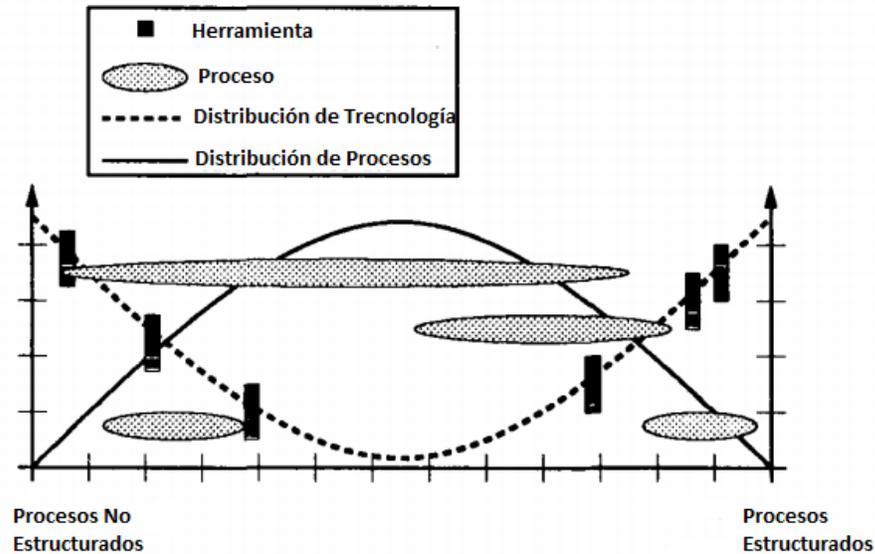


Figura 2.20: Soporte tecnológico para procesos de flujo de trabajo. Fuente: (Sheth, Georgakopoulos, Joosten, Rusinkiewicz, Scacchi, Wileden, & Wolf, 1997).

Sheth et al. (1997) describen cómo es que las tecnologías tradicionales de flujo de trabajo soportan los diferentes tipos de procesos, como se muestra en la *Figura 2.20*. De acuerdo con Sheth et al., la mayoría de los procesos se componen de procesos semi-estructurados, es decir se componen de procesos que tienen características que van desde procesos no estructurados hasta los estructurados. Sin embargo, Sheth et al. afirman que son poco frecuentes los procesos que se componen solamente de procesos puramente no-estructurados o puramente estructurados. Sin embargo, las tecnologías más frecuentes solo se centran en describir este tipo de procesos, como el CSCW para los puramente no estructurados, o el de flujo de trabajo para los puramente estructurados. Por tanto, el reto en el modelado de procesos de flujo de trabajo es describir formas de incorporar trabajo humano con tareas automatizadas. A continuación se estudia la relación entre LD y flujo de trabajo.

2.8.5 Complementariedad de LD y flujo de trabajo

Debido a la naturaleza de flujo de trabajo de la aproximación LD, una pregunta interesante sería saber ¿dónde estaría situado LD en la caracterización de flujo de trabajo bidimensional propuesta por Dayal et al. en la *Figura 2.19*. Para responder a esta pregunta, caracterizamos LD, primero en la dimensión de la estructura del proceso y después en la dimensión de la automatización de la tarea. Por una parte, en IMS LD se pueden especificar diseños de aprendizaje no estructurados, semi-estructurados y estructurados, por lo que IMS LD cubre muy bien todo el espectro de la dimensión de la estructura del proceso (ver *Figura 2.21*). Por otra parte, en IMS LD las actividades de aprendizaje deben ser realizadas por personas (con

2. Contexto de investigación y estado del arte

alguna intervención del sistema), mientras que las actividades de soporte deberían poder ser automatizadas por el sistema (lo cual no ocurre, al menos en lo que respecta al flujo de datos). Por tanto, en la dimensión de la automatización de la tarea, IMS LD cubre muy bien toda la parte del espectro de orientación a la persona, pero no de orientación a los sistemas (ver *Figura 2.21*).

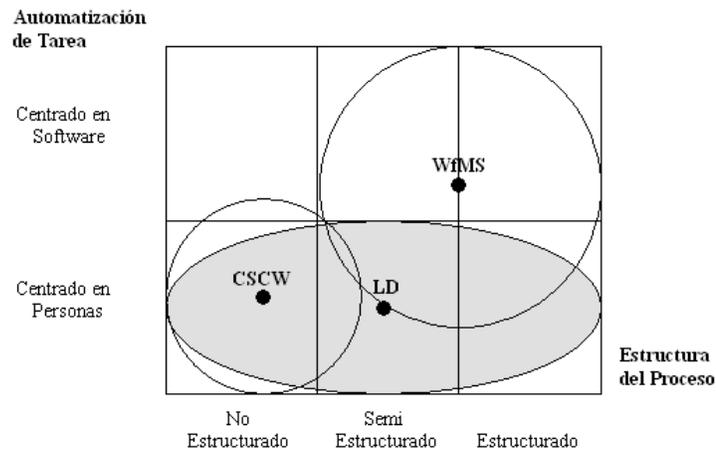


Figura 2.21: Categorizando LD (en gris) en la caracterización de flujo de trabajo propuesta por Dayal en (Dayal, Hsu & Ladin, 2001).

Del análisis realizado y como se puede ver en la *Figura 2.21*, al parecer LD y flujo de trabajo son dos aproximaciones complementarias entre sí del paradigma de flujo de trabajo aplicado en un espectro que va, desde procesos no estructurados y estructurados orientados a personas, como es el caso de LD, hasta procesos estructurados orientados a sistemas software, como es el caso de los sistemas de gestión de flujo de trabajo tradicionales.

Esta complementariedad identificada entre LD y flujo de trabajo toma especial relevancia cuando la relacionamos con el problema identificado en IMS LD, ya que las tecnologías de flujo de trabajo podrían contribuir ahí donde IMS LD falla: en especificar el flujo de datos a nivel herramientas para alcanzar la automatización del proceso. Pero, del mismo modo, IMS LD se podría utilizar ahí donde flujo de trabajo falla: especificación de tareas humanas, como Sheth y Dayal afirman que las tecnologías de flujo de trabajo debería de ser. Por tanto, una solución por composición de LD y de flujo de trabajo resulta interesante no solo porque puede utilizarse para resolver el problema del flujo de datos de LD, ya que el flujo de trabajo soporta la descripción del flujo de datos, sino también porque puede utilizarse como una solución intermedia hacia una futura integración de LD y flujo de trabajo.

2.8.6 Relación entre LD y Flujo de Trabajo

Existe una relación intrínseca entre LD y flujo de trabajo. Por un lado, IMS LD es el lenguaje estándar de LD usado para describir la parte declarativa del diseño de aprendizaje de un curso

2. Contexto de investigación y estado del arte

(Vogten, 2005) que luego es interpretado por un motor (de flujo de aprendizaje) con el objetivo de repetir el curso una y otra vez para diferentes usuarios y situaciones (Koper, 2007). La idea central de IMS LD para describir diseños de aprendizaje son las actividades. IMS LD utiliza la metáfora de la obra de teatro para modelar el flujo de aprendizaje. Un *play* que se compone de actos y cada acto que se compone de roles y partes de roles. Los roles se asocian a las actividades y los usuarios concretos a los roles. Pero son los roles y las actividades (no los usuarios concretos) los que constituyen la descripción del diseño de aprendizaje. Por otro lado, XPDL (lenguaje estándar de flujo de trabajo de la WfMC) define solo la parte declarativa de un proceso de flujo de trabajo que luego es interpretado por un motor (de flujo de trabajo) para poner en marcha uno o más casos (Baeyens, 2004). En XPDL se declaran roles y actividades, pero son las actividades el bloque fundamental que define el proceso. La idea central es que los roles se asocian a las actividades y las personas concretas a los roles. Pero son los roles y las actividades (no las personas concretas) las que constituyen la definición del proceso (Swenson & Irwin, 1995).

Baggetun et al. (2004) dibujan una relación entre procesos de flujo de trabajo y guiones pedagógicos:

“Existe una similitud entre el modelo de flujo de trabajo y un modelo de script. Un script CSCL es una secuencia de fases, tal que podemos verlo como un proceso de flujo de trabajo. Cada fase representa una actividad. Con las transiciones, podemos modelar el flujo de control y el flujo de datos entre las fases”

Más aún, la propia especificación IMS LD establece una relación entre IMS LD y flujo de trabajo:

“La coordinación de las actividades de los estudiantes en los distintos roles y su uso de recursos crea un 'flujo de aprendizaje', similar a la coordinación de actividades de un sistema de flujo de trabajo (pero sin el paso de documentos de un sistema de flujo de trabajo orientado a documentos)” (IMS LD, 2003).

Por tanto, no es raro que distintos autores en la literatura de LD se hayan acercado al campo de flujo de trabajo para proponer soluciones a distintos problemas identificados en LD. Por ejemplo, Caeiro et al. (2003) toman ideas del campo de flujo de trabajo para proponer nuevas estructuras de secuenciación de actividades en IMS LD. Vantroys y Peter (2003) proponen el lenguaje COWL, el cual es una extensión de XPDL, con el objetivo de poner en marcha procesos educativos especificados en IMS LD. Miao et al. (2005) proponen el lenguaje de guiones CSCL, el cual es un nuevo EML basado en la integración de IMS LD y XPDL para resolver distintos problemas identificados en IMS LD. Palomino-Ramírez et al. (2007) identifican complementariedad entre LD y flujo de trabajo.

A continuación se revisa el estado del arte relacionado con el problema identificado en esta Tesis Doctoral.

2.9 Trabajo relacionado

En esta sección se revisa el trabajo relacionado con el problema. Primero se revisa cómo se intenta dar la vuelta al problema en el contexto de IMS LD: primero a través de la adecuación de diseños de aprendizaje descritos con IMS LD y después a través de proponer una extensión a IMS LD. Una vez que se identifica que ambas aproximaciones tienen fuertes inconvenientes y debido a la relación identificada entre LD y flujo de trabajo, se revisa la posibilidad de describir el flujo de datos en el contexto de flujo de trabajo. Una vez hallados indicios de que la solución al problema puede estar en el campo de flujo de trabajo, se revisan las contribuciones de dicho campo a escenarios educativos en general y a LD en particular. Entonces, se revisan las contribuciones del campo de flujo de trabajo que podrían ser usadas para resolver el problema del flujo de datos a nivel de herramientas. Por último, se realiza una discusión general que toma en cuenta los inconvenientes de las soluciones actuales, lo cual dará pauta a la propuesta de esta Tesis Doctoral.

2.9.1 Adecuación de IMS LD

Algunos autores como Koper et al. (2009) y Miao et al. (2005) han propuesto formas indirectas de explotar la expresividad de IMS LD en procesos de aprendizaje colaborativo para incluir información de instancias del flujo de datos a nivel de actividades. Por ejemplo, la *Figura 2.22a* y la *Figura 2.22b* ilustran la aproximación usada por estos autores para representar las situaciones de aprendizaje colaborativo de revisión entre pares descritas anteriormente en la *Figura 2.10a* y *Figura 2.10b*, respectivamente. Esta aproximación se basa en adecuar el diseño para incluir información de los participantes en la forma de roles (Estudiante-x) de tal forma que se pueda discriminar qué datos son asignados a cada estudiante.

Primero, una aproximación por adecuación de las capacidades de IMS LD para especificar el flujo de datos a nivel de actividades. La principal ventaja de esta aproximación es que se mantiene interoperabilidad con la comunidad LD. Sin embargo, esta aproximación tiene tres importantes desventajas. La primera, es que el flujo de datos se especifica entre actividades, y no entre herramientas, lo que implica que el sistema no tiene suficiente información para entregar automáticamente los datos a las herramientas, por tanto requiriendo la intervención del participante (no se consigue el objetivo de automatización del proceso). Como consecuencia, los participantes sufren de una carga cognitiva adicional y situaciones propensas a error cuando manejan estos artefactos y herramientas (Palomino-Ramírez, Bote-Lorenzo, Asensio-Pérez, Dimitriadis & de la Fuente-Valentín, 2008). La segunda desventaja, se debe a que el diseño resultante no puede ser reutilizado en otras situaciones de aprendizaje en la que existen un diferente número de estudiantes o en la que el flujo de documentos es modificado. Este problema se debe a que el diseño de aprendizaje se

2. Contexto de investigación y estado del arte

describe no por su estructura, sino por una situación del flujo de datos (la particularización de la revisión entre pares al conjunto específico de estudiantes que van a tomar parte en la situación de aprendizaje, así como en la forma en la que éstos esperan intercambiar sus resultados). Por tanto, con esta aproximación se necesitaría un diseño de aprendizaje distinto para cada situación del flujo de datos. Por último, la tercera desventaja, es que esta aproximación tiene un efecto negativo en el diseño, y es que éste ya no describe la estructura del flujo de aprendizaje (una actividad después de otra) como en la *Figura 2.9*, sino más bien el flujo de actividades requerido para describir las interacciones colaborativas del flujo de datos a nivel de actividades (como finalmente ocurre en la *Figura 2.22*). Esto es, el diseño del flujo de aprendizaje es dirigido por la necesidad de especificar el flujo de datos a nivel de actividades.

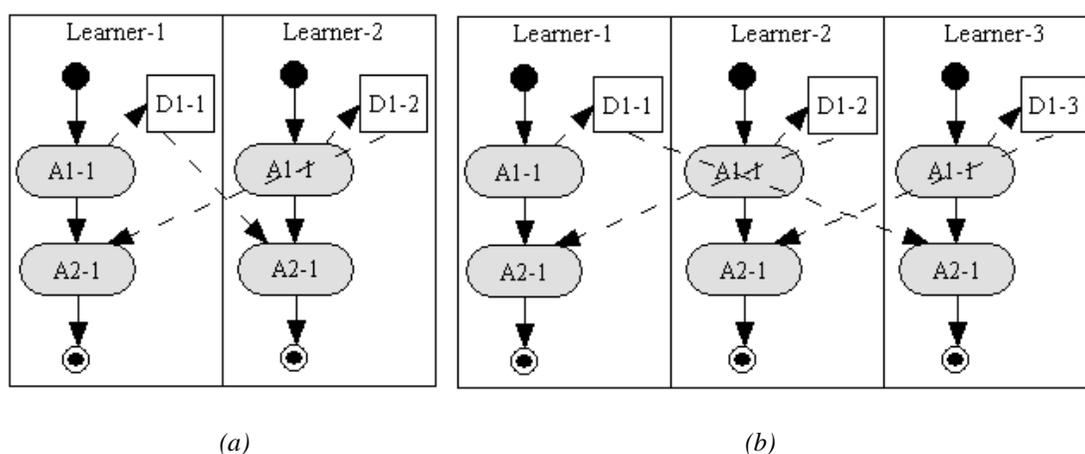


Figura 2.22: Aproximación usada en (Koper, 2009) y (Miao, Burgos, Griffiths & Koper, 2008) para especificar con IMS LD el flujo de datos a nivel de actividades del proceso de aprendizaje de revisión entre pares (a) entre dos estudiantes e (b) entre tres estudiantes.

2.9.2 Extensión de IMS LD

Algunos autores como König y Paramythis (2010) han propuesto una extensión de las capacidades de IMS LD con el objetivo de proporcionar el soporte suficiente para artefactos colaborativos que modele el flujo de datos a nivel de actividades. Ellos argumentan que un diseño de aprendizaje especificado con IMS LD no tiene información explícita (sino implícita dentro de recursos de contenido *imsldcontent*) sobre qué resultado va a ser creado en qué actividad, y dónde va a ser usado. Como se muestra en la *Figura 2.23*, un diseñador instruccional puede definir un artefacto como una colección (*collection*); o definir si el artefacto es visible en una cierta actividad (referenciado como *used-artefact*); o definir un artefacto como entrada (*input*) o salida (*output*) de una actividad. Con esta extensión propuesta a IMS LD, los autores pretenden modelar el flujo de datos a nivel de actividades, en donde los datos ya no son modelados como propiedades, sino a través de elementos del meta-modelo.

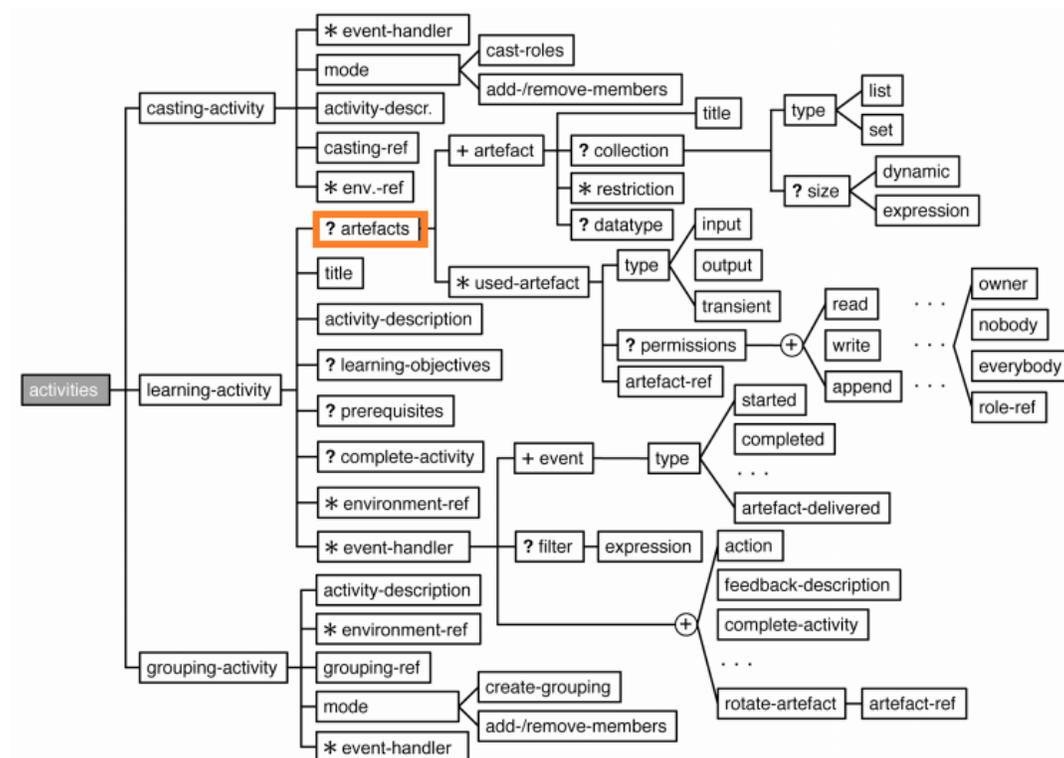


Figura 2.23: Extensión al elemento *activities* de IMS LD, resaltando la inclusión del elemento “*artefact*”. Fuente: (König & Paramythis, 2010).

Segundo, una aproximación por extensión de las capacidades de IMS LD. Desafortunadamente, esta aproximación solo tiene desventajas. La primera es que la modificación propuesta a IMS LD no permite la provisión de la información requerida para manejar automáticamente el flujo de datos a nivel de herramientas, sino solamente entre actividades, por tanto no se alcanza el objetivo de automatización del proceso. La segunda desventaja, al igual que en la aproximación anterior, es que el diseño resultante no puede ser reutilizado en otras situaciones del flujo de datos, ya que se necesitaría un diseño de aprendizaje distinto para cada situación derivada de la estructura del flujo de datos. Finalmente, la tercera desventaja es que la aproximación no sería interoperable con la comunidad LD.

2.9.3 Descripción del flujo de datos en el campo de flujo de trabajo

La descripción de procesos de flujo de trabajo se puede entender de manera efectiva desde dos perspectivas principales (van der Aalst, Hofstede, Kiepuszewski & Barros, 2003):

- **Control de flujo (*workflow control*).** Describe las actividades y su orden de ejecución, a través de distintos elementos que permiten controlar el flujo de ejecución, por ejemplo, secuencia, decisión, paralelismo, y sincronización.

2. Contexto de investigación y estado del arte

- **Flujo de datos (*workflow data*)**. Describe documentos de negocio y otros objetos que fluyen entre las actividades.

Desde la perspectiva del flujo de datos, existen, dos principales características que ocurren repetidamente en diferentes procesos de flujo de trabajo (Russell, ter Hofstede, Edmond & van der Aalst, 2005):

- **Visibilidad de los datos**. Se refiere al alcance de los datos respecto de los elementos activos de un proceso de flujo de trabajo.
- **Interacción de los datos**. Se refiere a la forma en que los datos se comunican entre los componentes activos dentro de un proceso de flujo de trabajo.

A partir de los patrones de flujo de datos (*workflow data patterns*) identificados en (Russell, ter Hofstede, Edmond & van der Aalst, 2005), se examinan varias formas comunes en que los datos pueden ser definidos y utilizados desde el contexto de un motor de flujo de trabajo. Típicamente, estas variaciones se relacionan con la forma en que los datos son declarados en el proceso principal de flujo de trabajo, y más importante todavía, es la forma en que los datos son usados. La definición de datos en el proceso de flujo de trabajo en la *Figura 2.24* es ilustrada por la frase *def var-name*. Dependiendo de dónde se definen los datos, la variable puede tener alcance de tarea, de bloque, de caso o de proceso, indicando el alcance o visibilidad de los datos. Los lugares donde el dato puede ser leído son ilustrados por la frase *use()*. De esta forma, los datos definidos a nivel de tarea solo son visibles (accesibles) por instancias de ejecución de la tarea (*Figura 2.24a*). Los datos definidos a nivel de bloque, solo son visibles por el *subworkflow* que compone el bloque (*Figura 2.24b*). Los datos definidos a nivel de instancia del proceso o caso, solo son visibles por los componentes de flujo de trabajo durante la ejecución del caso (*Figura 2.24c*). Los datos definidos a nivel del proceso de flujo de trabajo son visibles por todos y cada uno de los casos que están en el control del sistema de flujo de trabajo (*Figura 2.24d*).

A partir de los patrones del flujo de datos identificados en (Russell, ter Hofstede, Edmond & van der Aalst, 2005), se examinan varias formas comunes en que los datos pueden ser pasados entre componentes de un proceso de flujo de trabajo. De particular interés en este documento de tesis es la interacción de datos entre tareas y entre casos, esto es, la capacidad de comunicar datos entre dos instancias de tareas dentro del mismo caso y entre dos casos que se ejecutan concurrentemente. Por un lado, si las tareas representan herramientas, es de interés conocer la forma común de pasar datos entre herramientas. Por otro lado, si los casos representan instancias de datos e instancias de herramientas, es de interés conocer la forma común del paso de instancias de datos entre instancias de herramientas.

2. Contexto de investigación y estado del arte

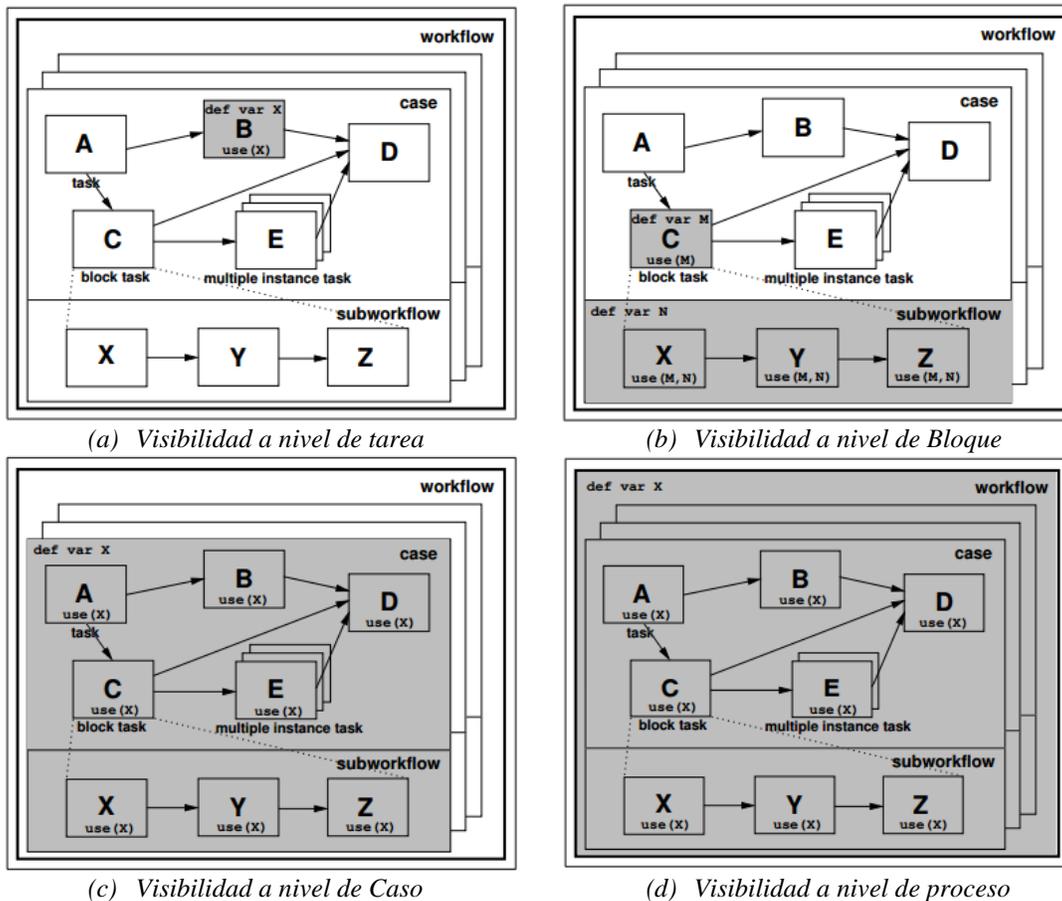


Figura 2.24: Visibilidad de datos de flujo de trabajo. Fuente: (Russell, ter Hofstede, Edmond & van der Aalst, 2005).

El paso de datos entre tareas es un aspecto fundamental de los sistemas de flujo de trabajo. En muchas situaciones tareas individuales se ejecutan en su propio espacio de direcciones y no comparten sus datos. Esto requiere la capacidad de moverlos entre distintas tareas. **Todos los motores de flujo de trabajo soportan la noción de paso de parámetros de una tarea a otra**, sin embargo, esto puede ocurrir de distintas formas dependiendo de la relación entre la perspectiva de datos y la perspectiva del control de flujo dentro del proceso de flujo de trabajo. Existen tres aproximaciones principales como se ilustra en la *Figura 2.25*:

1. **Canales integrados de datos y de control de flujo.** Ocurre cuando el control de flujo y los datos son pasados simultáneamente entre tareas utilizando el mismo canal, indicado en la *Figura 2.25a* por una flecha continua entre tareas.
2. **Canales de datos distintos.** Ocurre cuando los datos entre tareas de flujo de trabajo son pasados a través de un canal de datos explícito y distinto al canal de control de flujo, indicado en la *Figura 2.25b* por una flecha punteada entre tareas.

2. Contexto de investigación y estado del arte

3. **Almacenamiento de datos globales.** Ocurre cuando las tareas comparten los mismos datos y no se requiere paso de datos explícitos, típicamente a través de datos globales compartidos, como se muestra en la *Figura 2.25c*.

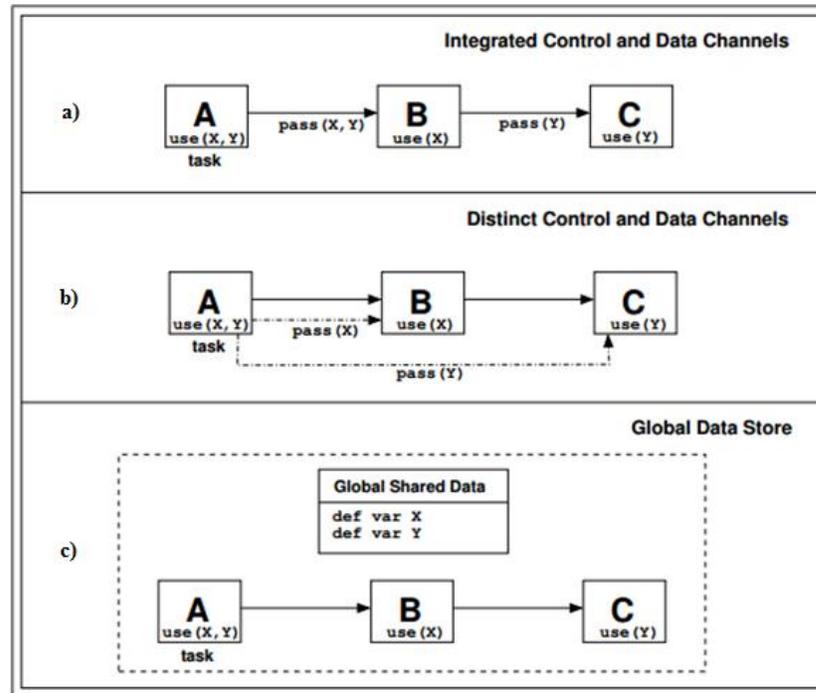


Figura 2.25: Aproximaciones de la interacción de datos entre tarea. Fuente: (Russell, ter Hofstede, Edmond & van der Aalst, 2005).

El paso de datos entre casos que se ejecutan concurrentemente también es de particular interés en este documento de tesis. De acuerdo con (Russell, ter Hofstede, Edmond & van der Aalst, 2005), en flujo de trabajo esto se puede conseguir respaldando en un repositorio de datos compartido los datos de los casos que están en comunicación. Esto puede ser una facilidad soportada a nivel de flujo de trabajo o un repositorio de datos externo. Ambos escenarios son ilustrados en la *Figura 2.26*. Ambas aproximaciones requieren que los casos que se comunican tengan conocimiento apriori de la ubicación de los datos que se van a pasar. También, requiere de la disponibilidad de una solución para resolver los potenciales problemas de concurrencia que pueden surgir cuando múltiples casos necesiten comunicarse unos con otros.

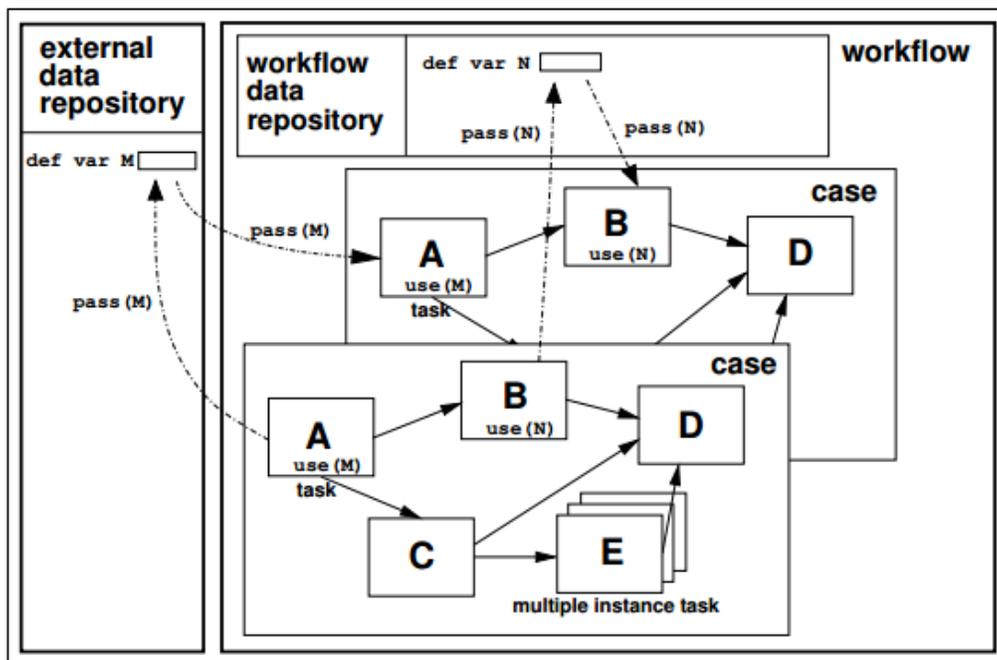


Figura 2.26: Interacción de datos entre casos de flujo de trabajo. Fuente: (Russell, ter Hofstede, Edmond & van der Aalst, 2005).

De la revisión anterior, parece ser que la descripción del flujo de datos, tanto para estructuras como para situaciones de aprendizaje colaborativo es factible en el campo de flujo de trabajo. A continuación, se lleva a cabo una revisión de las contribuciones del campo de flujo de trabajo, primero en la descripción de escenarios de aprendizaje en general, y después en la aproximación LD en particular.

2.9.4 Contribuciones de flujo de trabajo a escenarios de aprendizaje

Sistemas de gestión de flujo de trabajo (WfMS, *Workflow Managements Systems*, por sus siglas en inglés) han sido usados para soportar gestión de contenidos en *e-learning* (Simon, 2003), pero también usados para describir procesos educativos (Baggetun, Barros, Fesakis, Girardin, Hoeksema, Hämäläinen, & Vantroys, 2004).

La primera aplicación identificada de flujo de trabajo en educación es el sistema Flex-eL (Lin, Ho, Sadiq & Orłowska, 2002). Flex-eL es un entorno basado en flujo de trabajo usado para la entrega de cursos educativos a través de soportar la coordinación de módulos, rutas flexibles de actividades de aprendizaje y actividades de evaluación. Ellos afirman que la integración de herramientas que soportan el proceso de aprendizaje habilitada por tecnología de flujo de trabajo proporciona un entorno de aprendizaje efectivo y flexible. La idea de utilizar tecnologías de flujo de trabajo para gestionar actividades de enseñanza y de evaluación viene de la naturaleza de los procesos de estudio. Según Lin et al. (2002), el entorno del proceso de aprendizaje es transformado a tres dimensiones de una aplicación de flujo de trabajo:

2. Contexto de investigación y estado del arte

- *Procesos*. Los cursos disponibles son modelados en la forma de definiciones de procesos conteniendo un número de actividades de aprendizaje.
- *Organización*. El proceso de estudio involucra diferentes personas realizando diferentes tareas. Estudiantes pueden ser vistos como aprendices individuales y los educadores pueden compartir el mismo rol para gestionar un curso específico.
- *Infraestructura*. Las computadoras, el contenido de estudio, y otros artefactos se requiere para realizar las tareas del proceso para la aplicación de aprendizaje basada en flujo de trabajo.

También identificaron una transformación clara y lógica entre los requisitos de aplicaciones en *e-learning* y las tecnologías de flujo de trabajo. Según Lin et al. (2002), las tecnologías de flujo de trabajo ofrecen muchos beneficios que pueden potencialmente enriquecer entornos de aprendizaje, tales como:

- Automatizar el proceso de aprendizaje, lo que puede potencialmente mejorar la productividad del estudiante y/o del educador.
- Proporcionar monitoreo continuo a todos los participantes.
- Soportar la planeación de las actividades de aprendizaje así como de los recursos que la soportan.
- Trabajo al propio ritmo de los estudiantes, ya que ellos pueden elegir sus rutas de aprendizaje preferidas.
- Gestión compartida de la información y del conocimiento.

Similarmente a Flex-eL, en (Cesarini, Monga & Tedesco, 2004) se presenta Virtual Campus, la cual es una plataforma que proporciona un entorno de diseño y ejecución que son usados para definir y poner en marcha cursos educativos como flujos de trabajo. El entorno de autoría permite a los educadores definir y adecuar rutas de aprendizaje y publicarlas como flujos de trabajo. El entorno de ejecución permite poner en marcha los flujos de trabajo y guiar a los estudiantes a través de rutas de aprendizaje relacionadas. Esta aproximación toma ventaja de reglas de procedimientos para definir rutas de aprendizaje. Cesarini et al. (2004) proponen la arquitectura de un sistema de gestión de e-learning en el que se describe el proceso de aprendizaje como un proceso de flujo de trabajo, entonces las actividades realizadas por los estudiantes son guiadas por un motor de flujo de trabajo. Ellos afirman que un flujo de trabajo se compone de participantes, acciones, documentos y un conjunto de reglas procedurales que, una vez codificadas en una definición de proceso, pueden ser usadas para dirigir la puesta en marcha del flujo de trabajo. Esencialmente, los participantes en un entorno de aprendizaje son estudiantes y educadores, las acciones son las actividades de aprendizaje requeridas para los estudiantes o las actividades de corrección realizadas por los educadores, los documentos son los artefactos que los estudiantes pueden producir, ya sea en laboratorios, realizando las tareas o tomando una evaluación. Las reglas de procedimientos son relaciones entre los temas del curso. Según Cesarini et al. (2004), para aplicar completamente un WfMS a un escenario educativo se debe resolver un problema de sincronización.

2. Contexto de investigación y estado del arte

Si bien tanto Flex-eL como Campus Virtual son entornos para que educadores diseñen sus cursos centrados en el proceso de aprendizaje y no solamente en la entrega y gestión de contenidos educativos, sin embargo, ninguno de los dos considera además del objetivo de automatización del proceso, los objetivos de reutilización e interoperabilidad. Más aún, ninguno de los dos entornos considera aspectos del aprendizaje colaborativo, tales como el trabajo en grupo, el uso de artefactos compartidos, la distinción entre describir estructuras y situaciones de aprendizaje, etc.

Hasta aquí, se ha revisado el uso de tecnologías de flujo de trabajo para describir escenarios de aprendizaje. A continuación, se revisa el uso de tecnologías de flujo de trabajo conjuntamente con tecnologías educativas de LD (en concreto IMS LD) para describir escenarios de aprendizaje colaborativo en general y describir el flujo de datos en particular.

2.9.5 Contribuciones de flujo de trabajo en LD

Vantroys y Peter (2003) identifican una relación entre LD y flujo de trabajo y proponen aprovechar lo mejor de los dos mundos: usar IMS LD como lenguaje de alto nivel para especificar procesos educativos (estructuras de aprendizaje), y después usar una extensión del lenguaje de flujo de trabajo XPDL como lenguaje de bajo nivel para la puesta en marcha de situaciones de aprendizaje. La idea de los autores es simple: tomar ventaja de la facilidad de expresar procesos educativos con IMS LD y tomar ventaja de usar motores tradicionales existentes del campo de flujo de trabajo para su ejecución. Con este objetivo, Vantroys y Peter proponen una transformación (ver *Figura 2.27.*) de diseños de aprendizaje especificados en IMS LD hacia documentos en el lenguaje COWL (Vantroys & Peter, 2003), el cual es una extensión de la expresividad de XPDL, y que se usa para poner en marcha situaciones de aprendizaje colaborativo.

Para usar modelos IMS LD (diseños de aprendizaje) en la plataforma COW (*Cooperative Open Workflow*, por sus siglas en inglés) (Vantroys & Peter, 2003), Peter y Vantroys definieron reglas basadas en XSLT (Clark, 1999) para transformar de IMS LD al lenguaje COWL.

De acuerdo a la *Figura 2.27*, el concepto de actividad en IMS LD y actividad en XPDL es relativamente idéntico; el concepto de entorno de IMS LD lo relacionaron con los conceptos de aplicaciones y datos de flujo de trabajo; el concepto de rol de IMS LD lo relacionaron con el concepto de participante de XPDL; los conceptos *play* y método de IMS LD los relacionaron con los conceptos de proceso y transiciones de XPDL. Algunos aspectos del aprendizaje colaborativo, tales como el soporte de grupos y la sincronización de actividades en grupos los modelaron a través de subprocesos de XPDL. Sin embargo, para la descripción de actividades colaborativas, tuvieron que proponer una modificación a XPDL. Según Peter y Vantroys, cuando en una actividad hay asignado un único participante se crea un único *workitem* (una única instancia de la actividad). Sin embargo, en aprendizaje colaborativo, los participantes pueden desempeñar múltiples roles, lo que significa que una misma actividad tendrá asociado múltiples *workitems* (uno por cada participante que

2. Contexto de investigación y estado del arte

desempeña el mismo rol). Entonces, para poder modelar interacciones en actividades colaborativas Peter y Vantroys extendieron XPDL para describir *workitems* (instancias de actividades asociadas a los participantes). Por tanto, para poner en marcha diseños de aprendizaje en la plataforma COW (*Cooperative Open Workflow*, por sus siglas en inglés) (Vantroys & Peter, 2003), el modelo IMS LD es transformado a un modelo en el lenguaje COWL.

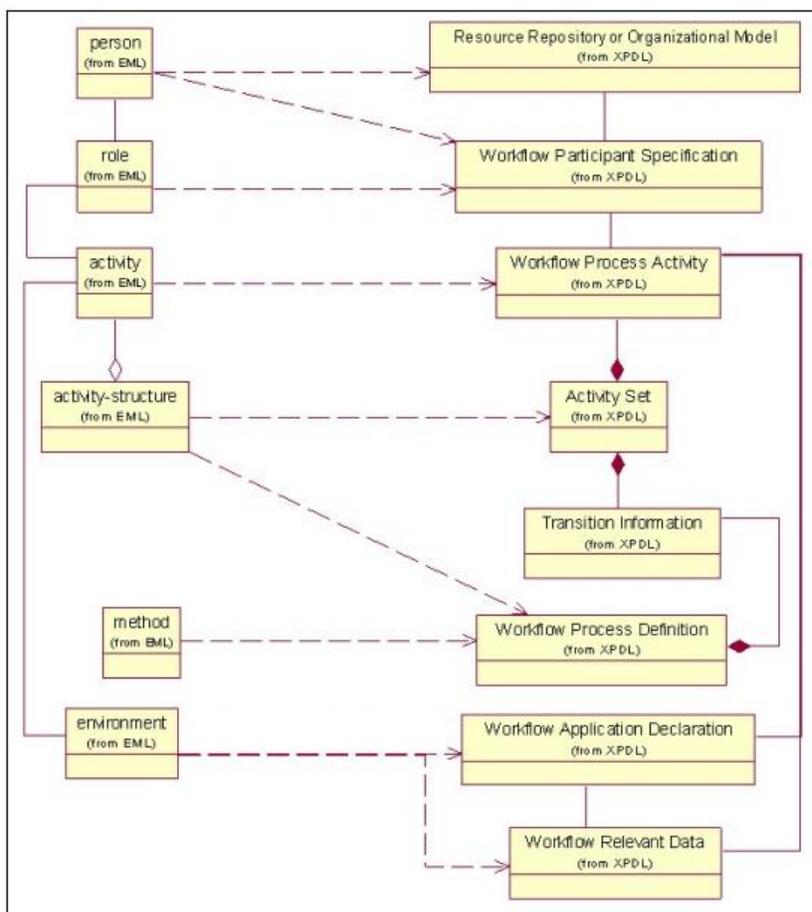


Figura 2.27: Transformación de elementos de IMS LD hacia COWL. Fuente: (Vantroys & Peter, 2003).

Sin embargo, esta aproximación tiene algunas desventajas. La primera se debe al hecho de que el flujo de datos de un diseño de aprendizaje corresponde al mundo educativo, y por tanto, de acuerdo con Peter y Vantroys, el flujo de datos debería especificarse en IMS LD y no en COWL, pero esto es justamente lo que IMS LD no puede hacer. Suponiendo que se alcanzara el objetivo de automatización del proceso, esto es, describir el flujo de datos a nivel de herramientas en el lenguaje COWL, entonces la segunda desventaja sería que no se alcanzaría el objetivo de reutilización, ya que el diseño resultante especificaría una situación de aprendizaje y no una estructura de aprendizaje, lo cual dificultaría su reutilización para distintas situaciones de aprendizaje. La tercera desventaja es que el diseño resultante no sería

2. Contexto de investigación y estado del arte

interoperable con herramientas y sistemas desarrollados en la comunidad de flujo de trabajo, ya que COWL no es un lenguaje estándar dicha comunidad.

Con el objetivo de resolver distintos problemas de expresividad de IMS LD en escenarios de aprendizaje colaborativo, entre ellos, el problema de especificar el flujo de datos a nivel de herramientas, Miao et al. (2005) proponen un nuevo EML llamado lenguaje de guiones CSCL, el cual está basado en el meta-modelo que se muestra en la *Figura 2.28*.

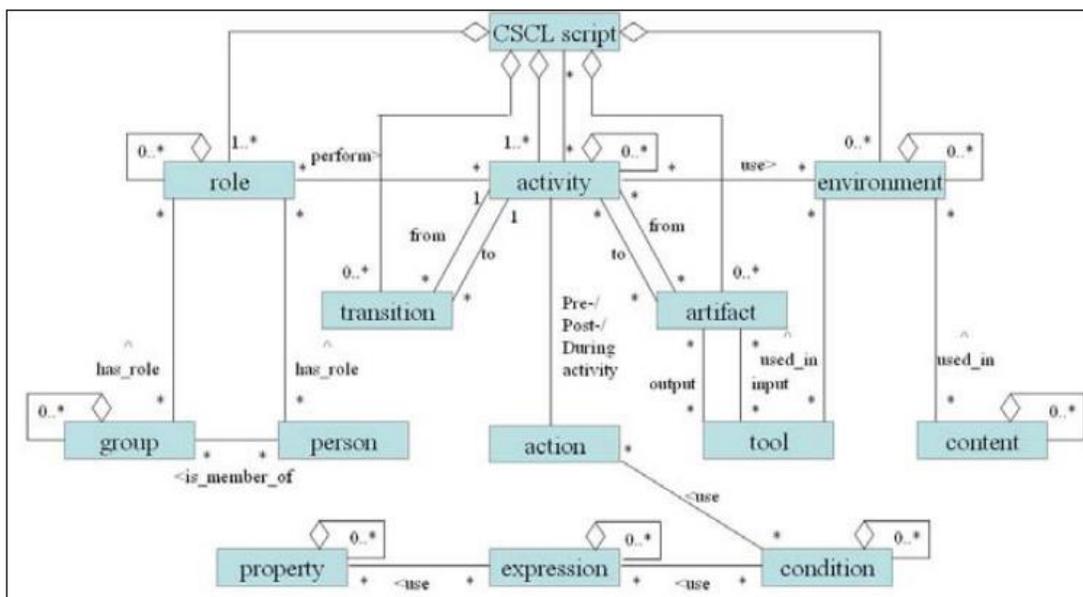


Figura 2.28: Meta-modelo del lenguaje de guiones CSCL. Fuente:(Miao, Hoeksema, Hoppe & Harrer, 2005).

Un análisis de este meta-modelo revela que se trata en realidad de la integración de elementos de los meta-modelos de IMS LD (ver *Figura 2.1*) y XPDL (Norin, 2002) (ver *Figura 2.29*). Siendo XPDL un lenguaje más expresivo que IMS LD (Miao, Hoeksema, Hoppe & Harrer, 2005), Miao et al. los usan como base de su nuevo meta-modelo, que después combina con elementos de IMS LD que son propios del dominio educativo, para crear así un nuevo meta-modelo con lo mejor de los dos mundos.

Si bien esta solución podría ser usada para resolver el problema de automatización del proceso (describir el flujo de datos), Miao et al. no dicen cómo resolver el problema de reutilización del diseño de aprendizaje resultante, esto es, el problema de distinguir entre estructuras y situaciones de aprendizaje colaborativo. Si el flujo de datos a nivel de herramientas describe solamente una situación de aprendizaje, entonces el diseño resultante no se podría reutilizar para distintas situaciones de aprendizaje. Más aún, una desventaja adicional sería que el diseño resultante no sería interoperable con la comunidad LD, ya que el lenguaje de guiones CSCL no es un lenguaje estándar en dicha comunidad.

2. Contexto de investigación y estado del arte

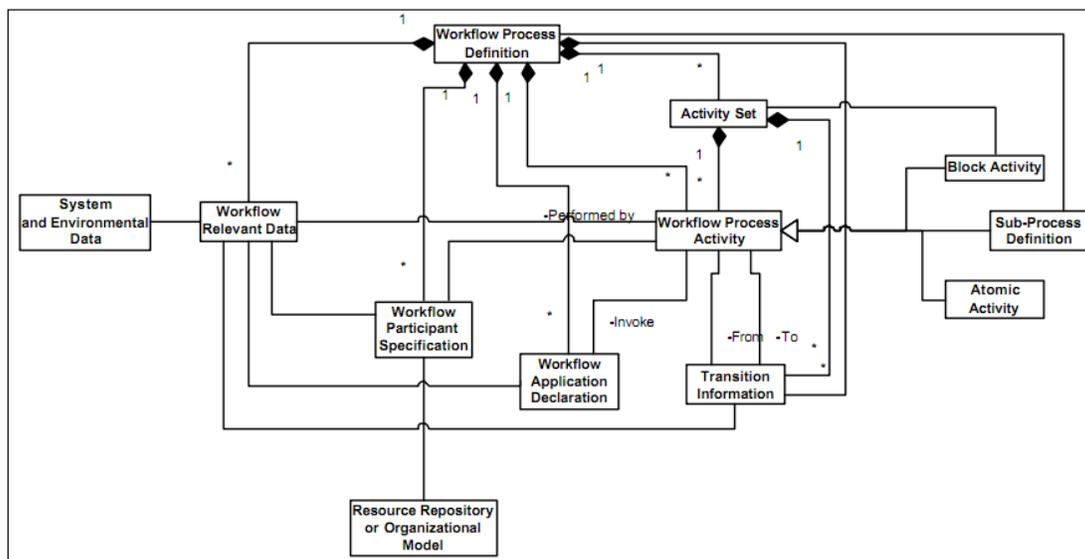


Figura 2.29: Meta-modelo de XPD 1.0. Fuente: (Norin, 2002).

2.9.6 Discusión

Existe una relación inherente entre LD y flujo de trabajo que se pone de manifiesto en el hecho de que varios investigadores han realizado contribuciones del campo de flujo de trabajo a LD. De esta afirmación emergen preguntas interesantes: ¿es posible prescindir de un lenguaje de LD en favor de usar un lenguaje de flujo de trabajo? ¿Cuáles son las ventajas y desventajas de usar flujo de trabajo para describir escenarios de aprendizaje?

Tanto IMS LD como XPD tienen en común que son lenguajes de modelado de procesos basados en actividad. Este tipo de lenguajes se componen de un meta-modelo que contiene todos los elementos del dominio usados para definir procesos. Mientras que XPD tiene elementos del dominio de procesos de negocio, IMS LD tiene elementos propios del dominio educativo. Por esta razón, algunos investigadores creen que lenguajes de flujo de trabajo no pueden reemplazar a lenguajes del dominio educativo, tal como IMS LD. Por ejemplo, para Marjanovic (2005), a pesar de que los lenguajes de flujo de trabajo están orientados a procesos, éstos no pueden ser usados directamente para describir procesos educativos, por varias razones. Primero, el principal objetivo de los lenguajes de flujo de trabajo es mejorar la eficiencia del proceso a través de la coordinación de tareas, mientras que el principal objetivo de los procesos educativos es conseguir los objetivos de aprendizaje de las personas. Segundo, los modelos de procesos de negocios son muy diferentes de los diseños educativos. Entre otras cosas, es más fácil de crear modelos de procesos de negocio que diseños de aprendizaje orientados a procesos. Tercero, los modelos de diseños de aprendizaje orientados a procesos necesitan ser mucho más flexibles que los modelos de procesos de negocio de flujo de trabajo. Sin embargo, para otros investigadores esto no es así, por ejemplo, para Baggetun et al. (2004) la vista de flujo de trabajo de procesos de aprendizaje es relativamente cercana al punto de vista de IMS LD. Considerando solo la parte

estructural de la especificación, la parte pedagógica de IMS LD, por ejemplo, los “objetivos” no son de utilidad para la puesta en marcha, la cual se enfoca principalmente en la estructura de actividades. Por lo tanto, para Baggetun et al., el modelo de IMS LD y el modelo de flujo de trabajo están relativamente cercanos entre sí: ambos se enfocan en la coordinación de actividades.

Con respecto a las ventajas y desventajas de usar tecnologías de flujo de trabajo para describir escenarios de aprendizaje, éstas pueden ser similares a las de describir procesos de flujo de trabajo (ver sección 2.8.3): automatización del proceso (automatizar la entrega de actividades de aprendizaje, artefactos y herramientas), reutilización de la definición del proceso (reutilizar el diseño de aprendizaje para distintas situaciones y contextos), y aceleración del proceso de desarrollo (poner en marcha situaciones de aprendizaje); mientras que las desventajas podrían ser: la necesidad de nuevos estándares (describir nuevos elementos en el meta-modelo debido a nuevas aproximaciones pedagógicas), la aplicación a dominios específicos (aplicación a dominios o pedagogías educativas específicas), la rigidez de la definición del proceso (necesidad de redefinir el proceso de aprendizaje en tiempo de ejecución), y la necesidad de definir la instanciación del proceso (la necesidad de describir situaciones de aprendizaje concretas).

A partir de la revisión de la literatura tanto de LD como de flujo de trabajo, en esta Tesis Doctoral se ha tomado en cuenta dos aspectos. Primero, que la comunidad de LD es una comunidad estable y madura, con muchas herramientas y sistemas desarrollados conforme al estándar IMS LD. Por lo tanto, es razonable que una investigación en la dirección de describir el flujo de aprendizaje (de diseños de aprendizaje) usando el estándar de LD (IMS LD) es relevante y significativa. Segundo, debido al enfoque orientado al aprendizaje de las personas de LD, y debido al enfoque orientado a la automatización de procesos de flujo de trabajo, y puesto que se identificó que el flujo de datos se puede describir en este último campo, entonces es razonable afirmar que la solución a los problemas identificados en esta Tesis Doctoral se puede conseguir a través de la contribución de estos campos.

2.10 Conclusiones

El contexto de investigación de esta Tesis Doctoral es el CSCL, el cual es un campo de estudio dedicado a la investigación de la aplicación de las TIC en educación, en particular el aprendizaje colaborativo, el cual se caracteriza por la interacción entre los miembros de un grupo. Procesos de aprendizaje colaborativo se pueden asistir de forma efectiva a través de estructurar las interacciones de los estudiantes en guiones colaborativos. En esta tesis se tratará con guiones colaborativos de tipo *macro-scripts* CSCL integrados, denominados aquí guiones colaborativos. Un guion colaborativo describe un flujo de actividades de aprendizaje que integra actividades individuales y colaborativas, actividades en grupos pequeños y de toda la clase, actividades virtuales y físicas, así como la gestión del flujo de datos que tiene lugar durante el proceso de aprendizaje. El flujo de datos, es tema central en esta tesis, y tiene que ver con la descripción de artefactos (datos) y herramientas, con la descripción de la

2. Contexto de investigación y estado del arte

secuenciación de herramientas, y con la descripción de la relación entre datos y herramientas que ocurre cuando artefactos creados por herramientas son después usados como entradas de otras herramientas en la misma o en distintas actividades de aprendizaje.

La práctica educativa efectiva es central a los conceptos de reutilización de guiones colaborativos y a su representación computacional. Por un lado, la distinción entre estructuras y situaciones de aprendizaje colaborativo puede ser usada para que el guion colaborativo pueda ser fácilmente adaptado y reutilizado en distintos entornos y contextos. Por otro lado, los lenguajes de modelado educativos (EML) pueden ser usados para que los guiones colaborativos puedan ser entendibles por ordenador. Aunque existen muchos EML, IMS LD es el lenguaje que ha despertado mucho interés en la comunidad investigadora y es considerado el impulsor del campo de Diseño de Aprendizaje. Un guion colaborativo en el contexto del campo de Diseño de Aprendizaje es llamado diseño de aprendizaje. Si bien es cierto IMS LD no ha sido ampliamente adoptado todavía debido a distintas razones organizacionales y tecnológicas (Griffiths & Liber, 2008). A pesar de ello, existe un renovado interés en usar IMS LD como un formato de interoperabilidad para desplegar diseños de aprendizaje producidos por herramientas de autoría existentes. Este renovado interés en IMS LD, la madurez del paradigma subyacente al marco que soporta a IMS LD, junto con el gran número de herramientas de autoría existentes y que son conformes con IMS LD, sugieren que resolver problemas de expresividad de IMS LD, al mismo tiempo que se trata de mantener la validez de herramientas IMS LD existentes, tanto como sea posible, es todavía un reto de investigación significativo.

Diseño de Aprendizaje es un campo multidisciplinario complejo que involucra teorías de aprendizaje, teorías de diseño instruccional, las mejores prácticas educativas y las TIC. Existen distintas formas de aproximar e implementar el campo de Diseño de Aprendizaje. Una forma de implementarlo es con IMS LD, el cual es un EML usado para describir el contenido y el proceso de una variedad de aproximaciones pedagógicas incluyendo el aprendizaje colaborativo. La relación inherente entre IMS LD y el campo bien conocido, estable y maduro de Flujo de Trabajo ayudaron a delimitar el contexto de este trabajo de investigación. El enfoque orientado a la coordinación de actividades que tienen en común la descripción de macro-scripts CSCL integrados en el campo CSCL, la descripción de diseños de aprendizaje en LD y la descripción de procesos de flujo de trabajo en el campo de Flujo de Trabajo, es la razón por la que se ha denominado en este documento de tesis: aproximación de Diseño de Aprendizaje basada en flujo de trabajo (LD) a la aproximación usada para implementar los objetivos del campo LD.

La aproximación LD se ha seleccionado como marco conceptual en esta tesis por varias razones: porque es de naturaleza de flujo de trabajo, el cual es un campo bien conocido, estable y maduro; porque es el marco conceptual que soporta la especificación IMS LD, que es considerado por muchos investigadores el lenguaje estándar de facto de la comunidad de Diseño de Aprendizaje; y porque IMS LD fue creado con la promesa de soportar diversidad pedagógica; compartición y reutilización de diseños de aprendizaje, así como la promesa de

2. Contexto de investigación y estado del arte

interoperabilidad entre una diversidad de herramientas y sistemas existentes y que son conformes con IMS LD. En resumen, en esta Tesis Doctoral se lleva a cabo una investigación en el campo LD, a través de una aproximación de LD basada en flujo de trabajo, y denominada aquí LD, y cuyo lenguaje de implementación estándar es IMS LD.

Los objetivos de describir diseños de aprendizaje con LD son los mismos que se buscan en la definición de procesos de flujo de trabajo: automatización del proceso, reutilización de la definición del proceso e interoperabilidad entre herramientas. La representación computacional de diseños de aprendizaje siguiendo la aproximación LD es esencial para alcanzar la automatización y reutilización del proceso. La distinción entre estructuras y situaciones de aprendizaje colaborativo es esencial para que un diseño de aprendizaje pueda ser fácilmente adaptado y reutilizado en distintos entornos y contextos. Si además, un diseño de aprendizaje se especifica utilizando un EML estándar (IMS LD), entonces es posible alcanzar también la interoperabilidad entre herramientas y sistemas software.

Un diseño de aprendizaje se caracteriza por la descripción de un flujo de aprendizaje y un flujo de datos. Aunque IMS LD se ha usado con relativo éxito para describir el flujo de aprendizaje en escenarios de aprendizaje colaborativo, sin embargo esto no se ha alcanzado para el flujo de datos. Distintos autores refieren el problema de especificar con IMS LD los artefactos de entrada y salida de las herramientas que soportan las actividades de aprendizaje. Este es un problema relevante, ya que sin la representación computacional del flujo de datos no sería posible alcanzar la automatización del proceso, al menos en lo que respecta al flujo de datos.

En este capítulo se llevó a cabo un análisis del problema de especificar el flujo de datos con IMS LD. Primero a nivel del modelo conceptual de IMS LD. Después, a nivel de dos ejemplos ilustrativos y por último a nivel de un caso de estudio auténtico. Del análisis realizado, se identificaron distintos problemas para describir el flujo de datos con IMS LD. El primero de ellos es que IMS LD no permite especificar la relación entre datos y herramientas abstractas, ni la secuenciación de herramientas abstractas en una misma actividad de aprendizaje. Segundo, IMS LD tampoco permite especificar la relación entre instancias de datos e instancias de herramientas, ni la secuenciación de instancias de herramientas en una misma actividad de aprendizaje. Por tanto, esta dificultad impide alcanzar el objetivo de automatización del proceso (en lo que respecta al flujo de datos). Tercero, en aprendizaje colaborativo se requiere describir no solamente la estructura de aprendizaje colaborativo (el flujo de aprendizaje y el flujo de datos), sino también describir situaciones de aprendizaje, a través de describir una o más instancias de la estructura del flujo de datos y de describir la asignación de los participantes a las actividades de aprendizaje y a las instancias de herramienta. Sin embargo, la especificación de la instanciación de procesos basados en flujo de trabajo es un problema abierto tanto en LD como en el campo de flujo de trabajo. Nuevamente, esta dificultad impide alcanza otro de los objetivos de LD: la reutilización de la definición del proceso.

2. Contexto de investigación y estado del arte

Soluciones actuales han sido propuestas para intentar resolver el problema. Primero, una adecuación por extensión de las capacidades de IMS LD que sin embargo solo especifica el flujo de datos a nivel de actividades, pero no a nivel de herramientas. Después, una aproximación por adecuación de IMS LD, que sin embargo ocurre entre actividades y el diseño resultante describe solamente una situación de aprendizaje en lugar de describir una estructura de aprendizaje, lo cual no permite que el diseño resultante pueda ser reutilizado en distintas situaciones de aprendizaje. Tercero, una aproximación por transformación de IMS LD hacia una extensión de las capacidades de XPDL, que sin embargo no es interoperable con la comunidad de flujo de trabajo; y por último una aproximación por combinación de IMS LD y XPDL en un nuevo EML llamado lenguaje de guiones CSCL, que sin embargo no es interoperable con la comunidad de LD.

En este espacio de soluciones del problema se identificó una posibilidad para resolverlo. Se trata de una aproximación por composición de LD y flujo de trabajo, es decir usar IMS LD para describir el flujo de aprendizaje y usar un lenguaje estándar de flujo de trabajo (por ejemplo XPDL) para describir el flujo de datos a nivel de herramientas. Esta aproximación fue seleccionada para resolver el problema de describir el flujo de datos a nivel de herramientas en LD al mismo tiempo que se mantiene interoperabilidad con la comunidad de LD y flujo de trabajo. En el siguiente capítulo se describe el método propuesto basado en esta aproximación.

3 LeadFlow4LD: Definición y Puesta en Marcha del Método Propuesto

En el capítulo anterior se analizó el problema de describir el flujo de datos en LD. Después de plantear el problema y revisar aproximaciones existentes en la literatura para resolverlo, en este capítulo se identifican primero los requisitos que debería cumplir una solución que resuelva los problemas identificados de acuerdo a los objetivos de LD. A continuación, se propone un espacio de soluciones de donde se elige la mejor aproximación para resolver el problema: una aproximación por composición de LD y flujo de trabajo. Entonces, se justifica la selección del lenguaje XPDL para especificar el flujo de datos a nivel de herramientas. A continuación, y tomando en cuenta la aproximación por composición propuesta, se describe una vista general del método propuesto llamado LeadFlow4LD. Después, se describe cada uno de los documentos que conforman un diseño de aprendizaje especificado conforme a LeadFlow4LD. Finalmente, se discuten diversas alternativas tecnológicas para la puesta en marcha de LeadFlow4LD.

3.1 Introducción

En el capítulo 2 de esta tesis se analizó el problema de representar computacionalmente en LD el flujo de datos a nivel de herramientas, tanto en procesos y situaciones de aprendizaje colaborativo. Más concretamente, y como resultado de dicho análisis, se identificaron distintas dificultades para describir el flujo de datos en LD, tales como la relación entre datos y herramientas, la secuenciación de herramientas en una misma actividad de aprendizaje (tanto para herramientas abstractas como para instancias de herramientas), así como para describir la asignación de participantes a las actividades de aprendizaje y a las instancias de herramientas, lo cual permite describir situaciones de aprendizaje colaborativo concretas.

Así también, en el capítulo 2 se identificó una relación inherente entre LD y flujo de trabajo. En particular, resulta interesante la complementariedad entre ambos, ya que si tomamos en cuenta los principales objetivos de LD, como son la automatización y reutilización del proceso, y la interoperabilidad con sistemas y herramientas, parece ser que una solución por composición podría ser usada para resolver los problemas anteriormente mencionados.

Estas ideas son la base utilizada en el presente capítulo para proponer una solución al problema de representación computacional del flujo de datos en procesos y situaciones de aprendizaje colaborativo, de forma que se consiga, en la medida de lo posible, los principales objetivos de LD: automatización y reutilización del proceso, así como interoperabilidad con sistemas y herramientas

3. LeadFlow4LD: El método propuesto

Por tanto, a partir de la revisión del estado del arte, se define un espacio de soluciones del problema que puede ser usado para resolver los problemas identificados. Se identifican las siguientes aproximaciones:

1. *Adecuación de LD*. Consiste en adecuar la expresividad de IMS LD para describir el flujo de datos en una situación de aprendizaje colaborativo.
2. *Extensión de LD*. Consiste en proponer una extensión a IMS LD para describir el flujo de datos a nivel de herramientas.
3. *Mapeo de LD a flujo de trabajo*. Consiste en transformar diseños de aprendizaje especificados en IMS LD en una extensión de un lenguaje estándar de flujo de trabajo (llamado COWL, el cual es una extensión del estándar de flujo de trabajo XPDL).
4. *Combinación de LD y flujo de trabajo*. Consiste en proponer un nuevo EML (llamado lenguaje de guionado CSCL), cuyo meta-modelo estaría formado por la integración de los meta-modelos de IMS LD y XPDL.
5. *Composición de LD y flujo de trabajo*. Consiste en crear diseños de aprendizaje descritos por la composición del flujo de aprendizaje especificado en el EML estándar de LD, y el flujo de datos especificado en un lenguaje estándar de flujo de trabajo.

De este espacio de soluciones del problema, en este capítulo se justifica la selección de la mejor de estas aproximaciones para posteriormente proponer una solución concreta al problema.

3.2 Selección de una aproximación para la solución del problema

A partir de los principales objetivos de LD, y por tanto que la solución debe de cumplir, la *Tabla 3.1* resume el análisis realizado en el capítulo anterior de las distintas aproximaciones que podrían ser usadas para resolver el problema del flujo de datos en LD.

Aproximaciones	Automatización	Reutilización	Interoperabilidad
Adecuación de LD	-	-	+
Extensión de LD	-	-	-
Transformación de LD a un lenguaje de flujo de trabajo	+	+	-
Combinación de LD y un lenguaje de flujo de trabajo	+	+	-
Composición de LD y un lenguaje estándar de flujo de trabajo	+	+	+

Tabla 3.1: Evaluación de aproximaciones que podrían ser usadas para resolver el problema del flujo de datos en LD.

Como se puede ver en la *Tabla 3.1*, las aproximaciones por adecuación y extensión de IMS LD quedan descartadas, ya que no permiten especificar el flujo de datos a nivel de herramientas, sino solamente a nivel de actividades y por tanto no sería posible alcanzar el

objetivo de automatización del flujo de datos. Las aproximaciones por transformación y definición de un nuevo EML podrían ser usadas para alcanzar los objetivos de automatización y reutilización del proceso. Sin embargo, en ambos casos no se consigue interoperabilidad con estándares actuales, ya sea en la comunidad LD o en la comunidad de flujo de trabajo. Por un lado, está la transformación de LD hacia un lenguaje propietario de flujo de trabajo (COWL) (Vantroys & Peter, 2003), el cual no es interoperable con estándares de esta comunidad. Por otro lado, está el lenguaje de guionado CSCL (Miao, Hoeksema, Hoppe & Harrer, 2005), el cual no es interoperable con la comunidad LD.

Por tanto, parece ser más razonable usar una aproximación por composición de lo mejor de dos mundos. Describir el flujo de aprendizaje usando el lenguaje estándar de LD (IMS LD), y describir el flujo de datos a nivel de herramientas usando un lenguaje estándar de flujo de trabajo. La principal ventaja de esta aproximación es que se trata de una solución interoperable con estándares existentes, ampliamente conocidos y estables en ambas comunidades (LD y flujo de trabajo); y que por tanto se toma ventaja de herramientas existentes que son conformes o soportan los estándares de LD y flujo de trabajo (editores, motores, VLE, WfMS, etc.). Una dificultad, sin embargo, sería la necesidad de especificar (usando quizás un lenguaje propietario) la coordinación o relación entre procesos del flujo de aprendizaje y flujo de trabajo, ya que sería necesario entregar y devolver el control durante la ejecución de ambos procesos. Afortunadamente esta desventaja es superada por la ventaja de mantener interoperabilidad con los procesos que describen el escenario educativo en sí. Esto es, el flujo de aprendizaje y el flujo de datos, seguirían siendo especificados, instanciados y ejecutados por lenguajes, herramientas y motores que son conformes a un estándar, ya sea de la comunidad LD, o de la comunidad de flujo de trabajo. Por tanto, la aproximación por composición del flujo de aprendizaje y el flujo de datos se puede considerar una solución interoperable con estándares actuales.

Habiendo seleccionado del espacio de soluciones la aproximación por composición de LD y flujo de trabajo, a continuación se selecciona el lenguaje estándar de flujo de trabajo que sería el más adecuado para describir el flujo de datos a nivel de herramientas.

3.3 Selección del lenguaje de flujo de trabajo para describir el flujo de datos

La comparación de lenguajes de especificación de procesos de flujo de trabajo es intrínsecamente difícil, debido a la falta de un criterio estándar de expresividad (Abiteboul, Bourhis & Vianu, 2012). Sin embargo, es posible realizar la comparación siguiendo los criterios de evaluación que interesan: la capacidad de los lenguajes de flujo de trabajo para expresar el flujo de datos a nivel de herramientas. Entonces, primero se evalúa la expresividad de distintos lenguajes para expresar un conjunto de patrones de control de flujo de trabajo (van der Aalst, 2004) (Russell, ter Hofstede, van der Aalst & Mulyar, 2006). Después, se evalúa la expresividad de distintos lenguajes para expresar, entre otros conceptos, el manejo de datos (Mendling, Neuman & Nüttgens, 2005). Por último, se revisa

en (Muehlen, 2007) la clasificación que hace de los distintos lenguajes de flujo de trabajo, de acuerdo a los estándares típicamente usados para especificar, intercambiar y poner en marcha procesos de flujo de trabajo.

3.3.1 Lenguajes de Flujo de Trabajo

En los últimos años, un gran número de lenguajes de modelado de procesos de flujo de trabajo, notaciones gráficas, motores de ejecución, etc., han emergido en el contexto de gestión de sistemas de información y arquitecturas orientadas a servicios (Rossi & Turrini, 2006). A continuación, se enuncian los principales lenguajes de flujo de trabajo:

- **BPEL** (*The Business Process Execution Language*, Lenguaje de Ejecución de Procesos de Negocios) (Andrews, Curbeta, Dholakia, Goland, Klein & Leymann, 2003). Es un lenguaje de modelado de procesos basado en XML diseñado con el fin de definir una notación para especificar un comportamiento de procesos de negocio basado en la coordinación de Servicios Web. BPEL es un lenguaje ejecutable usado para modelar tareas como llamadas a Servicios Web, cuyas entradas y salidas son especificadas por mensajes. Nació originalmente como una iniciativa de un consorcio formado por principales firmas de software (IBM, Microsoft, BEA, etc.), y se ha movido hacia OASIS (*Organization for the Advancement of Structured Information Standards*, Organización para el Adelanto de Estándares de Información Estructurada), la cual es un consorcio global que dirige el desarrollo y adopción de estándares de negocios y Servicios Web.
- **BPML** (*The Business Process Modeling Language*, Lenguaje de Modelado de Procesos de Negocios) (Mendling & Muller, 2003). Es un lenguaje de definición de procesos de negocio basado en XML y que usa el mismo paradigma que los lenguajes de programación estructurada. BPML permite especificar múltiples procesos en un documento XML, así como la comunicación relacionada entre estos procesos (Mendling, Neuman, & Nüttgens, 2005). Este lenguaje fue propuesto inicialmente por BPMI (*The Business Process Management Initiative*, Iniciativa para la Gestión de Procesos de Negocios). Sin embargo, a partir de su unión con OMG (*Object Management Group*, Grupo para la Gestión de Objetos), se dejó de dar soporte al lenguaje, en favor de BPEL y BPMN.
- **BPMN** (*Business Process Modeling Notation*, Notación para el Modelado de Procesos de Negocio) (White, 2004). Es un lenguaje que busca integrar diferentes notaciones gráficas de procesos de negocios. El objetivo de BPMN es cerrar la brecha entre el diseño y la implementación del proceso. Esto lo consigue proporcionando una notación que sea fácilmente entendible por todos los participantes del negocio, desde analistas de negocios que crean los borradores iniciales del proceso, hasta los técnicos desarrolladores responsables de implementar la tecnología que pondrán en marcha estos procesos y finalmente, la gente del negocio que los gestionará y monitoreará (Mendling, Neuman & Nüttgens, 2005).

Desarrollado inicialmente por BPMI, BPMN ha sido soportado por la OMG desde que ambas organizaciones se unieron en 2005.

- **ebXML** (*The Business Process Specification Schema*, Esquema de Especificación de Procesos de Negocio) (Clark, Casanave, Kanaskie, Harvey, Clark & Smith, 2001). Es un lenguaje basado en XML usado para especificar la coreografía de Servicios Web. El lenguaje soporta la especificación y coreografía de transacciones de negocios que se pueden implementar usando uno o más patrones estándares disponibles. El lenguaje ebXML es soportado por OASIS.
- **PNML** (*Petri Net Markup Language*, Lenguaje de Etiquetas basado en Redes de Petri) (Billington, Christensen, van Hee, Kindler, Kummer & Petrucci, 2003). Es un lenguaje que soporta la sintaxis básica de los elementos de Redes de Petri que puede ser extendido para representar tipos arbitrarios de Redes de Petri. PNML es una propuesta académica para un formato de intercambio basado en XML.
- **UML2-AD** (*Unified Modeling Language Activity Diagram*, Diagrama de Actividad UML2) (OMG, 2004). Es un lenguaje gráfico usado para modelar conceptos que incluye actividades, roles, control de flujo, manejos de datos, etc. El diagrama de actividad UML2 fue propuesto en el campo de la ingeniería de software orientada a objetos y es un estándar gestionado por la OMG.
- **WS-CDL** (*Web Service Coreography Description Language*, Lenguaje de Descripción de Coreografías de Servicios Web) (Kavantzias, Burdett & Ritzinger, 2004). Es un lenguaje basado en XML que describe colaboraciones P2P (*Peer-To-Peer*) definiendo, desde un punto de vista global, el intercambio ordenado de mensajes para conseguir un objetivo de negocio común. WSDL es soportado y gestionado por el W3C (*World Wide Web Consortium*).
- **XPDL** (*XML Process Definition Language*, Lenguaje definición de Procesos basados en XML) (XPDL, 2008). Es un lenguaje definición de procesos de negocio basado en XML usado como estándar de intercambio de modelos de procesos entre componentes, productos y herramientas. El lenguaje XPDL es soportado y gestionado por la WfMC.

A continuación se elige el lenguaje que será usado para describir el flujo de datos a nivel de herramientas en estructuras y situaciones de aprendizaje colaborativo.

3.3.2 Evaluación de lenguajes de flujo de trabajo

Van der Aalst (2004) realiza una evaluación de distintos lenguajes de flujo de trabajo sobre el conjunto de patrones de control de flujo definidos en (van der Aalst, Hofstede, Kiepuszewski & Barros, 2003). Estos patrones de control de flujo pertenecen al campo de flujo de trabajo y se describen en la *Tabla 3.2*. Aunque Russell et al. (2006) lleva a cabo una revisión de esta lista de patrones original y la extiende a una lista con más patrones, se trata de patrones específicos del campo de flujo de trabajo, tales como ciclo estructurado (*structured loop*), recursión (*recursion*), etc. Para reducir complejidad, y puesto que esta lista de patrones avanzados del campo de flujo de trabajo difícilmente podrían ser utilizados en el campo LD,

3. LeadFlow4LD: El método propuesto

se decidió tratar con la lista de patrones original. A continuación se discute si la expresividad de tales patrones es adecuada para describir la secuenciación de herramientas en un contexto educativo, es decir, en escenarios de aprendizaje colaborativo.

De acuerdo a la *Tabla 3.2*, los patrones de secuencia y división paralela podrían ser usados para describir la ejecución secuencial y paralela de herramientas, respectivamente. Un caso particular al patrón de división paralela es el patrón de elección múltiple, en el que se requiere de una condición para la ejecución paralela de herramientas, por ejemplo, la condición de que un usuario termine una herramienta podría ser usada para describir el lanzamiento de dos nuevas herramientas. El patrón de sincronización podría ser usado para describir la terminación del flujo de datos (compuesto por una secuencia de instancias de herramientas), por ejemplo, una vez que todas las instancias de herramientas finalizaron (de otra forma no se devuelve el control al flujo de aprendizaje). Por último, el patrón de elección exclusiva podría ser utilizado para describir la oportunidad de que el estudiante elija entre una de varias posibles herramientas (en función de la preferencia del estudiante).

Patrón de Flujo de Trabajo a Evaluar	Descripción	Ejemplo de Uso en LD
Secuencia (<i>Sequence</i>)	Una actividad en un proceso de flujo de trabajo se habilita después de completar otra actividad en el mismo proceso.	Invocación secuencial de un editor de textos para generar un informe y de una herramienta de cuestionarios para indicar dudas sobre el proceso de escritura.
División paralela (<i>Parallel Split</i>)	Un punto en el proceso, donde un hilo individual de control se divide en múltiples hilos de control que pueden ser ejecutados en paralelo, permitiendo a las actividades ser ejecutadas simultáneamente.	Se pide a los miembros de un mismo grupo que lean individualmente un texto y rellenen un cuestionario sobre el mismo cuyas respuestas serán posteriormente compartidas por todos los miembros del grupo.
Sincronización (<i>Synchronization</i>)	Un punto en el proceso donde múltiples ramas paralelas convergen en un simple hilo de control, por tanto sincronizando múltiples hilos.	Hasta que todos los miembros de grupos hayan revisado las respuestas de todos sus pares, se puede continuar con la siguiente actividad de aprendizaje.
Elección exclusiva (<i>Exclusive choice</i>)	Un punto en el proceso, donde, basado en una decisión o control de datos de flujo de trabajo, se elige entre una de varias ramas posibles.	Cuando un participante tiene la opción de elegir entre un informe de uno de sus pares.
Elección múltiple (<i>Multi-Choice</i>)	Un punto en el proceso, donde, basado en una decisión o control de datos de flujo de trabajo, se escoge algunas ramas y se ejecutan como hilos paralelos.	Cuando un participante tiene la opción de elegir entre dos o más informes al mismo tiempo.

Tabla 3.2: Patrones de control de flujo a evaluar en los lenguajes de flujo de trabajo seleccionados. Fuente: (van der Aalst, 2004).

Como se puede ver en la *Tabla 3.2*, los ejemplos de uso en LD para los patrones de control flujo cubren, un conjunto significativo de posibles escenarios educativos. Sin embargo, tomando en consideración que el campo de flujo de trabajo es más general que el campo educativo, y que el control de flujo se modela como un grafo dirigido (por tanto admite muchas posibilidades de modelado), entonces podemos concluir que la expresividad

3. LeadFlow4LD: El método propuesto

de estos patrones de control de flujo puede ser suficiente para modelar el flujo de datos a nivel de herramientas en un amplio rango de escenarios de aprendizaje colaborativo.

van der Aalst, realiza la comparación de distintos lenguajes de flujo de trabajo sobre este conjunto de patrones de control de flujo, de acuerdo a los siguientes criterios de evaluación. Si un lenguaje de flujo de trabajo soporta directamente el patrón seleccionado, éste es evaluado con +, de lo contrario se evalúa con -. Los resultados se resumen en la *Tabla 3.3*.

Patrón de Flujo de Trabajo a Evaluar	XPDL	UML2-AD	BPEL	PML	WSCI
Secuencia	+	+	+	+	+
División paralela	+	+	+	+	+
Sincronización	+	+	+	+	+
Elección exclusiva	+	+	+	+	+
Elección múltiple	+	-	+	-	-

Tabla 3.3: Evaluación de los lenguajes de flujo de trabajo seleccionados con respecto a los patrones de control de flujo descritos en la Tabla 3.2. Fuente: (van der Aalst, 2004).

Desafortunadamente, esta evaluación solo considera patrones de control de flujo que serían usados para describir la secuenciación de herramientas, pero no considera patrones de datos (Russell, ter Hofstede, Edmon & van der Aalst, 2005), esto es, la relación entre datos y herramientas que tiene lugar durante el manejo o gestión del flujo de datos, que junto con los patrones de control de flujo serían usados para describir el flujo de datos a nivel de herramientas. Por esta razón, se revisó la comparación de lenguajes de flujo de trabajo realizada por Mendling et al. (2005), en donde sí considera la expresividad de los lenguajes de flujo de trabajo para describir tanto el control de flujo como el manejo de datos. La evaluación de Mendling et al. tienen el objetivo de comparar la completitud de distintos lenguajes de flujo de trabajo con respecto a un marco de conceptos extraídos de los meta-modelos de estos lenguajes. Un resumen de los conceptos que interesan para describir el flujo de datos a nivel de herramientas en escenarios de aprendizaje colaborativo, se describe en la *Tabla 3.4*.

A continuación se analiza si la expresividad de estos conceptos es necesaria para describir el flujo de datos en escenarios de aprendizaje colaborativo. En este caso, la localización de servicios sería usada para describir la localización de herramientas. El control de flujo, al igual que en la evaluación anterior, sería usado para describir el flujo de herramientas (la secuenciación de herramientas). El manejo de datos que no fue considerado en la evaluación anterior, sería usado para describir la relación entre datos y herramientas, esto es, para describir que los datos de salida de herramientas sean usados después como entradas de otras herramientas (ver ejemplo de uso en LD en la *Tabla 3.4*). Los roles, serían usados para describir el papel que desempeñan los participantes durante el manejo de herramientas. Por último los eventos podrían ser usados para describir cambios de estado en el flujo de datos, por ejemplo, iniciar el flujo de datos, terminarlo, etc. Por tanto, como en el caso anterior, se concluye que la expresividad de estos conceptos puede ser usados para describir el flujo de datos a nivel de herramientas en escenarios de aprendizaje colaborativo.

3. LeadFlow4LD: El método propuesto

Nuevamente, tomando en cuenta los mismos argumentos anteriores (la generalización del campo de flujo de trabajo vs el campo LD, y la representación del control de flujo como un grafo dirigido), podemos concluir también que la expresividad de este marco de conceptos de flujo de trabajo puede ser suficiente para modelar el flujo de datos a nivel de herramientas en un amplio rango de escenarios de aprendizaje colaborativo.

Concepto a Evaluar	Descripción	Ejemplos de Uso en LD
Localización de servicios	Dónde o cómo puede ser localizado un servicio para realizar una tarea	Para describir la localización (URLs) de editores de texto, de mapas conceptuales, visualizadores de documentos PDF, etc.
Control de flujo	Define las relaciones entre diferentes tareas	Para describir que después de invocar un editor de texto donde se escribe un informe, se invoca otro editor para revisarlo.
Manejo de datos	VARIABLES usadas en una instancia de un proceso	Para describir que el informe creado por uno de los grupos, usando una herramienta de edición, se usa después como entra por otro editor, para su revisión por otro de los grupos.
Roles	Abstracción de los participantes en un proceso. Los roles son asignados a tareas, y los participantes a roles.	Para distinguir entre el rol que desempeñan los estudiantes cuando crean un informe, del que desempeñan cuando revisan los informes creados por sus pares.
Eventos	Representan cambios del mundo real	Para describir condiciones de inicio o de terminación del flujo de datos de una actividad de aprendizaje.

Tabla 3.4: Conceptos a evaluar en los lenguajes de flujo de trabajo seleccionados. Fuente: (Mendling, Neuman & Nüttgens, 2005).

Los resultados de la evaluación realizada por Mendling et al. se resume en la *Tabla 3.5*, siendo los criterios de evaluación los siguientes: si un lenguaje de flujo de trabajo soporta directamente el concepto a evaluar, este es evaluado con +, de lo contrario se evalúa con -.

Concepto a Evaluar	BPEL	BPML	BPMN	ebXML	PNML	UML2-AD	WS-CDL	WSCl	XPDL
1. Localización de Tareas	+	+	+	-	-	-	+	+	+
2. Control del Flujo	+	+	+	+	+	+	+	+	+
3. Manejo de Datos	+	+	+	-	-	+	+	-	+
4. Roles	+	+	+	+	-	+	+	+	+
5. Eventos	+	+	+	-	-	-	-	-	+

Tabla 3.5: Evaluación de los lenguajes de flujo de trabajo seleccionados con respecto a los conceptos descritos en la Tabla 3.3. Fuente: (Mendling, Neuman & Nüttgens, 2005).

Por tanto, si combinamos los resultados de evaluación obtenidos en la *Tabla 3.3* con los de la *Tabla 3.5*, resulta claro que los cuatro lenguajes de flujo de trabajo que podrían ser

3. LeadFlow4LD: El método propuesto

utilizados para describir el flujo de datos en escenarios de aprendizaje colaborativo son: BPEL, BPML, BPMN, y XPDL.

Ahora bien, con el objetivo de mantener interoperabilidad con estándares actuales de flujo de trabajo (se requiere que el flujo de datos sea puesto en marcha por un motor de flujo de trabajo estándar), nos interesa centrarnos en el trabajo realizado por las tres organizaciones de estándares: WfMC, OMG y BPMI. Entonces, desde la fusión de BPMI con OMG, se ha dejado de dar soporte a BPML en favor de BPMN y BPEL. El primero (BPMN), porque junto con XPDL proporcionan el estándar de facto para modelar procesos de negocio (Muehlen, 2007); el segundo (BPEL), porque es probablemente el lenguaje de ejecución de procesos de negocio más frecuentemente usado y ampliamente aceptado en la industria (Vasko & Dustdar, 2006). Además, de acuerdo a la clasificación de los lenguajes estándares de flujo de trabajo realizado por Muehlen (2007), como se muestra en la *Figura 3.1*, XPDL es considerado el lenguaje que proporciona el estándar de intercambio de procesos de negocios (XPDL, 2008).

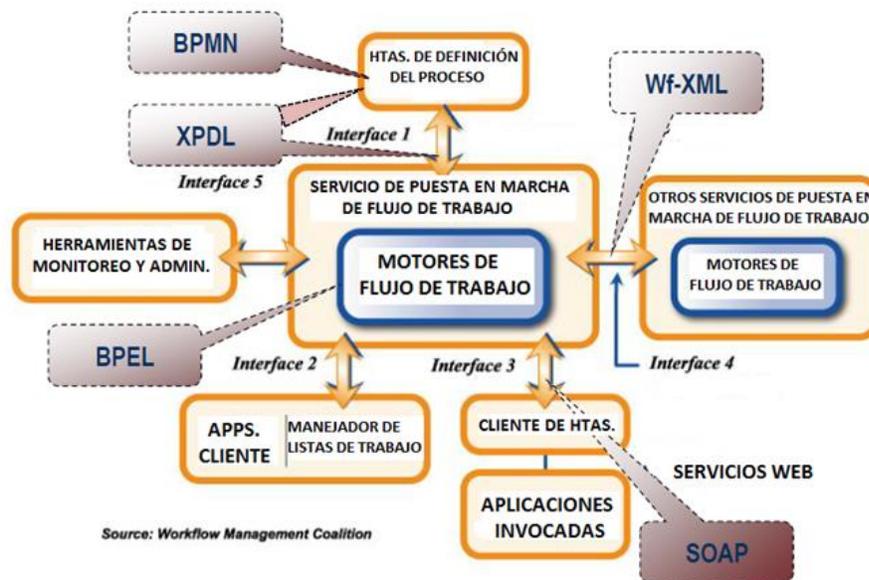


Figura 3.1: Estándares de flujo de trabajo. Fuente: (Muehlen, 2007).

Por tanto, tomando en cuenta esta última clasificación de estándares de flujo de trabajo, en esta Tesis Doctoral se ha seleccionado a XPDL como el lenguaje más adecuado para especificar y poner en marcha el flujo de datos. Primero, porque es considerado un estándar tanto para definir como para intercambiar procesos de flujo de trabajo. Y segundo, porque herramientas usadas tanto para la definición de procesos (usando el estándar BPMN), como para la ejecución de procesos (usando el estándar BPEL) serán, en general, interoperables con procesos definidos en XPDL. Esta última decisión de elegir entre estándares de flujo de trabajo, sin tomar en cuenta el contexto del problema (escenarios de aprendizaje) no debería ser relevante, ya que esto último ya se tomó en cuenta durante la evaluación de los patrones

3. LeadFlow4LD: El método propuesto

de control de flujo y los patrones de datos. A continuación se revisa el lenguaje de descripción de procesos de flujo de trabajo XPDL.

3.3.3 XPDL

XPDL (Norin, 2002) es un lenguaje definición de procesos basado en actividad (Myers & Berry, 1999) que define en un documento XML solamente la parte declarativa del proceso, es decir, lo que debe hacer, pero no cómo debe hacerlo. En este documento se declaran principalmente variables, actividades, participantes o roles abstractos, pero son las actividades el bloque fundamental que define la lógica del proceso, a través de la definición secuencial, paralela, selectiva o iterativa del orden de ejecución de las actividades.

El diagrama de la *Figura 3.2* describe el meta-modelo conceptual de XPDL, el cual se usa como marco conceptual para definir procesos de flujo de trabajo. Los principales elementos del meta-modelo son: actividad (*<Activity>*), participante (*<Participant>*), aplicación (*<Application>*), datos (*<DataField>*), y transiciones (*<Transition>*).

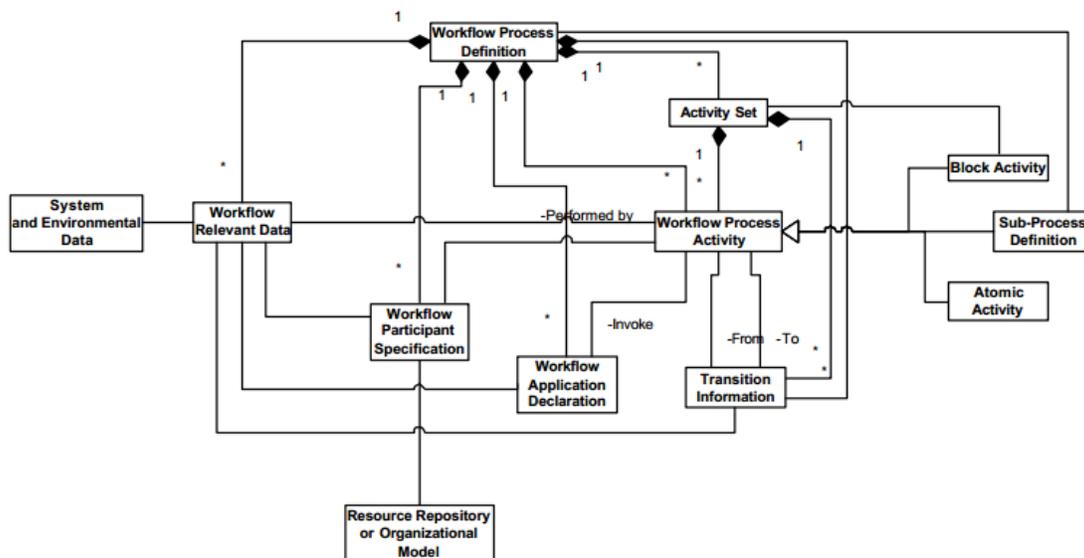


Figura 3.2: Meta-modelo de XPDL 1.0. Fuente: (Norin, 2002).

El resultado de describir un proceso con XPDL es un flujo de actividades de flujo de trabajo junto con los recursos que se necesita para realizarlas. Como se muestra en la *Figura 3.3*, la estructura de un proceso de flujo de trabajo descrito con XPDL se compone del elemento *<Package>* que sirve como contenedor de la definición del proceso; el elemento *<WorkflowProcess>* que define el comportamiento del proceso, es decir, qué actividades hacen qué y en qué orden; las actividades (*<Activity>*) que son el bloque fundamental de construcción de un proceso de flujo de trabajo y que representan las acciones a realizar, típicamente involucrando recursos, tales como roles o participantes (*<Participants>*), aplicaciones software (*<Applications>*) y datos (*<DataFields>*); las actividades pueden ser

3. LeadFlow4LD: El método propuesto

automáticas o manuales; las actividades manuales requieren de la intervención del participante y no pueden ser implementadas por el motor de flujo de trabajo, mientras que las actividades automáticas pueden implementar (<Implementation>) tareas (<Task>) automáticas (<TaskApplication>) que invocan aplicaciones software (<Application>). El flujo de datos a nivel de herramientas se puede definir relacionando los datos (<DataFields>) definidos en el proceso de flujo de trabajo, con los parámetros reales (<ActualParameter>) y formales (<FormalParameters>) de las aplicaciones (<Application>).

```

<Package>
  <Participants>
    <Participant>
      ...
    </Participant>
    ...
  </Participants>
  ...
  <WorkflowProcess>
    <WorkflowProcess>
      <Applications>
        ...
      </Applications>
      <Application>
        <FormalParameters>
          ...
        </FormalParameters>
      ...
    </DataFields>
    ...
  </DataFields>
  ...
  <Activities>
    <Activity>
      <Implementation>
        <Task>
          <TaskApplication>
            <ActualParameters>
              <ActualParameter>
                ...
              </ActualParameter>
            ...
          </ActualParameters>
        </TaskApplication>
      </Task>
    </Implementation>
  </Activity>
  ...
</Activities>
<Transitions>
  <Transition>
    ...
  -

```

Figura 3.3: Estructura de un proceso de flujo de trabajo definido en XPDL.

El control de flujo en XPDL se define a través de transiciones (<Transitions>) y las restricciones de transición (<TransitionRestrictions>). Las transiciones son en realidad arcos dirigidos entre las actividades de flujo de trabajo, mientras que las restricciones de transición

son expresiones lógicas que representan la condición de disparo de la transición (<AND>, <OR>, <XOR>, etc.).

3.4 Especificación de un diseño de aprendizaje con LeadFlow4LD

En esta sección se describe el método propuesto. Primero se presenta una vista general de LeadFlow4LD, después se describe cómo especificar la estructura de aprendizaje colaborativo y a continuación se describe cómo especificar una instancia de dicha estructura. En ambos casos, se presenta para un ejemplo ilustrativo simple pero significativo. Se trata de una situación de aprendizaje basada en el patrón del proceso de aprendizaje colaborativo de revisión entre pares descrito en el capítulo anterior (sección 2.7.1). La situación de aprendizaje seleccionada es una típica situación de aprendizaje colaborativo de revisión entre dos estudiantes, en la que primero cada estudiante, edita individualmente su propio artefacto durante la actividad de edición, y después cada estudiante revisa individualmente el artefacto creado por su par durante la actividad de revisión (ver *Figura 2.10a*). Por último, se describe el empaquetamiento de una UoLeadFlow4LD para su distribución.

3.4.1 Vista general de LeadFlow4LD

LeadFlow4LD es un método usado para describir diseños de aprendizaje en escenarios de aprendizaje colaborativo que combina el uso del lenguaje estándar de LD (IMS LD) y el lenguaje estándar de flujo de trabajo seleccionado (XPDL), con el objetivo de proporcionar representaciones interpretables por ordenador para estructuras y situaciones de aprendizaje colaborativo. De acuerdo con la *Figura 3.4*, LeadFlow4LD involucra una serie de pasos para la creación, por un lado, de tres documentos que describen la estructura de un proceso de aprendizaje colaborativo, y por otro lado, de dos documentos que describen una instancia de dicha estructura, es decir, que describen una situación de aprendizaje colaborativo.

Esta separación de un diseño de aprendizaje especificado en cinco diferentes documentos, tiene como objetivo fomentar la reutilización del diseño: los documentos del flujo de aprendizaje, flujo de datos y coordinación de flujos, que reflejan la estructura de aprendizaje colaborativo, pueden ser reutilizados en distintas situaciones de aprendizaje; ya que solo el documento de instanciación de grupos y/o el documento de población de grupos necesitarían ser modificados. Si además, estos 5 documentos son empaquetados en una sola UoLeadFlow4LD, es más fácil su intercambio y reutilización. A continuación, se describen los pasos que se deben seguir para alcanzar tales descripciones.

3. LeadFlow4LD: El método propuesto

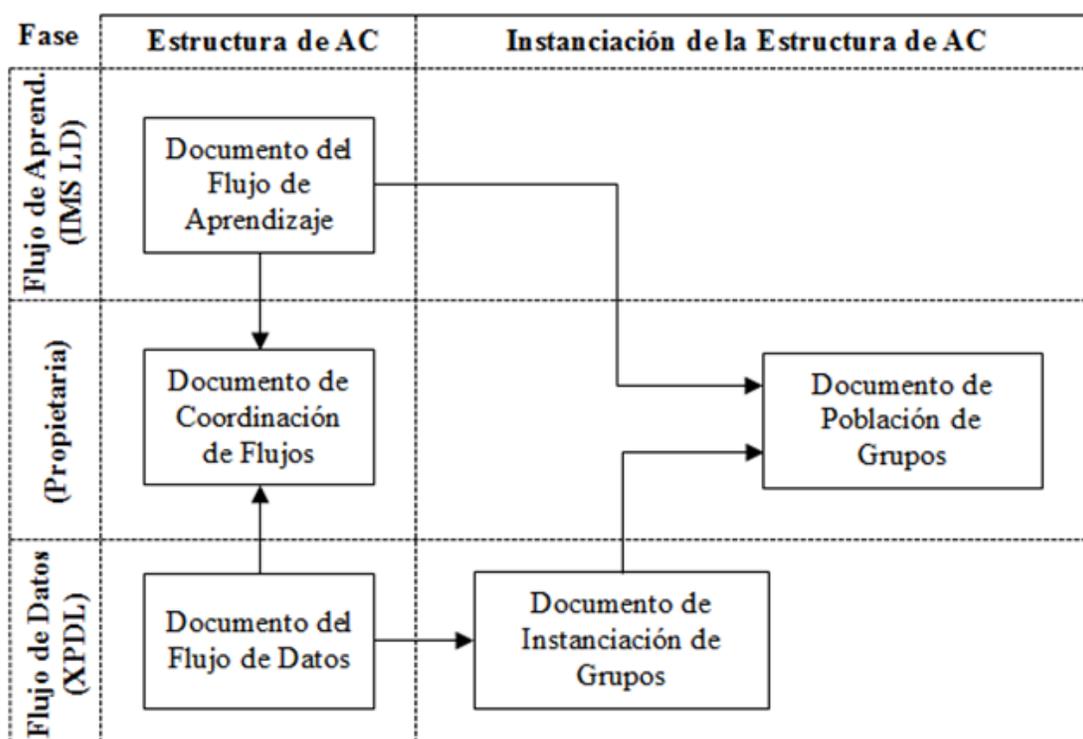


Figura 3.4: Vista general del método propuesto: LeadFlow4LD.

3.4.2 Especificación de la estructura de aprendizaje colaborativo

Como se muestra en la *Figura 3.4*, una estructura de aprendizaje colaborativo se especifica en 3 documentos: el documento que describe el flujo de actividades de aprendizaje o llamado simplemente el flujo de aprendizaje, el documento que describe el flujo de datos y el que describe la coordinación entre ambos, tareas que son llevadas a cabo por el diseñador instruccional, por su puesto, con el soporte de las correspondientes herramientas de autoría. A continuación, se describen cada uno de estos documentos.

Documento del Flujo de Aprendizaje

El documento del flujo de aprendizaje describe la secuencia de actividades de aprendizaje que van a ser realizadas por los distintos participantes (educadores, tutores, estudiantes, etc.), así como los roles que van a interpretar. Sin embargo, no se describen participantes concretos, sino solamente roles abstractos (editor, revisor, etc.). IMS LD es el lenguaje empleado para codificar el documento del flujo de aprendizaje debido a su amplia adopción en la comunidad LD, así como por su capacidad para expresar en un alto nivel de abstracción, propios del dominio educativo, todos los conceptos involucrados.

Para especificar un diseño de aprendizaje se puede usar la guía de mejores prácticas de IMS LD (*IMS LD Best Practices*, 2003) así como las herramientas de autoría correspondientes. Esta guía describe la secuencia de pasos a seguir para la especificación de un diseño de aprendizaje que culmine en una unidad de aprendizaje (UoL). Como se indica

3. LeadFlow4LD: El método propuesto

en la *Tabla 3.6*, el primer paso consiste en analizar el problema educativo como un caso de uso y convertirlo a un escenario con objetivos de aprendizaje. A continuación, usando el lenguaje natural, se describe una narrativa del proceso de aprendizaje describiendo el orden básico de las actividades. Después, esta narrativa se hace corresponder con un diagrama de actividad UML, que se usa como punto de partida para crear el documento XML que implementa la especificación IMS LD (llamado manifiesto IMS LD) y que representa propiamente el llamado diseño de aprendizaje. Una alternativa para especificar un diseño de aprendizaje con IMS LD es su interpretación como una obra de teatro, la cual fue descrita en la sección 2.5.5. Finalmente, para su distribución efectiva, el documento de diseño de aprendizaje se empaqueta junto con todos los recursos educativos en una UoL.

Fase del Proceso de un Diseño de Aprendizaje		Especificación en IMS LD
1	Definir objetivos de aprendizaje.	Especificar el elemento <i>learning-objectives</i> de IMS LD.
2	Desarrollar una narrativa del proceso de aprendizaje usando lenguaje natural.	No definida en IMS LD.
3	A partir de la narrativa del proceso, diseñar el flujo de aprendizaje usando un diagrama de actividad UML.	Especificar roles, actividades y el método usando los elementos <i>play</i> , <i>act</i> y <i>role-part</i> .
4	Asignar recursos: contenidos educativos, herramientas y servicios.	Especificar entornos y recursos de contenido.
5	Resultado del proceso de diseño (diseño de aprendizaje).	Manifiesto IMS LD.
6	Empaquetamiento del diseño de aprendizaje junto con todos los recursos para su distribución.	UoL

Tabla 3.6: Guía de mejores prácticas para especificar un diseño de aprendizaje con IMS LD.

Fuente: (IMS LD Best Practices, 2003).

De esta forma, el diseñador instruccional puede usar esta guía de mejores prácticas de IMS LD, además de un conjunto de herramientas de autoría, para producir UoLs que representen la estructura del flujo de aprendizaje de procesos de aprendizaje colaborativo. A manera de ejemplo, a continuación se usa esta de guía de mejores prácticas para especificar con IMS LD el documento del flujo de aprendizaje para el ejemplo ilustrativo. La *Tabla 3.8* describe las dos primeras fases de esta guía: los objetivos de aprendizaje y la narrativa del proceso usando lenguaje natural.

Fase del Proceso de un Diseño de Aprendizaje		Ejemplo Ilustrativo (Revisión entre Pares)
1	Objetivos de aprendizaje:	Buscar que los estudiantes aprendan del trabajo realizado por sus pares, que aprendan de la retroalimentación recibida y dada por/a sus pares, así como de su propia reflexión individual.
2	Narrativa del proceso:	Durante la actividad de edición, un estudiante o grupo de estudiantes editan primero un artefacto, utilizando para ello una herramienta individual o colaborativa. Después, durante la actividad de revisión, otro estudiante o grupo de estudiantes lo revisan, también con el soporte de una herramienta individual o colaborativa.

Tabla 3.7: Guía de mejores prácticas para especificar un diseño de aprendizaje con IMS LD.

Fuente: (IMS LD Best Practices, 2003).

3. LeadFlow4LD: El método propuesto

Entonces, a partir de la narrativa del proceso, se diseña el flujo de aprendizaje, el cual se describe usando un diagrama de actividad UML, como se muestra en la *Figura 3.5*. El diagrama describe la secuencia de actividades de aprendizaje: la actividad de edición (A1) asociada al rol editor, seguida de la actividad de revisión (A2), asociada al rol revisor.

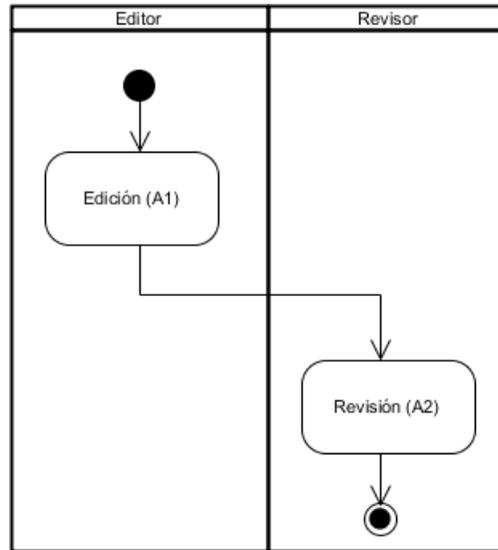


Figura 3.5: Diagrama de actividad UML que describe la narrativa del proceso de aprendizaje indicada en la Tabla 3.7.

La *Figura 3.6* y la *Figura 3.9* muestran detalles de las descripciones de actividades de edición y revisión para el ejemplo ilustrativo de revisión entre dos estudiantes. La *Tabla 3.8* resume la asignación de recursos educativos.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<html xmlns:ld="http://www.imslobal.org/xsd/imsld_v1p0"
xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Actividad de Edición</title>
</head>
<body bgcolor="white" text="black" link="#0000FF" vlink="#840084" alink="#0000FF">
<h2 class="title" style="clear: both">Edición de Mapa Conceptual</h2>
<h3>Actividad de Aprendizaje Individual</h3>
<P>
En esta actividad vas a leer el artículo localizado en la ventana de entorno del
sistema.
A continuación, crea un mapa conceptual sobre este tema.
Usa la ventana de entorno del sistema para acceder al servicio de edición.
Al finalizar el servicio de edición, tu mapa conceptual será almacenado
automáticamente en el repositorio del sistema.
</P>
<P>
Cuando termines pasa a la siguiente actividad.
</P>
```

Figura 3.6: Recurso de contenido (edition-desc.html) usado para describir la actividad de aprendizaje de edición del ejemplo ilustrativo de revisión entre dos estudiantes.

3. LeadFlow4LD: El método propuesto

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<html xmlns:ld="http://www.imsglobal.org/xsd/imsld_v1p0"
xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Actividad de Revisión</title>
</head>
<body bgcolor="white" text="black" link="#0000FF" vlink="#840084" alink="#0000FF">
<h2 class="title" style="clear: both">Revisión de Mapa Conceptual</h2>
<h3>Actividad de Aprendizaje Colaborativo</h3>
<P>
En esta actividad vas a revisar el mapa conceptual que ha realizado tu compañero en
la actividad anterior.
Usa la ventana de recursos del sistema para acceder al servicio de revisión.
</P>
</body>
</html>
```

Figura 3.7: Recurso de contenido (edition-desc.html) usado para describir la actividad de aprendizaje de edición del ejemplo ilustrativo de revisión entre dos estudiantes.

Recurso	Tipo	Asignación
edition-desc.html	Descripción de la actividad de edición.	Actividad de Edición
revisión-desc.html	Descripción de la actividad de revisión.	Actividad de Revisión
Cloud.pdf	Contenido educativo.	Actividad de Edición

Tabla 3.8: Asignación de recursos educativos a las actividades de aprendizaje del ejemplo ilustrativo de revisión entre dos estudiantes.

De acuerdo a la guía de mejores prácticas, a continuación sigue la especificación del diseño de aprendizaje usando IMS LD. Como es lógico, no se puede esperar que los diseñadores instruccionales o los educadores escriban directamente el XML. Para eso hay herramientas de autoría de muchos tipos: de más bajo o alto nivel de abstracción, más gráficas o menos gráficas, etc. (Prieto, Dimitriadis, Craft, Derntl, Émin, Katsamani & Villasclaras, 2013).

La *Figura 3.8* muestra parte del documento del flujo de aprendizaje que describe sus componentes estáticos (roles, actividades, entornos), mientras que su comportamiento dinámico se describe en la etiqueta *imsld:method* en la *Figura 3.9*, en la que se describe una obra de teatro compuesta de actos y guiones.

Documento del Flujo de Datos

El documento del flujo de datos describe la secuencia de invocaciones de herramientas que tendrá lugar durante el proceso de aprendizaje, los artefactos o datos generados por las herramientas, así como los datos de entrada y salida de las herramientas. Adicionalmente, describe también los roles de los participantes en las invocaciones de herramientas. XPDL es el lenguaje empleado para codificar el documento del flujo de datos debido a que es un estándar ampliamente aceptado por la comunidad de flujo de trabajo y por su capacidad de expresar todos los conceptos involucrados.

3. LeadFlow4LD: El método propuesto

```

<imsld:components>
<imsld:roles>
<imsld:learner identifier="role-Editor">
<imsld:title>Editor</imsld:title>
</imsld:learner>
<imsld:learner identifier="role-Revisor">
<imsld:title>Revisor</imsld:title>
</imsld:learner>
</imsld:roles>
<imsld:activities>
<imsld:learning-activity identifier="A1" isvisible="true">
<imsld:title>Edición</imsld:title>
<imsld:environment-ref ref="env-1" />
<imsld:activity-description>
<imsld:item identifier="item-1" identifierref="res-1" isvisible="true" />
</imsld:activity-description>
<imsld:complete-activity>
<imsld:user-choice />
</imsld:complete-activity>
</imsld:learning-activity>
<imsld:learning-activity identifier="A2" isvisible="true">
<imsld:title>Revisión</imsld:title>
<imsld:activity-description>
<imsld:item identifier="item-2" identifierref="res-2" isvisible="true" />
</imsld:activity-description>
<imsld:complete-activity>
<imsld:user-choice />
</imsld:complete-activity>
</imsld:learning-activity>
</imsld:activities>
<imsld:environments>
<imsld:environment identifier="env-1">
<imsld:title>Recursos</imsld:title>
<imsld:learning-object identifier="lo-1" isvisible="true">
<imsld:title>cloud-pdf</imsld:title>
<imsld:item identifier="item-3" identifierref="res-3" isvisible="true" />
</imsld:learning-object>
</imsld:environment>
</imsld:environments>
</imsld:components>

```

Figura 3.8: Parte del documento del flujo de aprendizaje que especifica sus componentes estáticos (roles, actividades y entornos), correspondiente a la técnica de aprendizaje colaborativo de revisión entre pares.

Existen en la literatura de flujo de trabajo algunas recetas de buenas prácticas para la creación de procesos de negocio (Miers, 2006) (Miers, 2006b). Sin embargo, estas recetas no son del dominio educativo, si no específicas de flujo de trabajo, y están centradas además en el control de flujo, y no en el flujo de datos, que es lo que nos interesa especificar en un contexto educativo. Por tanto, basándonos en las buenas prácticas para especificar con IMS LD el flujo de aprendizaje, en la *Tabla 3.9* indicamos cómo se debería especificar en XPDL el documento del flujo de datos en un contexto educativo.

3. LeadFlow4LD: El método propuesto

```

<imsld:method>
<imsld:play identifier="play-1" isvisible="true">
<imsld:title>Play</imsld:title>
<imsld:act identifier="act-1">33
<imsld:title>Acto-1</imsld:title>
<imsld:role-part identifier="rp-1">
<imsld:title>Role Part</imsld:title>
<imsld:role-ref ref="role-Editor" />
<imsld:learning-activity-ref ref="A1" />
</imsld:role-part>
<imsld:complete-act>
<imsld:when-role-part-completed ref="rp-1" />
</imsld:complete-act>
</imsld:act>
<imsld:act identifier="act-2">
<imsld:title>Acto-2</imsld:title>
<imsld:role-part identifier="rp-2">
<imsld:title>Role Part</imsld:title>
<imsld:role-ref ref="role-Revisor" />
<imsld:learning-activity-ref ref="A2" />
</imsld:role-part>
<imsld:complete-act>
<imsld:when-role-part-completed ref="rp-2" />
</imsld:complete-act>
</imsld:act>
<imsld:complete-play>
<imsld:when-last-act-completed />
</imsld:complete-play>
</imsld:play>
<imsld:complete-unit-of-learning>
<imsld:when-play-completed ref="play-1" />
</imsld:complete-unit-of-learning>
</imsld:method>

```

Figura 3.9: Parte del documento del flujo de aprendizaje que especifica su comportamiento dinámico (play, act, role-part), correspondiente a la estructura de aprendizaje colaborativo de revisión entre pares.

Fase del Diseño del Flujo de Datos		Especificación en XPDL
1	Definir objetivos del flujo de datos	No definida en XPDL.
2	Desarrollar una narrativa del proceso del flujo de datos a nivel de herramientas usando lenguaje natural.	No definida en XPDL.
3	A partir de la narrativa del proceso, diseñar el flujo de datos usando un diagrama de actividad UML.	Especificar roles, actividades, aplicaciones, datos y transiciones.
4	Resultado del proceso de diseño.	Definición del proceso de flujo de trabajo especificado en XPDL.
5	Empaquetamiento del flujo de datos en un fichero de proceso de negocio (BPR, <i>Buisness Process Archive</i>).	Fichero BPR.

Tabla 3.9: Guía propuesta para especificar en XPDL el flujo de datos en un contexto educativo.

De esta forma, el diseñador instruccional puede usar esta guía para producir BPRs que representen la estructura del flujo de datos de procesos de aprendizaje colaborativo. A manera de ejemplo, a continuación se usa esta de guía para especificar con XPDL el documento del flujo de datos para el ejemplo ilustrativo. La *Tabla 3.10* describe las dos primeras fases de esta guía.

3. LeadFlow4LD: El método propuesto

Fase para Definir el Flujo de Datos		Ejemplo Ilustrativo (Revisión entre Pares)
1	Objetivos del flujo de datos:	Utilización de herramientas de edición y revisión para el soporte de las actividades de aprendizaje y la gestión de artefactos de entrada y salida de las herramientas.
2	Narrativa del proceso del flujo de datos:	Durante la actividad de edición, se lanza automáticamente la herramienta de edición, usado para editar individual o colaborativamente un artefacto. Después, durante la actividad de revisión, se lanza automáticamente la herramienta de revisión individual o colaborativa, usada para revisar el artefacto creado por su par en la actividad anterior, el cual debe ser cargado automáticamente por la herramienta.

Tabla 3.10: Guía propuesta para especificar con XPD L el flujo de datos para el ejemplo ilustrativo.

Entonces, a partir de la narrativa del proceso, el diseñador instruccional diseña el flujo de datos, el cual se describe usando un diagrama de actividad UML, como se muestra en la *Figura 3.10*. El diagrama describe la secuencia de invocaciones de herramientas abstractas, así como la relación entre datos y herramientas. Por ejemplo, para el caso ilustrativo, la secuencia de herramientas se describe por la invocación de la herramienta de edición (H1) seguida de la invocación de la herramienta de revisión (H2), mientras que la relación entre datos y herramientas se describe por la condición de que el artefacto de salida (D1) de la herramienta de edición se usa después como entrada de la herramienta de revisión. Ambos, la secuenciación de herramientas, y la relación entre datos y herramientas definen en conjunto el flujo de datos.

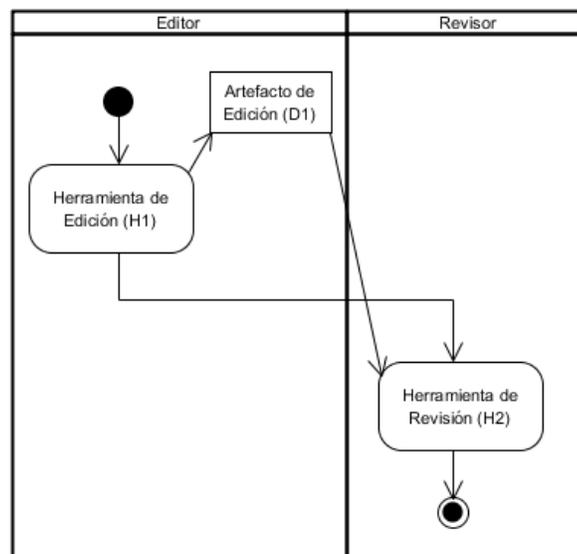


Figura 3.10: Diagrama de actividad UML que describe la narrativa del flujo de datos indicada en la Tabla 3.10.

El siguiente paso es la especificación del flujo de datos usando XPD L. El flujo de datos se especifica en XPD L separando los componentes estáticos de su comportamiento dinámico, de forma similar a como se especifica el flujo de aprendizaje en IMS LD, excepto que XPD L

3. LeadFlow4LD: El método propuesto

no usa la metáfora de la obra de teatro. Por un lado (en la parte estática), se definen roles, datos, herramientas, e invocaciones de herramientas. Por otro lado (en la parte dinámica), se define la secuencia de invocaciones de herramientas y la relación entre datos y herramientas. Los roles se describen en XPDL como participantes *<Participants>*, los artefactos o datos como variables de flujo de trabajo *<DataFields>*, las herramientas como aplicaciones *<Applications>*, y las invocaciones de herramientas como actividades de flujo de trabajo *<Activities>* de tipo implementación *<Implementation>*. La secuencia de invocaciones de herramientas se describe definiendo transiciones *<Transitions>* entre actividades, mientras que la relación entre datos y herramientas se describe al relacionar parámetros formales (*<FormalParameters>*) de herramientas con los argumentos reales *<ActualParameters>* que se describen con datos que almacenan los artefactos creados por las herramientas.

La *Figura 3.11* muestra parte del documento del flujo de datos que describe los componentes estáticos para el ejemplo ilustrativo: variables de flujo de trabajo, roles y herramientas abstractas definidas en términos de sus parámetros formales.

```

<xpdl:DataFields>
<xpdl:DataField Id="D1" IsArray="FALSE" Name="D1">
<xpdl:DataType>
<xpdl:BasicType Type="STRING"/>
</xpdl:DataType>
</xpdl:DataField>
</xpdl:DataFields>
<xpdl:Participants>
<xpdl:Participant Id="Editor" Name="Editor">
<xpdl:ParticipantType Type="ROLE"/>
</xpdl:Participant>
<xpdl:Participant Id="Revisor" Name="Revisor">
<xpdl:ParticipantType Type="ROLE"/>
</xpdl:Participant>
</xpdl:Participants>
<xpdl:Applications>
<xpdl:Application Id="App-H1">
<xpdl:FormalParameters>
<xpdl:FormalParameter Id="param-D1" Mode="OUT">
<xpdl:DataType>
<xpdl:BasicType Type="STRING"/>
</xpdl:DataType>
</xpdl:FormalParameter>
</xpdl:FormalParameters>
</xpdl:Application>
<xpdl:Application Id="App-H2">
<xpdl:FormalParameters>
<xpdl:FormalParameter Id="param-D1" Mode="IN">
<xpdl:DataType>
<xpdl:BasicType Type="STRING"/>
</xpdl:DataType>
</xpdl:FormalParameter>
</xpdl:FormalParameters>
</xpdl:Application>
</xpdl:Applications>

```

Figura 3.11: Parte del documento del flujo de datos que especifica sus componentes estáticos (datos, roles y herramientas), correspondientes al ejemplo ilustrativo de revisión entre pares.

Por otro lado, en la *Figura 3.12* se muestra parte del mismo documento del flujo de datos en el que se especifican, para el caso del ejemplo ilustrativo, las invocaciones de

3. LeadFlow4LD: El método propuesto

herramientas, la secuenciación de invocaciones de herramientas y la relación entre datos y herramientas. Note, por ejemplo, que para especificar que el artefacto D1 es una salida de la herramienta H1 y una entrada de la herramienta H2, primero se especifican las herramientas H1 y H2 con parámetros de salida (OUT) y entrada (IN), respectivamente. Después, cuando se especifican las invocaciones de herramientas, se especifica también el valor de los argumentos con el que se llaman las herramientas (la variable de flujo de trabajo D1, en ambos casos).

```

<xpdl:Activities>
<xpdl:Activity Id="H1" Name="H1">
<xpdl:Implementation>
<xpdl:Tool Id="App-H1" Type="APPLICATION">
<xpdl:ActualParameters>
<xpdl:ActualParameter>D1</xpdl:ActualParameter>
</xpdl:ActualParameters>
</xpdl:Tool>
</xpdl:Implementation>
<xpdl:Performer>Editor</xpdl:Performer>
<xpdl:Documentation>Herramienta de Edición</xpdl:Documentation>
<xpdl:ExtendedAttributes>
<xpdl:ExtendedAttribute Name="JaWE_GRAPH_PARTICIPANT_ID" Value="Editor"/>
<xpdl:ExtendedAttribute Name="JaWE_GRAPH_OFFSET" Value="56,108"/>
</xpdl:ExtendedAttributes>
</xpdl:Activity>
<xpdl:Activity Id="H2" Name="H2">
<xpdl:Implementation>
<xpdl:Tool Id="App-H2" Type="APPLICATION">
<xpdl:ActualParameters>
<xpdl:ActualParameter>D1</xpdl:ActualParameter>
</xpdl:ActualParameters>
</xpdl:Tool>
</xpdl:Implementation>
<xpdl:Performer>Revisor</xpdl:Performer>
<xpdl:Documentation>Herramienta de Revisión</xpdl:Documentation>
<xpdl:ExtendedAttributes>
<xpdl:ExtendedAttribute Name="JaWE_GRAPH_PARTICIPANT_ID" Value="Revisor"/>
<xpdl:ExtendedAttribute Name="JaWE_GRAPH_OFFSET" Value="50,202"/>
</xpdl:ExtendedAttributes>
</xpdl:Activity>
</xpdl:Activities>

<xpdl:Transitions>
<xpdl:Transition From="H1" Id="newpkg1_wpl_tral" To="H2">
<xpdl:ExtendedAttributes>
<xpdl:ExtendedAttribute Name="JaWE_GRAPH_TRANSITION_STYLE"
Value="NO_ROUTING_BEZIER"/>
</xpdl:ExtendedAttributes>
</xpdl:Transition>
</xpdl:Transitions>

```

Figura 3.12: Parte del documento del flujo de datos que especifica las invocaciones de herramientas, las relaciones entre datos y herramientas y la secuenciación de herramientas, correspondiente al ejemplo ilustrativo de revisión entre pares.

La especificación del flujo de datos no la lleva a cabo el diseñador instruccional directamente en XML, sino que típicamente es soportada por una herramienta de autoría, tal como la herramienta *Together Workflow Editor* (XPDL Editor), cuya interfaz gráfica se muestra en la *Figura 3.12*. Desafortunadamente esta herramienta pertenece al dominio de flujo de trabajo y no al dominio educativo, y por tanto tiene un impacto en el diseñador

instruccional, lo cual es discutido en el siguiente capítulo de evaluación de esta Tesis Doctoral.

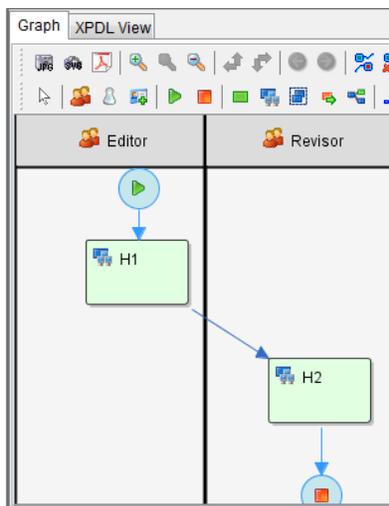


Figura 3.13: Herramienta Together Workflow Editor usada para especificar el flujo de datos con XPDL.

3.4.3 El documento de coordinación de flujos

Una vez que el flujo de aprendizaje y el flujo de datos han sido especificados en sus correspondientes documentos, se debe describir la información que establece la relación entre actividades de aprendizaje y las actividades de flujo de trabajo que representan invocaciones de herramientas, ya que ambos flujos son interdependientes.

Puesto que las herramientas son usadas para dar soporte a los participantes para conseguir los objetivos de aprendizaje, es razonable que LeadFlow4LD utilice un mecanismo de coordinación maestro-esclavo en el que el flujo de aprendizaje juega el rol maestro y el flujo de trabajo juega el rol esclavo, con el objetivo de decidir cuál de los dos flujos está guiando realmente las acciones de los estudiantes. En consecuencia, el flujo de aprendizaje debe identificar las actividades de aprendizaje en las que los participantes involucrados inician las invocaciones de herramientas. Del mismo modo, se deben identificar las actividades de flujo de trabajo en que el flujo de datos es iniciado y terminado (el momento de iniciar y regresar el control al flujo de aprendizaje, respectivamente).

Por lo tanto, una forma simple de codificar esta relación entre actividades de aprendizaje maestras y actividades de flujo de trabajo esclavas, es definiendo una lista de transiciones maestras y esclavas. En este contexto, una transición maestra define una conexión que va desde una actividad de aprendizaje hacia una actividad de flujo de trabajo que representa la invocación de la primera herramienta (de una secuencia de herramientas), mientras que una transición esclava define una conexión que va desde la última herramienta (de la secuencia de herramientas) hacia la actividad de aprendizaje a la que está asociada la secuencia de

3. LeadFlow4LD: El método propuesto

herramientas. Esta coordinación de flujos a nivel estructural es definida por el diseñador instruccional en un documento llamado el documento de coordinación de flujos, el cual es codificado en una especificación propietaria basada en el modelo conceptual mostrado en la *Figura 3.14*.

La desventaja de especificar la coordinación de flujos usando una especificación propietaria, se debe a que lo que se quiere modelar (la relación entre actividades IMS LD y actividades de flujo de trabajo) no es soportado ni por IMS LD ni por XPDL. Además, es el precio que hay que pagar para poder seguir manteniendo interoperabilidad con herramientas, motores y sistemas de ambas comunidades.

El meta-modelo de la *Figura 3.14* muestra los elementos conceptuales que integran cualquier documento que describe la coordinación de flujos, así como las relaciones que estos elementos guardan entre sí. Los elementos que componen un documento de coordinación de flujos son del tipo: manifiesto, información de la UoLeadFlow4LD, actividades (de aprendizaje), transiciones (maestras y esclavas), y herramientas. Un elemento del tipo transición maestra se relaciona con exactamente un elemento de tipo actividad de aprendizaje y uno o más elementos del tipo herramienta, mientras que un elemento del tipo transición esclava se relaciona con exactamente un elemento de tipo herramienta y exactamente con un elemento de tipo actividad de aprendizaje.

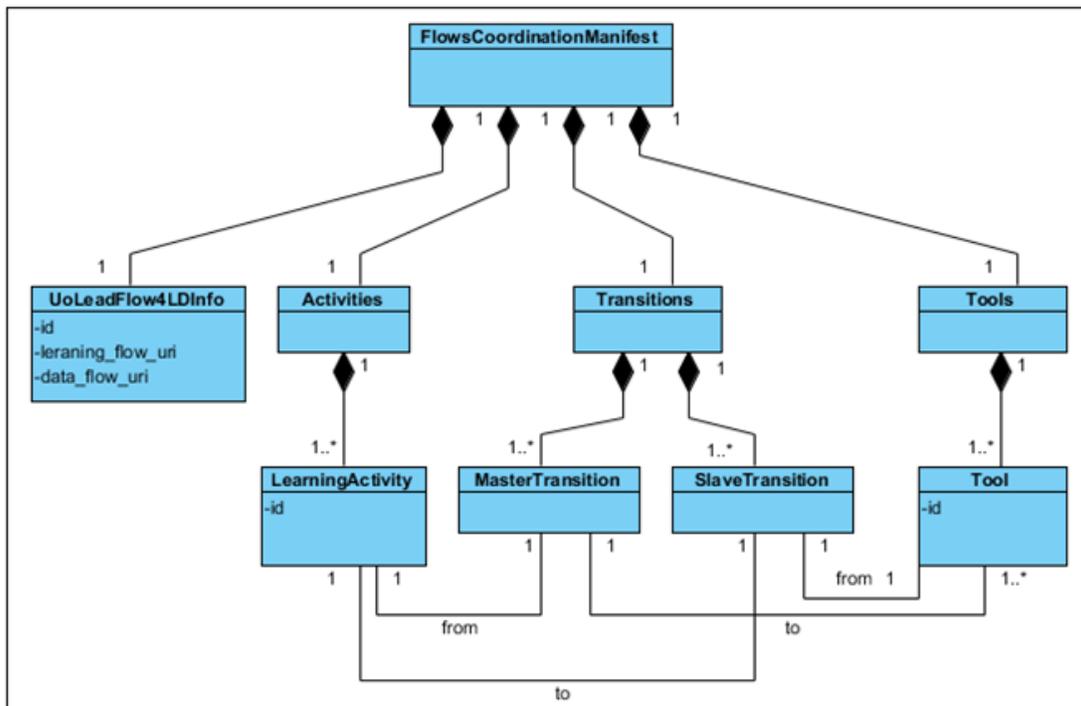


Figura 3.14: Modelo conceptual propuesto para especificar la coordinación de flujos.

3. LeadFlow4LD: El método propuesto

Los elementos y relaciones del meta-modelo que representa la coordinación de flujos (el flujo de aprendizaje y el flujo de datos) se recogen en el esquema XML que se puede consultar en el apéndice B.1. La *Figura 3.15* ilustra la relación entre las actividades de aprendizaje y el flujo de datos para el ejemplo ilustrativo. Esta tarea es realizada por el diseñador instruccional y debe entenderse como los momentos en que las herramientas dan el soporte necesario a las actividades de aprendizaje, así como los momentos en que dejan de hacerlo.

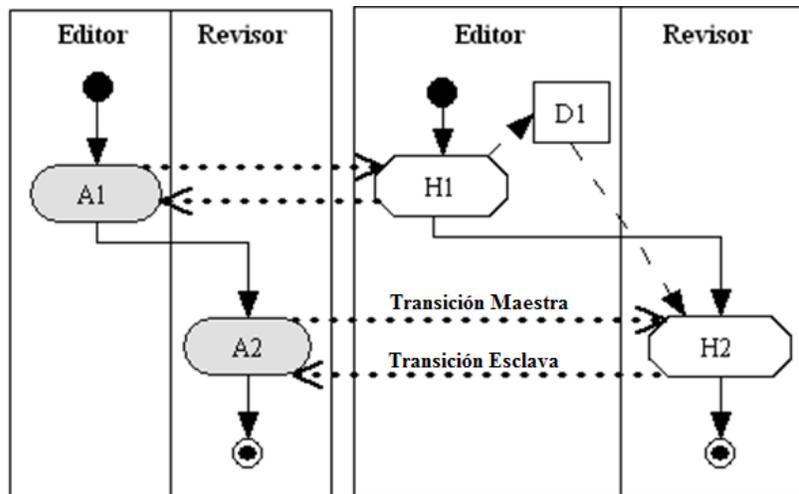


Figura 3.15: Diseño de la coordinación entre el flujo de aprendizaje y el flujo de datos correspondiente a la técnica de aprendizaje colaborativo de revisión entre pares.

Por ejemplo, la *Figura 3.16* describe la especificación de la *Figura 3.15* usando la especificación propuesta. Note que se describen dos transiciones maestras y dos esclavas, las cuales corresponden a las flechas punteadas en el diseño de la *Figura 3.15*.

3.4.4 Instanciación de la estructura de aprendizaje colaborativo

La simple asignación de participantes a la estructura del flujo de aprendizaje es suficiente para poner en marcha situaciones de aprendizaje, esto es, crear casos en terminología de flujo de trabajo. Sin embargo, esto no ocurre con la instanciación de la estructura del flujo de datos. En escenarios de aprendizaje colaborativo, la simple asignación de participantes a herramientas abstractas no es suficiente para poner en marcha situaciones de aprendizaje colaborativo concretas. En lugar de esto, se requiere definir primero cuántas instancias de datos y herramientas habrá, así como definir la relación entre éstas, y después, definir cómo se distribuyen los participantes en las instancias de herramientas. Por tanto, esto se puede conseguir en dos pasos: definiendo primero el número de grupos y las interacciones entre grupos, y en un segundo paso definiendo la población de los grupos. El número de grupos puede usarse para determinar cuántas instancias de datos y herramientas habrá, pero no dice cuántos participantes están en cada grupo. De esta forma, una interacción definida entre dos

3. LeadFlow4LD: El método propuesto

grupos no dice si es una interacción entre participantes individuales o entre grupos de participantes.

```

<activities>
<learning-activity id="A1" />
<learning-activity id="A2"/>
</activities>
<tools>
<tool id="H1" />
<tool id="H2" />
</tools>
<transitions>
<master-transition>
<from activity-ref="A1" />
<to tool-ref="H1" />
</master-transition>
<master-transition>
<from activity-ref="A2" />
<to tool-ref="H2" />
</master-transition>
<slave-transition>
<from tool-ref="H1" />
<to activity-ref="A1" />
</slave-transition>
<slave-transition>
<from tool-ref="H2" />
<to activity-ref="A2" />
</slave-transition>
</transitions>
</flows-coordination-manifest>

```

Figura 3.16: Parte del documento de coordinación de flujos que especifica la relación entre el flujo de aprendizaje y el flujo de datos, correspondiente al ejemplo ilustrativo de revisión entre pares.

Por ejemplo, la *Figura 3.17* muestra dos instancias posibles de la estructura del flujo de datos de la *Figura 3.10*. Por un lado, la *Figura 3.17a* describe dos instancias de datos y herramientas de edición y dos instancias de datos y herramientas de revisión, junto con la relación entre instancias de datos y herramientas mostrada. Esto se puede entender como dos grupos que participan en la actividad de edición y dos grupos en la actividad de revisión, con la interacción entre grupos ilustrada por la relación entre datos y herramientas. Por otro lado, la *Figura 3.17b* describe tres instancias de datos y herramientas de edición y tres instancias de datos y herramientas de revisión, junto con la relación entre instancias de datos y herramientas mostrada. Esto se puede entender como tres grupos que participan en la actividad de edición y tres grupos en la actividad de revisión, con la interacción entre grupos indicada.

Sin embargo, para definir una situación de aprendizaje colaborativo concreta, se debe definir además la distribución de los participantes en las actividades de aprendizaje y en las instancias de herramientas, lo cual se puede entender también como definir la población de los grupos. Por ejemplo, para el caso ilustrativo, la *Tabla 3.11* describe la asignación de los participantes tanto a las actividades del flujo de aprendizaje de la *Figura 3.5*, como a las instancias de herramientas de la instancia del flujo de datos de la *Figura 3.17a*. Note que cada participante *p1* y *p2* desempeñan el rol editor y revisor, mientras que cada participante usa su propia instancia de herramienta de edición y revisión. Similarmente, esto también se puede

3. LeadFlow4LD: El método propuesto

entender como se muestra en la *Tabla 3.12*, en donde que en cada actividad de aprendizaje se crean dos grupos, los grupos interaccionan como se indica en la *Figura 3.17a*, y en el que hay un participante por grupo.

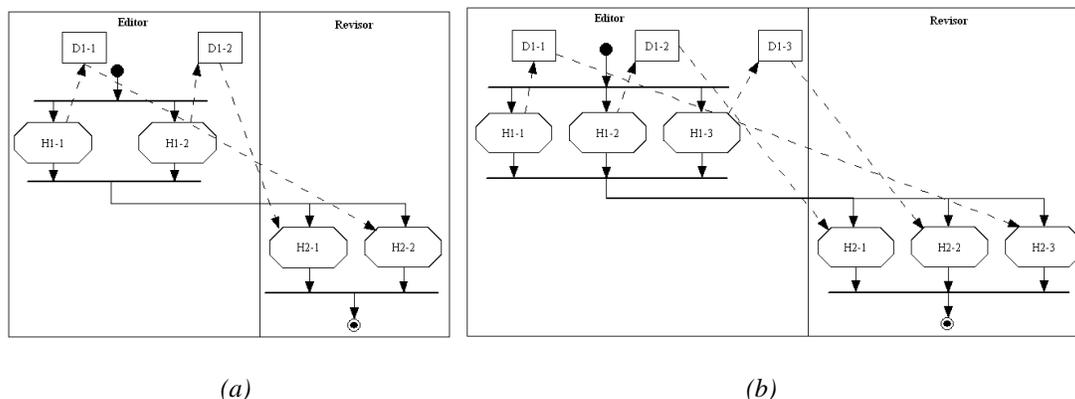


Figura 3.17: Dos instancias de la estructura del flujo de datos correspondiente al ejemplo ilustrativo de revisión entre pares (Figura 3.10): a) dos instancias de datos y herramientas b) tres instancias de datos y herramientas, así como la relación entre éstas.

Editor Revisor		Instancias de Herramientas del Documento de Instanciación de Grupos			
Editor	Revisor	H1-1	H1-2	H2-1	H2-2
p1,p2	p1,p2	p1	p2	p1	p2

Tabla 3.11: Asignación de participantes a las actividades de aprendizaje y a las instancias de herramientas, correspondiente al ejemplo ilustrativo de revisión entre pares.

Actividad de Aprendizaje	#Grupos	Grupo (id)	Participantes
Edición (A1)	2	1	p1
		2	p2
Revisión (A2)	2	1	p1
		2	p2

Tabla 3.12: Situación de aprendizaje del ejemplo ilustrativo de revisión entre pares definida por el número de grupos en cada actividad de aprendizaje (dos grupos), la interacción entre grupos (no mostrada en la Tabla) y la población de los grupos.

Como debe ser claro de este ejemplo, muchas otras situaciones de aprendizaje se pueden definir para la misma estructura del flujo de datos de la *Figura 3.10*, tan solo con definir el número de grupos de la actividad, cómo interaccionan los grupos, y cómo se pueblan los grupos. Por esta razón, LeadFlow4LD distingue entre la estructura de aprendizaje colaborativo y su instanciación. A diferencia de un típico diseño de aprendizaje especificado conforme a IMS LD, donde la instanciación no se especifica en absoluto, ya que la especificación de la instanciación no esté definida en diseño de aprendizaje (Caeiro-Rodríguez, 2008), ni siquiera en flujo de trabajo (Decker & Mendling, 2008). En LeadFlow4LD, la especificación de la instanciación de la estructura de aprendizaje

colaborativo se lleva a cabo en dos documentos separados: el documento de instanciación de grupos y el documento de población de grupos, los cuales se describen a continuación.

3.4.5 Documento de instanciación de grupos

El documento de instanciación de grupos es definido a partir del documento del flujo de datos, pero tomando en cuenta las interacciones entre instancias de datos y herramientas que describen la interacción colaborativa de los grupos participantes en la situación de aprendizaje. Es importante aclarar que aunque el documento es llamado documento de instanciación de grupos, en realidad no se especifican instancias de grupos, sino instancias de datos de datos y herramientas, así como las relaciones que guardan entre sí. Es decir, a partir de la estructura del flujo de datos, el educador o el diseñador instruccional deben definir tantas instancias de datos y herramientas como el número de grupos participantes, y después definir la relación entre éstas de acuerdo a la relación entre los grupos (por ejemplo, especificar qué grupo revisa el artefacto de qué grupo). Este documento es especificado en XPDL, con el soporte de herramientas de autoría, al igual que el documento del flujo de datos, del cual se deriva.

Por tanto, a partir de la estructura del flujo de datos de la *Figura 3.10* el diseñador instruccional o el educador pueden definir dos grupos que interaccionan como se indica en la *Figura 3.17a*. Este diseño es especificado en XPDL, con el soporte de herramientas de autoría, en un documento de instanciación de grupos, y que se puede consultar en el apéndice C.

3.4.6 Documento de población de grupos

La instanciación de la estructura de aprendizaje colaborativo no solo se describe a nivel estructural, como es el caso del documento de instanciación de grupos, el cual define una instancia de la estructura del flujo de datos, sino que se describe también a nivel situacional. Para esto es precisamente el documento de población de grupos, el cual es especificado por el educador con el objetivo de describir las particularidades de la situación de aprendizaje colaborativo, tales como la asignación de los participantes a las actividades de aprendizaje (vía la asignación de participantes a roles) y la asignación de los participantes a las instancias de herramientas (asignación de recursos a las actividades de flujo de trabajo).

La especificación del documento de población de grupos no sigue una especificación estándar, ya que, como se indicó anteriormente, la instanciación de procesos basados en el paradigma de flujo de trabajo es un problema abierto (Caeiro-Rodríguez, 2008) incluso en el campo de flujo de trabajo (Decker & Mendling, 2008). En su lugar, LeadFlow4LD propone el uso de una representación XML basada en un modelo conceptual y esquema propuesto por iCollage (ver *Figura 3.18*) (Hernández-Gonzalo, Villasclaras-Fernández, Hernández-Leo, Asencio-Pérez & Diminitriadis, 2008), la cual es usada para especificar la instanciación de grupos como instancias de roles, así como la asignación de participantes a grupos. Sin embargo, LeadFlow4LD ha extendido este meta-modelo para incluir la asignación de

3. LeadFlow4LD: El método propuesto

participantes a instancias de herramientas, según se muestra en el modelo conceptual de la *Figura 3.19* (iLeadFlow4LD).

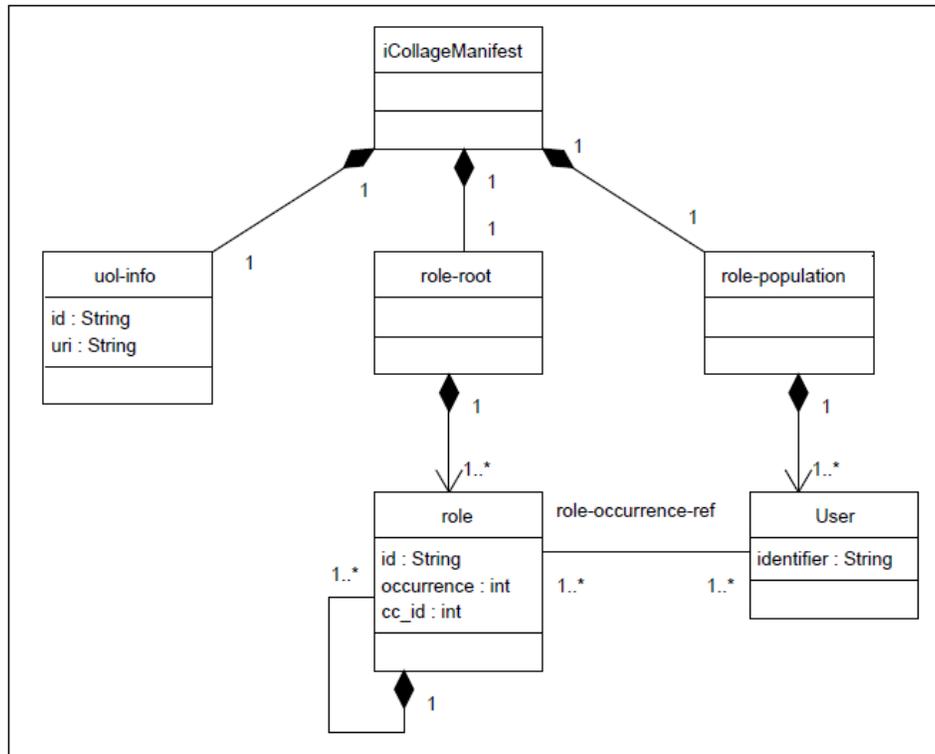


Figura 3.18: Modelo conceptual de iCollage. Fuente: (Hernández-Gonzalo, Villasclaras-Fernández, Hernández-Leo, Asencio-Pérez & Diminiadis, 2008).

Esta especificación propietaria del documento de población de grupos es una desventaja del método propuesto, sin embargo es una necesidad, ya que no existe a la fecha un estándar para especificar la instanciación de procesos basados en flujo de trabajo. En su lugar, cada implementación de flujo de trabajo toma decisiones de implementación propietarias para la asignación de recursos en tiempo de instanciación. Por tanto, esta desventaja es contrarrestada por la ventaja de LeadFlow4LD de mantener interoperabilidad con estándares actuales de LD y flujo de trabajo.

El meta-modelo propuesto de la *Figura 3.19* muestra los elementos conceptuales que integran cualquier documento que describe la población de grupos, así como las relaciones que guardan éstos elementos entre sí. Tomando como base los mismos elementos de iCollage (ver *Figura 3.18*): *uol-info* (llamado ahora *uUoLeadFlow4LDInfo*), *role-root* (llamado ahora *RoleRoot*), *role-population* (llamado ahora *GroupPopulation*), *role* (llamado ahora *Role*) y *User* (se mantuvo el nombre). Este meta-modelo se ha extendido para incluir capacidades de inicialización de datos (llamado *DataInstances*), por ejemplo para inicializar localizadores de artefactos, y de asignación de usuarios a instancias de herramientas (llamado *ToolInstances*). Los elementos y sus relaciones en el meta-modelo que representan una situación de

3. LeadFlow4LD: El método propuesto

aprendizaje basada en la asignación de usuarios a las actividades de aprendizaje y a las instancias de herramientas se recogen en el esquema XML que se incluye en el apéndice B.2.

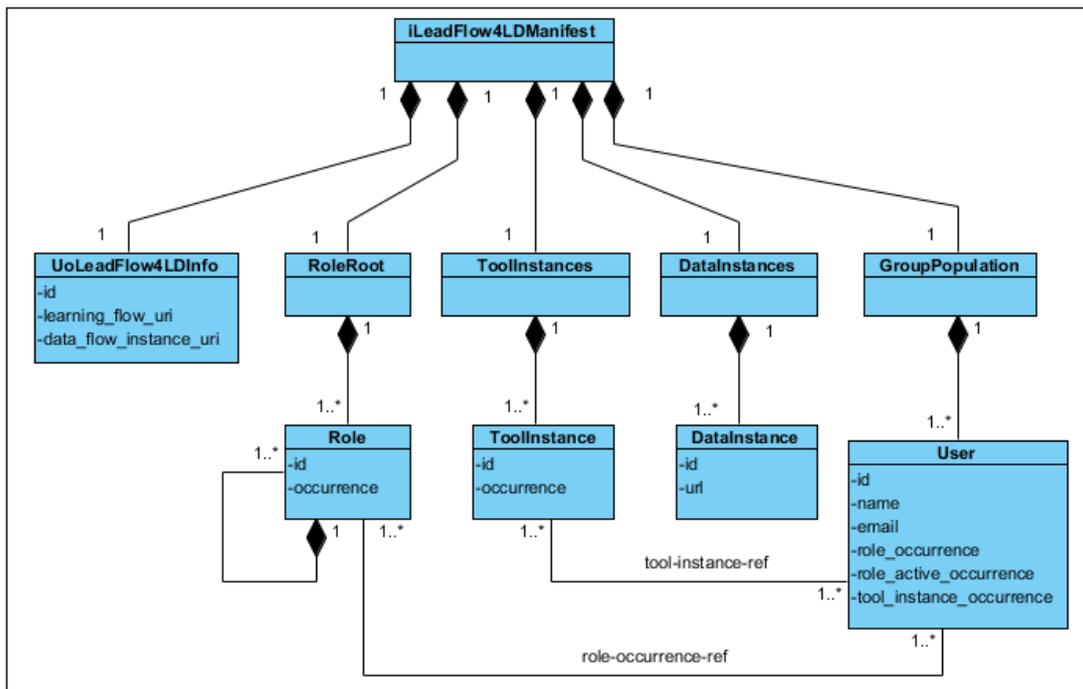


Figura 3.19: Modelo conceptual de iLeadFlow4LD como una extensión del de iCollage.

Considere la situación de aprendizaje del ejemplo ilustrativo en el que participan dos estudiantes y en el que cada estudiante revisa individualmente (con el soporte de una instancia de herramienta individual) el artefacto creado por su par en la actividad anterior. La *Tabla 3.11* describe conceptualmente la asignación de participantes a las actividades de aprendizaje y a las instancias de herramientas, mientras que una parte del documento de población de grupos especificado en el lenguaje propietario iCollage extendido se muestra en la *Figura 3.20*. El resto del documento de población de grupos es referido al apéndice C. Note en la *Figura 3.20*, en la etiqueta de población de grupos, que cada usuario participante $p1$ y $p2$ es asignado a un rol de aprendizaje, sí como a dos instancias de herramientas, una edición y otra de revisión.

3.4.7 Distribución de diseños de aprendizaje conformes a LeadFlow4LD: UoLeadFlow4LD

Con el propósito de una distribución más efectiva de un diseño de aprendizaje que ha sido especificado conforme a LeadFlow4LD, los cinco documentos que se resumen en la *Tabla 3.13* pueden ser empaquetados siguiendo la especificación IMS de empaquetamiento de contenidos (IMS CP, 2004) dentro de una unidad de aprendizaje llamada UoLeadFlow4LD. Esta aproximación alentaría a los educadores a mejorar un diseño de aprendizaje

3. LeadFlow4LD: El método propuesto

empaquetado en una UoLeadFlow4LD, a través de la agregación de nuevas estructuras y situaciones de aprendizaje colaborativo.

```

<role-root>
<role id="Editor" occurrence="1" />
<role id="Revisor" occurrence="2" />
</role-root>
<tool-instances>
<tool-instance id="H1-1" />
<tool-instance id="H1-2" />
<tool-instance id="H2-1" />
<tool-instance id="H2-2" />
</tool-instances>
<data-instances>
<data-instance id="D1-1"
url="http://www.dabbleboard.com/draw?b=Guest724137&amp;i=3&amp;c=0069f27a78fcb7ce96759
061674564c8176c863d" />
<data-instance id="D1-2"
url="http://www.dabbleboard.com/draw?b=Guest724137&amp;i=4&amp;c=4d074aacc70ada12de060
blf75f198e9a50aca05" />
</data-instances>
<group-population>
<user id="p1" name="Juan" email="p1@gmail.com">
<role-occurrence-ref ref="1"></role-occurrence-ref>
<role-occurrence-ref ref="2"></role-occurrence-ref>
<role-active-occurrence-ref ref="1" />
<tool-instance-id-ref ref="H1-1" />
<tool-instance-id-ref ref="H2-1" />
</user>
<user id="p2" name="Miguel" email="p2@gmail.com">
<role-occurrence-ref ref="3"></role-occurrence-ref>
<role-occurrence-ref ref="4"></role-occurrence-ref>
<role-active-occurrence-ref ref="3" />
<tool-instance-id-ref ref="H1-2" />
<tool-instance-id-ref ref="H2-2" />
</user>
</group-population>

```

Figura 3.20: Parte del documento de población de grupos correspondiente a la situación de aprendizaje 1.1.

Documento	Descripción	Especificación	Responsable
Flujo de Aprendizaje	Define la secuencia de actividades de aprendizaje	IMS LD	Diseñador Instruccional
Flujo de Datos	Define la secuencia de invocaciones de herramientas y la relación entre datos y herramientas	XPDL	Diseñador Instruccional
Coordinación de Flujos	Define la relación entre actividades de aprendizaje e invocaciones de herramientas.	Propietaria	Diseñador Instruccional
Instanciación de Grupos	Define las instancias de datos y herramientas, así como las relaciones entre éstas.	XPDL	Diseñador Instruccional / Educador
Población de Grupos	Define la asignación de participantes a las actividades de aprendizaje y a las instancias de herramientas.	Propietaria	Educador

Tabla 3.13: Resumen de los documentos que especifican un diseño de aprendizaje a través del método propuesto (LeadFlow4LD).

Otra alternativa podría ser empaquetar todos los documentos en un formato de archivo de proceso de negocio (BPR, *Buisness Process Archive*). Sin embargo, puesto que estos

empaquetamientos serían compartidos entre diseñadores instrucciones y educadores, es más razonable utilizar una especificación del dominio educativo, como es el caso de IMS CP.

3.5 Aproximaciones de implementación del sistema de puesta en marcha de una UoLeadFlow4LD

Hasta aquí, una UoLeadFlow4LD contiene la especificación de la estructura y situaciones de aprendizaje colaborativo desde el punto de vista del diseñador instruccional y del educador. Sin embargo, ésta no puede ser interpretada todavía por motores que sean conformes a IMS LD y a XPDL, ya que se requiere de la ejecución coordinada de ambos flujos. Por tanto, con el propósito de poner en marcha una UoLeadFlow4LD, se debe implementar un sistema que sea capaz de ejecutar unidades de aprendizaje especificadas con el método propuesto. Con este propósito, se han identificado tres diferentes aproximaciones para su implementación, las cuales se describen a continuación.

La primera aproximación es llamada aproximación de implementación LeadFlow4LD, la cual consiste en interpretar directamente una UoLeadFlow4LD a través de un motor que sea conforme con el método propuesto. La ventaja de esta aproximación es que los documentos de una UoLeadFlow4LD pueden ser directamente interpretados por dicho motor; sin embargo, la desventaja de esta aproximación es que no es una solución interoperable con motores estándares existentes, sin contar con la dificultad que implica su implementación. Por tanto, para tomar ventaja de la existencia de motores existentes para poner en marcha una UoLeadFlow4LD, se distinguen otras dos posibilidades.

La segunda aproximación de implementación es llamada flujo de trabajo maestro y flujo de aprendizaje y flujo de datos esclavos. La idea de esta aproximación, como se ilustra en la *Figura 3.21* es usar un motor de flujo de aprendizaje existente (que sea conforme a IMS LD) que ponga en marcha una instancia del proceso que describe el flujo de aprendizaje. De forma similar, utilizar un motor de flujo de trabajo existente (que sea conforme con XPDL) usado con un doble propósito: para poner en marcha una instancia del flujo de datos y para poner en marcha una instancia de un proceso que sería usado para la coordinación entre flujos. La ventaja de esta aproximación, es que el documento de coordinación entre flujos se especificaría en un lenguaje de flujo de trabajo estándar, tal como XPDL. La desventaja, sin embargo, es que dicho documento requeriría de un mapeo para referir a las actividades del flujo de aprendizaje.

3. LeadFlow4LD: El método propuesto

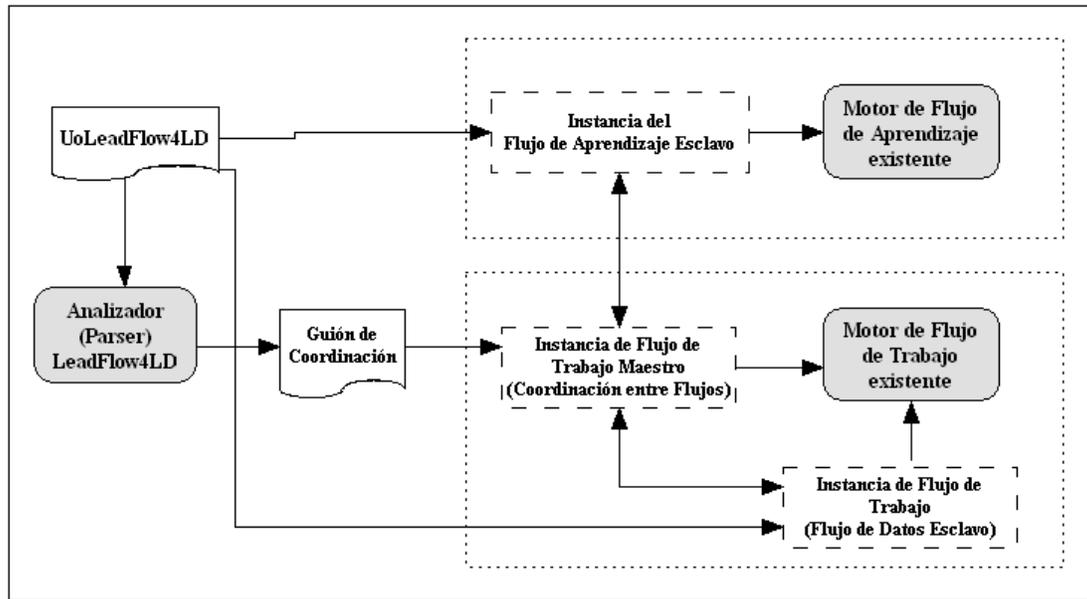


Figura 3.21: Aproximación de implementación flujo de trabajo maestro y flujo de aprendizaje y flujo de datos esclavos.

La tercera aproximación de implementación es llamada flujo de aprendizaje maestro y flujo de trabajo esclavo. La idea de esta aproximación, como se ilustra en la *Figura 3.22*, es usar un motor existente de flujo de aprendizaje como motor maestro y usar un motor existente de flujo de trabajo como esclavo. Con esta idea, a partir del análisis de la *UoLeadFlow4LD* se deberían generar documentos del flujo de aprendizaje y de flujo de trabajo que estén sincronizados entre sí, con el objetivo de que motores existentes de flujo de aprendizaje y flujo de trabajo puedan iniciar, detener y continuar la ejecución sincronizada de cada flujo. La ventaja de esta aproximación es que no requiere poner en marcha una instancia de un proceso de coordinación entre flujos, sino que el documento de coordinación de flujos sería usado solamente para conocer los momentos en el flujo de aprendizaje y en el flujo de datos, en donde cada motor (conforme a IMS LD y a XPDL) debe detener y reanudar su ejecución. La desventaja, sin embargo, es que se requiere la generación de documentos del flujo de aprendizaje y flujo de trabajo que estén sincronizados entre sí.

En esta Tesis Doctoral se propone esta última aproximación, ya que, primero, se trata de una solución interoperable con motores estándares actuales; segundo, porque no requiere poner en marcha una instancia de un proceso de coordinación entre flujos, que además tendría que ser especificado en un lenguaje del dominio de flujo de trabajo; y tercero, porque el documento de coordinación de flujos podría ser especificado en un lenguaje, que aunque propietario, podría ser más cercano al dominio educativo (en comparación con XPDL).

3. LeadFlow4LD: El método propuesto

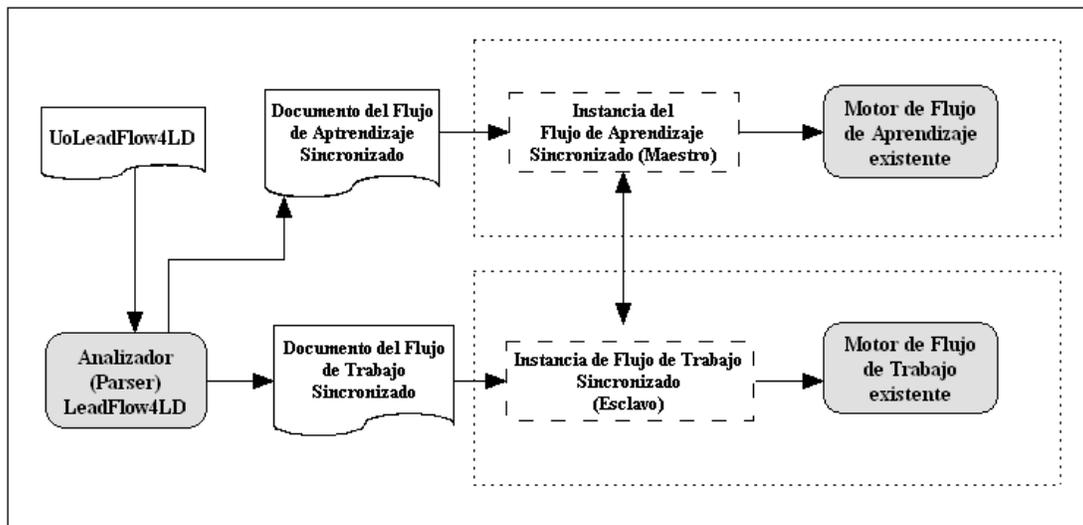


Figura 3.22: Aproximación de implementación flujo de aprendizaje maestro y flujo de trabajo esclavo.

3.6 Sistema de puesta en marcha de una UoLeadFlow4LD

El sistema de puesta en marcha de una UoLeadFlow4LD, de acuerdo a la aproximación maestro-esclavo de la Figura 3.22, sigue la arquitectura básica de un sistema basado en flujo de trabajo (ver Figura 3.24), el cual se basa en el modelo de referencia de flujo de trabajo propuesto por la WfMC (WfMC Terminology & Glossary, 1999) y mostrado en la Figura 3.23.

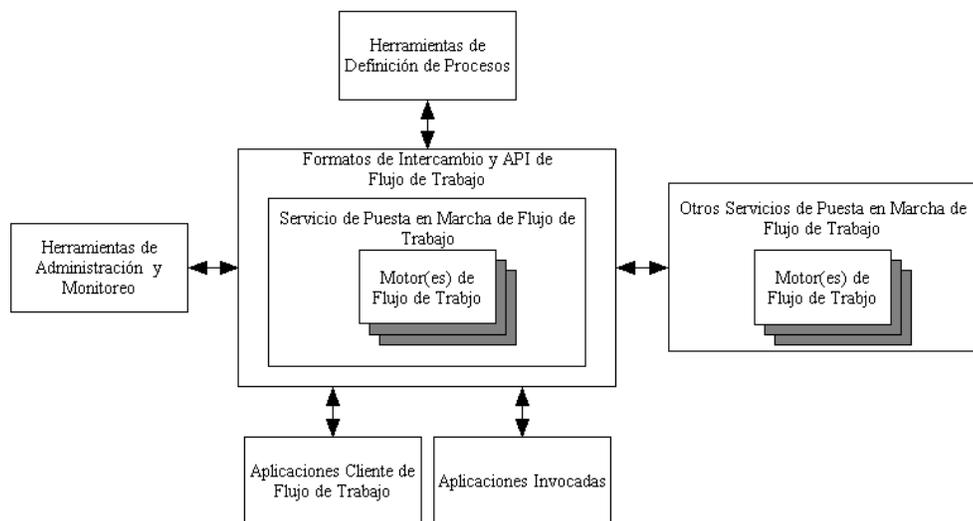


Figura 3.23: El modelo de referencia de flujo de trabajo. Fuente: (WfMC Terminology & Glossary, 1999).

3. LeadFlow4LD: El método propuesto

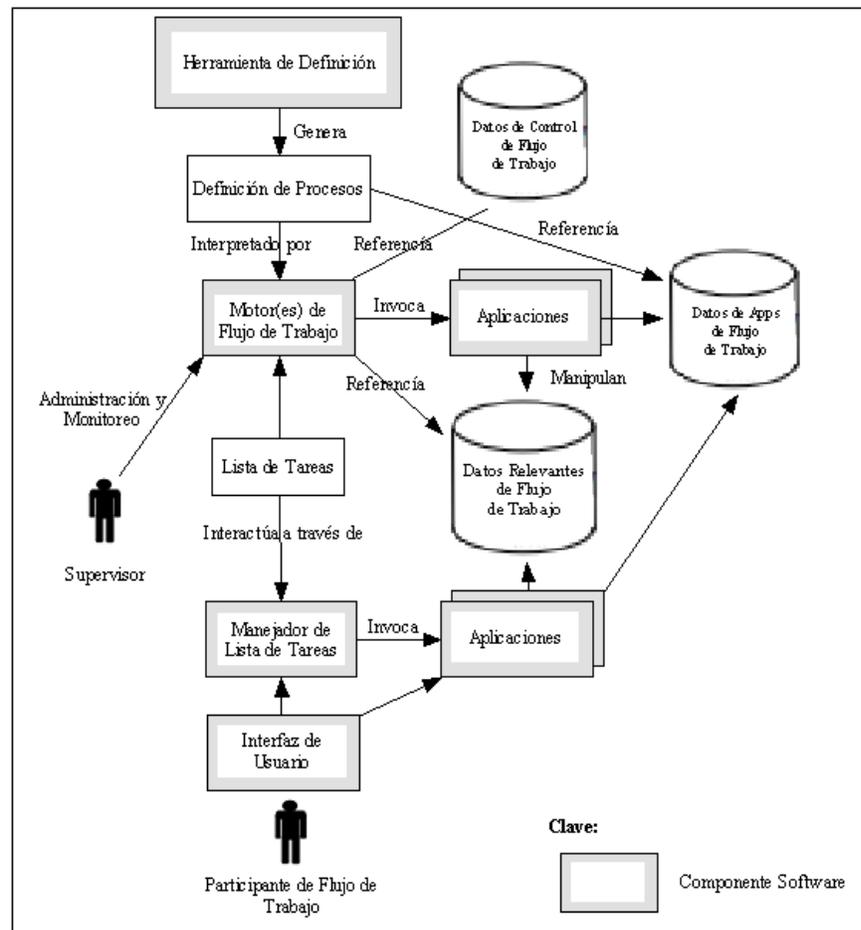


Figura 3.24: Arquitectura básica de un sistema de gestión de flujo de trabajo. Fuente: (WfMC Terminology & Glossary, 1999).

De acuerdo a esta arquitectura básica de la *Figura 3.24*, herramientas de flujo de trabajo son usadas para generar la definición (representación computacional) de procesos, los cuales son interpretados por motores de flujo de trabajo. Los supervisores son los encargados de poner en marcha y gestionar los procesos de flujo de trabajo, lo cual incluye la instanciación, administración y monitoreo de procesos. La ejecución de un proceso de flujo de trabajo implica la ejecución de una lista de tareas (*work items*), así como la invocación de aplicaciones y la referenciación de datos del proceso. La lista de tareas se refiere al trabajo a realizar por los participantes en las actividades de flujo de trabajo, y es gestionado a través del manejador de la lista de tareas. El manejador de la lista de tareas interactúa, por un lado, con el motor de flujo de trabajo (a través de un API de flujo de trabajo) controlando la progresión de las tareas e invocando las aplicaciones definidas en el proceso, y por otro lado, interactúa también con los participantes. Los participantes interactúan con el manejador de la lista de tareas (por medio de una interfaz de usuario) notificando, por ejemplo, que se inició o terminó una tarea, y también son los responsables de manejar las aplicaciones invocadas. Los datos pueden ser de distintos tipos: datos de control de flujo de trabajo (por ejemplo,

instancias de procesos), datos relevantes de flujo de trabajo (por ejemplo, variables de procesos) y datos generados por las aplicaciones de flujo de trabajo (por ejemplo, ficheros de configuración). Los datos relevantes de flujo de trabajo representan variables del proceso que pueden ser modificadas en tiempo de ejecución.

3.6.1 Arquitectura del sistema propuesto

De acuerdo a esta arquitectura básica de un sistema basado en flujo de trabajo, y tomando en cuenta la naturaleza de flujo de trabajo de LeadFlow4LD, la arquitectura del sistema propuesto de puesta en marcha de una UoLeadFlow4LD se muestra en la *Figura 3.25* y se describe a continuación. Herramientas de autoría son usadas por diseñadores instruccionales y/o educadores para especificar los cinco documentos que definen por completo una UoLeadFlow4LD: documentos del flujo de aprendizaje y flujo de datos, documento de coordinación de flujos, y documentos de instanciación y población de grupos. Esta UoLeadFlow4LD es analizada automáticamente por un componente software llamado componente analizador LeadFlow4LD, el cual tiene como objetivo generar automáticamente dos nuevos documentos: el **documento del flujo de aprendizaje sincronizado** y el **documento del flujo de trabajo sincronizado**. La forma en que estos documentos se generan automáticamente se explica más adelante. Cada documento es interpretado independientemente por un motor de flujo que es conforme con estándares de los campos LD y Flujo de Trabajo, y que son llamados respectivamente, motores de flujo de aprendizaje y flujo de trabajo. Cada motor interactúa con el administrador del sistema, con participantes (estudiantes, profesores), con instancias y variables de ambos procesos, de distintas maneras.

Por un lado, el motor de flujo de aprendizaje interactúa con el administrador del sistema a través de herramientas de administración y monitoreo, las cuales son usadas para crear, gestionar y monitorear instancias de procesos del flujo de aprendizaje. Una vez iniciada la ejecución de una instancia de un proceso del flujo de aprendizaje, los estudiantes participantes pueden utilizar la interfaz de usuario de LeadFlow4LD para acceder a su entorno de aprendizaje e interactuar con el manejador de la lista de tareas, esto es, para interactuar con la secuencia de actividades de aprendizaje (según el API del motor de flujo de aprendizaje seleccionado). El motor del flujo de aprendizaje puede manipular no solo el estado de sus propias variables de proceso (a través de referenciar las variables o datos del proceso), sino también el estado de variables de procesos en el flujo de trabajo, a través de la invocación de adaptadores que interactúan con la interfaz del motor de flujo de trabajo (llamados adaptadores del motor de flujo de trabajo).

Similarmente, y por otro lado, el motor del flujo de trabajo interactúa con el administrador del sistema a través de herramientas de administración y monitoreo, las cuales son usadas para crear, gestionar y monitorear instancias de procesos de flujo de trabajo (que describen propiamente el flujo de datos). Una vez iniciada la ejecución de una instancia de un proceso de flujo de trabajo, los estudiantes participantes pueden utilizar la interfaz de usuario de LeadFlow4LD para interactuar con el manejador de las lista de tareas que representa la

3. LeadFlow4LD: El método propuesto

secuencia de invocaciones de herramientas. El motor de flujo de trabajo puede invocar herramientas (aplicaciones) y referenciar variables de flujo de trabajo que representan artefactos de entrada y salida de herramientas. El motor de flujo de trabajo puede manipular no solo el estado de sus propias variables de proceso (a través de referenciar las variables o datos del proceso), sino también el estado de variables de procesos en el flujo de aprendizaje, a través de la invocación de adaptadores que interactúan con la interfaz del motor de flujo de aprendizaje (llamados adaptadores del motor de flujo de aprendizaje). Como debe ser claro, esta característica de que un motor pueda invocar adaptadores que manipulen el estado de variables de procesos en otros sistemas de puesta en marcha, hace posible delegar el control entre motores de flujo, lo que habilita la coordinación entre flujos.

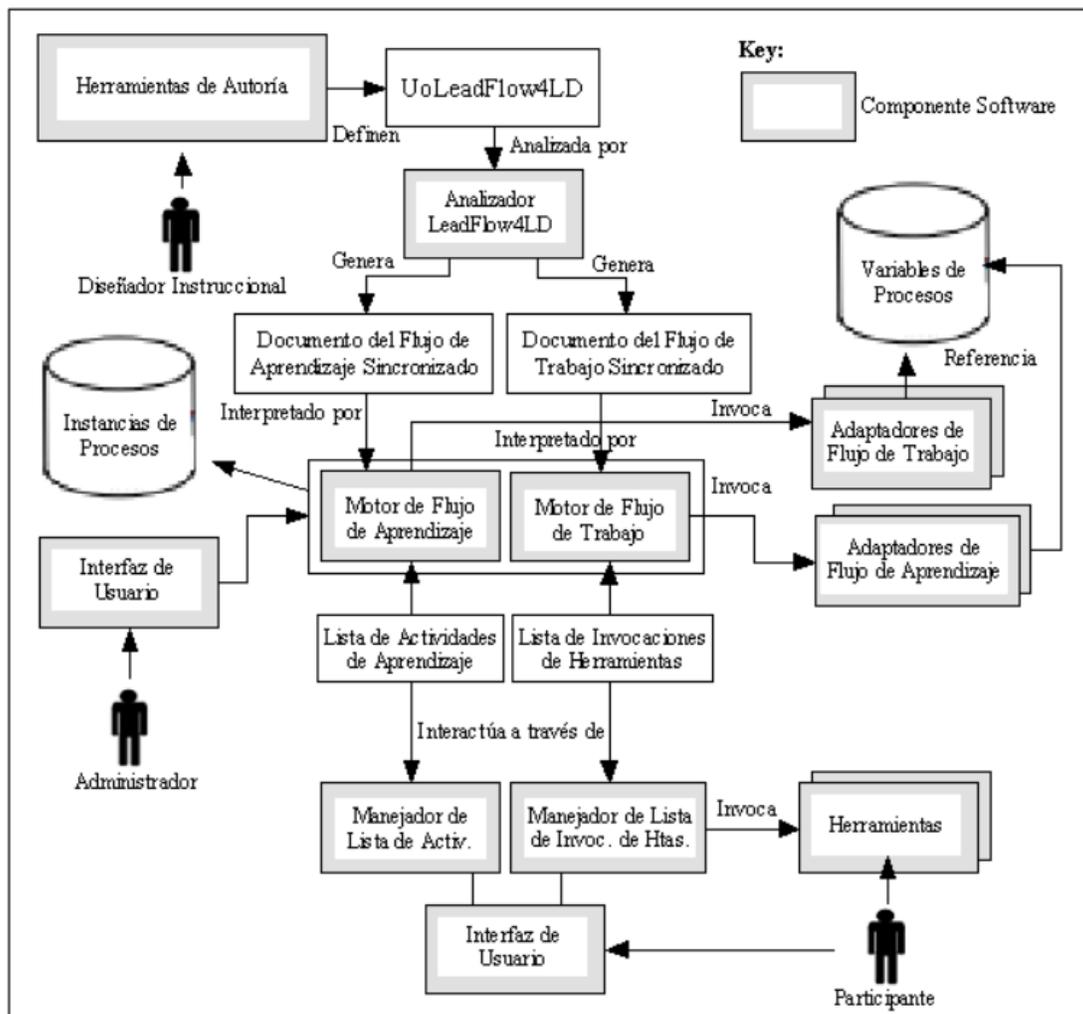


Figura 3.25: Arquitectura del sistema propuesto de puesta en marcha de una UoLeadFlow4LD.

La coordinación entre instancias de procesos del flujo de aprendizaje y flujo de trabajo sincronizados, se describe en el diagrama de secuencia UML de la Figura 3.26. Primero, un participante inicia (o reinicia) una actividad de aprendizaje a través de la interfaz que

3. LeadFlow4LD: El método propuesto

despliega la lista de actividades del flujo de aprendizaje. Cuando el participante inicia el flujo de datos (usando esta misma interfaz) que da soporte a la actividad de aprendizaje, el motor del flujo de aprendizaje lanza un adaptador del motor de flujo de trabajo con el propósito de avisar el inicio (o reinicio) de la ejecución del flujo de datos. Lo que en realidad hace el adaptador es modificar (vía el API del motor de flujo de trabajo) el valor de las variables de flujo de trabajo que representan el inicio del flujo de datos para esa actividad de aprendizaje. A continuación, el motor de flujo de trabajo lanza la herramienta que es usada por los estudiantes participantes para dar soporte a la actividad de aprendizaje. Después, cuando los participantes terminan de usar la última herramienta de la secuencia de herramientas de una actividad de aprendizaje (según lo definido en el flujo de datos), el motor de flujo de trabajo lanza un adaptador del motor del flujo de aprendizaje con el propósito de avisar que debe continuar su ejecución. Lo que en realidad hace el adaptador es modificar (vía el API del motor de flujo de aprendizaje) el valor de las variables del flujo de aprendizaje que representan la terminación de una actividad de aprendizaje.

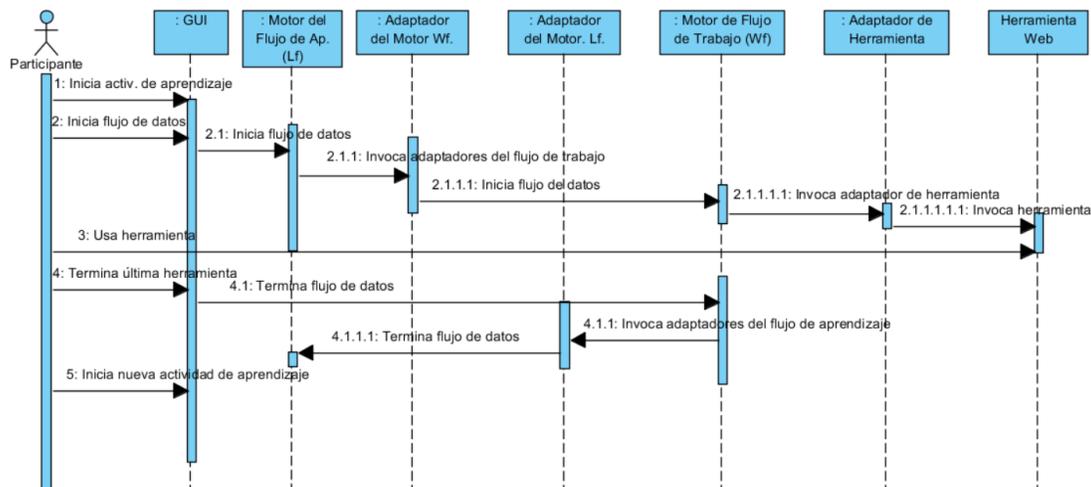


Figura 3.26: Diagrama de secuencia de la ejecución coordinada de los motor de flujo de aprendizaje y flujo de trabajo.

Detalles sobre la implementación de estos adaptadores se darán más adelante. Antes, se describe la funcionalidad del componente analizador LeadFlow4LD responsable de generar automáticamente los documentos del flujo de aprendizaje y flujo de trabajo sincronizados, los cuales son necesarios para la coordinación entre flujos.

3.6.2 Funcionalidad del Componente Analizador LeadFlow4LD

La principal tarea del componente analizador LeadFlow4LD es analizar una UoLeadFlow4LD, con el objetivo de generar automáticamente los documentos del flujo de aprendizaje y flujo de trabajo sincronizados. La ejecución independiente de ambos documentos por motores estándares de flujo de aprendizaje y flujo de trabajo, producirá la ejecución coordinada de ambos flujos. En esta sección se describe lo que debe hacer el

3. LeadFlow4LD: El método propuesto

componente analizador LeadFlow4LD, mientras que en una sección posterior se describirá cómo lo debe llevar a cabo.

Para que la puesta en marcha del flujo de aprendizaje y flujo de trabajo sincronizados produzca una ejecución alternada entre flujos, la principal tarea del componente analizador LeadFlow4LD es la de incorporar en ambos documentos un mecanismo que permita esta ejecución coordinada. Una forma de conseguirlo es a través de agregar a cada uno de los documentos, estados de sincronización y acciones que permitan alcanzar dichos estados. Para entender este mecanismo, considere el diagrama de estados que caracteriza la puesta en marcha (no coordinada) del flujo de aprendizaje y el flujo de datos para el caso del ejemplo ilustrativo de revisión entre 2 estudiantes. Por un lado, la *Figura 3.27a* muestra los estados que caracterizan el flujo de aprendizaje, esto es, los estados existentes durante la realización de las actividades de edición y revisión para cada uno de los dos participantes. Por otro lado, la *Figura 3.27b* muestra los estados que caracterizan el flujo de datos, esto es, los estados existentes durante la utilización de las herramientas de edición y revisión de cada una de las actividades de aprendizaje.

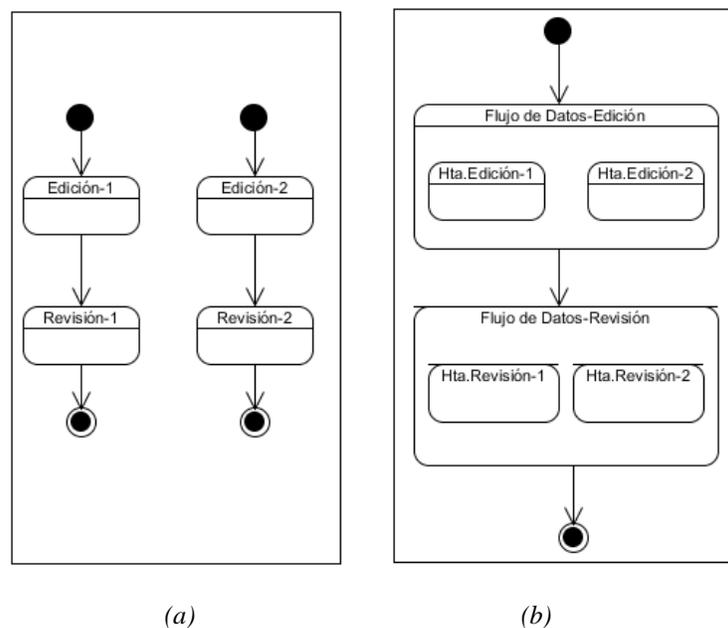


Figura 3.27: Diagrama de estados que caracterizan (a) el flujo de aprendizaje y (b) el flujo de datos (no coordinados), correspondientes a la situación de aprendizaje del ejemplo ilustrativo de revisión entre 2 estudiantes.

Para la coordinación del flujo de aprendizaje y el flujo de datos, se requiere de la interacción entre estados de ambos flujos. Una forma de conseguirlo sin alterar los estados existentes, es a través de agregar estados intermedios, llamados estados de sincronización, tanto en el flujo de aprendizaje como en el flujo de datos, así como las acciones que permitan alcanzar dichos estados, como se ilustra en el diagrama de estados de la *Figura 3.28* para el

3. LeadFlow4LD: El método propuesto

caso del ejemplo ilustrativo (estados de sincronización están sombreados, mientras que las acciones son ilustradas con flechas y la etiqueta “a”).

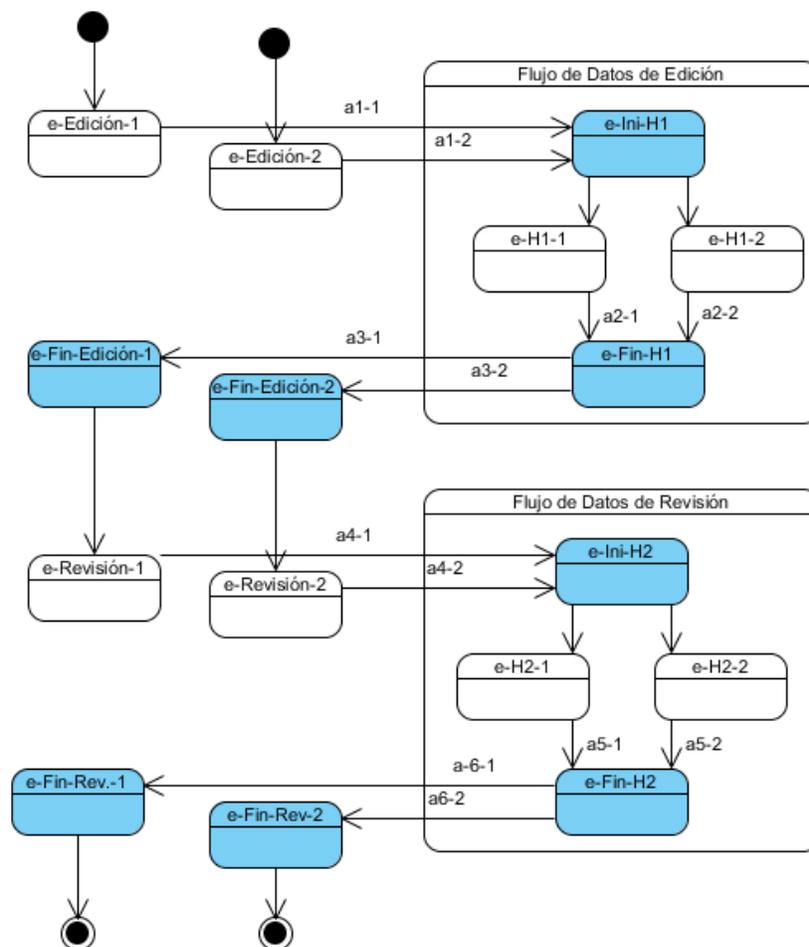


Figura 3.28: Diagrama de estados característico en LeadFlow4LD en el que interaccionan estados del flujo de aprendizaje con estados del flujo de datos, correspondiente al ejemplo ilustrativo de revisión entre 2 estudiantes. Los estados de sincronización se muestran sombreados.

Cuando la ejecución del flujo de aprendizaje es independiente del flujo de datos, del estado de edición se pasa directamente al estado de revisión (ver Figura 3.27a). Sin embargo, cuando ambos flujos están coordinados entre sí como se muestra en la Figura 3.28, se deben representar estados de sincronización en ambos flujos. Por un lado, en el flujo de aprendizaje, estados de sincronización pueden ser usados para describir la terminación de las actividades de aprendizaje. Esto permitiría representar los momentos en que el flujo de aprendizaje puede avanzar a la siguiente actividad de aprendizaje. De otra forma, un participante podría iniciar con la actividad de revisión sin que el otro participante haya terminado aún con el flujo de datos de la actividad de edición (entonces, un participante podría no tener un artefacto para revisar). Similarmente, por otro lado, en el flujo de datos, estados de sincronización pueden

ser usados para describir el inicio y fin del flujo de datos de cada actividad de aprendizaje. Esto permitiría representar los momentos en el flujo de datos del inicio y terminación de la secuencia de invocaciones de herramientas de cada una de las actividades de aprendizaje. De otra forma, un participante que haya terminado de usar la herramienta de edición podría empezar a usar la herramienta de revisión, pero sin haber iniciado con la actividad de revisión en el flujo de aprendizaje. Adicionalmente, como se puede ver en la *Figura 3.28*, las acciones son el mecanismo usado para alcanzar los distintos estados de sincronización entre flujos.

En esta tesis se propone, por un lado, que los estados de sincronización sean modelados a través de variables de proceso (propiedades locales personales en el caso del flujo de aprendizaje especificado con IMS LD), donde el valor de cada variable se usa para determinar si una actividad de aprendizaje ha terminado o no, o si una secuencia de herramientas ha iniciado o terminado. Por otro lado, se propone que las acciones para alcanzar los distintos estados de sincronización puedan ser modeladas a través de la invocación de adaptadores. Un adaptador del motor de flujo de trabajo puede ser invocado por el motor de flujo de aprendizaje para modificar el valor de las variables de proceso que representan el inicio del flujo de datos. Similarmente, un adaptador del motor de flujo de aprendizaje puede ser invocado por el motor de flujo de trabajo para modificar el valor de las variables de proceso que representan la terminación de actividades de aprendizaje.

Por tanto, la principal funcionalidad del componente analizador LeadFlow4LD es la de agregar automáticamente, dentro del documento del flujo de aprendizaje sincronizado, las variables de proceso y las condiciones que representan la terminación de cada una de las actividades de aprendizaje; agregar automáticamente, dentro del documento del flujo de trabajo sincronizado, las variables de proceso y las condiciones que representan el inicio y el fin del flujo de datos; y agregar también automáticamente y en ambos documentos, los adaptadores que permitirán alcanzar dichos estados. Es importante hacer notar que una versión distinta del componente analizador LeadFlow4LD podría generar documentos válidos para otras soluciones como, por ejemplo, una solución basada en el lenguaje de guiones CSCL (Miao, Hoeksema, Hoppe & Harrer, 2005). Por tanto, la propuesta de modelado de LeadFlow4LD también podría ser válida para otras soluciones técnicas de combinación (o no) de flujos.

3.6.3 Procedimiento a seguir por el componente analizador LeadFlow4LD

Una vez decida la infraestructura tecnológica, a continuación se describe el procedimiento a través del cual el componente analizador LeadFlow4LD debe generar los documentos del flujo de aprendizaje y flujo de trabajo sincronizados, a partir de analizar los distintos documentos que conforman una UoLeadFlow4LD.

Representación de los Estados de Sincronización

Para delegar el control al motor de flujo de trabajo e iniciar una secuencia de herramientas correspondiente a una actividad de aprendizaje, se requiere representar en el

3. LeadFlow4LD: El método propuesto

documento del flujo de trabajo sincronizado, el estado que representa el inicio de una secuencia de herramientas (inicio del flujo de datos). Tal estado se puede representar definiendo variables globales de flujo de trabajo, así como condiciones de inicio, para cada una de las instancias de herramientas que dan inicio al flujo de datos, y por cada uno de los participantes que usan o comparten una instancia de la herramienta. Estas variables pueden ser nombradas por el componente analizador LeadFlow4LD con el identificador $V-Hi-j-ini-pk$, donde i representa el identificador de la herramienta, j la instancia de la herramienta y pk el participante que usa la herramienta. Es importante darse cuenta de que si dos participantes pa y pb comparten una misma herramienta, entonces se requieren definir dos variables para representar el inicio de la secuencia de herramientas: $V-Hi-j-ini-pa$ y $V-Hi-j-ini-pb$, así como una condición de inicio (por ejemplo, que ambas variables se evalúen a *True*). Estas variables de flujo de trabajo y condiciones que representan el inicio del flujo de datos, pueden ser generadas automáticamente por el componente analizador LeadFlow4LD y definidas en el documento del flujo de trabajo sincronizado, a partir de analizar los siguientes documentos de una UoLeadFlow4LD: flujo de aprendizaje, instanciación de grupos, coordinación de flujos, y población de grupos.

Considere el caso del ejemplo ilustrativo en el que el flujo de datos de la actividad de edición inicia con dos instancias de la herramienta de edición ($H1-1$ y $H1-2$) y que son utilizadas individualmente por los participantes $p1$ y $p2$. Entonces, después de analizar una UoLeadFlow4LD, el componente analizador LeadFlow4LD debe definir las variables globales de flujo de trabajo: $V-H1-1-ini-p1$ y $V-H1-2-ini-p2$. Estas variables representan el estado de inicio de la secuencia de herramientas en la actividad de edición: si el valor de la variable $V-H1-1-ini$ es *True*, entonces la instancia de la herramienta $H1-1$ puede ser iniciada por el participante $p1$. La *Tabla 3.14* muestra todas las variables de flujo de trabajo y las condiciones que representan el estado de inicio del flujo de datos para cada una de las actividades de aprendizaje del ejemplo ilustrativo.

Instancia de Herramienta	Variables de Flujo de Trabajo	Valor Inicial	Condición de Inicio del Flujo de Datos
$H1-1$	$V-H1-1-ini-p1$	<i>False</i>	$V-H1-1-ini-p1 == True$
$H1-2$	$V-H1-2-ini-p2$	<i>False</i>	$V-H1-2-ini-p2 == True$
$H2-1$	$V-H2-1-ini-p1$	<i>False</i>	$V-H2-1-ini-p1 == True$
$H2-2$	$V-H2-2-ini-p2$	<i>False</i>	$V-H2-2-ini-p2 == True$

Tabla 3.14: Variables de flujo de trabajo y condiciones que representan el inicio del flujo de datos para el ejemplo ilustrativo de revisión entre dos estudiantes.

Similarmente, para delegar el control al motor del flujo de aprendizaje, se requiere representar en el documento del flujo de trabajo sincronizado el estado de finalización de la secuencia de herramientas de una actividad de aprendizaje. Tal estado se puede representar definiendo variables globales de flujo de trabajo y condiciones de terminación para cada una de las instancias de herramientas que finalizan el flujo de datos (las últimas herramientas de la secuencia). Estas variables pueden ser nombradas por el componente analizador LeadFlow4LD con el identificador $V-Hi-j-fin$, donde i representa el número de la herramienta y j la instancia de la herramienta. De esta forma, si el valor de la variable $V-Hi-j-fin$ es *True*,

3. LeadFlow4LD: El método propuesto

entonces significa que la instancia de la última herramienta en la secuencia (H_{i-j}) ha terminado. Estas variables de flujo de trabajo y condiciones que representan la terminación del flujo de datos, pueden ser generadas automáticamente por el componente analizador LeadFlow4LD y definidas en el documento del flujo de trabajo sincronizado, a partir de analizar los siguientes documentos de una UoLeadFlow4LD: flujo de aprendizaje, instanciación de grupos, coordinación de flujos, y población de grupos.

Considere el ejemplo ilustrativo durante la actividad de edición. Aquí las instancias de herramientas de edición $H1-1$ y $H1-2$, usadas respectivamente por los participantes $p1$ y $p2$, terminan la secuencia de herramientas (de hecho también la inician, ya que solo hay una herramienta en la secuencia). Entonces, después de analizar una UoLeadFlow4LD, el componente analizador LeadFlow4LD debe definir las variables globales de flujo de trabajo: $V-H1-1-fin$ y $V-H1-2-fin$. Estas variables representan el estado de finalización de la secuencia de herramientas en la actividad de edición: si el valor de la variable $V-H1-1-fin$ es *True*, entonces la instancia de la herramienta $H1-1$ ha sido terminada por el participante $p1$. Sin embargo, para que el control de flujo se pueda delegar al flujo de aprendizaje se requiere que el valor de la variable $V-H1-2-fin$ del participante $p2$ también sea *True*. La *Tabla 3.14* muestra todas las variables de flujo de trabajo y condiciones que representan el estado de finalización del flujo de datos para el ejemplo ilustrativo.

Actividad de Fin de Herramienta	Variable Agregada	Valor	Condiciones Agregadas de Fin de Herramienta
$H1-1$	$V-H1-1-fin$	<i>False</i>	$V-H1-1-fin == True$
$H1-2$	$V-H1-2-fin$	<i>False</i>	$V-H1-2-fin == True$
$H2-1$	$V-H2-1-fin$	<i>False</i>	$V-H2-1-fin == True$
$H2-2$	$V-H2-2-fin$	<i>False</i>	$V-H2-2-fin == True$

Tabla 3.15: Variables de flujo de trabajo y condiciones que representan la terminación del flujo de datos correspondiente al ejemplo ilustrativo de revisión entre dos estudiantes.

Por último, con el propósito de mantener bloqueado el flujo de aprendizaje hasta que todos los participantes hayan terminado la secuencia de herramientas de la actividad, se requiere representar en el documento del flujo de aprendizaje sincronizado, el estado de terminación de todas las actividades de aprendizaje donde existe una invocación al flujo de datos. Tal estado se puede representar en el documento del flujo de aprendizaje sincronizado, definiendo propiedades IMS LD de tipo local personal, así como condiciones de terminación de actividad para cada una de las actividades de aprendizaje que realizan una invocación al flujo de datos. Estas propiedades pueden ser nombradas por el componente analizador LeadFlow4LD con el identificador $P-A_i$, donde i representa el número de la actividad. El valor de estas variables indican si una actividad de aprendizaje ha sido completada o no: si en la actividad de aprendizaje A_i participan los usuarios $p1, p2, \dots, pn$, entonces sólo si el valor de las propiedades locales personales $P-A_i(p1), P-A_i(p2), \dots, P-A_i(pn)$ es *True*, significa que la actividad A_i ha sido completada. Estas propiedades IMS LD y condiciones que representan la terminación de una actividad de aprendizaje, pueden ser generadas automáticamente por el componente analizador LeadFlow4LD y definidas en el documento del flujo de aprendizaje

3. LeadFlow4LD: El método propuesto

sincronizado, a partir de analizar los siguientes documentos de una UoLeadFlow4LD: flujo de aprendizaje, coordinación de flujos, y población de grupos.

Considere la actividad de edición del ejemplo ilustrativo, donde participan los estudiantes $p1$ y $p2$. Para que la actividad de edición $A1$ esté completada se requiere que los valores de las propiedades $P-A1(p1)$ y $P-A1(p2)$ sean ambas *True*. La *Tabla 3.16* muestra todas las propiedades IMS LD y las condiciones que representan el estado de terminación de las actividades de aprendizaje para el ejemplo ilustrativo.

Actividad de Aprendizaje	Propiedad Local Personal			Condición para Completar la Actividad
	Fin de Actividad	Participante	Valor inicial	
Edición (A1)	$P-A1$	$p1$	<i>False</i>	$P-A1(p1) == True \text{ and } P-A1(p2) == True$
		$p2$	<i>False</i>	
Revisión (A2)	$P-A2$	$p1$	<i>False</i>	$P-A2(p1) == True \text{ and } P-A2(p2) == True$
		$p2$	<i>False</i>	

Tabla 3.16: Propiedades y condiciones agregadas al documento del flujo de aprendizaje sincronizado usadas para bloquear el flujo de aprendizaje, correspondiente al ejemplo ilustrativo de revisión entre dos estudiantes.

Representación de Acciones

Una vez representados, dentro de los documentos del flujo de aprendizaje y flujo de trabajo sincronizados, los estados de sincronización requeridos para la coordinación entre flujos, la siguiente tarea del componente analizador LeadFlow4LD es representar, también en ambos documentos, las acciones necesarias para que se puedan alcanzar dichos estados. En general, son tres tipos de acciones que se deben representar: las acciones para delegar el control del flujo de aprendizaje al flujo de datos, esto es, las acciones para alcanzar el estado de inicio del flujo de datos; las acciones para alcanzar el estado final del flujo de datos, y las acciones para devolver el control al flujo de aprendizaje, esto es, las acciones para alcanzar el estado de la siguiente actividad de aprendizaje.

Acciones para Delegar el Control al Flujo de Datos

Para delegar el control al motor del flujo de datos (desde el flujo de aprendizaje), se requiere agregar en el documento del flujo de aprendizaje sincronizado invocaciones de adaptadores del motor de flujo de trabajo que permitan modificar las variables que representan el inicio del flujo de datos. Estos adaptadores pueden ser nombrados por el componente analizador LeadFlow4LD con el identificador *AppWf-Hi-j-ini-pk*, donde *i* representa el número de la herramienta que da inicio al flujo de datos, *j* la instancia de la herramienta, y *pk* el participante que usa la herramienta. Los identificadores de las variables a modificar se pueden pasar como argumentos en cada invocación. Esta invocación de adaptadores del motor de flujo de trabajo, junto con sus argumentos, pueden ser generados automáticamente por el componente analizador LeadFlow4LD y agregados en el documento del flujo de aprendizaje sincronizado, a partir de analizar los siguientes documentos de una

3. LeadFlow4LD: El método propuesto

UoLeadFlow4LD: flujo de aprendizaje, instanciación de grupos, coordinación de flujos, y población de grupos.

Considere el caso del ejemplo ilustrativo, cuando el participante $p1$ en la actividad de edición decide iniciar el flujo de datos. En este caso, el valor de la variable que representa el inicio de la secuencia de herramientas de edición ($V-H1-1-ini-1$) debe cambiar al valor *True*, lo cual habilita al motor de flujo de trabajo a invocar la herramienta de edición $H1-1$ en la que interviene dicho participante ($p1$). De la misma forma, cuando el participante $p2$ en la actividad de edición decide iniciar el flujo de datos, la variable $V-H1-2-ini-2$ debe cambiar al valor *True* para habilitar la invocación de la herramienta de edición $H1-2$ (usada por $p2$). La *Tabla 3.17* resume las invocaciones de adaptadores del motor de flujo de trabajo, junto con sus argumentos, requeridos para el caso del ejemplo ilustrativo.

Sin embargo, para ejemplo ilustrativo y el caso de IMS LD, estos adaptadores del motor de flujo de trabajo no pueden ser agregados directamente al documento del flujo de aprendizaje sincronizado (que está representado en IMS LD). En su lugar, deben ser agregados dentro de recursos de contenido de tipo *imsl-d-content* y asociados a entornos (*environments*) de las actividades de aprendizaje. Es importante darse cuenta de que estos recursos de contenido son distintos para cada estudiante participante. Por ejemplo, en la actividad de edición, el participante $p1$ requiere que se invoque el adaptador: *AppWf-H1-1-ini-1*, pero el participante $p2$ requiere que se invoque el adaptador: *AppWf-H1-2-ini-2*.

Adaptador del Motor de Flujo de Trabajo	Actividad de Aprendizaje donde se Invoca	Argumentos	
		Variable ID	Valor
<i>AppWf-H1-1-ini-1</i>	<i>A1</i>	<i>V-H1-1-ini-1</i>	<i>True</i>
<i>AppWf-H1-2-ini-1</i>	<i>A1</i>	<i>V-H1-2-ini-2</i>	<i>True</i>
<i>AppWf-H2-1-ini-1</i>	<i>A2</i>	<i>V-H2-1-ini-1</i>	<i>True</i>
<i>AppWf-H2-2-ini-1</i>	<i>A2</i>	<i>V-H2-2-ini-2</i>	<i>True</i>

Tabla 3.17: Características de la invocación de adaptadores del motor del flujo de trabajo, usados para delegar el control al flujo de trabajo para el ejemplo ilustrativo.

Para controlar el contenido que se va a mostrar en tiempo de ejecución a los distintos participantes de las actividades de aprendizaje, se requieren definir condiciones IMS LD (*if-then-else*) en función del valor real de la propiedad *P-user* que representa a cada participante. De esta forma, durante la generación de los recursos de contenido, se utiliza la etiqueta *div-class* para especificar lo que se va a mostrar u ocultar a cada uno de los participantes. En la *Tabla 3.18* se listan las características que deben tener los recursos *imsl-d-content* usados para referenciar las aplicaciones de invocación del motor de flujo de trabajo para el caso del ejemplo ilustrativo.

3. LeadFlow4LD: El método propuesto

Actividad de Aprendizaje	Recurso <i>ImSldContent</i>	div (class)	(Href)
Edición (A1)	<i>res-IMS LD-content-A1.xml</i>	<i>p1</i>	<i>AppWf-H1-1-ini-1</i>
		<i>p2</i>	<i>AppWf-H1-2-ini-1</i>
Revisión (A2)	<i>res-IMS LD-content-A2.xml</i>	<i>p1</i>	<i>AppWf-H2-1-ini-1</i>
		<i>p2</i>	<i>AppWf-H2-2-ini-1</i>

Tabla 3.18: Recursos *imsld-content* agregados al documento del flujo de aprendizaje sincronizado para el ejemplo ilustrativo de revisión entre dos estudiantes.

Acciones para Finalizar el Flujo de Datos

Para alcanzar el estado de finalización del flujo de datos, se requiere que al terminar la última herramienta de una secuencia de herramientas en una actividad de aprendizaje, se modifique el estado de las variables de flujo de trabajo que representan la terminación del flujo de datos. Sin embargo, las herramientas (típicamente servicios de terceros) invocadas por el motor de flujo de trabajo no informan a éste cuándo son terminadas por los participantes (a menos que la herramienta se diseñe a la medida para este propósito). Para resolver esta dificultad, se pueden usar interfaces de usuario que interactúan con el motor de flujo de trabajo. De esta forma, los participantes pueden usar esta interfaz para indicar el momento de finalización de una herramienta, a través de la cual sería posible modificar el valor de la variable que representa la terminación del flujo de datos. Estas interfaces usadas para indicar la finalización del flujo de datos, junto con las variables a modificar, pueden ser nombradas por el componente analizador LeadFlow4LD con los identificadores *Form-Hi-j*, y *V-Hi-j-fin*, respectivamente, donde *i* representa el número de la herramienta de terminación del flujo de datos, y *j* la instancia de la herramienta. Ambos (formularios de finalización del flujo de datos junto con las variables a modificar), pueden ser generados automáticamente por el componente analizador LeadFlow4LD y referenciados en el documento del flujo de trabajo sincronizado, a partir de analizar los siguientes documentos de una UoLeadFlow4LD: instanciación de grupos y coordinación de flujos.

Considere el caso del ejemplo ilustrativo, cuando el participante *p1* en la actividad de edición termina de usar la herramienta de edición, el valor de la variable que representa el fin de la herramienta (*V-H1-1-fin*) debe cambiar al valor *True*. Similarmente, cuando el participante *p2* termina de usar la herramienta de edición, el valor de la variable *V-H1-2-fin* debe cambiar al valor *True*. Esto habilita al motor de flujo de trabajo a delegar el control al flujo de aprendizaje. La Tabla 3.19 muestra las interfaces de usuario a generar y las variables referenciadas en este caso. Por ejemplo, cuando el participante *p1* inicia el flujo de datos de la actividad de edición, el motor de flujo de trabajo lanza al mismo tiempo una interfaz de usuario y la instancia de la herramienta de edición *H1-1*. A través de la interfaz de usuario el participante puede indicar el momento en que termina de usar la herramienta. Esta acción

3. LeadFlow4LD: El método propuesto

provoca un cambio en el valor de la variable $V-H1-1-fin$ que representa la terminación de la herramienta.

Nombre de la Interfaz de Usuario	Actividad de Flujo de Trabajo Asociada	Variable de Fin de Herramienta
<i>Form-H1-1</i>	<i>H1-1</i>	<i>V-H1-1-fin</i>
<i>Form-H1-2</i>	<i>H1-2</i>	<i>V-H1-2-fin</i>
<i>Form-H2-1</i>	<i>H2-1</i>	<i>V-H2-1-fin</i>
<i>Form-H2-2</i>	<i>H2-2</i>	<i>V-H2-2-fin</i>

Tabla 3.19: Interfaces de usuario de finalización del flujo de datos, junto con las variables a modificar, para el caso del ejemplo ilustrativo.

Acciones para Delegar el Control al Flujo de Aprendizaje

Para delegar el control al motor del flujo de aprendizaje, se requiere agregar en el documento del flujo de trabajo sincronizado invocaciones de adaptadores del motor del flujo de aprendizaje que permitan modificar las propiedades IMS LD que representan la terminación de las actividades de aprendizaje. Estos adaptadores pueden ser nombrados por el componente analizador LeadFlow4LD con el identificador $AppLf-Hi-j-fin$, donde i representa el número de la herramienta que finaliza el flujo de datos, y j la instancia de la herramienta. Los identificadores de las propiedades IMS LD a modificar se pueden pasar como argumentos en cada invocación. Esta invocación de adaptadores del motor de flujo de aprendizaje, junto con sus argumentos, pueden ser generados automáticamente por el componente analizador LeadFlow4LD y agregados en el documento del flujo de trabajo sincronizado, a partir de analizar los siguientes documentos de una $UoLeadFlow4LD$: flujo de aprendizaje, instanciación de grupos, coordinación de flujos, y población de grupos.

Considere el caso del ejemplo ilustrativo, cuando el participante $p1$ en la actividad de edición, termina el flujo de datos de edición (termina de usar la herramienta $H1-1$). En tal caso, el valor de la propiedad IMS LD $P-A1(p1)$, que representa la terminación de la actividad de aprendizaje del participante $p1$, debe cambiar al valor *True*. Sin embargo, cuando esto ocurre, el flujo de aprendizaje aún no queda desbloqueado, ya que aún resta que el participante $p2$ termine de usar la herramienta $H1-2$ en la actividad de edición, de forma que el valor de la propiedad $P-A1(p2)$ también sea *True*. La Tabla 3.20 muestra la invocación de adaptadores del motor del flujo de aprendizaje, junto con la lista de sus argumentos, para el caso del ejemplo ilustrativo. Por ejemplo, cuando el participante $p1$ termina el flujo de datos ($H1-1-fin$ es *True*), la invocación del adaptador $AppLf-H1-1-fin$ es el responsable de modificar la propiedad IMS LD $P-A1$ al valor *True*, correspondiente al participante $p1$.

3. LeadFlow4LD: El método propuesto

Fin de Herramienta	Adaptador del Flujo de Aprendizaje	Parámetros Actuales (Argumentos)	
		Property	UserList
<i>H1-1-fin</i>	<i>AppLf-H1-1-fin</i>	<i>P-A1</i>	<i>p1</i>
<i>H1-2-fin</i>	<i>AppLf-H1-2-fin</i>	<i>P-A1</i>	<i>p2</i>
<i>H2-1-fin</i>	<i>AppLf-H2-1-fin</i>	<i>P-A2</i>	<i>p1</i>
<i>H2-2-fin</i>	<i>AppLf-H2-2-fin</i>	<i>P-A2</i>	<i>p2</i>

Tabla 3.20: Características de la invocación de adaptadores del motor del flujo de aprendizaje, junto con sus argumentos, usadas para delegar el control al flujo de aprendizaje, para el ejemplo ilustrativo.

Para más detalles acerca de la implementación de los adaptadores usados en la invocación de los motores de flujo de aprendizaje y flujo de trabajo de acuerdo a la infraestructura tecnológica seleccionada, puede consultarse los apéndices D.5 y D.6.

3.6.4 Arquitectura del sistema prototipo de puesta en marcha

El componente analizador LeadFlow4LD no ha podido ser implementado en esta tesis por razones de tiempo. Sin embargo, para evaluar el método propuesto es necesario poner en marcha una UoLeadFlow4LD. Con este propósito, se ha creado un sistema prototipo (ver *Figura 3.29*) que puede ser usado como una prueba de concepto para ilustrar que la puesta en marcha de una UoLeadFlow4LD es posible. Sin embargo, el sistema prototipo tiene algunas limitaciones respecto del sistema “ideal” propuesto, las cuales se describen a continuación.

Como se puede ver en la *Figura 3.29*, en comparación con la arquitectura “ideal” propuesta en la *Figura 3.25*, el componente analizador LeadFlow4LD ha sido sustituido por el desarrollador, quien en este caso es el responsable de generar “manualmente” los documentos del flujo de aprendizaje y flujo de trabajo sincronizados para cada situación de aprendizaje. En general, el sistema prototipo es menos amigable para el usuario que la versión “ideal” propuesta en la *Figura 3.25*, ya que los participantes requerirán interactuar con interfaces de usuario tanto del motor de flujo de aprendizaje como de flujo de trabajo, en lugar de interactuar a través de una única interfaz de usuario de LeadFlow4LD.

Para más información sobre la implementación del sistema prototipo, incluyendo detalles del procedimiento “manual” que debe seguir el desarrollador para generar los documentos del flujo de aprendizaje y flujo de trabajo sincronizados; configuración del servidor de aplicaciones, herramientas de monitoreo de instancias de procesos, así como la implementación de adaptadores de los motores de flujo de aprendizaje y flujo de trabajo, puede consultarse el apéndice D. En el siguiente capítulo, diferentes situaciones de aprendizaje colaborativo se han puesto en marcha usando este sistema prototipo.

3. LeadFlow4LD: El método propuesto

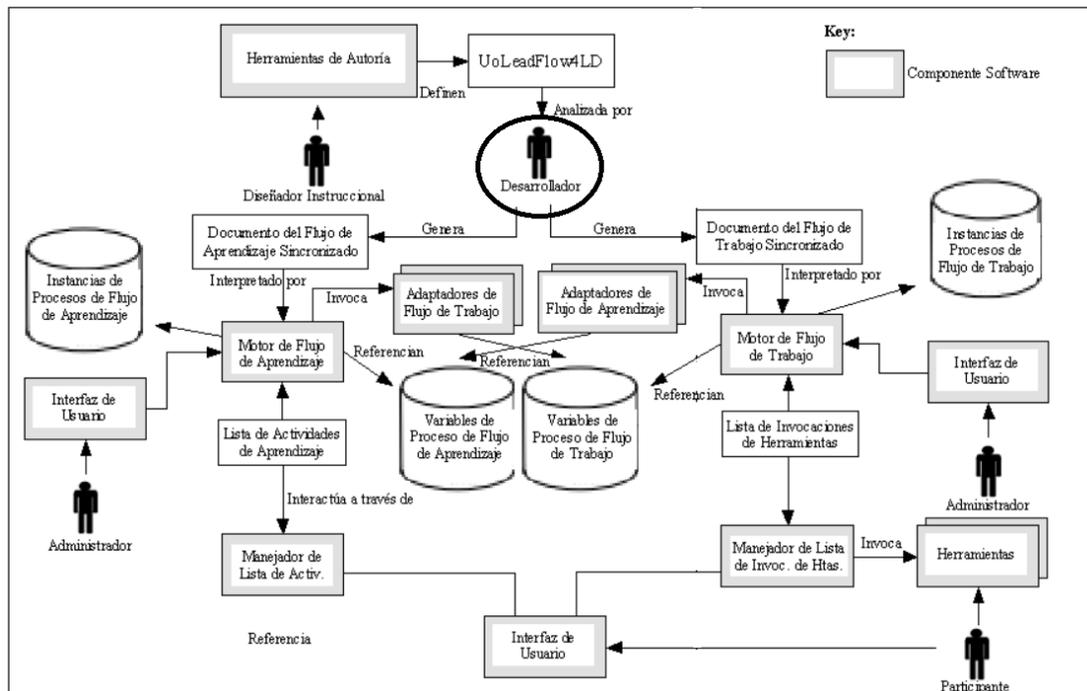


Figura 3.29: Arquitectura del sistema prototipo de puesta en marcha de una UoLeadFlow4LD. Al centro de la figura (encerrado en el círculo) se puede ver que el componente analizador LeadFlow4LD ha sido sustituido por el desarrollador.

3.6.5 Selección de una infraestructura tecnológica para un prototipo de puesta en marcha

Una infraestructura tecnológica ha sido seleccionada para implementación del sistema de puesta en marcha de una UoLeadFlow4LD. Por un lado, Coppercore (Vogten & Martens, 2005), como el motor de ejecución del flujo de aprendizaje capaz de interpretar el documento del flujo de aprendizaje sincronizado. Por otro lado, Bonita *Open Solution* (BOS) (Surhome, Tennoe & Henssonow, 2010), como el motor de ejecución de flujo de trabajo capaz de interpretar el documento del flujo de trabajo sincronizado. El primero, porque es el motor de referencia usado por la especificación IMS LD y el segundo, porque es de código abierto, estable y ampliamente usado por la comunidad de flujo de trabajo. Otras alternativas de motores de ejecución de flujo de trabajo de código abierto en sistemas de gestión de procesos de negocio son Intalio y ProcessMaker. Sin embargo, se optó por BOS, ya que a diferencia de Intalio y ProcessMaker, BOS está basado en Web, lo que facilita la integración de herramientas basadas en Web usadas por el entorno de aprendizaje.

Para la implementación de los adaptadores de cada uno de los motores, y que son usados para delegar el control de flujo, se utilizaron tecnologías REST y SOAP de servicios Web. Por un lado, para implementar los adaptadores del motor de flujo de trabajo BOS, se utilizó tecnología REST aprovechando el API REST que ofrece BOS. La justificación de esta

selección obedece a que las necesidades de este estilo de arquitectura (desempeño de sistemas dominado por comunicación en red e interacción de componentes basados en comunicación de datos, más que en realizar tareas de cómputo intensivo) (Fielding & Taylor, 2002), coincide con la necesidad de comunicar datos y con la posibilidad de que el sistema se convierta eventualmente en una aplicación Web, donde los motores de flujo pueden estar incluso geográficamente distribuidos. Por otro lado, para implementar los adaptadores del motor de flujo de aprendizaje (Coppercore), se utilizó tecnología SOAP, ya que Coppercore no define por ahora un API REST. Los requisitos para las herramientas de aprendizaje invocadas por el motor de flujo de trabajo (BOS) es la de aplicaciones Web invocadas a través de URLs, con la funcionalidad de persistencia de datos y con acceso individual y/o colaborativo.

3.6.6 Discusión

En esta sección se discutirán distintos aspectos relacionados con la especificación de una UoLeadFlow4LD, su puesta en marcha, y la factibilidad de utilizarla en producción. Además, se discutirán dificultades relacionadas con la naturaleza de flujo de trabajo del método propuesto.

Primero, está la aparente complejidad del método propuesto. A primera vista, podría parecer que cinco documentos para representar computacionalmente situaciones de aprendizaje colaborativo es demasiado, si se compara, por ejemplo, con un diseño de aprendizaje especificado en IMS LD, o con un proceso de flujo de trabajo especificado en XPDL, los cuales sólo requieren especificar un único documento. Sin embargo, se deben considerar distintas ventajas de especificar una UoLeadFlow4LD a través de distintos documentos. La primera, y la más importante en esta Tesis, es que no es posible especificar el flujo de aprendizaje y el flujo de datos en un único documento, ya sea especificado en IMS LD o en XPDL. Segundo, con LeadFlow4LD se especifica no sólo la estructura de aprendizaje colaborativo, sino también una situación de aprendizaje particular, analogía que no existe actualmente en los campos de LD o Flujo de Trabajo. A través de especificar separadamente un diseño de aprendizaje en documentos estructurales y situacionales, es posible que distintas situaciones de aprendizaje se puede derivar fácilmente de la misma estructura de aprendizaje colaborativo, reutilizando de distintas maneras y a distintos niveles los documentos que conforman una UoLeadFlow4LD. Y por último, y no por ello menos importante, la ventaja de mantener interoperabilidad con estándares actuales, tanto en LD como en Flujo de Trabajo.

También está la dificultad para el diseñador instruccional y el educador de especificar un diseño de aprendizaje conforme a LeadFlow4LD. Desafortunadamente al día de hoy, existe una falta de herramientas de autoría para especificar la mayoría de los documentos que conforman una UoLeadFlow4LD. Por un lado, para especificar los documentos del flujo de datos y de instanciación de grupos (instanciación del flujo de datos), la única posibilidad es utilizar herramientas de autoría propias del dominio de flujo de trabajo. Sin embargo, éstas no

3. LeadFlow4LD: El método propuesto

son adecuadas para ser usadas por diseñadores instruccionales y educadores. Por otro lado, para especificar los documentos de coordinación de flujos y de población de grupos, no existen en este momento herramientas de autoría disponibles, ya que dichas especificaciones son propietarias de LeadFlow4LD. Por tanto, para la especificación de una UoLeadFlow4LD por parte de diseñadores instruccionales y educadores, se requiere invertir en el desarrollo de herramientas de autoría para especificar los documentos del flujo de datos, de coordinación de flujos, y de instanciación y población de grupos. Un ejemplo claro en esta dirección es WebCollage (Villasclaras-Fernández, Hernández-Leo, Asensio-Pérez & Dimitriadis, 2013), la cual es una herramienta de autoría usada para especificar la instanciación de grupos, así como la asignación de participantes a grupos, aunque en un contexto del flujo de aprendizaje y no del flujo de datos.

También está la complejidad para poner en marcha una UoLeadFlow4LD, y por tanto, su dificultad de su aplicación en producción. La creación automática de los documentos del flujo de aprendizaje y flujo de trabajo sincronizados no es trivial, lo cual se puso en evidencia durante el procedimiento “manual” empleado para poner en marcha una UoLeadFlow4LD, ya que el componente analizador LeadFlow4LD, encargado de dicha tarea, no ha sido implementado todavía. Por tanto, para su uso en producción de una UoLeadFlow4LD, la implementación del componente analizador LeadFlow4LD es necesaria. Un ejemplo claro en esta dirección es Glue!PS (Alario-Hoyos, Muñoz-Cristóbal, Prieto-Santos, Bote-Lorenzo, Asensio-Pérez, Gómez-Sánchez, Vega-Gorgojo & Dimitriadis, 2012), el cual despliega diseños de aprendizaje que, por ejemplo, son el resultado de combinar documentos IMS LD y documentos de población de grupos en Moodle, Blackboard, LAMS, etc.

Otro ejemplo de la falta de infraestructura de puesta en marcha actual de una UoLeadFlow4LD, es que no soporta una forma automática de gestionar instancias de herramientas (creación, modificación, y eliminación de instancias de herramientas). Esto implica un incremento significativo en la carga de especificación cuando se trate de un número considerablemente grande de instancias de herramientas. Con el sistema prototipo actual, dichas instancias deben ser gestionadas manualmente. Sin embargo, actualmente se está planeando enriquecer la infraestructura de puesta en marcha de LeadFlow4LD con el sistema de integración Glue!, el cual también es propuesto por el grupo de investigación (Alario-Hoyos, Bote-Lorenzo, Gómez-Sánchez, Asensio-Pérez, Vega-Gorgojo & Ruiz-Callejas, 2013), de forma que sea factible la gestión automática de instancias de herramientas de terceros.

Por último, otras limitaciones identificadas son más bien limitaciones generales propias del paradigma de flujo de trabajo, tal como la dificultad de modificar el diseño de aprendizaje en tiempo de ejecución y la necesidad continua de crear nuevos estándares. Por un lado, está la dificultad de modificar un diseño de aprendizaje una vez que éste se ha puesto en marcha, lo cual es sin duda una desventaja en escenarios de aprendizaje reales. Por otro lado, está la dificultad para poner de acuerdo a toda una comunidad de práctica. Nuevas necesidades en el modelado de procesos, implican nuevos elementos del meta-modelo, por tanto nuevos

lenguajes de modelado (o modificación de los existentes) y la necesidad de nuevos estándares.

3.7 Conclusiones

En este capítulo se propuso una solución al problema de especificar el flujo de datos a nivel de herramientas en LD. Partiendo del problema identificado en el capítulo anterior y de la revisión del estado del arte, se plantearon los objetivos que debería cumplir la solución, y entonces se propuso un espacio de soluciones para resolver el problema. Después de evaluar la mejor solución y de seleccionar el lenguaje de flujo de trabajo que sería usado para especificar el flujo de datos (XPDL), se propuso un método para resolver el problema basado en la aproximación por composición del flujo de aprendizaje y el flujo de datos. LeadFlow4LD es un método usado para describir estructuras y situaciones de aprendizaje colaborativo basado en especificar el flujo de aprendizaje usando el estándar de LD (IMS LD), y el flujo de datos usando un lenguaje estándar de flujo de trabajo (XPDL).

El método se describe desde la perspectiva del diseñador instruccional y del educador y se compone de dos partes principales: la definición de la estructura de aprendizaje colaborativo y la definición de la instanciación de dicha estructura. Por un lado, la estructura de aprendizaje colaborativo requiere la especificación separada de la estructura del flujo de aprendizaje (en el documento del flujo de aprendizaje), la del flujo de datos (en el documento del flujo de datos) y la de la coordinación entre ambas estructuras (en el documento de coordinación de flujos). Por otro lado, la especificación de la instanciación de la estructura de aprendizaje colaborativo se lleva a cabo a dos niveles de abstracción: estructural y situacional. A nivel estructural, LeadFlow4LD propone la especificación de una instancia de la estructura del flujo de datos (en el documento de instanciación de grupos), la cual es necesaria para describir la interacción colaborativa del flujo de datos entre los grupos participantes.

Una instancia de la estructura del flujo de datos representa en realidad una situación de aprendizaje genérica, ya que solo especifica el número de instancias de datos y herramientas (de los grupos participantes), así como la interacción entre éstas (interacción entre grupos, es decir, qué grupo revisa el artefacto de qué grupo). Sin embargo, no especifica información sobre cómo se distribuyen los participantes en los grupos. Por tanto, distintas situaciones de aprendizaje concretas se pueden derivar de la misma situación de aprendizaje genérica, en función de cómo se asignen participantes concretos a los grupos. Esto es precisamente lo que se especifica a nivel situacional, pero no solo para el flujo de datos, sino también para el flujo de aprendizaje. De esta forma, en el documento de población de grupos se especifica la asignación de participantes a las actividades de aprendizaje y a las instancias de herramientas.

Además, con el objetivo de permitir el intercambio entre educadores y entre herramientas, no solo de la estructura de aprendizaje colaborativo, sino también de la situación de aprendizaje concreta, LeadFlow4LD propone una especificación propietaria para

3. LeadFlow4LD: El método propuesto

especificar tales situaciones de aprendizaje. El documento usado para este propósito es llamado el documento de población de grupos. La necesidad de LeadFlow4LD de incluir documentos con especificaciones propietarias, es una desventaja del método propuesto. Sin embargo, aún con esto, algunos de los documentos de una UoLeadFlow4LD mantienen interoperabilidad con estándares actuales. Además, se contribuye a especificar la instanciación de procesos de flujo de trabajo en un contexto educativo, lo cual no es soportado actualmente ni en el campo LD ni el campo de Flujo de Trabajo.

Por último, después de revisar distintas aproximaciones de implementación para poner en marcha una situación de aprendizaje especificada con el método propuesto, se seleccionó la aproximación de implementación flujo de aprendizaje maestro y flujo de trabajo esclavo y se propuso un sistema y procedimiento de puesta en marcha basado en esta aproximación. La idea es usar el motor de flujo de aprendizaje como motor maestro y el motor de flujo de trabajo como esclavo. Entonces, a partir de una UoLeadFlow4LD y un procedimiento definido, un componente software llamado analizador LeadFlow4LD puede generar automáticamente los así llamados documentos del flujo de aprendizaje y de flujo de trabajo sincronizados. La ejecución independiente de ambos documentos por motores conformes a IMS LD y XPDL, respectivamente, da como resultado la puesta en marcha de una UoLeadFlow4LD.

Aunque a la fecha no se ha implementado el componente analizador LeadFlow4LD, se ha propuesto un procedimiento de cómo dicho componente debería de generar automáticamente los documentos del flujo de aprendizaje y de flujo de trabajo sincronizados. En el siguiente capítulo se presenta la evaluación del método propuesto.

4 Evaluación de LeadFlow4LD

En el capítulo 2 se planteó el problema de especificar con IMS LD el flujo de datos en procesos de aprendizaje colaborativo, y se propuso una primera aproximación para su solución: aproximación por composición de LD y Flujo de Trabajo. A continuación, en el capítulo 3 se propuso un método para resolver el problema planteado basado en la aproximación mencionada y que fue llamado LeadFlow4LD. En este capítulo se presentan los resultados de evaluar el método propuesto. Son tres los objetivos que se busca evaluar de LeadFlow4LD. Primero, la expresividad para especificar la automatización del flujo de datos en estructuras y situaciones de aprendizaje colaborativo. Esto permitiría su interpretación por ordenador para automatizar la invocación de herramientas y los datos de entrada y salida de las herramientas, así como la secuenciación de invocaciones de herramientas en una misma actividad de aprendizaje. Segundo, evaluar la reutilización de un diseño de aprendizaje especificado conforme al método propuesto. Esto permitiría la reutilización de una UoLeadFlow4LD para distintas situaciones y contextos. Y tercero, evaluar la interoperabilidad del método propuesto con estándares actuales. Para alcanzar estos objetivos se utiliza una metodología de evaluación basada en ejemplos, que es reconocida como buena práctica en el campo de la ingeniería de software (Shaw, 2002). El proceso de evaluación selecciona primero un conjunto de situaciones de aprendizaje colaborativo significativas no triviales, y que están basadas en distintos patrones de procesos o técnicas de aprendizaje colaborativo. Estas situaciones de aprendizaje seleccionadas son justificadas por los objetivos del método propuesto de expresividad para especificar el flujo de datos y reutilización del diseño de aprendizaje resultante. Después, cada una de las situaciones de aprendizaje colaborativo se especifica conforme a LeadFlow4LD y se ponen en marcha, utilizando para ello el sistema prototipo desarrollado con este propósito y descrito en el capítulo anterior. Entonces, se discute la capacidad de LeadFlow4LD para automatizar el flujo de datos en cada una de las situaciones de aprendizaje, así como la reutilización del diseño resultante y se compara con otras soluciones alternativas. Por último, se discute globalmente hasta qué punto es posible afirmar que la solución es expresiva y reutilizable, y se enumeran las ventajas y desventajas de la solución propuesta.

4.1 Introducción

En el capítulo 2 se discutió el problema de representar computacionalmente el flujo de datos a nivel de herramientas en procesos y situaciones de aprendizaje colaborativo, identificando los principales objetivos que una solución basada en flujo de trabajo debería de cumplir, y que coinciden con los objetivos de esta tesis: automatización del proceso, reutilización de la definición del proceso e interoperabilidad con sistemas y herramientas. Como resultado, en el capítulo 3 se propuso una solución interoperable con LD y Flujo de Trabajo llamada LeadFlow4LD, la cual está basada en la composición del flujo de aprendizaje especificado en IMS LD y el flujo de datos especificado en XPDL. Un diseño de aprendizaje especificado conforme a LeadFlow4LD separa la especificación de la estructura de aprendizaje colaborativo de su instanciación. La estructura de aprendizaje colaborativo describe el proceso de aprendizaje, mientras que una instancia de dicha estructura, describe una situación de aprendizaje colaborativo. De esta forma, es razonable que a través de especificar el flujo

de datos se consiga el objetivo de automatización del proceso; que a través de especificar separadamente la estructura del proceso de aprendizaje de una instancia de dicha estructura, se consiga el objetivo de reutilización del proceso; y que a través de la utilización de lenguajes estándares de LD y flujo de trabajo, se consiga el objetivo de interoperabilidad con sistemas y con herramientas. A continuación, se describe la metodología particular empleada para la evaluación de LeadFlow4LD.

4.2 Metodología de evaluación

Los resultados de investigación en ciencias e ingenierías bien establecidas recaen en el proceso de cómo se obtienen los resultados y en la validación de los mismos. Sin embargo, en ingeniería de software es permanente la preocupación por cómo enfocar los procesos de investigación, incluyendo los métodos de validación (Shaw, 2002), lo cual se puede deber a la naturaleza y complejidad propias de la disciplina (Brooks, 1987).

Mientras algunos autores como Zelkowitz y Wallace (1997, 1998), Tichy et al. (1995), y Tichy (1998), critican la falta de validación experimental en las estrategias de investigación usadas en ingeniería de software, en comparación con el paradigma experimental clásico que siguen otras ingenierías, Shaw (2002) exploró las estrategias de investigación que el campo de la ingeniería de software reconoce como buenas prácticas. Así, Shaw propone un modelo basado en estrategias de investigación que han mostrado ser exitosas en el campo de la ingeniería de software. El modelo se basa en un estudio que clasifica los tipos de preguntas de investigación realizadas, los tipos de resultados producidos, y el tipo de validación obtenida, mismo que se resume en la *Tabla 4.1*.

Así por ejemplo, la *Tabla 4.1(a)*, agrupa los tipos de preguntas de investigación más comunes identificadas en el campo de la ingeniería de software, dentro del marco de estudio realizado por Shaw. En dicho estudio, Shaw identificó que las preguntas de investigación más comunes están relacionadas con mejoras de los métodos de desarrollo software y de análisis de exactitud del software (pruebas y verificación). Por ejemplo, ¿cómo crear X?, o ¿cómo evaluar la calidad de X? En dicho estudio, Shaw identificó también que los tipos de preguntas cambian conforme el campo madura. Así por ejemplo, conforme pasa el tiempo en un área de investigación, es más común encontrar investigación basada en métodos formales que investigación orientada solamente a la factibilidad de tareas (dado X, ¿es posible implementar X?).

La *Tabla 4.1(b)*, agrupa los resultados de investigación más comunes identificados en el campo de la ingeniería de software, también dentro del marco de estudio realizado por Shaw. Shaw identificó que los resultados más comunes son los que describen nuevos procedimientos de desarrollo o análisis software, así como los que describen modelos analíticos y descriptivos. Los resultados pueden ser tanto generales representados por un modelo (el cual puede ser de muchos grados de precisión y formalidad), o pueden ser la solución a un problema o análisis específico. Resultados basados en modelos empíricos y

4. Evaluación de LeadFlow4LD

patrones que aparecen repetidamente en la literatura. En la *Tabla 4.2* se muestran un listado de buenas estrategias de investigación, de acuerdo al estudio realizado por Shaw, donde se combinan las preguntas de investigación, los resultados obtenidos y los tipos de validación más comunes encontrados en la literatura de ingeniería de software.

Pregunta	Resultado	Validación
Método de desarrollo	Procedimiento	Análisis
Método de desarrollo	Procedimiento	Experiencia
Método de desarrollo	Procedimiento	Ejemplo
Método de desarrollo	Modelo cualitativo	Experiencia
Método de desarrollo	Modelo cualitativo	Persuasión
Método de desarrollo	Modelo analítico	Experiencia
Método de desarrollo	Notación o herramienta	Análisis
Método de desarrollo	Notación o herramienta	Experiencia
Método de desarrollo	Notación o herramienta	Ejemplo
Método de análisis	Procedimiento	Análisis
Método de análisis	Procedimiento	Experiencia
Método de análisis	Procedimiento	Ejemplo
Método de análisis	Modelo analítico	Análisis
Método de análisis	Modelo analítico	Experiencia
Método de análisis	Modelo analítico	Ejemplo
Método de análisis	Herramienta	Ejemplo
Evaluación de instancia	Análisis específico	Análisis
Evaluación de instancia	Análisis específico	Ejemplo
Evaluación de instancia	Respuesta	Análisis

Tabla 4.2: Lista de buenas estrategias de investigación en ingeniería de software. Estudio realizado por Shaw (2002). En negrilla se ubican las estrategias de investigación usadas en esta tesis en el marco de buenas prácticas del estudio realizado por Shaw.

Estos patrones o buenas estrategias de investigación fueron identificados, según Shaw, a partir de explorar los resultados individuales de los artículos consultados durante su estudio. Sin embargo, los principales resultados se obtuvieron, según Shaw, con el paso del tiempo, conforme artículos sucesivos mejoraban progresivamente la calidad de sus resultados y mejoraban en credibilidad. Por tanto, Shaw concluye que evaluar el significado de resultados en el campo de la ingeniería de software, debe hacerse en este contexto mayor, conforme madura el área de investigación. Por tanto, reportes iniciales en el área pueden ser informales y cualitativos, presentando validaciones persuasivas para investigación exploratoria, mientras reportes posteriores presentan modelos experimentales y hasta alcanzar modelos formales. Este patrón de crecimiento es consistente, según Shaw, con el modelo de madurez de tecnologías de la ingeniería de software propuesto por Redwine et al. (1984). Ellos encontraron que típicamente le toma a una tecnología de software de 15-20 años evolucionar desde la formulación del concepto hasta el punto donde la tecnología está lista para popularización. Por tanto, concluye Shaw, una estrategia de investigación debe tomar en cuenta la acumulación de evidencia en el tiempo.

En esta tesis, usaremos el marco del estudio realizado por Shaw para evaluar LeadFlow4LD. El primer paso de la metodología de evaluación es identificar dentro del marco de buenas estrategias de investigación de Shaw (2002), y que se muestra en la *Tabla*

4.2, el tipo de investigación llevada a cabo en esta tesis. Así, dentro de este marco, el primer tipo de pregunta de investigación llevada a cabo en esta tesis es una del tipo de método de desarrollo: ¿cómo se puede automatizar el flujo de datos a nivel de herramientas en un diseño de aprendizaje? Además de una segunda pregunta de investigación del tipo de método de análisis: ¿cómo se puede evaluar la reutilización e interoperabilidad de un diseño de aprendizaje, en escenarios de aprendizaje colaborativo? Por otra parte, el resultado obtenido en esta tesis, dentro de este mismo marco, es uno del tipo procedimiento: LeadFlow4LD es un método para especificar el flujo de aprendizaje y el flujo de datos a nivel de herramientas en escenarios de aprendizaje colaborativo.

Finalmente, el tipo de validación usada en esta tesis, también dentro del marco descrito por Shaw, será uno basado en ejemplos. La validación basada en ejemplos puede considerarse una buena validación inicial de LeadFlow4LD en el campo de la ingeniería de software, tomando en cuenta la poca madurez del problema del flujo de datos en diseño de aprendizaje (Palomino-Ramírez, Martínez-Monés, Bote-Lorenzo, Asensio-Pérez, & Dimitriadis, 2007). Más aún, tomando en cuenta que no se abordarán ejemplos triviales (*toy examples*, como refiere Shaw), sino más bien ejemplos bien documentados en la literatura, compuestos de una diversidad de técnicas bien conocidas de aprendizaje colaborativo, y que van de lo simple a lo complejo, aunque a decir verdad realizados en un entorno controlado.

Esta metodología propuesta para evaluar LeadFlow4LD, basada en el marco de Shaw, es contrastada a continuación con ideas de (Kitchenham, Linkman & Law, 1997) sobre la evaluación de metodologías y herramientas en ingeniería de software. Kitchenham clasifica los ejercicios de evaluación de métodos y herramientas en ingeniería de software en dos tipos: cuantitativos y cualitativos, además de la dimensión que describe la forma en que se llevará a cabo la evaluación: experimento formal, caso de estudio y encuesta. De acuerdo con Kitchenham, los métodos cuantitativos tienen como objetivo establecer efectos medibles de usar un método o herramienta, mientras que los métodos cualitativos tienen como objetivo establecer si el método o herramienta es el adecuado a través de análisis. En un experimento formal, a los distintos desarrolladores se les pide realizar una tarea (o variedad de tareas) usando los diferentes métodos ó herramientas que están bajo evaluación. En un caso de estudio, cada método o herramienta bajo evaluación es tratado como un proyecto real basado en el procedimiento de desarrollo estándar usado en el campo. Por último, en una encuesta, los usuarios que han usado un método o herramienta específicos en otros proyectos, se les pide información sobre el método o herramienta que está bajo evaluación.

De acuerdo con Kitchenham, las metodologías cuyo impacto no puede ser medido directamente en un solo proyecto individual (por ejemplo, un único diseño de aprendizaje en el caso de LeadFlow4LD), no pueden ser evaluadas a través de casos de estudio cuantitativos, ya que una salida individual no puede ser usada para dar medidas cuantitativas de probabilidad. Por otra parte, también según Kitchenham, las encuestas pueden ser usadas para evaluar metodologías donde el impacto medible es indirecto, pero sólo si las metodologías son de uso extendido (lo cual no es el caso de LeadFlow4LD). Por último, de acuerdo con

Kitchenham, los experimentos formales pueden ser descartados cuando es difícil identificar un caso de estudio que actúe como base para llevar a cabo las comparaciones (como puede ser la selección de un único diseño de aprendizaje a ser especificado conforme a LeadFlow4LD), así como cuando la cantidad de elementos de la metodología impida que se ajuste bien a un experimento formal (por ejemplo, la complejidad inherente de LeadFlow4LD).

Por tanto, de acuerdo con estas ideas de Kitchenham usadas para evaluar metodologías y herramientas en el campo de la ingeniería de software, una metodología de análisis cualitativo basado en casos de estudio puede ser un método razonable para evaluar LeadFlow4LD, lo cual está alineado con la metodología de evaluación propuesta en el marco de Shaw, si sustituimos los casos de estudio con participantes reales por ejemplos significativos no triviales y participantes simulados o ficticios, tomando en cuenta la novedad de la propuesta y la poca madurez del campo. La validación de LeadFlow4LD basada en casos de estudio reales realizados por terceros (en lugar de participantes simulados o ficticios), se deja para un trabajo futuro de esta tesis.

El segundo paso de la metodología de evaluación de esta tesis es seleccionar y justificar los ejemplos que serán usados para evaluar LeadFlow4LD. Estos ejemplos incluyen tanto las situaciones de aprendizaje colaborativo que serán usadas para ilustrar las bondades del método propuesto, en lo que respecta a la expresividad de especificar la automatización del flujo de datos y la reusabilidad del diseño de aprendizaje resultante, así como las técnicas de aprendizaje colaborativo de las que se derivan dichas situaciones. El criterio de interoperabilidad a evaluar no tiene implicación en la selección de unas u otras situaciones y técnicas de aprendizaje.

El siguiente paso de la evaluación es la especificación con el método propuesto, tanto de la estructura de aprendizaje colaborativo como de instancias de dicha estructura, para cada una de las situaciones de aprendizaje seleccionadas. Después, se usa el sistema prototipo desarrollado en esta tesis para poner en marcha las distintas situaciones de aprendizaje. El propósito aquí es observar y discutir las bondades y posibles limitaciones de LeadFlow4LD al poner en marcha las distintas situaciones de aprendizaje, con respecto a su capacidad de automatizar el flujo de datos, con respecto al nivel de reutilización del diseño de aprendizaje resultante, y con respecto al nivel de interoperabilidad del diseño de aprendizaje con sistemas y herramientas de estándares actuales. A continuación, se compara la solución obtenida con otras posibles soluciones, para de esta forma mejorar el entendimiento del mérito obtenido, como sugiere Young (1993). Por último, se discute globalmente hasta qué punto es posible afirmar que el método propuesto es suficientemente expresivo para automatizar el flujo de datos, hasta qué punto es posible la reusabilidad e interoperabilidad de los diseño de aprendizaje resultantes, y además de resumir las ventajas y desventajas de la solución propuesta.

4.3 Selección y justificación de situaciones de aprendizaje

Para la selección de situaciones de aprendizaje colaborativo usadas en la evaluación de LeadFlow4LD, se tomó en cuenta dos de los tres objetivos de esta tesis. Primero, la expresividad de LeadFlow4LD para representar computacionalmente el flujo de datos, incluyendo la invocación automática de herramientas, la automatización de los datos de entrada y salida de las herramientas, así como la secuenciación automática de herramientas en una misma actividad de aprendizaje. Y segundo, la reutilización del diseño de aprendizaje resultante (UoLeadFlow4LD) en distintas situaciones de aprendizaje colaborativo y contextos. La interoperabilidad de un diseño de aprendizaje con estándares actuales, que es el tercer objetivo de esta tesis, como se comentó anteriormente, no tiene implicación directa en la selección de las situaciones y técnicas de aprendizaje colaborativo usadas en la evaluación de LeadFlow4LD.

Todas las situaciones de aprendizaje colaborativo seleccionadas pertenecen a distintos patrones de procesos o técnicas de aprendizaje colaborativo que están bien documentados en la literatura, los cuales van de menor a mayor complejidad, con el propósito, por un lado, de facilitar el entendimiento de la propuesta, y por otro lado, de ilustrar el alcance o capacidad expresiva de la solución propuesta, esto es, mostrar que LeadFlow4LD no está limitado solamente a la especificación de ejemplos triviales (*toy examples*). Un resumen de las situaciones de aprendizaje colaborativo seleccionadas en la evaluación de LeadFlow4LD se muestra en la *Tabla 4.3*. En esta tabla se distingue entre las características que se busca ilustrar en una cierta situación de aprendizaje, de aquellas que aunque son soportadas por LeadFlow4LD no son detalladas en el ejemplo, o bien, no son aplicables para el ejemplo. Note que para todas las situaciones de aprendizaje seleccionadas, ninguna de las características, ya sea de automatización del flujo de datos, o de reutilización del diseño, es no soportada por LeadFlow4LD.

El primer ejemplo seleccionado, es un patrón simple pero representativo de escenarios de aprendizaje colaborativo: la técnica de aprendizaje colaborativo de revisión entre pares (Bartels, 2003) (Barkley, Cross & Major, 2005). De esta técnica fueron seleccionadas cuatro situaciones de aprendizaje. Dos de ellas tienen el propósito de ilustrar la automatización del flujo de datos, particularmente, la automatización de la invocación de herramientas, así como los datos de entrada y salida de las herramientas. De hecho, la primera de estas situaciones fue usada como ejemplo ilustrativo de la propuesta en el capítulo anterior. Las otras dos situaciones fueron seleccionadas para ilustrar la reutilización del diseño de aprendizaje resultante.

El segundo ejemplo seleccionado, es un patrón de proceso de aprendizaje colaborativo llamado convergencia del conocimiento (Weinberger, Stegmann & Fischer, 2005). Este patrón se puede entender como una generalización de la técnica de revisión entre pares (por lo que es más complejo que el primero). De esta técnica se seleccionó una situación de

4. Evaluación de LeadFlow4LD

aprendizaje que tiene el propósito de ilustrar la expresividad de LeadFlow4LD para especificar la secuenciación de herramientas en una misma actividad de aprendizaje.

El tercer y último ejemplo seleccionado es el llamado caso MOSAIC . El caso MOSAIC (Palomino-Ramírez, Bote-Lorenzo, Asensio-Pérez, Dimitriadis & de la Fuente-Valentín, 2008), es una situación de aprendizaje colaborativo basada en una jerarquía de técnicas de aprendizaje colaborativo que son bien conocidas: la técnica de aprendizaje colaborativo pirámide (Hernández-Leo, 2007), la técnica de revisión entre pares antes mencionada, y la técnica jigsaw (Barkley, Cross & Major, 2005). El propósito de usar el caso MOSAIC es ilustrar, por un lado, la expresividad de LeadFlow4LD para especificar el flujo de datos, lo cual incluye la invocación automática de herramientas, la automatización de los datos de entrada y salida de las herramientas, y la secuenciación automática de herramientas en una misma actividad de aprendizaje. Y por otro lado, para ilustrar la aplicación del método propuesto en escenarios de aprendizaje colaborativo complejos (no triviales).

Técnica de Aprendizaje Colaborativo	Sit. ID	Característica de la Expresividad de LeadFlow4LD			Reutilización de una UoLeadFlow4LD (Reutilización del Diseño de Aprendizaje Resultante)
		Automatización del Flujo de Datos			
		Invocación Automática de Herramientas	Automatización de los Datos de Entrada y Salida de las Herramientas	Secuenciación de Herramientas en una misma Actividad de Aprendizaje	
1. Revisión entre Pares	1.1	✓	✓	○	○
	1.2	✓	✓	○	○
	1.3	-	-	○	✓
	1.4	-	-	○	✓
2. Convergencia del Conocimiento	2.1			✓	○
		-	-		
3. MOSAIC	3.1	✓	✓	✓	-

Tabla 4.3: Características de LeadFlow4LD que se busca ilustrar con cada una de las situaciones de aprendizaje. Claves: ✓ Característica soportada por LeadFlow4LD, y además es el propósito del ejemplo. – Característica soportada por LeadFlow4LD, pero no es detallada en el ejemplo, por no ser su principal propósito. ○ característica soportada por LeadFlow4LD, pero no es aplicable para el ejemplo. ✗ Característica no soportada por LeadFlow4LD para el ejemplo.

Una vez justificadas las situaciones de aprendizaje colaborativo usadas en la evaluación del método propuesto, a continuación se describe su especificación y puesta en marcha, a través del sistema prototipo desarrollado para este propósito y descrito en el capítulo anterior. Entonces, como resultado de su puesta en marcha, se discute la expresividad de LeadFlow4LD y reusabilidad de una UoLeadFlow4LD.

4.4 Ejemplo ilustrativo: revisión entre pares

Como se mencionó en el capítulo 2, la técnica de aprendizaje colaborativo de revisión entre pares, es un típico proceso de aprendizaje colaborativo en el que estudiantes individuales o grupos de estudiantes editan artefactos con el soporte de herramientas individuales o colaborativas. Después, los artefactos así creados, ya sean individuales o compartidos, son

revisados de forma cruzada por los estudiantes participantes. En la *Tabla 4.4* se describen cuatro situaciones de aprendizaje concretas derivadas de esta técnica y que han sido seleccionadas con el propósito de evaluar, del método propuesto, la invocación automática de herramientas, la automatización del flujo de datos a nivel de herramientas, y la reusabilidad del diseño de aprendizaje resultante.

Por una parte, en las situaciones de aprendizaje 1.1 y 1.2 tiene lugar una típica revisión cruzada entre pares. Mientras en la situación 1.1 participan estudiantes en actividades individuales soportadas por herramientas individuales (la colaboración se produce porque cada estudiante revisa el trabajo de su par), en la situación 1.2 participan parejas de estudiantes realizando actividades colaborativas y soportadas por herramientas colaborativas. Es importante darse cuenta, que en cada actividad de aprendizaje, las herramientas que soportan la actividad deben ser invocadas automáticamente, esto es, los estudiantes no deben preocuparse de localizar y lanzar las herramientas. Además, los estudiantes tampoco deben preocuparse de almacenar o recuperar los artefactos creados por las herramientas. Así, durante la actividad de revisión, la herramienta de revisión, de cada participante o grupo de participantes, debe estar precargada con el artefacto creado por su par en la actividad anterior, lo cual es el principal propósito a ilustrar en estas dos situaciones de aprendizaje.

Por otra parte, en las situaciones de aprendizaje 1.3 y 1.4 también ocurre revisión cruzada entre pares, pero en esta ocasión participan 3 estudiantes en actividades individuales soportadas por herramientas individuales y 3 parejas de estudiantes en actividades colaborativas soportadas por herramientas colaborativas, respectivamente. El propósito de seleccionar estas dos situaciones de aprendizaje es ilustrar, que situaciones de aprendizaje similares, basadas en la misma estructura (mismo patrón de aprendizaje colaborativo y misma instancia de la estructura del flujo de datos) se pueden reutilizar muy fácilmente con LeadFlow4LD.

4.4.1 Especificación de la estructura de aprendizaje colaborativo

Puesto que las situaciones de aprendizaje colaborativo 1.1 a 1.4 se derivan de la misma técnica de aprendizaje colaborativo (el patrón del proceso de revisión entre pares), todas estas situaciones comparten la misma estructura de aprendizaje colaborativo (ver *Figura 3.10*) y por tanto comparten los mismos documentos estructurales: el documento del flujo de aprendizaje, el documento del flujo de datos y el documento de coordinación de flujos. Todos estos tres documentos estructurales ya han sido especificados anteriormente, ya que corresponden al ejemplo ilustrativo analizado y discutido en el capítulo anterior.

Situación	Descripción
1.1	Dos estudiantes editan individualmente un artefacto y después cada estudiante revisa el artefacto creado por su par.
1.2	Cuatro estudiantes editan en parejas su propio artefacto y después cada pareja revisa colaborativamente el artefacto creado por su par.
1.3	Tres estudiantes editan individualmente un artefacto y después el primer estudiante revisa el artefacto creado por el segundo estudiante, el segundo estudiante revisa el del tercero y el tercero el del primero.
1.4	Seis estudiantes editan en parejas sus propios artefactos y después, también en parejas pero de distintos estudiantes, revisan colaborativamente el artefacto del tercero.

Tabla 4.4: Descripción de situaciones de aprendizaje colaborativo seleccionadas para la evaluación de LeadFlow4LD, derivadas de la técnica de aprendizaje colaborativo de revisión entre pares.

4.4.2 Especificación de la instanciación de la estructura

Para especificar las situaciones de aprendizaje colaborativo 1.1 a 1.4 derivadas de la técnica de aprendizaje colaborativo de revisión entre pares, se deben especificar una o más instancias de la estructura del flujo de datos mostrada en la *Figura 3.10*, lo cual se lleva a cabo en dos documentos, el documento de instanciación de grupos y el documento de población de grupos.

Documentos de Instanciación de Grupos

Para especificar las situaciones de aprendizaje 1.1 y 1.2 se requiere definir, a partir de la estructura del flujo de datos de la *Figura 3.10*, dos instancias tanto del artefacto de edición (*D1*) como de las herramientas de edición (*H1*) y de revisión (*H2*), para de esta forma producir las interacciones colaborativas entre datos y herramientas que caracterizan a ambas situaciones de aprendizaje. Esto se refleja en el diseño de la instancia de la estructura del flujo de datos realizada para el ejemplo ilustrativo en el capítulo anterior y mostrada en la *Figura 3.17a*.

De forma similar, para especificar las situaciones de aprendizaje 1.3 y 1.4 a partir de la estructura del flujo de datos de la *Figura 3.10*, se requiere definir tres instancias de datos y herramientas como se muestra en la *Figura 3.17b*. Aquí, el diseñador instruccional ha definido tres instancias tanto del artefacto de edición (*D1*) como de las herramientas de edición (*H1*) y de revisión (*H2*), mientras que las interacciones colaborativas entre instancias de datos y herramientas son las que caracterizan a ambas situaciones de aprendizaje. Los documentos de instanciación de grupos que describen las cuatro situaciones de aprendizaje se pueden consultar en el apéndice C.

Documentos de Población de Grupos

La especificación concreta de las situaciones de aprendizaje 1.1 a 1.4, corresponde al educador y consiste en la asignación de participantes tanto a las actividades de aprendizaje como a las instancias de herramientas. Un resumen de esta asignación se muestra en la *Tabla 4.5*.

Situación (Id)	Roles del Flujo de Aprendizaje		Instancias de Herramientas del Documento de Instanciación de Grupos					
	Editor	Revisor	H1-1	H1-2	H1-3	H2-1	H2-2	H2-3
1.1	p1,p2	p1,p2	p1	p2	-	p1	p2	-
1.2	p1,p2,p3,p4	p1,p2,p3,p4	p1,p2	p3,p4	-	p1,p2	p3,p4	-
1.3	p1,p2,p3	p1,p2,p3	p1	p2	p3	p1	p2	p3
1.4	p1,p2,p3, p4,p5,p6	p1,p2,p3, p4,p5,p6	p1,p2	p3,p4	p5,p6	p1,p5	p2,p3	p4,p6

Tabla 4.5: Asignación de participantes a las actividades de aprendizaje y a las instancias de herramientas, correspondientes a las situaciones de aprendizaje 1.1-1.4.

Como podemos ver en la *Tabla 4.5*, las cuatro situaciones de aprendizaje colaborativo requieren que todos los participantes se desempeñen en los roles editor y revisor, lo que significa que todos los participantes intervienen tanto en las actividades de edición como en las de revisión. Así también, la *Tabla 4.5* muestra que los participantes son asignados a instancias de herramientas individuales o colaborativas. Por ejemplo en la situación de aprendizaje 1.1, el participante *p1* es asignado a la instancia de la herramienta de edición individual *H2-1*, mientras que en la situación 1.4, los participantes *p5* y *p6* son asignados para compartir la instancia de la herramienta de edición colaborativa *H1-3*. Los documentos de población de grupos de las cuatro situaciones de aprendizaje se pueden consultar en el apéndice C.

4.4.3 Puesta en marcha de las situaciones de aprendizaje

En esta sección se describe la puesta en marcha de las situaciones de aprendizaje 1.2 a 1.4, ya que la puesta en marcha de la situación 1.1 es usada como ejemplo y detallada en el apéndice D. Es importante darse cuenta de lo siguiente. Primero, que para la puesta en marcha de todas y cada una de las situaciones de aprendizaje se ha usado el sistema prototipo, el cual tiene algunas limitaciones respecto del sistema “ideal” propuesto (ver secciones 3.6.1 y 3.6.5). Por ejemplo, que los participantes deben usar distintas interfaces para interactuar con los motores del flujo de aprendizaje y flujo de trabajo, en lugar de usar una única interfaz amigable. Segundo, que la puesta en marcha y discusión de cada situación de aprendizaje tiene un principal objetivo. Mientras que el principal objetivo de las situaciones 1.1 y 1.2 es ilustrar la expresividad de LeadFlow4LD para especificar la automatización del flujo de datos (invocación automática de herramientas y automatización de los datos de entrada y salida de herramientas), el principal objetivo de las situaciones de aprendizaje 1.3 y 1.4 es ilustrar la reusabilidad del diseño de aprendizaje resultante. Por ello, se debe esperar que sólo se ilustren ciertos momentos de cada puesta en marcha.

Automatización del Flujo de Datos

La puesta en marcha de la situación de aprendizaje 1.2 se describe a continuación. Cuando el participante *p1* inicia el flujo de datos desde la actividad de edición, éste no tiene que esperar a que su compañero, el participante *p2*, inicie también el flujo de datos de la actividad de edición. En tal caso, el motor del flujo de aprendizaje habilita al motor de flujo

4. Evaluación de LeadFlow4LD

de trabajo a iniciar el flujo de datos correspondiente a dicha actividad de aprendizaje. Éste es el momento en que el motor de flujo de trabajo lanza automáticamente, de acuerdo a lo especificado en el flujo de datos, la pizarra colaborativa del participante $p1$. Después, cuando el participante $p2$ inicia el flujo de datos de la actividad de edición, la misma instancia de la pizarra colaborativa del participante $p1$ es compartida ahora con el participante $p2$.

Es importante darse cuenta de que esta instancia de la pizarra colaborativa es invocada automáticamente por el motor de flujo de trabajo, pero que sin embargo, puede ser cerrada, relanzada o terminada a petición de cada participante. Aquí se debe distinguir la diferencia entre cerrar y terminar una herramienta. Mientras un participante puede cerrar una herramienta para relanzarla más tarde, también es posible terminar la herramienta, para advertir así al motor de flujo de trabajo que continúe con la secuencia de herramientas especificadas en el flujo de datos, pero siempre con relación a la misma actividad de aprendizaje, o bien, terminar el flujo de datos de la actividad, si es que la herramienta terminada fue la última herramienta de la secuencia, y en tal caso, devolver el control al motor del flujo de aprendizaje.

Un participante puede cerrar una herramienta a través de la interfaz de la propia herramienta; o puede relanzar una herramienta a través de los formularios de flujo de trabajo mostrados en la *Figura 4.1* (oprimiendo el botón “continuar” mostrado en la figura); o bien, un participante puede terminar una herramienta, seleccionando la casilla “Terminar Herramienta” y oprimiendo el botón “continuar” mostrados también en los formularios de la *Figura 4.1*.

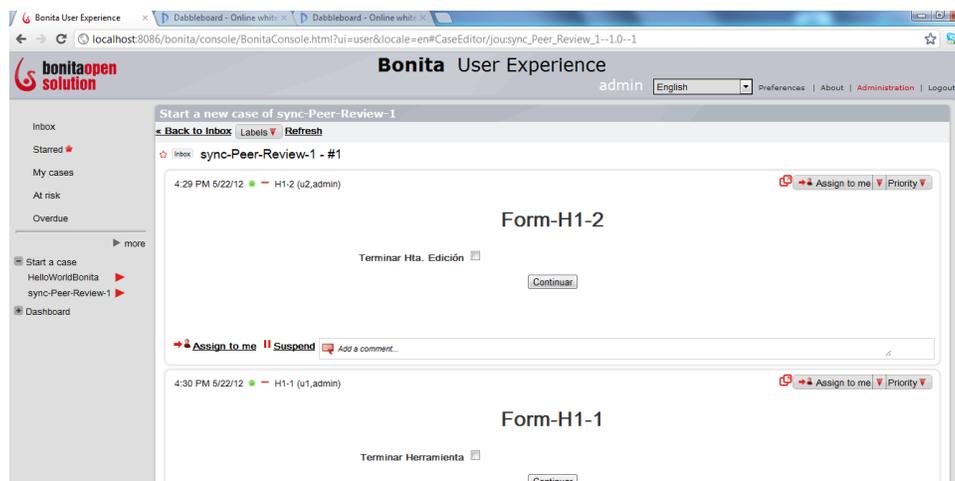


Figura 4.1: Formulario de flujo de trabajo usado por cada participante ($p1, p2$) de la actividad de edición, para relanzar la instancia del editor H1, o bien para terminar la herramienta y pasar a la siguiente herramienta de la secuencia, que en este caso es la terminación del flujo de datos de edición.

En resumen, por un lado, el motor de flujo de trabajo es el responsable de lanzar automáticamente una herramienta, así como de crear un formulario para cada participante que comparte la herramienta, desde donde puede relanzar o terminar una herramienta. Por otro

4. Evaluación de LeadFlow4LD

lado, son los participantes los responsables de cerrar o terminar una herramienta. Cuando una herramienta es cerrada, ésta puede ser relanzada más tarde a través del formulario. Pero, cuando una herramienta es terminada, ésta es la forma que se tiene de avisar al motor de flujo de trabajo que se ha terminado de usar una herramienta, y que debe continuar con la secuencia de herramientas, o bien terminar el flujo de datos de la actividad, según sea el caso.

Una vez que los participantes terminan de usar la secuencia de herramientas en el flujo de datos de edición, el motor de flujo de trabajo devuelve el control al motor del flujo de aprendizaje, permitiendo continuar con la siguiente actividad de aprendizaje: la actividad de revisión. Después, cuando los participantes inician el flujo de datos en la actividad de revisión, el motor del flujo de aprendizaje devuelve nuevamente el control al motor de flujo de trabajo, el cual lanza automáticamente las pizarras colaborativas que soportan la actividad de revisión. Tal es el caso de la pizarra colaborativa presentada a la pareja $\{p1, p2\}$ mostrada en la *Figura 4.2*, pero que además como se puede ver en la misma figura, está inicializada con el mapa conceptual creado por la pareja $\{p3, p4\}$ en la actividad anterior (lo cual ilustra la automatización de los datos de entrada y salida de herramientas). La *Figura 4.2* ilustra también el chat integrado en la herramienta colaborativa y que se usa para la comunicación entre pares. Similarmente, el motor de flujo de trabajo entrega también la pizarra colaborativa a la pareja $\{p3, p4\}$, pero ésta vez inicializada con el mapa conceptual creado por la pareja $\{p1, p2\}$ en la actividad anterior.

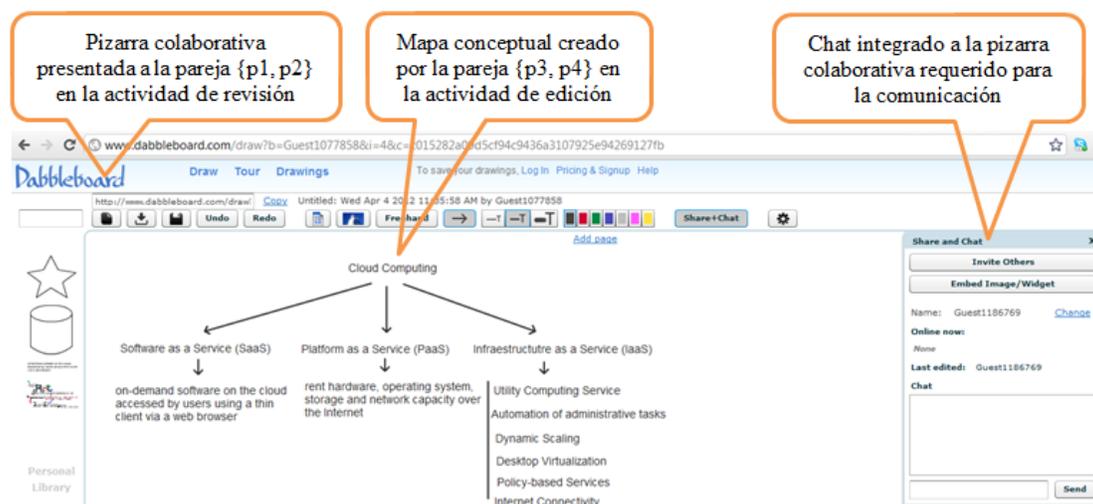


Figura 4.2: Pizarra colaborativa entregada automáticamente por el sistema a la pareja $\{p1, p2\}$ e inicializada con el mapa conceptual de la pareja $\{p3, p4\}$ creado en la actividad anterior.

Reutilización del Diseño (UoLeadFlow4LD)

De la misma forma que ocurrió con las situaciones de aprendizaje 1.1 y 1.2, la puesta en marcha de las situaciones de aprendizaje 1.3 y 1.4 mostró, como se esperaba, la invocación automática de herramientas y la automatización de los datos de entrada y salida de las herramientas. Sin embargo, lo interesante de esta puesta en marcha es la reutilización de la UoLeadFlow4LD. En efecto, la puesta en marcha de las situaciones de aprendizaje 1.3 y 1.4 ilustraron, que con solo modificar el documento de población de grupos, esto es, la forma en que se asignan participantes a las actividades de aprendizaje y a las instancias de herramientas, se pudieron poner en marcha distintas situaciones de aprendizaje.

4.4.4 Discusión

A través de la puesta en marcha de las cuatro situaciones de aprendizaje (1.1 a 1.4) se pudo observar, por una parte, que los distintos participantes, ya sea que trabajen en actividades individuales (situaciones 1.1 y 1.3) o trabajen colaborativamente (situaciones 1.2 y 1.4), no tienen que preocuparse de las herramientas (individuales o colaborativas) que requieren usar en una actividad de aprendizaje, ya que el motor de flujo de trabajo se encarga de invocarlas automáticamente cuando las necesitan (de acuerdo a lo especificado en el flujo de datos). Así también, se pudo observar que los participantes no tienen que preocuparse de localizar y recuperar los artefactos (individuales o compartidos) que requieren las herramientas para llevar a cabo las actividades de aprendizaje, ya que éstas se inician automáticamente con los artefactos especificados como entradas de las herramientas. Más aún, los participantes tampoco tienen que preocuparse de guardar los artefactos creados por las herramientas, ya que éstos se almacenan automáticamente de acuerdo a lo especificado en el flujo de datos. Por tanto, de lo observado durante la puesta en marcha de las situaciones de aprendizaje 1.1 a 1.4, es razonable afirmar que LeadFlow4LD es expresivo para especificar la automatización del flujo de datos, lo cual incluye la invocación automática de herramientas (individuales o colaborativas), así como la automatización de los datos (individuales o compartidos) de entrada y salida de las herramientas.

Por otra parte, con la puesta en marcha de las situaciones de aprendizaje 1.3 y 1.4 se pudo observar claramente que solamente uno de cinco documentos, el documento de población de grupos, fue necesario modificar para poner en marcha cada una de las situaciones de aprendizaje. Esto permite que distintas situaciones de aprendizaje se puedan especificar con tan sólo cambiar la forma en que se distribuyen los participantes, ya sea en las actividades de aprendizaje o en las instancias de herramientas. También se observó que las instancias de la estructura del flujo de datos que caracterizan las situaciones de aprendizaje {1.1, 1.2} y {1.3, 1.4} se crearon a partir de la misma estructura del flujo de datos. Esto permite que distintas instancias se puedan derivar de la misma estructura del flujo de datos con tan solo definir el número de instancias de datos y herramientas y la forma en que éstas se relacionan entre sí. Más aún, incluso el mismo documento de población de grupos se puede reutilizar con tan solo cambiar los participantes concretos que intervienen en la

situación de aprendizaje. Esto permite crear situaciones de aprendizaje en la que intervienen distintos estudiantes, pero manteniéndose el mismo número y tamaño de los grupos, así como las mismas interacciones entre grupos. Por tanto, de lo observado durante la puesta en marcha de las cuatro situaciones de aprendizaje, es razonable afirmar que es posible reutilizar una UoLeadFlow4LD para distintas situaciones de aprendizaje colaborativo.

4.4.5 Comparación con otras soluciones

Una solución alternativa para describir las situaciones de aprendizaje colaborativo de revisión entre pares 1.1 y 1.3, es la que proponen Koper (2009) y Miao et al. (2005). Ellos proponen incluir información de los participantes en la forma de roles (Estudiante-x) de tal forma que se pueda discriminar qué datos son asignados a cada estudiante, como se ilustró en la *Figura 2.22a* y *Figura 2.22b* en la sección 2.9.1. Sin embargo, como se comentó en dicha sección, esta aproximación tiene varias desventajas. La primera es que no especifica el flujo de datos a nivel de herramientas, sino solamente el flujo de datos a nivel de actividades. En tal caso, la invocación de herramientas, y los datos de entrada y salida de las herramientas no pueden ser automatizados, lo cual es el primer principal objetivo de esta tesis. La segunda desventaja es que el diseño resultante no describe la estructura de aprendizaje colaborativo de revisión entre pares, sino solamente una instancia de dicha estructura. Con esta aproximación, sería necesario crear un diseño de aprendizaje distinto para especificar cada una de las situaciones de aprendizaje 1.1 a 1.4, con lo que el diseño de aprendizaje resultante no se podría reutilizar en distintas situaciones de aprendizaje, lo cual es el segundo principal objetivo de esta tesis.

En cambio, con el método propuesto, se consiguió especificar y poner en marcha tanto la invocación automática de herramientas, y la automatización de los datos de entrada y salida de las herramientas, así como la reutilización de la estructura de aprendizaje colaborativo de revisión entre pares. Por tanto, la comparación de las bondades ofrecidas por LeadFlow4LD con respecto a la solución alternativa mencionada, contribuye a entender mejor el mérito de la solución propuesta.

4.5 Ejemplo ilustrativo: convergencia del conocimiento

Como se mencionó en el capítulo 2, este es un patrón de proceso de aprendizaje colaborativo bien documentado en la literatura (Weinberger, Stegmann & Fischer, 2005) y que es en realidad una generalización de la técnica de aprendizaje colaborativo de revisión entre pares. Primero, en la actividad de edición, un grupo de estudiantes editan informes individuales o compartidos sobre un tema o caso. Después, en la actividad de revisión, los informes son revisados y criticados en turnos por el resto de los miembros del grupo. Por último, en la actividad de reflexión, los informes iniciales junto con las críticas realizadas regresan a los autores originales, quienes convergen sus propias ideas junto con las de sus compañeros de grupo para producir así las conclusiones o reflexiones finales del caso. Por supuesto, también se pueden plantear convergencias “en paralelo” de grupos de estudiantes, en donde cada estudiante de cada grupo revisa, por turnos, al resto de sus compañeros de grupo.

La diferencia entre la técnica de revisión entre pares y la técnica de convergencia del conocimiento, es que en esta última, durante la actividad de revisión, cada participante revisa en secuencia (en turnos) el informe creado por el resto de los participantes en la actividad anterior. Por ejemplo, en una situación con tres participantes, cada participante debe revisar y criticar, en secuencia, el informe creado por los otros dos participantes; con cuatro participantes, cada participante debe revisar el informe de los otros tres, etc. Por tanto, en una situación con sólo dos participantes, la técnica de convergencia del conocimiento se reduce a la técnica de revisión entre pares, a la que se le agrega una actividad final de reflexión.

En la situación de aprendizaje seleccionada (identificada como situación 2.1 en la *Tabla 4.4*) participan 3 estudiantes. Durante la actividad de edición, a cada estudiante se le pide escribir un informe sobre uno de 3 temas o casos (a cada estudiante le corresponde un caso diferente). La actividad de edición es soportada, para cada estudiante, por un visor (*browser*) usado para leer el caso, y un editor usado para escribir el informe sobre el caso. Después, durante la actividad de revisión, cada estudiante tiene que revisar y criticar, en turnos, el informe creado por cada uno de sus otros dos compañeros. Para realizar la actividad de revisión, cada estudiante debe tener acceso a los informes creados por sus otros dos compañeros y a sus respectivos casos. Por tanto, la actividad de revisión es soportada por dos tipos de herramientas: un editor (precargado con el informe que se va revisar) usado para leer y criticar el informe, y un visor del caso correspondiente. Durante la actividad de revisión, cada estudiante debe revisar y criticar primero el informe de uno de sus compañeros y luego el del otro (secuencia que continuaría si participaran n-estudiantes). Es importante darse cuenta de que durante la misma actividad de revisión, y por cada estudiante, el par de herramientas visor-editor debe invocarse en secuencia, lo cual es el principal propósito a ilustrar en esta situación de aprendizaje. Por último, en la actividad de reflexión, cada informe, junto con las dos críticas de sus compañeros de grupo, regresa al autor original, en donde cada estudiante agrega sus conclusiones finales del caso, a través de converger sus propias ideas con las ideas de sus compañeros de grupo.

4.5.1 Especificación de la estructura de aprendizaje colaborativo

Documento del Flujo de Aprendizaje

En la *Figura 4.3* se muestra el diseño de la estructura del flujo de aprendizaje creado por el diseñador instruccional correspondiente a la técnica de convergencia del conocimiento. Aquí, el flujo de actividades se compone de la secuencia de las actividades de edición (A1), revisión (A2) y reflexión (A3), asociadas respectivamente a los roles editor, revisor y de reflexión. El documento del flujo de aprendizaje correspondiente a este diseño se puede consultar en el apéndice C.

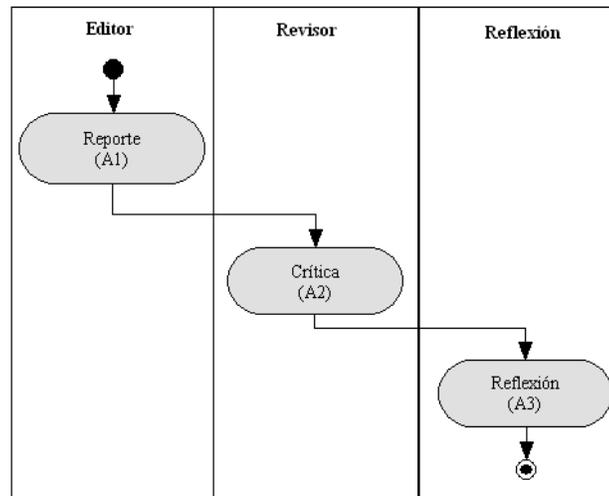


Figura 4.3: Diseño del flujo de aprendizaje correspondiente a la técnica de aprendizaje colaborativo basada en la convergencia del conocimiento.

Documento del Flujo de Datos

La Figura 4.4 muestra el diseño de la estructura del flujo de datos creado por el diseñador instruccional para ésta técnica de aprendizaje colaborativo. Primero, la secuencia de herramientas inicia con la invocación simultánea de dos herramientas asociadas a la actividad de edición. La primera herramienta (*H1*), se trata de un visor de documentos usado por los estudiantes participantes para leer el caso asignado (*D1*); y la segunda herramienta (*H2*), consiste en un editor de texto usado por los estudiantes para escribir su informe inicial sobre el caso (*D2*). Segundo, la secuencia de herramientas continúa con dos herramientas simultáneas asociadas a la actividad de revisión. La primera herramienta, igual que en la actividad anterior, se trata de un visor de documentos (*H3*) usado para leer el caso asignado; y la segunda herramienta (*H4*), se trata de un editor que tiene un doble propósito: acceder al informe inicial sobre el caso (*D2*) y escribir las críticas sobre dicho informe (*D2*). Por último, la secuencia de herramientas termina con la invocación de dos herramientas simultáneas asociadas a la actividad de reflexión. La primera herramienta (*H5*), se trata nuevamente de un visor de descripción de casos; y la segunda herramienta (*H6*), se trata de un editor usado para acceder al artefacto *D2* que contiene el reporte inicial del caso, junto con sus críticas, pero que también se usa para escribir la reflexión final sobre el caso.

La relación entre datos y herramientas del diseño de la Figura 4.4 se puede interpretar de la siguiente manera. El artefacto (*D1*) representa el documento que describe el caso, el cual es leído por los distintos visores de documentos en las tres actividades de aprendizaje (por tanto, *D1* es una entrada de las herramientas *H1*, *H3* y *H5*). El artefacto *D2* representa el informe inicial del caso (salida de *H2*), pero que también se usa para incluir la crítica sobre el informe del caso (salida de *H4*), así como para incluir las conclusiones finales del caso (salida de *H6*). En la Tabla 4.6 se resumen los artefactos y herramientas abstractas usados en el diseño de la

4. Evaluación de LeadFlow4LD

Figura 4.4, mientras que el documento del flujo de datos de esta situación de aprendizaje se puede consultar en el apéndice C.1

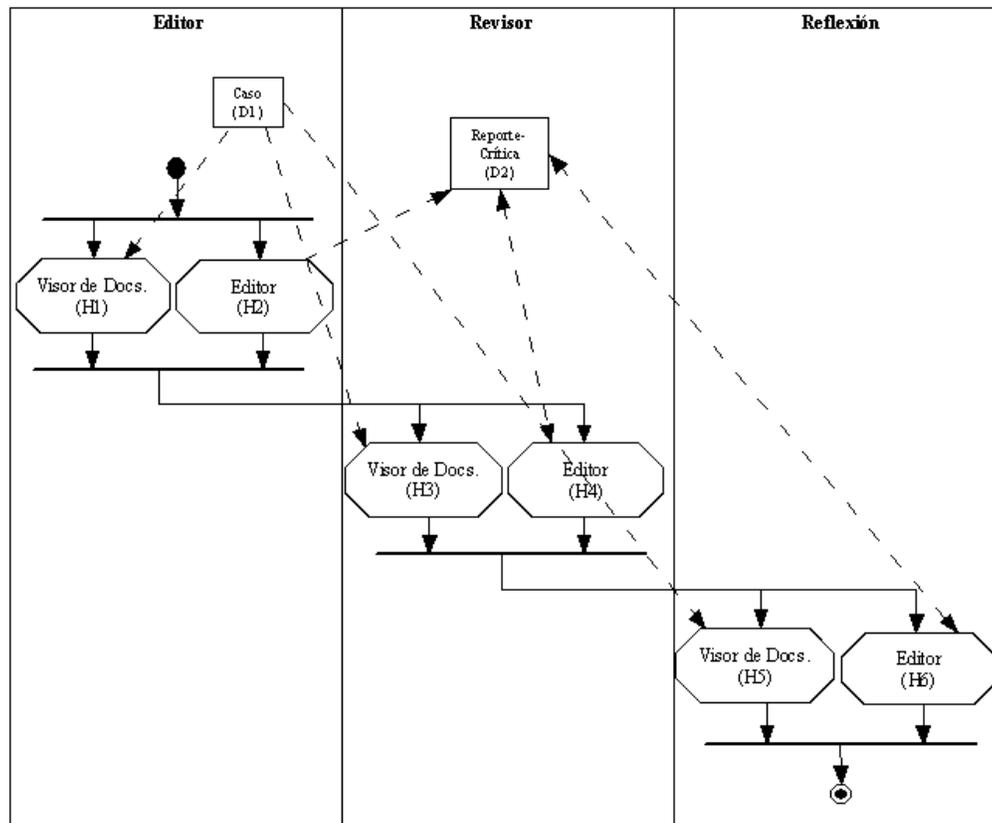


Figura 4.4: Diseño de la estructura del flujo de datos correspondiente a la técnica de aprendizaje colaborativo basada en la convergencia del conocimiento.

ID	Descripción	ID	Descripción
D1	Descripción del caso	D2	Informe sobre el caso + críticas + reflexión final
H1	Visor del caso durante la edición del reporte	H2	Editor del informe del caso
H3	Visor del caso durante la crítica del reporte	H4	Editor de críticas sobre el informe del caso
H5	Visor del caso durante la reflexión final	H6	Editor de la reflexión final

Tabla 4.6: Descripción de artefactos (datos) y herramientas abstractas referidas en el diseño del flujo de datos correspondiente a la técnica de aprendizaje colaborativo de convergencia del conocimiento.

Documento de la Coordinación de Flujos

La Figura 4.5 muestra la relación estructural entre el flujo de aprendizaje y el flujo de datos correspondiente a la técnica de convergencia del conocimiento. La especificación de parte de este documento se muestra en la Figura 4.6. Como se puede ver, se especifican tres transiciones maestras: de A1 a H1, de A2 a H3 y de A3 a H5, las cuales son usadas para especificar el inicio del flujo de datos. Similarmente, se especifican tres transiciones esclavas:

de $H2$ a $A1$, de $H4$ a $A2$ y de $H6$ a $A3$, las cuales son usadas para especificar la terminación del flujo de datos en cada una de las actividades de aprendizaje.

Es importante darse cuenta de que no es en esta estructura del flujo de datos donde se ilustra la secuenciación de herramientas en una misma actividad de aprendizaje, ya que como es claro en el diseño de la *Figura 4.6*, a ninguna actividad de aprendizaje está asociada una secuencia de herramientas (más de una herramienta en secuencia), ya que en todos los casos se trata de invocaciones concurrentes de herramientas. La secuenciación de herramientas, es ilustrada más adelante, cuando se definen instancias de la estructura del flujo de datos.

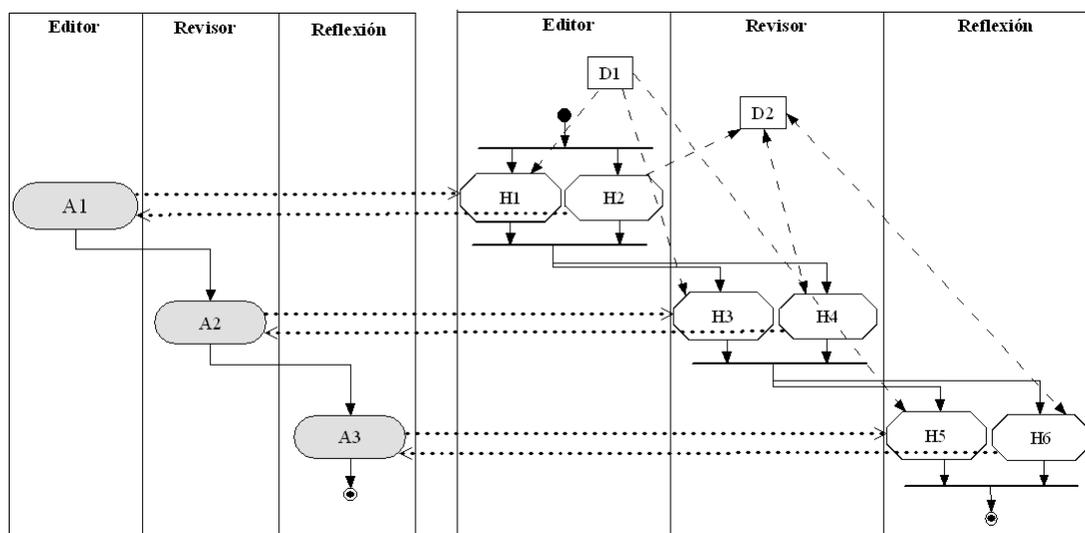


Figura 4.5: Diseño de la coordinación entre el flujo de aprendizaje y el flujo de datos correspondiente a la estructura de aprendizaje colaborativo basada en la convergencia del conocimiento.

4.5.2 Especificación de la instanciación de la estructura

Documento de Instanciación de Grupos

Para especificar la situación de aprendizaje 2.1 se requiere que el diseñador instruccional reflexione sobre las interacciones colaborativas que ocurren en el flujo de datos. Por tanto, a partir de la estructura del flujo de datos descrita en la *Figura 4.4* y tomando en cuenta que interactúan colaborativamente tres participantes en el grupo, donde a cada participante se le asigna un diferente caso, el diseñador instruccional define tres instancias tanto del artefacto que representa el documento del caso a estudiar ($D1$) como del artefacto que representa el informe, la crítica y la reflexión final sobre el caso ($D2$), como se refleja en la instancia de la estructura del flujo de datos mostrada en la *Figura 4.7*.

4. Evaluación de LeadFlow4LD

```

<activities>
<learning-activity id="A1" />
<learning-activity id="A2"/>
<learning-activity id="A3"/>
</activities>
<tools>
<tool id="H1" />
<tool id="H2" />
<tool id="H3" />
<tool id="H4" />
<tool id="H5" />
<tool id="H6" />
</tools>
<transitions>
<master-transition>
<from activity-ref="A1" />
<to tool-ref="H1" />
</master-transition>
<master-transition>
<from activity-ref="A2" />
<to tool-ref="H3" />
</master-transition>
<master-transition>
<from activity-ref="A3" />
<to tool-ref="H5" />
</master-transition>
<slave-transition>
<from tool-ref="H2" />
<to activity-ref="A1" />
</slave-transition>
<slave-transition>
<from tool-ref="H4" />
<to activity-ref="A2" />
</slave-transition>
<slave-transition>
<from tool-ref="H6" />
<to activity-ref="A3" />
</slave-transition>
</transitions>

```

Figura 4.6: Parte del documento de coordinación de flujos que especifica la coordinación entre el flujo de aprendizaje y el flujo de datos, correspondiente a la técnica de aprendizaje colaborativo basada en la convergencia del conocimiento.

Es importante hacer notar también las tres instancias definidas tanto de la herramienta de edición (*H3*) usada para escribir el informe del caso, como de la herramienta de edición (*H6*) usada en la reflexión final. Además, puesto que en cada grupo colaboran tres participantes cada uno con casos diferentes, cada participante debe criticar los informes de sus otros dos compañeros, por lo que se requiere de dos instancias de la herramienta de edición (*H4*) para la crítica de cada participante y dos instancias del visor de casos (*H3*). Es decir, en total se requieren de 6 instancias de *H3* y *H4*, 2 por cada participante. Esto último es importante, ya que se refleja en el diseño como una secuenciación de herramientas durante la actividad de revisión y que aparece encerrado en óvalos en la *Figura 4.7*. En la *Tabla 4.7* y en la *Tabla 4.8* se presenta un resumen de las instancias de datos y herramientas mostradas en la *Figura 4.7*. El documento de instanciación de grupos de la situación de aprendizaje 2.1 se puede consultar en el apéndice C.1.

4. Evaluación de LeadFlow4LD

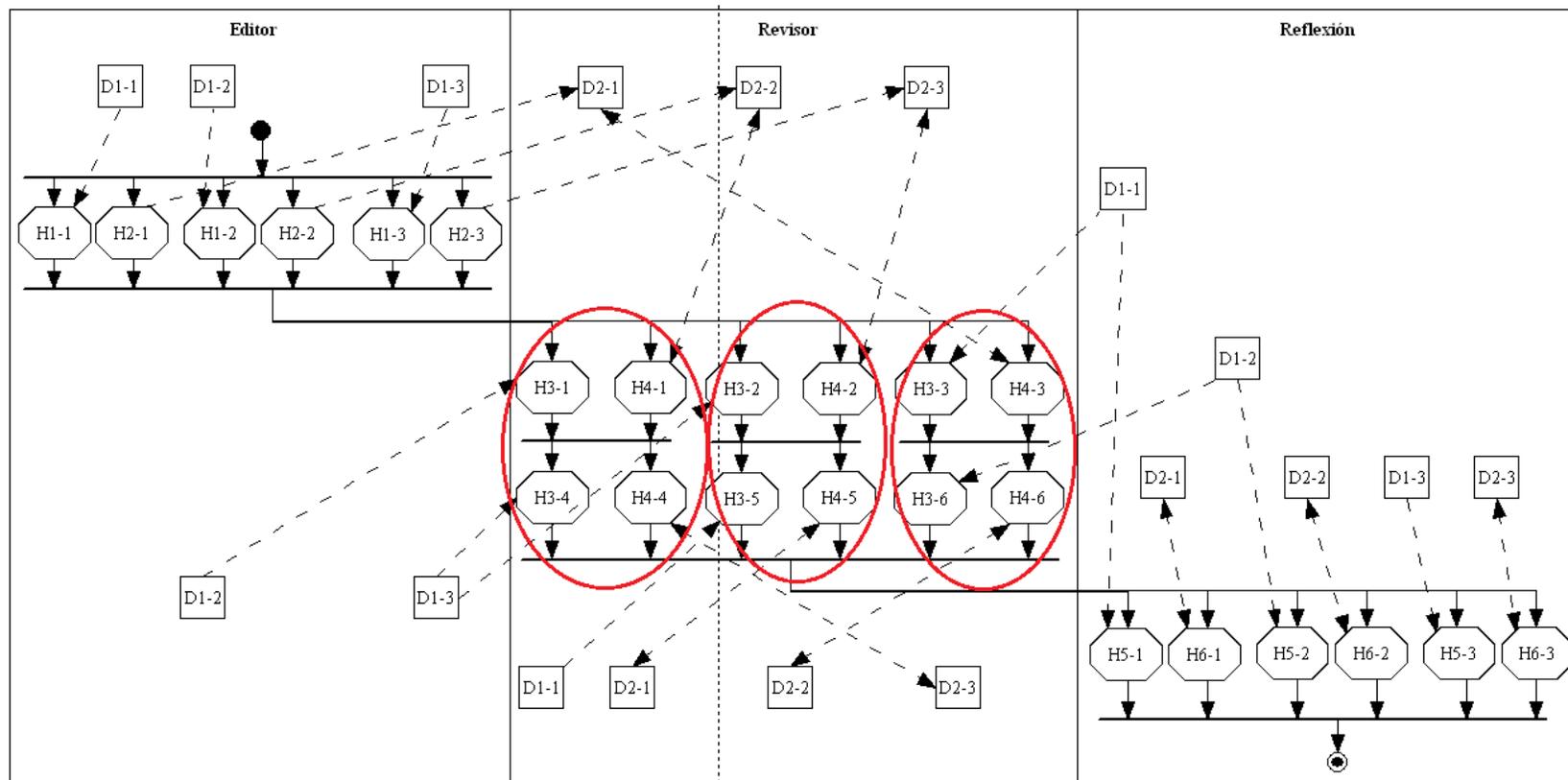


Figura 4.7: Instancia de la estructura del flujo de datos correspondiente a la situación de aprendizaje 2.1. Encerrado en óvalos se ilustra la secuenciación de herramientas de la actividad de revisión, lo cual es el principal propósito a ilustrar en esta situación de aprendizaje.

4. Evaluación de LeadFlow4LD

Dato	Instancias		Descripción	Id	Descripción
	ID	Cant.			
D1	3	D1-1	Descripción del caso 1	D1-2	Descripción del caso 2
		D1-3	Descripción del caso 3		
D2	3	D2-1	Informe del participante 1 + crítica del participante 3 + crítica del participante 2 + reflexión final del participante 1	D2-2	Informe del participante 2 + crítica del participante 1 + crítica del participante 3 + reflexión final del participante 1
		D2-3	Informe del participante 3 + crítica del participante 2 + crítica del participante 1 + reflexión final del participante 3		

Tabla 4.7: Descripción de instancias de datos definidas en el documento de instanciación de grupos de la situación de aprendizaje 2.1.

Hta	Instancias		Descripción	ID	Descripción
	ID	Cant.			
H1	3	H1-1	Visor del caso 1 por el participante 1	H1-2	Visor del caso 2 por el participante 1
		H1-3	Visor del caso 3 por el participante 3		
H2	3	H2-1	Editor del reporte del caso 1 por el participante 1	H2-2	Editor del reporte del caso 2 por el participante 2
		H2-3	Editor del reporte del caso 3 por el participante 3		
H3	6	H3-1	Visor del caso 2 por el participante 1	H3-2	Visor del caso 3 por el participante 2
		H3-3	Visor del caso 1 por el participante 3	H3-4	Visor del caso 3 por el participante 1
		H3-5	Visor del caso 1 por el participante 2	H3-6	Visor del caso 2 por el participante 3
H4	6	H4-1	Editor de la crítica del participante 1 al participante 2	H4-2	Editor de la crítica del participante 2 al participante 3
		H4-3	Editor de la crítica del participante 3 al participante 1	H4-4	Editor de la crítica del participante 1 al participante 3
		H4-5	Editor de la crítica del participante 2 al participante 1	H4-6	Editor de la crítica del participante 3 al participante 2
H5	3	H5-1	Visor del caso 1 por el participante 1	H5-2	Visor del caso 2 por el participante 2
		H5-3	Visor del caso 3 por el participante 3		
H6	3	H6-1	Editor de la reflexión final del participante 1	H6-2	Editor de la reflexión final del participante 2
		H6-3	Editor de la reflexión final del participante 3		

Tabla 4.8: Descripción de instancias de herramientas definidas en el documento de instanciación de grupos de la situación de aprendizaje 2.1.

Documento de Población de Grupos

Como se muestra en la *Tabla 4.9*, la situación de aprendizaje 2.1 requiere que todos los participantes se desempeñen en los tres roles, lo que significa que todos ellos intervienen en las actividades de edición, revisión y reflexión. Así también, en la *Tabla 4.10* se muestra la forma en que el educador asigna los participantes a las instancias de herramientas. El

documento de población de grupos de la situación de aprendizaje 2.1 se puede consultar en el apéndice C.

Situación	Roles del Flujo de Aprendizaje		
	Editor	Revisor	Reflexión
2.1	p1,p2,p3	p1,p2,p3	p1,p2,p3

Tabla 4.9: Asignación de participantes a roles de aprendizaje correspondiente a la situación de aprendizaje 2.1.

Instancia de Herramienta	Participantes	Instancia de Herramienta	Participantes	Instancia de Herramienta	Participantes
H1-1	p1	H1-2	p2	H1-3	p3
H2-1	p1	H2-2	p2	H2-3	p3
H3-1	p1	H3-2	p2	H3-3	p3
H3-4	p1	H3-5	p2	H3-6	p3
H4-1	p1	H4-2	p2	H4-3	p3
H4-4	p1	H4-5	p2	H4-6	p3
H5-1	p1	H5-2	p2	H5-3	p3
H6-1	p1	H6-2	p2	H6-3	p3

Tabla 4.10: Asignación de participantes a las instancias de herramientas de la situación de aprendizaje 2.1.

4.5.3 Puesta en marcha de la situación de aprendizaje

El Flujo de Aprendizaje

La Figura 4.8 muestra una vista general del flujo de aprendizaje que el sistema presenta a cada uno de los tres participantes de la situación de aprendizaje 2.1, y que está compuesta de una actividad de edición, otra de revisión y una más de reflexión.

Automatización del Flujo de Datos

Esta situación de aprendizaje fue seleccionada para ilustrar la expresividad del método propuesto en automatizar la secuenciación de herramientas en una misma actividad de aprendizaje. Con este propósito, la puesta en marcha de la situación de aprendizaje iniciará el recorrido que sigue el informe inicial creado por uno de los tres participantes (el participante p2) en la actividad de edición, después seguirá la secuenciación de herramientas que tiene lugar durante la actividad de revisión (ésta es la parte de interés de la puesta en marcha de esta situación de aprendizaje), y hasta la creación del informe final del participante p2 en la actividad de reflexión.

En la Figura 4.9a se muestra el momento en que el participante p2 selecciona e inicia el flujo de datos desde la actividad de edición. En ese momento, el motor del flujo de aprendizaje habilita al motor del flujo de trabajo a iniciar el flujo de datos correspondiente a la actividad de edición. De acuerdo a lo especificado, este inicia lanzando automáticamente una instancia del visor de documentos (ver Figura 4.9b) necesario para leer el documento del caso asignado, así como una instancia del editor de texto usado por el participante p2 para

escribir su informe inicial sobre el caso. Instancias similares de estas herramientas son lanzadas automáticamente por el motor de flujo de trabajo para el resto de los participantes.

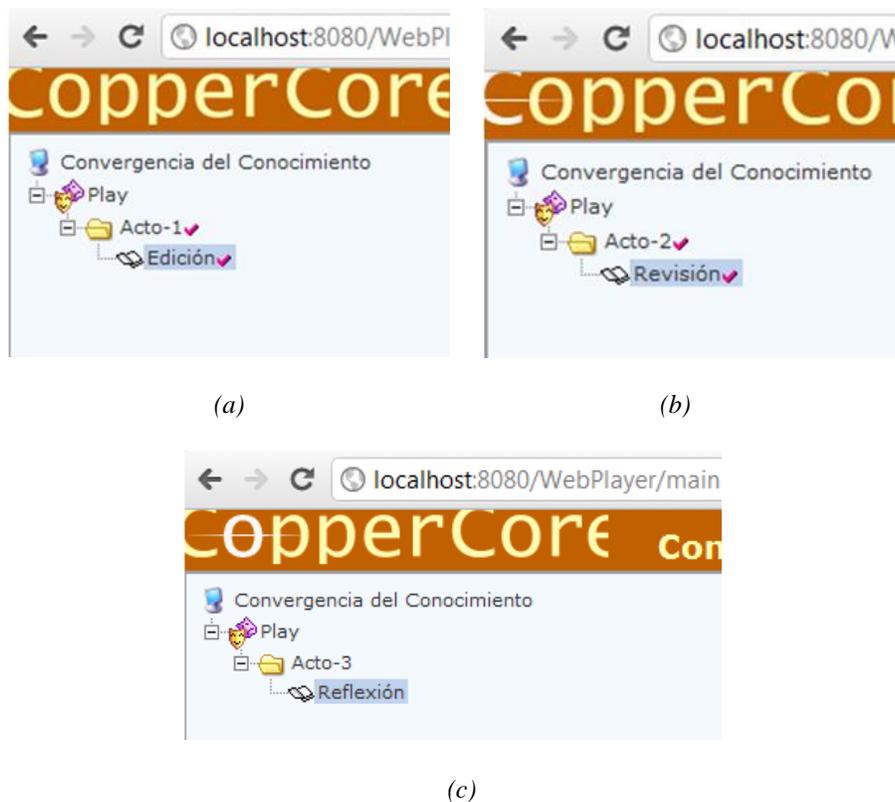


Figura 4.8: Vista general del flujo de aprendizaje presentado a cada uno de los tres participantes de la situación 2.1: (a) actividad de edición (b) actividad de revisión (c) actividad de reflexión.

Es importante darse cuenta de que el visor de documentos en la Figura 4.9b está inicializado con la descripción del caso que el participante *p2* requiere para escribir su informe inicial sobre el caso (aquí es donde se observa la automatización de los datos de entrada de herramientas). Similarmente, visores de documentos correspondientes al resto de los participantes se inicializan automáticamente con la descripción de diferentes casos.

A continuación, cada participante escribe su informe inicial sobre el caso. Esto se muestra en la Figura 4.10 para el participante *p2*. Una vez que los participantes finalizan sus herramientas de edición, termina el flujo de datos correspondiente a la actividad de edición. Los informes iniciales creados son automáticamente almacenados por las herramientas y el motor de flujo de trabajo devuelve el control al motor del flujo de aprendizaje, para de esta forma continuar con la siguiente actividad de aprendizaje: la actividad de revisión.

4. Evaluación de LeadFlow4LD

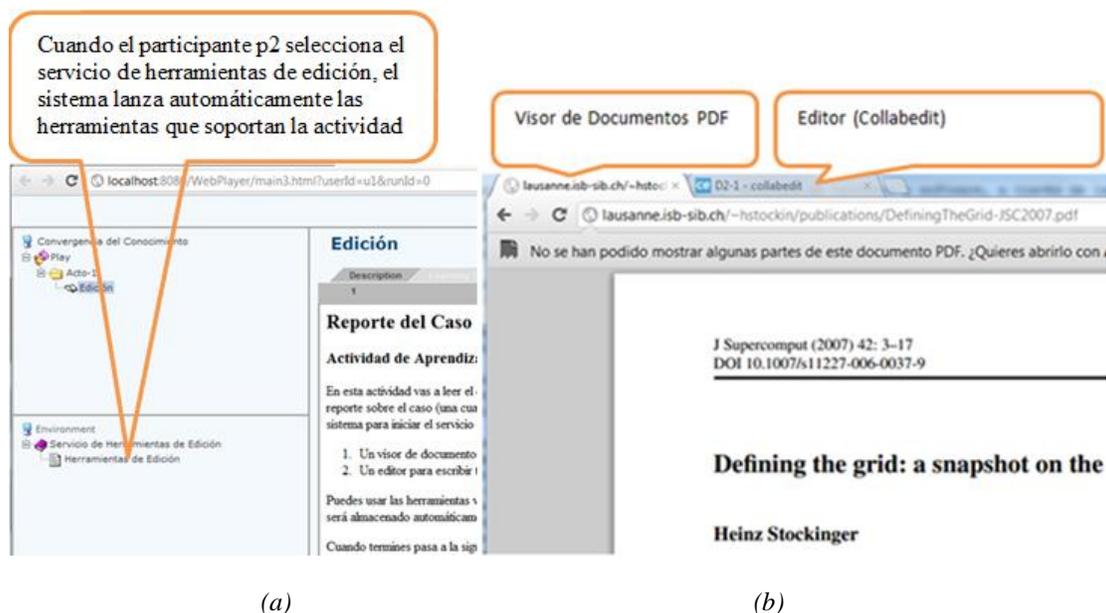


Figura 4.9: Momento en que (a) el participante $p2$ inicia el flujo de datos desde la actividad de edición e (b) el motor de flujo de trabajo lanza automáticamente una instancia del visor de documentos que ya incluye el caso asignado, así como una instancia del editor de texto para escribir su informe inicial sobre el caso.



Figura 4.10: Reporte del caso realizado por el participante $p2$, después de leer el artículo asignado.

Durante la actividad de revisión, cada participante critica en turnos o rondas, los informes creados por los otros dos participantes. Por tanto, el informe inicial del participante $p2$, es criticado por el participante $p1$ y también por el participante $p3$. A continuación se muestra la puesta en marcha de la primera ronda de críticas.

La Figura 4.11a muestra el momento en que el participante $p1$ inicia el flujo de datos desde la actividad de revisión, y como resultado, el motor del flujo de aprendizaje delega el control al motor de flujo de trabajo. Este lanza automáticamente tanto la herramienta de edición inicializada con el informe inicial creado por el participante $p2$ (aquí es donde se observa la invocación automática de herramientas y la automatización de los datos de entrada y salida de herramientas), como el visor del caso del informe que se va a criticar (ver Figura

4. Evaluación de LeadFlow4LD

4.11b). Similarmente, el resto de los participantes seleccionan e inician también el flujo de datos correspondiente a la actividad de revisión.

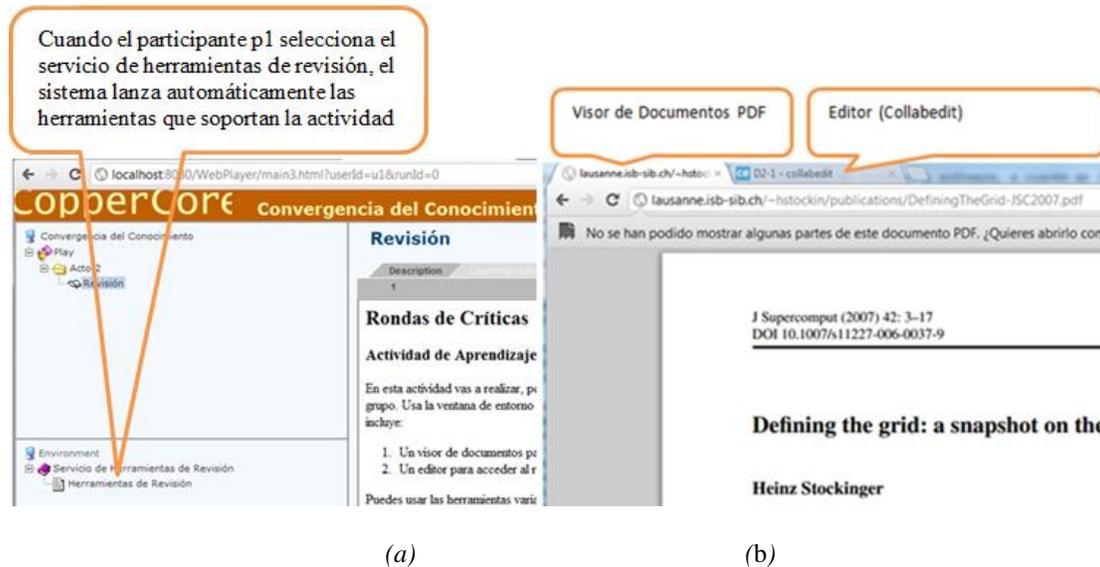


Figura 4.11: Momento en que (a) el participante p1 selecciona su servicio de herramientas en la actividad de revisión e (b) el motor de flujo de trabajo lanza automáticamente el visor de documentos inicializado con el caso asignado y el editor de texto inicializado con el informe inicial del participante p2.

La Figura 4.12 muestra el artefacto que contiene el informe inicial del participante p2, pero una vez que ha sido criticado por el participante p1. Similarmente, el participante p2 critica el informe del participante p3 y el participante p3 critica el informe del participante p1.

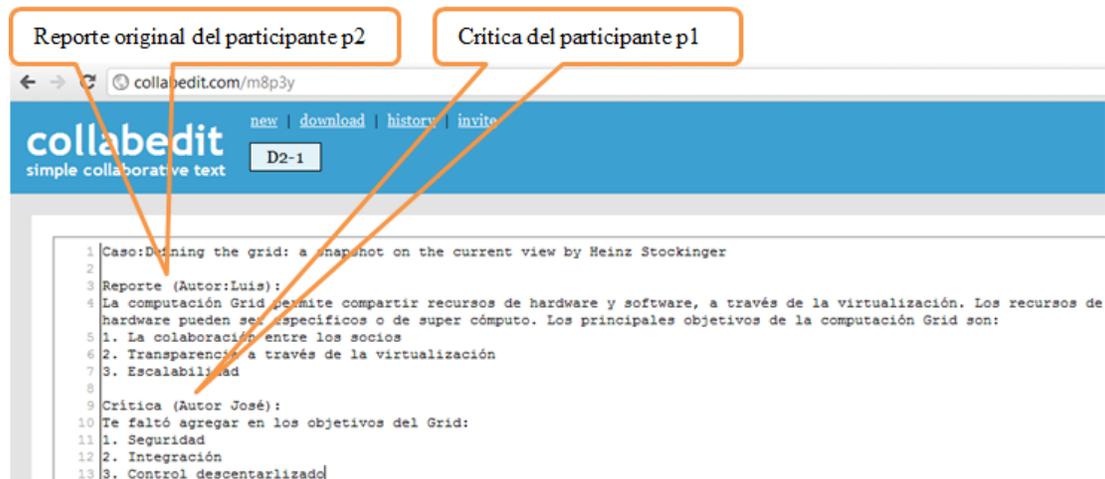


Figura 4.12: Artefacto que contiene el informe inicial del participante p2 más la crítica del participante p1.

Una vez que los tres participantes han llevado a cabo sus primeras rondas de críticas, la siguiente fase de la actividad de revisión es realizar la segunda y última ronda de críticas. Para ello, una vez que todos los participantes terminaron sus herramientas usadas en la primera ronda de críticas, el motor de flujo de trabajo, de acuerdo lo especificado, lanza automáticamente las instancias de herramientas que serán usadas en la segunda ronda de críticas (aquí es donde se observa la automatización de la secuenciación de herramientas en una misma actividad de aprendizaje, que en este caso es la actividad de revisión). La *Figura 4.13* muestra este escenario para el participante *p3* una vez que ha realizado su crítica al informe del participante *p2*. Aquí se puede ver que la herramienta de edición está inicializada con el artefacto que contiene el informe inicial del participante *p2*, más la crítica realizada por el participante *p1*, y a la que se le ha agregado la crítica realizada por el participante *p3* (aquí es donde se observa la invocación automática de herramientas y la automatización de los datos de entrada y salida de las herramientas). Similarmente, el resto de los participantes realiza también su segunda rondas de críticas: el participante *p1* critica el reporte del participante *p3* y el participante *p2* critica el reporte del participante *p1*.

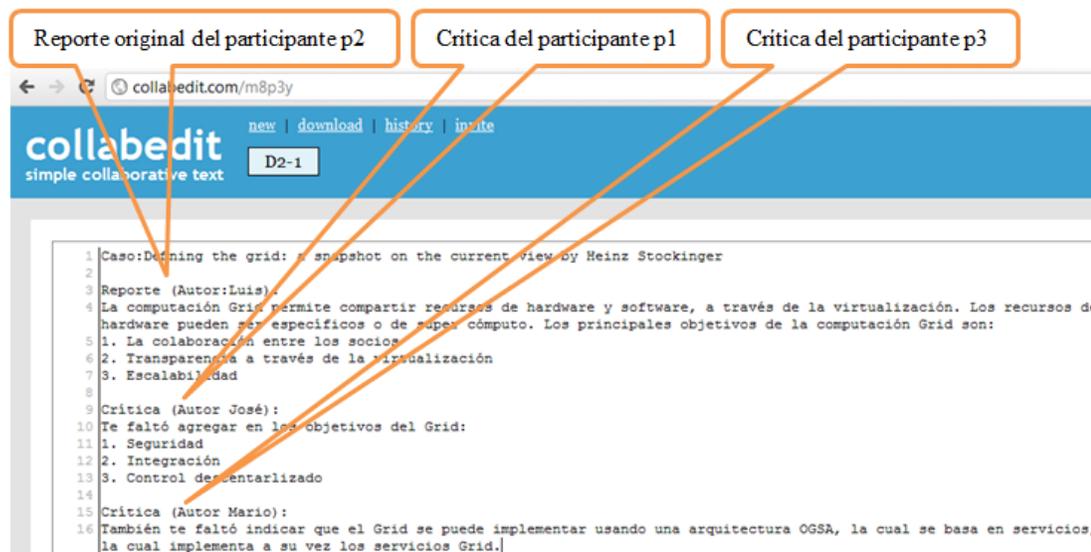


Figura 4.13: Crítica del participante p3 realizada al reporte inicial del participante p2.

Finalmente, una vez que los tres participantes han llevado a cabo su segunda ronda de críticas y han finalizado todos ellos sus herramientas, hasta entonces finaliza el flujo de datos correspondiente a la actividad de revisión. En ese momento el motor del flujo de trabajo delega el control al motor del flujo de aprendizaje, y se puede continuar con la siguiente y última actividad de aprendizaje: la actividad de reflexión.

Una vez que el participante *p2* selecciona e inicia el flujo de datos desde la actividad de reflexión, el motor del flujo de aprendizaje delega el control al motor de flujo de trabajo. Este, de acuerdo a lo especificado, lanza automáticamente tanto la herramienta que contiene el artículo del caso, como la herramienta de edición inicializada con el artefacto que contiene

4. Evaluación de LeadFlow4LD

el informe inicial del participante *p2* y las críticas de sus pares (los participantes *p1* y *p3*). La *Figura 4.14* muestra este artefacto, al que se le han agregado las conclusiones finales del propio participante *p2*. Artefactos similares son producidos también por el resto de los participantes.



Figura 4.14: Reflexiones del participante p2 de acuerdo a su reporte inicial y a las críticas de sus pares.

4.5.4 Discusión

A través de la puesta en marcha de la situación de aprendizaje 2.1 se pudo observar la secuenciación automática de invocaciones de herramientas durante la actividad de revisión. Aquí, cada uno de los tres participantes llevó a cabo sus dos rondas de críticas, por tanto instancias de la herramienta de edición fueron invocadas en secuencia. Los distintos participantes no tuvieron que preocuparse de invocar un editor, y luego otro editor para realizar sus dos rondas de críticas, sino que el motor de flujo de trabajo las invocó automáticamente de acuerdo a lo especificado en el flujo de datos. Por tanto, de la puesta en marcha de esta situación de aprendizaje podemos afirmar que LeadFlow4LD permitió automatizar la secuenciación de invocaciones de herramientas en una misma actividad de aprendizaje.

Durante la puesta en marcha de esta situación de aprendizaje también se observó la siguiente dificultad. Parece ser que los participantes pueden perder conciencia de las tareas asociadas con las herramientas que se ejecutan en secuencia en la misma actividad de aprendizaje. Por ejemplo, la actividad de revisión de la situación 2.1 se compone de dos

tareas: revisar y criticar el informe de un primer participante, y después, revisar y criticar el informe de un segundo participante. Para resolver esta dificultad parcialmente, se puede especificar la tarea a realizar por las herramientas en la descripción de las actividades de flujo de trabajo que representan invocaciones de herramientas (ver *Figura 4.1*), tarea que sin embargo debe ser realizada por el educador en tiempo de instanciación. Una solución definitiva a este problema puede estar en la dirección de descomponer la especificación de una actividad de grano grueso en sub-actividades, cada una de las cuales describiendo la tarea a realizar por una herramienta en la secuencia de invocaciones de herramientas.

Por último, si bien es cierto que esta situación de aprendizaje no fue seleccionada para ilustrar la reutilización de una UoLeadFlow4LD. Sin embargo, sería interesante discutir cómo reutilizar el diseño resultante para el caso de más participantes. Un caso ocurre, por ejemplo, con n estudiantes, donde cada uno participa en $n-1$ rondas de críticas durante la actividad de revisión, y por tanto $n-1$ instancias de herramientas de edición (H4) y visores de casos (H3) son invocadas en secuencia. Otro caso ocurre, por ejemplo, con $2n$ estudiantes, donde n estudiantes participan en $n-1$ rondas de críticas, y los otros n lo hacen con el resto ($n-1$) estudiantes. A nivel de diseño estructural de la técnica de convergencia del conocimiento, el diseñador instruccional no debe preocuparse de cuántos estudiantes participarán durante la interacción colaborativa del flujo de datos. Sino que, es hasta el momento de la instanciación de la estructura del flujo de datos (en el documento de instanciación de grupos), donde el diseñador instruccional debe decidir cuántas instancias de herramientas se van a crear y cómo se van a secuenciar éstas, lo que da una idea de la capacidad del método propuesto para reutilizar la estructura de un diseño de aprendizaje para derivar nuevas situaciones de aprendizaje similares.

4.5.5 Comparación con otras soluciones

Una solución alternativa para describir la situación de aprendizaje colaborativo basada en la técnica de convergencia del conocimiento, puede ser al igual que en el caso anterior, la aproximación propuesta por Koper (2009) y Miao et al. (2005) e ilustrada en la *Figura 2.22a* y en la *Figura 2.22b*, y que corresponden respectivamente a las situaciones de aprendizaje 1.1 y 1.3, respectivamente. Sin embargo, puesto que no es posible especificar la relación entre datos y herramientas, entonces tampoco sería posible especificar la secuenciación automática de invocaciones de herramientas en una misma actividad de aprendizaje. Además, el diseño de aprendizaje resultante describiría solamente una situación de aprendizaje particular, en lugar de describir una estructura de aprendizaje colaborativo que pueda ser reutilizada en distintas situaciones de aprendizaje.

Una segunda alternativa de solución, podría ser descomponer la actividad de revisión en una secuencia de actividades de aprendizaje de grano fino, de forma que se elimine la secuenciación de herramientas en la misma actividad de aprendizaje, y dejando las herramientas asociadas a sus propias actividades de aprendizaje. Por ejemplo, sería necesario descomponer la actividad de revisión en dos actividades para el caso de la situación 2.1, ya

que se deben realizar dos turnos de revisión. El problema de esta solución es que se debe descomponer actividades de grano grueso en un flujo de actividades de grano fino, lo que aumenta el nivel de coerción de las actividades, lo cual puede desmotivar el aprendizaje de los estudiantes (Dillenbourg, 2002). Además, de que situaciones de aprendizaje similares, donde participen 4, 5, ..., n estudiantes, requerirían de un nuevo y distinto diseño de aprendizaje, lo cual impide que un mismo diseño de aprendizaje se reutilice para distintas situaciones de aprendizaje.

En cambio, con el método propuesto, se consiguió especificar y poner en marcha la secuenciación automática de invocaciones de herramientas en una misma actividad de aprendizaje. Por tanto, la comparación de las bondades ofrecidas por LeadFlow4LD con respecto a las soluciones alternativas mencionadas, contribuye toda vía más a entender el mérito de la solución propuesta.

4.6 Ejemplo ilustrativo: caso MOSAIC

Como se mencionó en el capítulo 2 (consultar sección 2.7.3), el caso MOSAIC es la composición jerárquica de tres técnicas de aprendizaje colaborativo que son bien conocidas: pirámide de 3 niveles, *jigsaw* y revisión entre pares. La idea es utilizar la técnica de revisión entre pares para mejorar los resultados obtenidos durante cada uno de los tres niveles de la pirámide. El primer nivel de la pirámide se compone de 3 fases. La primera fase es individual y es donde los participantes se hacen expertos en un tema (el universo del problema es dividido en temas o partes). Después, la segunda fase es colaborativa, y es donde participantes expertos en un tema trabajan juntos como grupos de expertos, mejorando los resultados individuales creados en la actividad anterior. Finalmente, en la tercera fase del primer nivel de la pirámide, participantes expertos en distintos temas trabajan juntos en grupos de *jigsaw*, con el propósito de mejorar los resultados anteriores. A continuación, en el segundo nivel de la pirámide, grupos de *jigsaw* trabajan juntos para producir nuevos y mejores resultados consensuados. Por último, en el tercer nivel de la pirámide, todos los participantes se juntan en un solo grupo para discutir y presentar sus conclusiones finales. A continuación se describe una situación de aprendizaje concreta derivada de esta jerarquía de técnicas de aprendizaje colaborativo.

La situación de aprendizaje seleccionada (identificada como situación 3.1 en la *Tabla 4.4*) se compone de una pirámide 3 niveles en la que participan 12 estudiantes, y en la que el problema se ha dividido en 3 temas o partes. En la primera fase del primer nivel de la pirámide, cada estudiante lee individualmente uno de 3 artículos seleccionados (4 estudiantes leen el mismo artículo) con el propósito de producir un mapa conceptual individual, así como una lista de preguntas o dudas sobre el artículo analizado. En la segunda fase, todos los estudiantes que leyeron el mismo artículo trabajan juntos como un grupo de expertos (se forman 3 grupos de 4 expertos cada uno). Primero realizan una revisión entre pares de su trabajo previo (retroalimentación del mapa conceptual y la lista de preguntas de estudiantes individuales) y como consecuencia producen un nuevo mapa conceptual y una lista de

preguntas de los grupos de expertos. En la tercera fase del primer nivel de la pirámide, se forman grupos de *jigsaw* (se forman 4 grupos de *jigsaw* de 3 estudiantes cada uno) quienes primero retroalimentan a los grupos de expertos sobre sus mapas conceptuales y listas de preguntas, y después crean su propio mapa conceptual y lista de preguntas mejorado y consensuado. En el segundo nivel de la pirámide, grupos de *jigsaw* trabajan juntos (se forman 2 grupos de 6 estudiantes cada uno), primero realizando una revisión entre pares del trabajo previo y después creando un mapa conceptual y una lista de preguntas consensuadas. Por último, en el tercer nivel de la pirámide todos los estudiantes trabajan juntos para presentar el mapa conceptual definitivo y sus conclusiones finales.

Es importante recordar que esta situación de aprendizaje fue seleccionada con el propósito de ilustrar, por un lado, la expresividad de LeadFlow4LD para automatizar el flujo de datos, y por otro lado, como debe ser claro de la descripción anterior, para ilustrar la aplicación del método propuesto en escenarios de aprendizaje colaborativo complejos.

4.6.1 Especificación de la estructura de aprendizaje colaborativo

Documento del Flujo de Aprendizaje

En la *Figura 4.15* se muestra el diseño de la estructura del flujo de aprendizaje correspondiente al conjunto de técnicas de aprendizaje colaborativo que componen el llamado caso MOSAIC. Aquí, el flujo de actividades inicia en el rol estudiante con la actividad (A1) a través de la cual los participantes conocen el entorno de aprendizaje. Después, en el rol de estudiantes expertos, el flujo de aprendizaje inicia con una actividad de estudio individual (A2), que pretende hacerlos expertos sobre un tema o caso y que, como resultado, producirán un mapa conceptual individual así como una lista de preguntas o dudas del mismo; el flujo de actividades de aprendizaje continúa con una actividad individual de revisión entre pares (A3), en la que los estudiantes retroalimentan de forma individual a sus compañeros expertos sobre los artefactos creados por ellos en la actividad anterior (mapa conceptual y listado de preguntas individuales); y por último, el flujo de actividades en el rol de estudiante experto termina con una actividad colaborativa de estudio de expertos (A4), en la que los estudiantes trabajan juntos para crear un nuevo mapa conceptual así como una nueva lista de preguntas o dudas del mismo, pero ahora como grupos de expertos. A continuación, en el rol de *jigsaw*, el flujo de aprendizaje continúa con una actividad de revisión entre pares (A5), en la que los estudiantes trabajan juntos para retroalimentar a sus compañeros de *jigsaw* sobre los artefactos creados por ellos en la actividad anterior; después trabajan juntos en una actividad colaborativa (A6) para crear un nuevo mapa conceptual así como una nueva lista de preguntas o dudas del mismo, pero ahora como grupos de *jigsaw*. Después, el flujo de actividades en el nivel 2 de la pirámide continúa de forma similar al anterior, primero con una actividad de revisión entre pares (A7) para retroalimentar a sus compañeros sobre los artefactos creados por ellos en la actividad anterior, y después con una actividad colaborativa (A8) para crear un nuevo mapa conceptual así como una nueva lista de preguntas o dudas, pero ahora como grupos extendidos de *jigsaw*. Por último, el flujo de

4. Evaluación de LeadFlow4LD

aprendizaje termina en el nivel 3 de la pirámide, con una actividad colaborativa (A9) en la que todos los estudiantes crean en consenso un solo mapa conceptual, discuten y presentan sus conclusiones finales.

Un resumen de la descripción de las actividades de aprendizaje del caso MOSAIC se muestra en la *Tabla 4.11*, mientras que el documento del flujo de aprendizaje de este diseño puede ser consultado en el apéndice C.

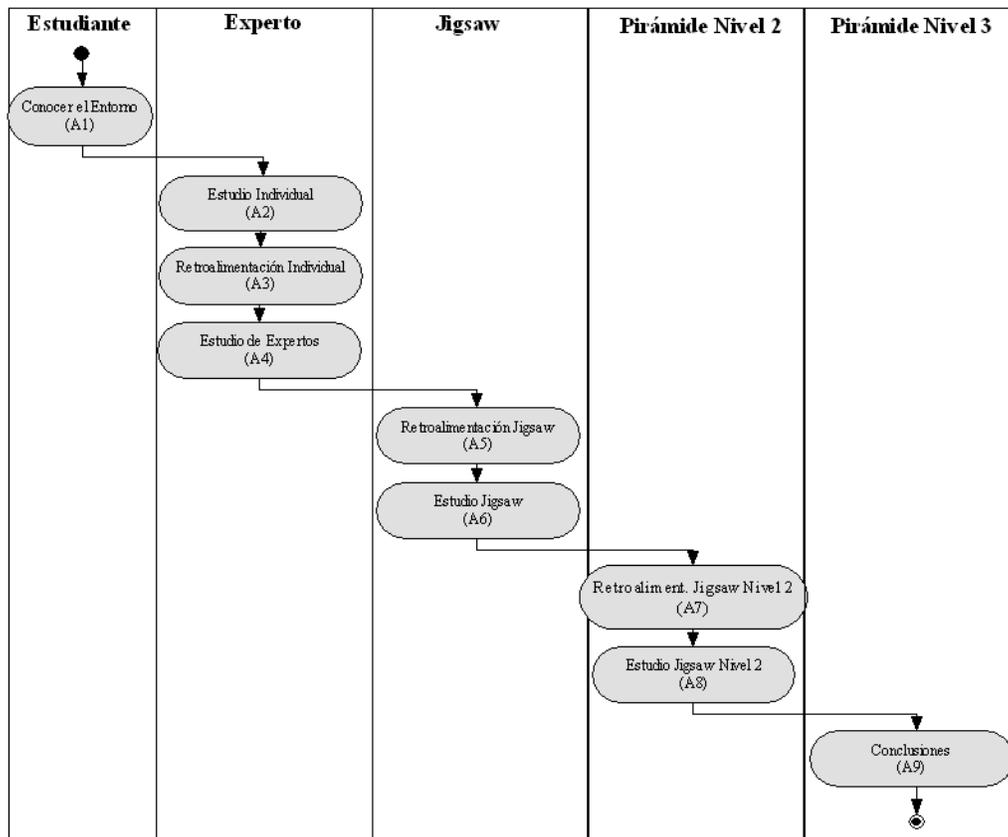


Figura 4.15: Diseño del flujo de aprendizaje correspondiente a la técnica de aprendizaje colaborativo basada en el caso MOSAIC.

Documento del Flujo de Datos

La *Figura 4.16* muestra el diseño de la estructura del flujo de datos correspondiente a la técnica de aprendizaje colaborativo basada en el caso MOSAIC. Aquí se describe la secuencia de invocaciones de herramientas, así como la relación entre datos y herramientas. En la secuencia de herramientas de la *Figura 4.16* se muestran primero las tres herramientas simultáneas asociadas a la actividad de crear el mapa conceptual individual: el visor de documentos (H1) usado para acceder al caso asignado (D1); la pizarra individual (H2) usada para crear el mapa conceptual individual (D2); y el editor de texto individual (H3) usado para escribir las preguntas o dudas individuales (D3). A continuación en la secuencia, están las tres herramientas simultáneas asociadas a la actividad de retroalimentación individual: el

4. Evaluación de LeadFlow4LD

visor de documentos (*H4*) usado para visualizar el caso asignado; la pizarra individual (*H5*) usada para retroalimentar el mapa conceptual realizado por los estudiantes en la actividad anterior; y el editor de texto individual (*H6*) usado para responder las dudas o preguntas realizadas por los estudiantes en la actividad anterior. Después en la secuencia, están las 5 herramientas simultáneas asociadas a la actividad de crear un mapa conceptual de expertos: el visor de documentos (*H7*), las pizarras colaborativas (*H8* y *H10*) y el editor de texto colaborativo (*H9* y *H11*). Las herramientas *H10* y *H11* son usadas para crear los mapas conceptuales (*D4*) y preguntas o dudas de los expertos (*D5*), mientras que las herramientas *H8* y *H9* son usadas solamente para consultar los mapas conceptuales y las dudas creadas por los estudiantes expertos en la actividad anterior.

Activ. ID	Nombre	Tipo	Descripción
A1	Conociendo el entorno	Indiv.	Vista general del proceso, manejo de interfaz, recursos, etc.
A2	Estudio individual	Indiv.	Estudiantes crean un mapa conceptual del caso asignado y una lista de preguntas o dudas
A3	Revisión individual	Indiv.	Estudiantes retroalimentan a expertos del mismo caso sobre su mapa conceptual y lista de preguntas de la actividad anterior
A4	Estudio de expertos	Colab.	Expertos analizan retroalimentación recibida de otros expertos y juntos crean un mapa conceptual del caso asignado y una lista de preguntas o dudas
A5	Revisión jigsaw	Colab.	Grupos <i>jigsaw</i> retroalimentan a grupos de expertos sobre su mapa conceptual y lista de preguntas de la actividad anterior
A6	Estudio <i>jigsaw</i>	Colab.	Grupos de <i>jigsaw</i> analizan retroalimentación recibida de otros grupos <i>jigsaw</i> y juntos crean un mapa conceptual del caso asignado y una lista de preguntas o dudas
A7	Revisión <i>jigsaw</i> nivel 2	Colab.	Grupos extendidos de <i>jigsaw</i> retroalimentan a grupos <i>jigsaw</i> sobre su mapa conceptual y lista de preguntas de la actividad anterior
A8	Estudio <i>jigsaw</i> nivel 2	Colab.	Grupos extendidos de <i>jigsaw</i> analizan retroalimentación recibida de otros grupos extendidos <i>jigsaw</i> y juntos crean un mapa conceptual del caso asignado y una lista de preguntas o dudas
A9	Conclusiones	Colab.	Todos los estudiantes analizan la retroalimentación recibida por grupos extendidos <i>jigsaw</i> y juntos crean un mapa conceptual consensuado y las conclusiones finales

Tabla 4.11: Resumen de las actividades de aprendizaje correspondientes a la técnica de aprendizaje colaborativo basada en el caso MOSAIC.

A continuación, ya en el rol de *jigsaw*, la secuencia de herramientas continúa con 3 herramientas simultáneas asociadas a la actividad de retroalimentación de *jigsaw*: el visor de documentos (*H12*), la pizarra colaborativa (*H13*), y el editor de texto colaborativo (*H14*). Las herramientas *H13* y *H14* son usadas para retroalimentar a los expertos sobre sus mapas conceptuales (*D4*) y sobre el listado de dudas o preguntas creados por ellos en la actividad anterior (*D5*). En la Tabla 4.12 se resume la descripción de artefactos abstractos que aparecen en el diseño de la Figura 4.16, mientras que en la Tabla 4.13 y en la Tabla 4.14 se resume la descripción de herramientas abstractas del diseño de la Figura 4.16. El documento del flujo de datos de esta situación de aprendizaje puede ser consultado en el apéndice C.

4. Evaluación de LeadFlow4LD

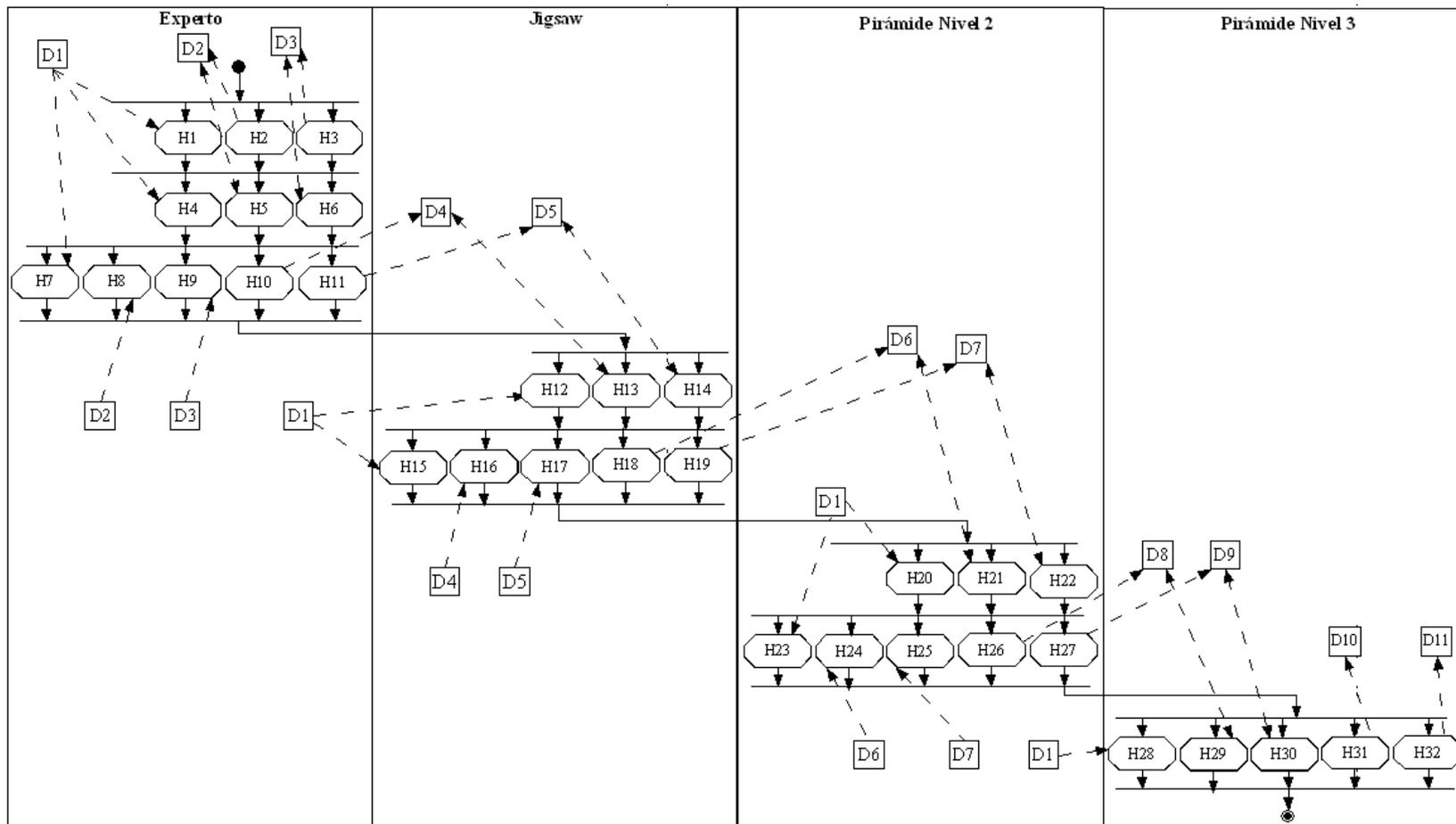


Figura 4.16: Diseño del flujo de datos correspondiente a la técnica de aprendizaje colaborativo basada en el caso MOSAIC.

4. Evaluación de LeadFlow4LD

Actividad	Dato	Descripción del Artefacto
A1. Conociendo el entorno	D1	Artículos
A2. Mapa Conceptual Individual de expertos	D2	Mapa conceptual individual de expertos + retroalimentación individual de expertos
	D3	Preguntas individuales de expertos + retroalimentación individual de expertos
A4. Mapa conceptual colaborativo del grupo de expertos	D4	Mapa conceptual colaborativo de expertos + retroalimentación del grupo <i>jigsaw</i>
	D5	Preguntas colaborativas de expertos + retroalimentación del grupo <i>jigsaw</i>
A6. Mapa conceptual colaborativo del grupo <i>jigsaw</i>	D6	Mapa conceptual colaborativo del grupo <i>jigsaw</i> + retroalimentación del grupo de nivel 2 de la pirámide
	D7	Preguntas colaborativas del grupo <i>jigsaw</i> + retroalimentación del grupo de nivel 2 de la pirámide
A8. Mapa conceptual colaborativo del nivel 2 de la pirámide	D8	Mapa conceptual colaborativo del grupo de nivel 2 de la pirámide + retroalimentación del grupo de nivel 3 de la pirámide (toda la clase)
	D9	Preguntas colaborativas del grupo de nivel 2 de la pirámide + retroalimentación del grupo de nivel 3 de la pirámide (toda la clase.)
A9. Conclusiones	D10	Mapa conceptual colaborativo consensual de toda la clase
	D11	Conclusiones finales

Tabla 4.12: Descripción de los artefactos abstractos referidos en el diseño del flujo de datos del caso MOSAIC.

Actividad	Hta.	Descripción de la Herramienta	Tipo de Herramienta
A2. Mapa Conceptual Individual de expertos	H1	Visor de documentos	Navegador
	H2	Editor de mapas conceptuales	Pizarra individual
	H3	Editor de texto	Editor individual
A3. Retroalimentación individual del grupo de expertos	H4	Visor de documentos	Navegador
	H5	Editor de mapas conceptuales	Pizarra individual
	H6	Editor de texto	Editor individual
A4. Mapa conceptual colaborativo del grupo de expertos	H7	Visor de documentos	Navegador
	H8	Editor de mapas conceptuales	Pizarra colaborativa
	H9	Editor de texto	Editor colaborativo
	H10	Editor de mapas conceptuales	Pizarra colaborativa
A5. Retroalimentación individual del grupo <i>jigsaw</i>	H11	Editor de texto	Editor colaborativo
	H12	Visor de documentos	Navegador
	H13	Editor de mapas conceptuales	Pizarra colaborativa
	H14	Editor de texto	Editor colaborativo

Tabla 4.13: Descripción de herramientas abstractas referidas en el diseño del flujo de datos del caso MOSAIC.

Documento de la Coordinación de Flujos

La Tabla 4.15 describe la relación entre el flujo de aprendizaje y el flujo de datos correspondiente al conjunto de técnicas de aprendizaje colaborativo que caracterizan el caso MOSAIC. La especificación de este documento se puede consultar en el apéndice C.

4. Evaluación de LeadFlow4LD

Actividad	Hta.	Descripción de la Herramienta	Tipo de Herramienta
A6. Mapa conceptual colaborativo del grupo <i>jigsaw</i>	H15	Visor de documentos	Navegador
	H16	Editor de mapas conceptuales	Pizarra colaborativa
	H17	Editor de texto	Editor colaborativo
	H18	Editor de mapas conceptuales	Pizarra colaborativa
	H19	Editor de texto	Editor colaborativo
A7. Retroalimentación colaborativa del nivel 2 de la pirámide	H20	Visor de documentos	Navegador
	H21	Editor de mapas conceptuales	Pizarra colaborativa
	H22	Editor de texto	Editor colaborativo
A8. Retroalimentación colaborativa del nivel 2 de la pirámide	H23	Visor de documentos	Navegador
	H24	Editor de mapas conceptuales	Pizarra colaborativa
	H25	Editor de texto	Editor colaborativo
	H26	Editor de mapas conceptuales	Pizarra colaborativa
A9. Conclusiones	H28	Visor de documentos	Navegador
	H29	Editor de mapas conceptuales	Pizarra colaborativa
	H30	Editor de texto	Editor colaborativo
	H31	Editor de mapas conceptuales	Pizarra colaborativa
	H32	Editor de texto	Editor colaborativo

Tabla 4.14: Descripción de herramientas abstractas referidas en el diseño del flujo de datos del caso MOSAIC (continuación).

Transiciones Maestras		Transiciones Esclavas	
De	A	De	A
Actividad de Aprendizaje Maestra	Actividad de Flujo de Trabajo Esclava	Actividad de Flujo de Trabajo Esclava	Actividad de Aprendizaje Maestra
A2	H1, H2, H3	H1, H2, H3	A2
A3	H4, H5, H6	H4, H5, H6	A3
A4	H7, H8, H9, H10, H11	H7, H8, H9, H10, H11	A4
A5	H12, H13, H14	H12, H13, H14	A5
A6	H15, H16, H17, H18, H19	H15, H16, H17, H18, H19	A6
A7	H20, H21, H22	H20, H21, H22	A7
A8	H23, H24, H25, H26, H27	H23, H24, H25, H26, H27	A8
A9	H28, H29, H30, H31, H32	H28, H29, H30, H31, H32	A9

Tabla 4.15: Coordinación entre el flujo de aprendizaje y flujo de datos correspondiente a la técnica de aprendizaje colaborativo basada en el caso MOSAIC.

4.6.2 Especificación de la instanciación de la estructura

Documento de Instanciación de Grupos

Para especificar la situación de aprendizaje 3.1 se requiere que el diseñador instruccional reflexione primero sobre la interacción colaborativa que tiene lugar en el flujo de datos. La Tabla 4.16 muestra la distribución de grupos en las distintas actividades de aprendizaje de acuerdo a los 12 estudiantes participantes y a la división del problema en 3 temas o partes.

Actividad	Id	Número de Grupos	Tamaño del Grupo
Conociendo el entorno	A1	12	1
Estudio individual	A2	12	1
Retroalimentación individual	A3	12	1
Estudio de expertos	A4	3	4
Retroalimentación <i>Jigsaw</i>	A5	4	3
Estudio de <i>Jigsaw</i>	A6	4	3
Retroalimentación nivel 2 de la pirámide	A7	2	6
Estudio nivel 2 de la pirámide	A8	2	6
Conclusiones finales	A9	1	12

Tabla 4.16: Número y tamaño de grupos participantes en la situación de aprendizaje 3.1.

Por tanto, a partir de la estructura del flujo de datos de la *Figura 4.16*, el diseñador instruccional define las instancias de datos y herramientas que se muestran en la *Tabla 4.17* y en la *Tabla 4.18*. Por ejemplo, como se muestra en la *Tabla 4.17*, el diseñador instruccional define tres instancias del artefacto que representa los casos o artículos de los expertos (D1); doce instancias del mapa conceptual individual (D2); tres instancias del mapa conceptual colaborativo creado por el grupo de expertos (D4); cuatro instancias del mapa conceptual colaborativo del grupo de *jigsaw*, etc. Similarmente, en la *Tabla 4.18*, el diseñador instruccional define doce instancias del editor del mapa conceptual individual (H2); tres instancias del editor colaborativo del mapa conceptual de expertos (H10), ya que se crean 3 grupos de expertos; cuatro instancias del editor colaborativo del mapa conceptual de *jigsaw* (H18), ya que se crea igual número de grupos de *jigsaw*, etc. Además, es importante darse cuenta de que también se requieren definir 48 instancias de la herramienta de edición del mapa conceptual individual (H5), ya que durante la actividad de retroalimentación individual cada participante debe revisar y retroalimentar individualmente a cada uno de sus tres compañeros expertos, además de poder acceder a su propio mapa conceptual. En resumen, para esta situación de aprendizaje colaborativo, el diseñador instruccional o el educador requieren definir un total de 47 instancias de datos y 300 instancias de herramientas, además de especificar la relación que guardan éstas entre sí, lo que da una idea de la complejidad de la situación de aprendizaje y la necesidad de su representación computacional. Más adelante se discute esta dificultad de especificar un gran número de instancias, por parte del diseñador instruccional y/o el educador.

Hasta aquí se han definido el número de instancias de datos y herramientas, pero aún falta definir la relación entre éstas, la cual se puede consultar en el apéndice C.2. Un ejemplo de los datos de entrada y salida de la herramienta individual *H2* (mapa conceptual) se muestra en la *Tabla 4.19*. Aquí, la instancia de la herramienta *H2-1* corresponde al editor del mapa conceptual individual del participante *p1*, por lo que la salida de esta herramienta es el artefacto *D1-1*. Sin embargo, en el documento de instanciación de grupos solo se especifica la relación entre instancias de datos y herramientas, mientras que la asignación de participantes a instancias de herramientas tiene lugar más adelante en el documento de población de grupos. El documento de instanciación de grupos de esta situación de aprendizaje puede ser consultado en el apéndice C.

4. Evaluación de LeadFlow4LD

Dato		Instancias	
ID	Cant.	Id	Descripción
D1	3	D1-1	Artículo del grupo experto 1
		D1-2	Artículo del grupo experto 2
		D1-3	Artículo del grupo experto 3
D2	12	D2-1	Mapa concep. del participante 1 + retroalimentación de los participantes 4, 7 y 10
		D2-2	Mapa concep. del participante 2 + retroalimentación de los participantes 5, 8 y 11
		D2-3	Mapa concep. del participante 3 + retroalimentación de los participantes 6, 9 y 12
		D2-4	Mapa concep. del participante 4 + retroalimentación de los participantes 1, 7 y 10
		D2-5	Mapa concep. del participante 5 + retroalimentación de los participantes 2, 8 y 11
		D2-6	Mapa concep. del participante 6 + retroalimentación de los participantes 3, 9 y 12
		D2-7	Mapa concep. del participante 7 + retroalimentación de los participantes 1, 4 y 10
		D2-8	Mapa concep. del participante 8 + retroalimentación de los participantes 2, 5 y 11
		D2-9	Mapa concep. del participante 9 + retroalimentación de los participantes 3, 6 y 12
		D2-10	Mapa concep. del participante 10 + retroalimentación de los participantes 1, 4 y 7
		D2-11	Mapa concep. del participante 11 + retroalimentación de los participantes 2, 5 y 8
		D2-12	Mapa concep. del participante 12 + retroalimentación de los participantes 3, 6 y 9
D3	12	D3-1	Preguntas del participante 1 + retroalimentación de los participantes 4, 7 y 10
		D3-2	Preguntas del participante 2 + retroalimentación de los participantes 5, 8 y 11
		D3-3	Preguntas del participante 3 + retroalimentación de los participantes 6, 9 y 12
		D3-4	Preguntas del participante 4 + retroalimentación de los participantes 1, 7 y 10
		D3-5	Preguntas del participante 5 + retroalimentación de los participantes 2, 8 y 11
		D3-6	Preguntas del participante 6 + retroalimentación de los participantes 3, 9 y 12
		D3-7	Preguntas del participante 7 + retroalimentación de los participantes 1, 4 y 10
		D3-8	Preguntas del participante 8 + retroalimentación de los participantes 2, 5 y 11
		D3-9	Preguntas del participante 9 + retroalimentación de los participantes 3, 6 y 12
		D3-10	Preguntas del participante 10 + retroalimentación de los participantes 1, 4 y 7
		D3-11	Preguntas del participante 11 + retroalimentación de los participantes 2, 5 y 8
		D3-12	Preguntas del participante 12 + retroalimentación de los participantes 3, 6 y 9
D4	3	D4-1	Mapa concep. colab. del grupo de expertos 1 + retro. de los grupos de <i>jigsaw</i> 1 a 4
		D4-2	Mapa concep. colab. del grupo de expertos 2 + retro. de los grupos de <i>jigsaw</i> 1 a 4
		D4-3	Mapa concep. colab. del grupo de expertos 3 + retro. de los grupos de <i>jigsaw</i> 1 a 4
D5	3	D5-1	Preguntas colab. del grupo de expertos 1 + retro. de los grupos de <i>jigsaw</i> 1 a 4
		D5-2	Preguntas colab. del grupo de expertos 2 + retro. de los grupos de <i>jigsaw</i> 1 a 4
		D5-3	Preguntas colab. del grupo de expertos 3 + retro. de los grupos de <i>jigsaw</i> 1 a 4
D6	4	D6-1	Mapa concep. colab. del gpo. <i>jigsaw</i> 1 + retro. del gpo. 1 del nivel 2 de la pirámide
		D6-2	Mapa concep. colab. del gpo. <i>jigsaw</i> 2 + retro. del gpo. 1 del nivel 2 de la pirámide
		D6-3	Mapa concep. colab. del gpo. <i>jigsaw</i> 3 + retro. del gpo. 2 del nivel 2 de la pirámide
		D6-4	Mapa concep. colab. del gpo. <i>jigsaw</i> 4 + retro. del gpo. 2 del nivel 2 de la pirámide
D7	4	D7-1	Preguntas colab. del grupo de <i>jigsaw</i> 1 + retro. del grupo 1 del nivel 2 de la pirámide
		D7-2	Preguntas colab. del grupo de <i>jigsaw</i> 2 + retro. del grupo 1 del nivel 2 de la pirámide
		D7-3	Preguntas colab. del grupo de <i>jigsaw</i> 3 + retro. del grupo 2 del nivel 2 de la pirámide
		D7-4	Preguntas colab. del grupo de <i>jigsaw</i> 4 + retro. del grupo 2 del nivel 2 de la pirámide
D8	2	D8-1	Mapa conceptual colaborativo del grupo 1 del nivel 2 de la pirámide
		D8-2	Mapa conceptual colaborativo del grupo 2 del nivel 2 de la pirámide
D9	2	D9-1	Preguntas colaborativas del grupo 1 del nivel 2 de la pirámide
		D9-2	Preguntas colaborativas del grupo 2 del nivel 2 de la pirámide
D10	1	D10-1	Mapa conceptual colaborativo consensual de toda la clase
D11	1	D11-1	Conclusiones consensuadas de toda la clase

Tabla 4.17: Resumen de instancias de artefactos definidas en la situación de aprendizaje 3.1.

4. Evaluación de LeadFlow4LD

Hta.	Inst. Cant.	Descripción	Hta.	Inst. Cant.	Descripción
H1	12	Visor de artículos individual	H17	12	Editor colaborativo de preguntas de expertos por grupo de <i>jigsaw</i> (3 x 4)
H2	12	Editor individual de mapa conceptual	H18	4	Editor colaborativo de mapa conceptual de grupos de <i>jigsaw</i>
H3	12	Editor individual de preguntas	H19	4	Editor colaborativo de preguntas de grupos de <i>jigsaw</i>
H4	12	Visor de artículos individual	H20	6	Visor de artículos por grupo del nivel 2 de la pirámide (3 x 2)
H5	48	Editor individual de mapa conceptual de expertos (12 x 4)	H21	4	Editor colaborativo de mapa conceptual de grupos de <i>jigsaw</i> por grupo del nivel 2 de la pirámide (2 x 2)
H6	48	Editor individual de preguntas de expertos (12 x 4)	H22	4	Editor colaborativo de preguntas de grupos de <i>jigsaw</i> por grupo del nivel 2 de la pirámide (2 x 2)
H7	3	Visor de artículos por grupo de expertos	H23	6	Visor de artículos por grupo del nivel 2 de la pirámide (3 x 2)
H8	12	Editor colaborativo de mapa conceptual de expertos por grupo de expertos (4 x 3)	H24	4	Editor colaborativo de mapa conceptual de grupos de <i>jigsaw</i> por grupo del nivel 2 de la pirámide (2 x 2)
H9	12	Editor colaborativo de preguntas de expertos por grupo de expertos (4 x 3)	H25	4	Editor colaborativo de preguntas de grupos de <i>jigsaw</i> por grupo del nivel 2 de la pirámide (2 x 2)
H10	3	Editor colaborativo de mapa conceptual por grupo de expertos	H26	2	Editor colaborativo de mapa conceptual por grupo del nivel 2 de la pirámide
H11	3	Editor colaborativo de preguntas por grupo de expertos	H27	2	Editor colaborativo de preguntas por grupo del nivel 2 de la pirámide
H12	12	Visor de artículos por grupo de <i>jigsaw</i> (3 x 4)	H28	3	Visor de artículos para toda la clase (3x1)
H13	12	Editor colaborativo de mapa conceptual de expertos por grupo de <i>jigsaw</i> (3 x 4)	H29	2	Editor colaborativo de mapa conceptual por grupo del nivel 2 de la pirámide
H14	12	Editor colaborativo de preguntas de expertos por grupo de <i>jigsaw</i> (3 x 4)	H30	2	Editor colaborativo de preguntas por grupo del nivel 2 de la pirámide
H15	12	Visor de artículos por grupo de <i>jigsaw</i> (3 x 4)	H31	1	Editor colaborativo de mapa conceptual para toda la clase
H16	12	Editor colaborativo de mapa conceptual de expertos por grupo de <i>jigsaw</i> (3 x 4)	H32	1	Editor colaborativo de preguntas para toda la clase

Tabla 4.18: Resumen de instancias de herramientas definidas en la situación de aprendizaje 3.1.

4. Evaluación de LeadFlow4LD

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H2	12	H2-1	-	D2-1	Editor del mapa conceptual del participante 1
		H2-2	-	D2-2	Editor del mapa conceptual del participante 2
		H2-3	-	D2-3	Editor del mapa conceptual del participante 3
		H2-4	-	D2-4	Editor del mapa conceptual del participante 4
		H2-5	-	D2-5	Editor del mapa conceptual del participante 5
		H2-6	-	D2-6	Editor del mapa conceptual del participante 6
		H2-7	-	D2-7	Editor del mapa conceptual del participante 7
		H2-8	-	D2-8	Editor del mapa conceptual del participante 8
		H2-9	-	D2-9	Editor del mapa conceptual del participante 9
		H2-10	-	D2-10	Editor del mapa conceptual del participante 10
		H2-11	-	D2-11	Editor del mapa conceptual del participante 11
		H2-12	-	D2-12	Editor del mapa conceptual del participante 12

Tabla 4.19: Instancias y argumentos de la herramienta H2 (editor de mapa conceptual) de la situación de aprendizaje 3.1.

Documento de Población de Grupos

La asignación de participantes a las actividades de aprendizaje de la situación de aprendizaje 3.1 se muestra en la Tabla 4.20, mientras que la asignación de participantes a las instancias de herramientas se puede consultar en el apéndice C. Un ejemplo de tal asignación se muestra en la Tabla 4.21, donde se muestra la asignación de cada uno de los tres grupos de expertos a las tres instancias de la pizarra colaborativa (H10) usadas para crear los mapas conceptuales durante la actividad del estudio de expertos. El documento de población de grupos de esta situación de aprendizaje se puede consultar en el apéndice C.

Situación ID	Roles del Flujo de Aprendizaje				
	Estudiante	Experto	Jigsaw	Pirámide Nivel 2	Pirámide Nivel 3
3.1	p1,p2,p3,p4, p5,p6,p7,p8, p9,p10,p11,p12	p1,p2,p3,p4, p5,p6,p7,p8, p9,p10,p11,p12	p1,p2,p3,p4, p5,p6,p7,p8, p9,p10,p11,p12	p1,p2,p3,p4, p5,p6,p7,p8, p9,p10,p11,p12	p1,p2,p3,p4, p5,p6,p7,p8, p9,p10,p11,p12

Tabla 4.20: Asignación de participantes a roles de aprendizaje correspondiente a la situación de aprendizaje 3.1.

Hta.	Id	Participantes
H10	H10-1	p1,p4,p7,p10
	H10-2	p2,p5,p8,p11
	H10-3	p3,p6,p9,p12

Tabla 4.21: Asignación de participantes a las instancias de herramientas H10 de la situación de aprendizaje 3.1.

4.6.3 Puesta en marcha de la situación de aprendizaje

El Flujo de Aprendizaje

La Figura 4.17 muestra una vista general del flujo de aprendizaje que el motor del flujo de aprendizaje entrega a cada uno de los doce participantes de la situación de aprendizaje. En

4. Evaluación de LeadFlow4LD

la *Figura 4.17* no se muestra el acto 1 que incluye la actividad de conocimiento del entorno. Los actos 2, 3, y 4 se componen de estructuras de actividad, las cuales a su vez se componen de una secuencia de actividades de aprendizaje, mientras que el acto 5 está formado solamente por la actividad de conclusiones.

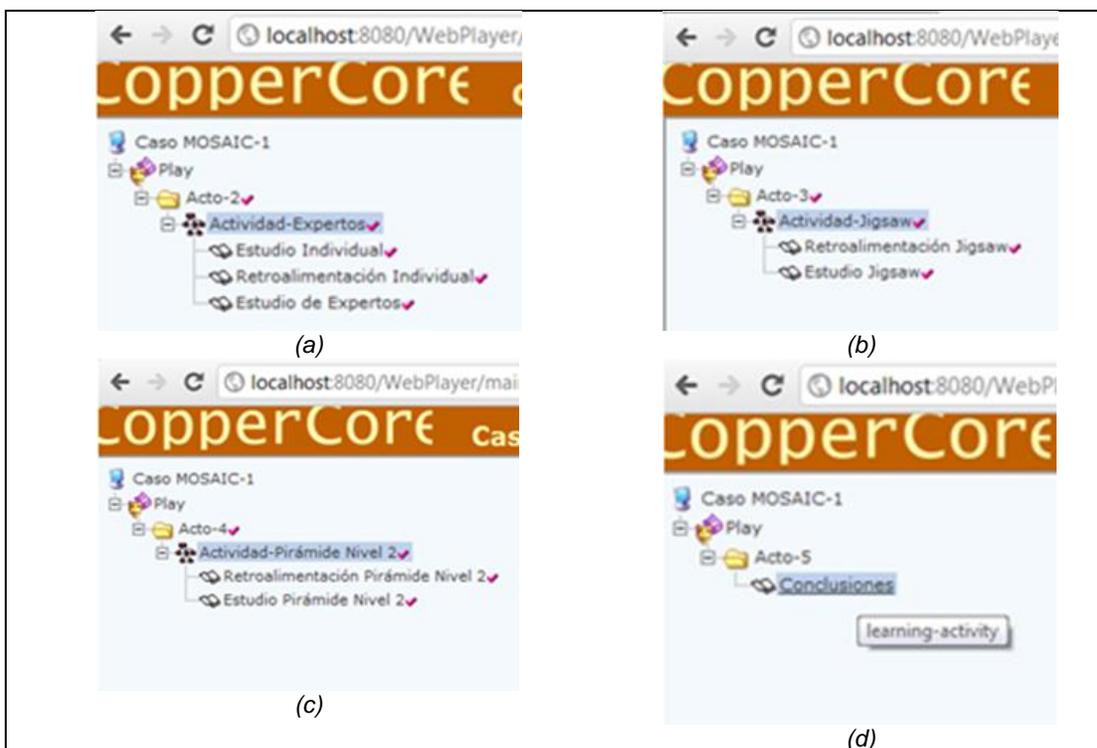


Figura 4.17: Vista general del flujo de aprendizaje presentado a cada uno de los doce participantes de la situación 3.1: (a) actividades de expertos (b) actividades de los grupos de jigsaw (c) actividades de grupos del nivel 2 de la pirámide e (d) conclusiones finales.

Automatización del Flujo de Datos

La automatización del flujo de datos de esta situación de aprendizaje, la ilustraremos en dos momentos de la puesta en marcha de la actividad de retroalimentación de los grupos de *jigsaw* (primera actividad del acto 3). En el primer momento ilustraremos la invocación automática de herramientas, así como la automatización de los datos de entrada y salida de las herramientas. Mientras que en el segundo momento ilustraremos la secuenciación de invocaciones de herramientas que ocurre en esta actividad de aprendizaje. Recordemos que en esta actividad, cada uno de los 4 grupos de *jigsaw*: $\{p1,p2,p3\}$, $\{p4,p5,p6\}$, $\{p7,p8,p9\}$, $\{p10,p11,p12\}$ deben revisar y retroalimentar cada uno de los 3 mapas conceptuales, así como a las listas de preguntas/dudas creados por los 4 grupos de expertos en la actividad anterior: $\{p1,p4,p7,p10\}$, $\{p2,p5,p8,p11\}$, $\{p3,p6,p9,p12\}$, $\{p3,p6,p9,p12\}$.

El primer momento de la automatización del flujo de datos se ilustra en la *Figura 4.18*. Cuando el participante *p1* inicia el flujo de datos desde la actividad de retroalimentación

4. Evaluación de LeadFlow4LD

jigsaw. En ese momento, el motor de flujo de trabajo lanza automáticamente una instancia del visor de documentos, una instancia de la pizarra colaborativa y una instancia del editor colaborativo (ver *Figura 4.18b*). Instancias similares de estas herramientas son lanzadas automáticamente por el motor de flujo de trabajo, después de que cualquiera de los integrantes del resto de los grupos de *jigsaw* inician su correspondiente flujo de datos.

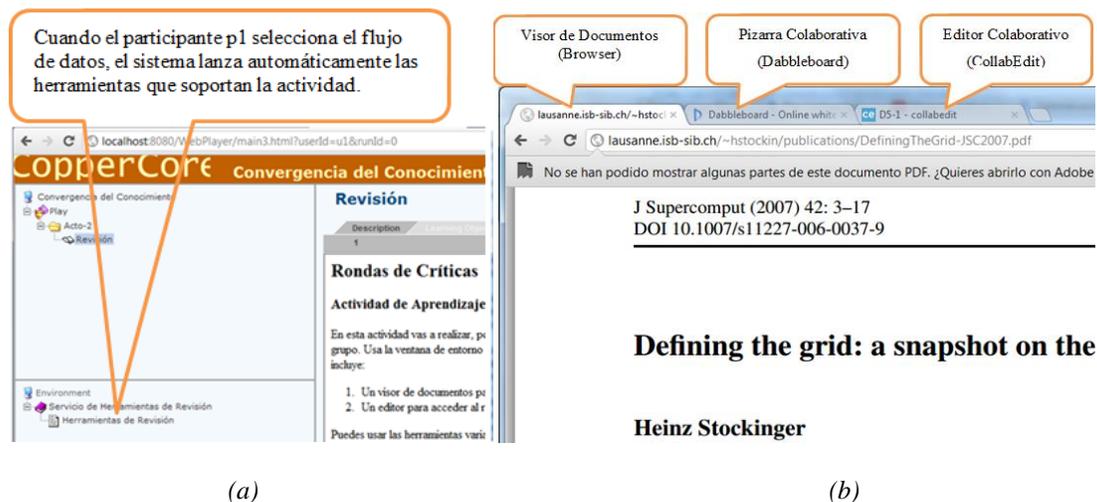


Figura 4.18: Captura del momento en que (a) el participante selecciona el flujo de datos (b) el sistema lanza automáticamente una instancia de un visor de documentos PDF, una instancia de una pizarra colaborativa, y una instancia de un editor colaborativo.

Es importante darse cuenta de que el visor de documentos en la *Figura 4.18b*, está inicializado con el artículo que el grupo de *jigsaw* $\{p1, p2, p3\}$ requiere consultar para mejorar su retroalimentación del mapa conceptual creado por el grupo de expertos $\{p1, p4, p7, p10\}$ en la actividad anterior. De modo similar, instancias de visores de documentos de otros participantes pertenecientes a otros grupos de *jigsaw* son inicializados automáticamente con otros artículos (en esta situación de aprendizaje son 3 artículos distintos, uno para cada grupo de expertos).

La *Figura 4.19* ilustra el momento en que el participante *p1* expande su pizarra colaborativa y advierte que ya está inicializada con el mapa conceptual que el grupo de *jigsaw* requiere para llevar a cabo la retroalimentación colaborativa. Similarmente, las pizarras de otros grupos de *jigsaw* son inicializadas automáticamente con los mapas conceptuales pertenecientes a otros grupos de expertos.

La *Figura 4.20* ilustra el momento en que el participante *p1* expande su editor colaborativo y advierte que ya está inicializado con la lista de preguntas/dudas que el grupo de *jigsaw* requiere para llevar a cabo la retroalimentación. Similarmente, los editores colaborativos de otros grupos de *jigsaw* se inicializan automáticamente con las preguntas/dudas correspondientes a otros grupos de expertos. Debe entenderse que esta inicialización de los editores ha sido especificada por los educadores en el documento de instanciación de grupos. Fue entonces que el diseñador instruccional y/o el educador

4. Evaluación de LeadFlow4LD

especificaron los artefactos de entrada y salida de cada una de las instancias de herramientas. Más adelante se discutirá el esfuerzo requerido para llevar a cabo tal especificación, por parte del diseñador instruccional y/o del educador.

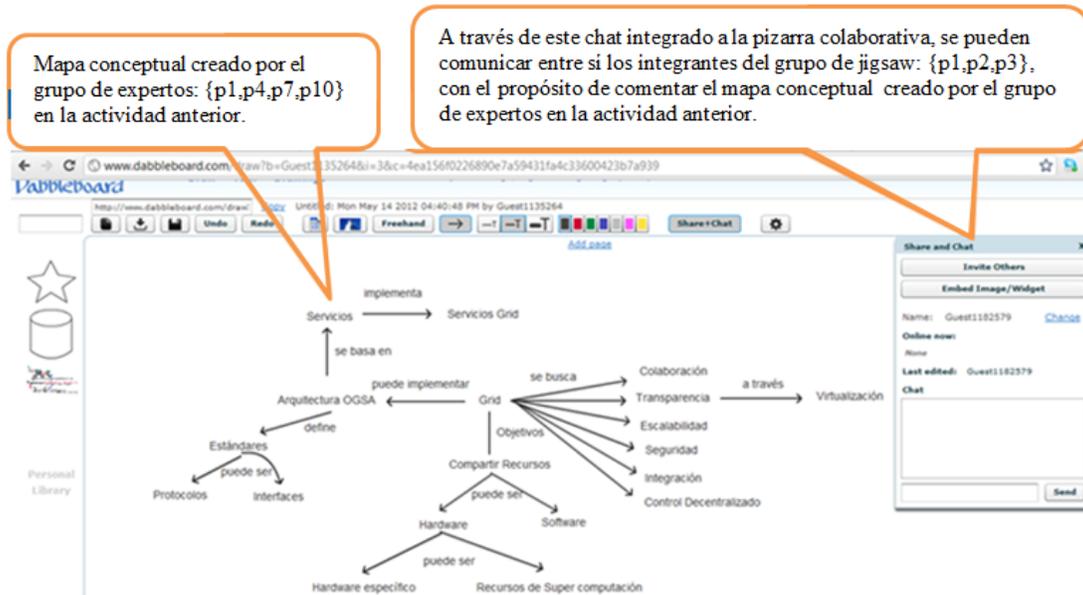


Figura 4.19: Captura de pantalla que muestra la pizarra colaborativa del grupo jigsaw {p1,p2,p3} automáticamente inicializada con el mapa conceptual del grupo de expertos {p1,p4,p7,p10}.

Lista de preguntas/dudas realizadas por el grupo de expertos: {p1,p4,p7,p10} en la actividad anterior.

A través de este chat integrado a la herramienta, se pueden comunicar entre si los integrantes del grupo de jigsaw: {p1,p2,p3}, con el objetivo de responder colaborativamente a las preguntas del grupo de expertos.

Figura 4.20: Captura de pantalla que muestra el editor colaborativo del grupo jigsaw {p1,p2,p3} automáticamente inicializado con la lista de preguntas/dudas del grupo de expertos {p1,p4,p7,p10}.

El segundo momento usado para ilustrar la automatización del flujo de datos se produce cuando cada uno de los grupos de *jigsaw* debe retroalimentar, por turnos, a cada uno de los tres grupos de expertos. En el primer turno, el grupo *jigsaw* {*p1,p2,p3*} retroalimenta primero al grupo de expertos {*p1,p4,p7,p10*}, después, en un segundo turno, al grupo de expertos {*p2,p5,p8,p11*} y finalmente, en un tercer turno, al grupo de expertos {*p3,p6,p9,p12*}. Al mismo tiempo y de forma similar, el grupo *jigsaw* {*p4,p5,p6*} retroalimenta primero al grupo de expertos {*p1,p4,p7,p10*}, después al grupo {*p2,p5,p8,p11*} y por último al grupo {*p3,p6,p9,p12*}. Similarmente, el grupo *jigsaw* {*p1,p2,p3*} retroalimenta primero al grupo de expertos {*p1,p4,p7,p10*}, después al grupo {*p2,p5,p8,p11*} y por último al grupo {*p3,p6,p9,p12*}.

Con este propósito, y de acuerdo con lo especificado, el motor de flujo de trabajo lanza automáticamente, y también por turnos, instancias de herramientas para cada uno de los cuatro grupos de *jigsaw*. Así, mientras unos grupos ya terminaron de revisar y retroalimentar a sus 3 grupos de expertos, otros quizás aún no terminan de hacerlo con el primero. Sin embargo, ninguno de los 12 participantes en la actividad de retroalimentación de *jigsaw* puede continuar con la siguiente actividad de aprendizaje (el estudio de *jigsaw*) hasta que los cuatro grupos de *jigsaw* hayan terminado todos y cada uno de ellos sus herramientas.

4.6.4 Discusión

A través de la puesta en marcha del caso MOSAIC se pudo observar, por un lado, que el motor de flujo de trabajo se encargó de automatizar la invocación de herramientas, así como los datos de entrada y salida de las herramientas, incluso en esta situación de aprendizaje compleja en donde intervienen una diversidad de técnicas de aprendizaje colaborativo, una cantidad masiva de instancias de datos y herramientas, así como una compleja relación entre éstas. Por otro lado, se pudo observar también la automatización de la secuenciación de invocaciones de herramientas en una misma actividad de aprendizaje. Por tanto, de la puesta en marcha del caso MOSAIC es razonable afirmar que el método propuesto (LeadFlow4LD) es suficientemente expresivo para especificar la automatización del flujo de datos incluso en un escenario de aprendizaje colaborativo complejo, como este caso.

Por otra parte, la puesta en marcha del caso MOSAIC también puso en evidencia la enorme carga de trabajo que deben realizar el diseñador instruccional y/o el educador, para especificar los documentos de instanciación y de población de grupos. El diseñador instruccional o el educador deben especificar, en función del número de grupos, el número de instancias de datos y herramientas, los datos de entrada y salida de cada una de las instancias de herramientas, la asignación de participantes a las actividades de aprendizaje y a cada una de las instancias de herramientas. De acuerdo con esto, es altamente improbable que en un escenario MOSAIC auténtico (con estudiantes y educadores reales), y tomando en cuenta el estado actual de LeadFlow4LD, el educador esté dispuesto a llevar a cabo tal especificación masiva de datos de instanciación. Sin embargo, una solución a este problema puede ser su especificación automática o semiautomática, a través del soporte de herramientas de autoría.

4. Evaluación de LeadFlow4LD

Un educador podría utilizar una herramienta de autoría para, a partir de la estructura del flujo de datos, el número de grupos y patrones (predefinidos) de intercambio entre instancias de datos y herramientas, especificar de manera abstracta, una instancia de dicha estructura. Similarmente, se podrían definir y utilizar patrones de asignación de participantes a roles y a instancias de herramientas. Por ejemplo, WebCollage (Villasclaras-Fernández, Hernández-Leo, Asensio-Pérez & Dimitriadis, 2013) es un ejemplo de una herramienta de autoría que lleva a cabo esta asignación automática de participantes para el flujo de aprendizaje, pero no para el flujo de datos. Por tanto, la implementación en herramientas de autoría de las estrategias mencionadas, permitirían especificar automáticamente los documentos de instanciación y población de grupos, con mínima intervención del educador. En cualquier caso, esta dificultad encontrada en la instanciación, no debe afectar la evaluación de principal objetivo de esta tesis: la automatización del flujo de datos de un diseño de aprendizaje en escenarios de aprendizaje colaborativos. La tarea de especificar automáticamente los documentos de instanciación y población de grupos, se deja como un trabajo futuro de investigación de esta tesis.

Otro aspecto a discutir después de la puesta en marcha del caso MOSAIC, es la reutilización del diseño resultante. Parece ser que cuanto más compleja es una estructura de aprendizaje colaborativo, más difícil podría ser derivar nuevas situaciones de aprendizaje. ¿Qué tanto se podría reutilizar la estructura MOSAIC para poner en marcha otras situaciones de aprendizaje similares? Para responder a esta pregunta, la *Tabla 4.22* resume otras dos posibles situaciones de aprendizaje basadas o derivadas de la estructura descrita por el caso MOSAIC. Por un lado, en la situación de aprendizaje 3.2 participan 18 estudiantes, el universo del problema se divide en 3 temas o partes, y se forman igual número de grupos de expertos, esto es, 3 grupos de expertos de 6 estudiantes cada uno. Por otro lado, en la situación de aprendizaje 3.3 participan 6 estudiantes, el universo del problema se divide sólo en 2 partes, y se forman 2 grupos de expertos de 3 estudiantes cada uno. Por tanto, es razonable afirmar que incluso estructuras de aprendizaje colaborativo complejas, como es el caso MOSAIC, se pueden reutilizar para derivar nuevas situaciones de aprendizaje similares, como las mostradas en las mostradas en la *Tabla 4.22*.

Rol	Situación 3.2			Situación 3.3		
	#Grupos	Tamaño Grupo	Total Estudiantes	#Grupos	Tamaño Grupo	Total Estudiantes
Pirámide Nivel 3	1	18	18	1	6	6
Pirámide Nivel 2	2	9	18	2	3	6
Jigsaw	6	3	18	3	2	6
Experto	3	6	18	2	3	6
Estudiante	18	1	18	6	1	6

Tabla 4.22: Características de otras situaciones de aprendizaje que podrían ser derivadas de la técnica basada en el caso MOSAIC.

Más aún, si analizamos con más detalle las tres situaciones de aprendizaje colaborativo derivados de la estructura MOSAIC (situaciones 3.1 a 3.3), se pueden identificar y discutir aspectos interesantes. Existe al parecer una relación entre el número de partes en que se

divide el universo del problema del caso MOSAIC, el número y tamaño de los grupos de expertos, el número y tamaño de los grupos de jigsaw, etc. Por ejemplo, en la situación de aprendizaje 3.1, el universo del problema se divide en 3 partes, hay 3 grupos de expertos de 4 integrantes cada uno, 4 grupos de jigsaw de 3 integrantes cada uno, 2 grupos de jigsaw en el segundo nivel de la pirámide de 6 integrantes cada uno, y al final un solo grupo de 12 integrantes. En la situación de aprendizaje 3.2, el universo del problema se divide en 3 partes, hay 3 grupos de expertos de 6 integrantes cada uno, 6 grupos de jigsaw de 3 integrantes cada uno, 2 grupos de jigsaw en el segundo nivel de la pirámide de 9 integrantes cada uno, y al final un solo grupo de 18 integrantes. Mientras que en la situación de aprendizaje 3.3 el universo del problema se divide en 2 partes, hay 2 grupos de expertos de 3 integrantes cada uno, 3 grupos de jigsaw de 2 integrantes cada uno, 2 grupos de jigsaw en el segundo nivel de la pirámide de 3 integrantes cada uno, y al final un solo grupo de 6 integrantes.

Estas relaciones entre partes, número y tamaño de grupos, resultan interesantes por dos motivos. Primero, porque se ve claramente que es posible derivar situaciones de aprendizaje similares para una misma estructura de aprendizaje colaborativo compleja, como es la estructura que caracteriza al caso MOSAIC. Segundo, porque al parecer es posible definir de forma abstracta una situación de aprendizaje, con tan solo definir el número de partes en que se divide el universo del problema, el número y tamaño de los grupos de expertos, el número y tamaño de los grupos de jigsaw, etc. Esto aliviaría la necesidad de que todo el detalle de especificar la instanciación del flujo de datos recayera directamente en el diseñador instruccional o en el educador. Además, es importante hacer notar que es posible trasladar el diseño general MOSAIC a otras situaciones que comparten similares objetivos de aprendizaje. Esto es, la misma composición de técnicas bien conocidas de aprendizaje colaborativo, pueden ser reutilizadas para aprender sobre distintos temas o contextos, tales como el *grid*, la nube, economía, legislación laboral, etc.

Por tanto, aunque el caso MOSAIC no fue seleccionado para ilustrar la reutilización de una UoLeadFlow4LD. De la discusión anterior, es razonable afirmar que el segundo principal objetivo de esta tesis, esto es, la reutilización del diseño de aprendizaje resultante, también es posible de alcanzar con el método propuesto.

4.6.5 Comparación con otras soluciones

Se discuten a continuación dos alternativas de solución del caso MOSAIC, respecto del método propuesto. La primera es, de hecho, la usada en el caso de estudio real (ver sección 2.7.3). La segunda, está basada en la aproximación propuesta por Koper (2009) y Miao et al. (2005), la cual ha sido usada también como alternativa de solución en los dos ejemplos ilustrativos anteriores.

Por un lado, la descripción del flujo de datos del caso de estudio MOSAIC (estudio real) se llevó a cabo directamente en la descripción textual de las actividades, a través de instruir a los estudiantes sobre la gestión manual del flujo de datos. Sin embargo, con ésta aproximación, no solo no se alcanzó la automatización del flujo de datos a nivel de

herramientas (el primer objetivo de esta tesis), sino que emergieron algunos problemas, tales como un exceso de carga cognitiva, así como situaciones propensas a error en los estudiantes y profesores (Palomino-Ramírez, 2008). Además, el análisis del caso de estudio identificó también que el diseño de aprendizaje resultante incluía información de instancias, lo cual impide la reutilización del diseño (el segundo objetivo de esta tesis). Por tanto, con esta aproximación de describir el flujo de datos directamente en la descripción textual de la actividad, no es posible alcanzar ninguno de los dos de tres objetivos de esta tesis: automatización del flujo de datos y reusabilidad del diseño resultante.

Por otro lado, está la posibilidad de utilizar la aproximación propuesta por Koper (2009) y Miao et al. (2005). Sin embargo, con el objetivo de especificar la interacción colaborativa entre grupos, esto es, la interacción entre datos y actividades, sería necesario modificar el diseño de forma que se especifiquen actividades y roles, de acuerdo al número de grupos indicados en la *Tabla 4.16*. Por ejemplo, sería necesario especificar 12 actividades de conocimiento del entorno (A1) asociadas a 12 roles de estudiantes individuales; 12 actividades de estudio individual (A2), 12 actividades de retroalimentación individual (A3) y 3 actividades de estudio de expertos (A4) asociadas todas ellas a sus respectivos números de roles de experto; 4 actividades de retroalimentación jigsaw (A5) y 4 actividades de estudio jigsaw (A6) asociadas a 4 roles de jigsaw; 2 actividades de retroalimentación (A7) y dos actividades de estudio (A8), ambas correspondientes al nivel 2 de la pirámide, y asociadas al mismo número de roles: Pirámide Nivel 2. Como debe ser claro de este análisis, ésta aproximación no solo no permite alcanzar la automatización del flujo de datos a nivel de herramientas (el primer objetivo de esta tesis), sino que el diseño resultante no se podría reutilizar para distintas situaciones de aprendizaje en el que intervengan un diferente número de grupos (otro de los objetivos de esta tesis).

En cambio, con el método propuesto, se consiguió especificar una situación de aprendizaje colaborativo compleja, que involucra una jerarquía de técnicas de aprendizaje colaborativo. Durante la puesta en marcha del diseño de aprendizaje del caso MOSAIC, se pudo observar la automatización del flujo de datos, que incluye la invocación automática de herramientas, la automatización de los datos de entrada y salida de las herramientas, así como la secuenciación de invocaciones de herramientas en una misma actividad de aprendizaje. Además, de la discusión realizada anteriormente, se puede afirmar que el método propuesto puede ser reutilizado para derivar situaciones de aprendizaje similares, a partir de la misma estructura de aprendizaje colaborativo, incluso si esta es compleja. Por tanto, la comparación de las bondades ofrecidas por LeadFlow4LD con respecto a las soluciones alternativas mencionadas, contribuye a entender mejor el mérito de la solución propuesta.

4.7 Discusión global

En esta sección se discute el alcance de la solución propuesta con respecto a los objetivos de automatización del flujo de datos, la reusabilidad del diseño de aprendizaje y la interoperabilidad con sistemas y herramientas que sean conformes a estándares actuales.

Automatización del Flujo de Datos

De la puesta en marcha de las distintas situaciones de aprendizaje, correspondientes a distintas técnicas de aprendizaje colaborativo y de distintos niveles de complejidad, se puede afirmar que LeadFlow4LD permite especificar la automatización del flujo de datos. Es decir, permite especificar la invocación automática de herramientas, la automatización de los datos de entrada y salida de las herramientas, y la automatización de la secuenciación de la invocación de herramientas en una misma actividad de aprendizaje. Pero, ¿es suficientemente expresivo LeadFlow4LD?, ¿se puede especificar cualquier flujo de herramientas?, ¿se puede especificar cualquier relación entre datos y herramientas?, ¿cuál es el límite para especificar el flujo de datos con LeadFlow4LD? A continuación se busca responder a estas preguntas.

Como se afirmó en la sección 3.4.2, el flujo de datos se especifica en un lenguaje estándar de flujo de trabajo: XPDL. XPDL es un lenguaje de definición de procesos de flujo de trabajo (*workflow*) basado en actividad. La actividad de flujo de trabajo es el bloque fundamental que define la lógica de un proceso de flujo de trabajo. Esta lógica, muchas veces llamada lógica de negocio, puede ser secuencial, paralela, selectiva o iterativa. Tomando en cuenta que invocaciones de herramientas en un contexto educativo son modeladas como actividades de flujo de trabajo en el contexto *workflow*, entonces es razonable afirmar que con el método propuesto es posible especificar una lógica de invocaciones de herramientas que sea, de igual manera, secuencial, paralela, selectiva o iterativa, lo cual debe dar una clara idea de la capacidad expresiva de LeadFlow4LD para especificar el flujo de datos en un contexto educativo.

Por otra parte, está la expresividad de LeadFlow4LD para describir la relación entre artefactos y herramientas, esto es, la expresividad para especificar los datos de entrada y salida de las herramientas. Como se afirmó en la sección 3.4.2, los datos se especifican en XPDL como variables de flujo de trabajo *<DataFields>*, las herramientas como aplicaciones (*<Application>*), mientras que la relación entre datos y herramientas se especifica relacionando los datos con los parámetros reales (*<ActualParameter>*) y formales (*<FormalParameters>*) de las aplicaciones que representan las herramientas. Entonces, puesto que una herramienta puede definirse en términos de cero o más parámetros formales de entrada y/o de salida, y un artefacto puede ser usado como referencia de cualquier parámetro real (*actual parameter*) de una herramienta, ya sea de entrada y/o de salida, entonces es razonable afirmar que con el método propuesto es posible especificar cualquier relación entre artefactos y herramientas, lo cual da una idea clara de la capacidad expresiva de LeadFlow4LD para especificar la relación entre datos y herramientas en un contexto educativo. Por tanto, del razonamiento anterior, podemos concluir que LeadFlow4LD es lo suficientemente expresivo para describir el flujo de datos en escenarios de aprendizaje colaborativo.

Reutilización del Diseño de Aprendizaje Resultante

La puesta en marcha de las situaciones de aprendizaje 1.3 y 1.4, así como la discusión llevada a cabo en el resto de las situaciones de aprendizaje, ilustraron la capacidad del método propuesto para reutilizar, a distintos niveles, el diseño de aprendizaje especificado conforme a LeadFlow4LD. Primero, una UoLeadFlow4LD puede ser reutilizada a nivel estructural, a partir de crear distintas instancias de la estructura del flujo de datos. En tal caso, ninguno de los tres documentos estructurales requiere ser modificado. Segundo, una UoLeadFlow4LD puede ser reutilizada a un nivel situacional, a partir de la forma en que se distribuyen los participantes en las actividades de aprendizaje y en las instancias de herramientas. En tal caso, sólo el documento de población de grupos requiere ser modificado. Tercero, una UoLeadFlow4LD puede ser reutilizada, incluso a un segundo nivel situacional, simplemente con cambiar los estudiantes participantes. En tal caso, el documento de población de grupos es modificado reemplazando únicamente los nombres de los participantes, pero ni el número de participantes, ni la distribución de participantes, como en el caso anterior. Y por último, una UoLeadFlow4LD también puede ser reutilizada a nivel contextual, donde la misma UoLeadFlow4LD puede ser reutilizada en distintos contextos educativos: economía, tecnología, legislación laboral, etc. Por tanto, del razonamiento anterior, podemos concluir que un diseño de aprendizaje especificado con LeadFlow4LD se caracteriza por contar con distintos niveles de reutilización: estructural, dos niveles situacionales y contextual.

Interoperabilidad del Diseño de Aprendizaje Resultante con Sistemas y Herramientas

Uno de los objetivos de la solución propuesta, LeadFlow4LD, es mantener la mayor interoperabilidad posible con estándares actuales. Si bien es cierto que no se consigue el 100% de interoperabilidad con el método propuesto, sí se consiguen distintos niveles: interoperabilidad a nivel del flujo de aprendizaje, a nivel del flujo de datos, y a nivel de unidades de aprendizaje.

Por una parte, está la interoperabilidad de LeadFlow4LD a nivel del flujo de aprendizaje, la cual permite que el documento del flujo de aprendizaje de una UoLeadFlow4LD, pueda ser editado por herramientas de autoría, y ejecutado por motores y sistemas que sean conformes con el estándar IMS LD. Por otra parte, está la interoperabilidad de LeadFlow4LD a nivel del flujo de datos, la cual permite que los documentos del flujo de datos y de instanciación de grupos (instanciación del flujo de datos) puedan ser editados por herramientas de autoría, y ejecutados por motores y sistemas que sean conformes con el estándar de flujo de trabajo XPDL. El valor de estos dos niveles de interoperabilidad está en el hecho de que LeadFlow4LD está principalmente basado en los estándares IMS LD y XPDL, lo cual permite, por un lado, una reducción significativa en el esfuerzo de desarrollo de herramientas de autoría (ya que las herramientas pueden ser reutilizadas), y por otro lado, una reducción de la curva de aprendizaje mientras se usan herramientas basadas en dichos estándares. Por último, está la interoperabilidad de LeadFlow4LD a nivel de unidades de aprendizaje, la cual permite que el contenido de una UoLeadFlow4LD pueda ser editado por herramientas de

autoría que sean conformes al estándar IMS CP. El valor de este nivel de interoperabilidad está en el hecho de que el empaquetamiento de contenidos de una UoLeadFlow4LD se basa en el estándar de empaquetamiento de contenidos educativos de *e-learning* IMS CP, lo cual permite que unidades de aprendizaje pueden ser distribuidas de manera efectiva, e intercambiadas entre diseñadores instruccionales y educadores.

Desafortunadamente, dos de los documentos que conforman una UoLeadFlow4LD no siguen una especificación estándar, como es el caso del documento de coordinación de flujos y el documento de población de grupos. Sin embargo, en el primer caso, una especificación propietaria es necesaria para describir la interacción entre flujos, los cuales se especifican en lenguajes distintos, mientras que en el segundo caso, la especificación de la instanciación de procesos basados en el paradigma de flujo de trabajo (requisito para la población de grupos) es un problema abierto en LD (Caeiro-Rodríguez, 2008), incluso en el campo de flujo de trabajo (Decker & Mendling, 2008).

Por último, recordar que la puesta en marcha de las distintas situaciones de aprendizaje ha sido realizada a través del sistema prototipo, el cual tiene algunas limitaciones respecto del sistema “ideal” propuesto en el capítulo anterior y que no se ha implementado por cuestión de tiempo (ver secciones 3.6.1 y 3.6.5). Es importante hacer notar que el principal objetivo del sistema prototipo es el de ilustrar la factibilidad de poner en marcha un diseño de aprendizaje especificado conforme a LeadFlow4LD, mientras que detalles, tales como la usabilidad de la interfaz de usuario, o la complejidad para generar los documentos del flujo de aprendizaje y flujo de trabajo sincronizados, no deben ser tomados en cuenta, ya que con una implementación a la medida, esto no sucedería.

4.7.1 Efectos colaterales

A continuación, y como consecuencia de alcanzar los objetivos de automatización y reutilización, se presentan y discuten los siguientes efectos colaterales que fueron identificados durante la puesta en marcha de las distintas situaciones de aprendizaje.

Flexibilidad del Uso de Herramientas

El primer efecto colateral identificado está relacionado con lo que podría ser una imposición de las herramientas. Al estar especificadas las herramientas en el flujo de datos, los participantes no pueden elegir usar sus propias herramientas en una actividad de aprendizaje, ya que el motor de flujo de trabajo les entrega automáticamente las herramientas que soportan la actividad. Esta problemática se podría abordar de distintas maneras. La primera, permitiendo que cada participante seleccione su herramienta antes de la puesta en marcha de la situación de aprendizaje. En lugar de que sea el diseñador instruccional quien seleccione la herramienta durante la preparación de la puesta en marcha del diseño de aprendizaje, sea cada participante quien lleva a cabo dicha selección. La segunda, especificando distintas opciones de herramientas en el flujo de datos. En lugar de especificar una única herramienta, se podrían especificar distintas herramientas en paralelo, donde el

4. Evaluación de LeadFlow4LD

usuario podría elegir (en tiempo de ejecución) la más adecuada. Y la tercera, permitiendo que cada participante utilice sus propias herramientas, las cuales no estarían especificadas en el flujo de datos. Sin embargo, sería necesario especificar servicios para incorporar al entorno de flujo de trabajo los artefactos producidos por las herramientas externas al flujo de trabajo y que fueron usadas por los usuarios.

Consciencia de Autoría de los Artefactos

El segundo efecto colateral identificado está relacionado con lo que podría ser una falta de consciencia de los participantes sobre el origen o propiedad de los artefactos inicializados automáticamente por las herramientas. Por ejemplo, en la *Figura 4.2* la herramienta presentada al participante no dice nada sobre quién es el propietario del mapa conceptual. Esto puede ser un problema, si el educador considera que la retroalimentación mejora si el estudiante conoce de antemano el propietario de un artefacto. Esta problemática se podría abordar de dos maneras. La primera, a través de revisar el contrato entre los motores de flujo de trabajo y las herramientas que invocan. Si la autoría de los artefactos es importante para el educador, se podría prever esto en el contrato, aunque aquí surgiría la problemática de cómo imponer requisitos a herramientas de terceros, como se discute en (Alario-Hoyos, 2012). Y la segunda, a través de solicitar directamente a los participantes que indiquen explícitamente la autoría del artefacto, como se ilustra en la *Figura 4.21* (señalado arriba a la izquierda).

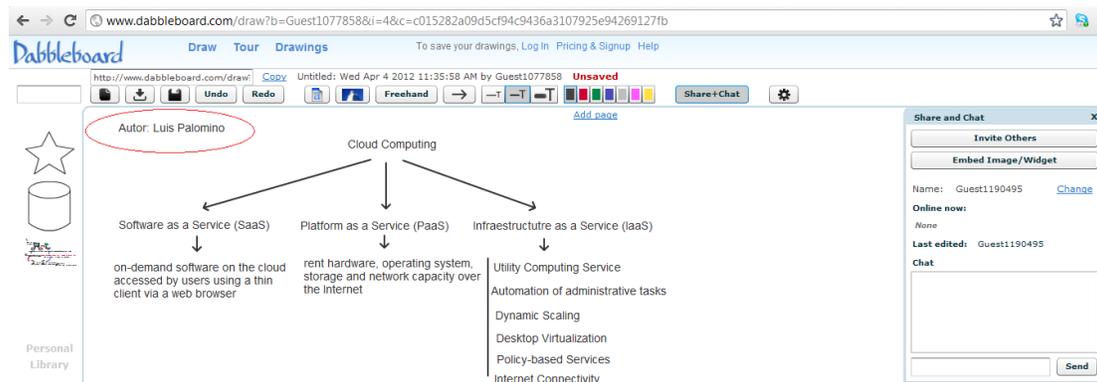


Figura 4.21: Pizarra colaborativa entregada automáticamente por el sistema a la pareja {p1,p2} e inicializada con el mapa conceptual de la pareja {p3,p4}. Note que en el mismo contenido se indica la autoría del mapa conceptual.

4.8 Conclusiones

En el capítulo 2 se introdujo el problema de especificar el flujo de datos en diseño de aprendizaje, a través de estudiar situaciones de aprendizaje basadas en tres técnicas de aprendizaje colaborativo: revisión entre pares, convergencia del conocimiento y el caso MOSAIC. Después, en el capítulo 3 se propuso un método basado en la composición del flujo de aprendizaje especificado en el lenguaje estándar de diseño de aprendizaje (IMS LD) y el flujo de datos, especificado en un lenguaje estándar de flujo de trabajo (XPDL). En este

capítulo, con el objetivo de evaluar la expresividad del método propuesto y la reutilización de los diseños de aprendizaje así creados, se seleccionaron, especificaron y se pusieron en marcha distintas situaciones de aprendizaje colaborativo derivadas de las técnicas antes mencionadas.

Los resultados de la puesta en marcha de las distintas situaciones de aprendizaje seleccionadas muestran que es factible poner en marcha un diseño de aprendizaje especificado conforme al método propuesto (LeadFlow4LD). En primer lugar, se puede afirmar que LeadFlow4LD es suficientemente expresivo para especificar la automatización del flujo de datos, en particular la invocación automática de herramientas, la automatización de los datos de entrada y salida de herramientas, así como la secuenciación de invocaciones de herramientas en una misma actividad de aprendizaje. Por tanto, se esperaría que los participantes en un escenario de aprendizaje colaborativo no tengan que preocuparse por localizar y recuperar los artefactos y las herramientas que requieren utilizar en la actividad, ya que el motor de flujo de trabajo se los entregaría automáticamente. En segundo lugar, se puede afirmar que una UoLeadFlow4LD puede ser reutilizada en distintas situaciones y contextos de aprendizaje colaborativo, ya que sólo los documentos de instanciación y/o de población de grupos necesitarían ser modificados. Y en tercer lugar, se puede afirmar, que si bien LeadFlow4LD no es 100% interoperable con estándares actuales, sí que mantiene distintos niveles de interoperabilidad con IMS CP, IMS LD y XPDL, tales como a nivel de intercambio de UoLeadFlow4LDs, a nivel de edición y puesta en marcha del flujo de aprendizaje, y a nivel de edición y puesta en marcha del flujo de datos.

Se puede concluir que el método propuesto (LeadFlow4LD) se caracteriza, respecto de otras aproximaciones existentes, por las siguientes:

Ventajas:

1. **Automatización de la invocación de herramientas.** Se esperaría que los participantes de una actividad de aprendizaje no tengan que preocuparse de localizar y recuperar las herramientas a utilizar en una actividad de aprendizaje, ya que el sistema se encargaría de entregarlas automáticamente.
2. **Automatización de los artefactos de entrada y salida de herramientas.** Se esperaría que los participantes de una actividad de aprendizaje no tengan que preocuparse de localizar, recuperar y almacenar los artefactos que requieren las herramientas en una actividad de aprendizaje, ya que el sistema se encargaría de entregarlos automáticamente a las herramientas, así como de almacenarlos automáticamente cuando termine la utilización de éstas.
3. **Reducción de la carga cognitiva de los participantes en las actividades de aprendizaje.** Se esperaría que la automatización del flujo de datos contribuya a reducir la carga cognitiva de los participantes de la actividad, permitiendo que éstos se centren en realizar las actividades de aprendizaje.
4. **Reducción de situaciones propensas de error.** Se esperaría que la automatización del flujo de datos elimine la posibilidad de errores por parte de los participantes al

tratar de recuperar los artefactos y las herramientas que requieren utilizar en una actividad de aprendizaje.

5. **Reutilización de la UoLeadFlow4LD.** Se esperaría que un diseño de aprendizaje especificado conforme a LeadFlow4LD pueda ser reutilizado para distintas situaciones y contextos de aprendizaje derivadas de la misma estructura de aprendizaje colaborativo.

Desventajas:

1. **Rigidez en la elección de herramientas.** La automatización de la invocación de herramientas en una actividad de aprendizaje, no facilitaría a los participantes la elección de sus propias herramientas.
2. **Pérdida de conciencia sobre la propiedad de los artefactos.** La automatización de los datos de entrada y salida de las herramientas no facilitaría a los participantes identificar el autor de los artefactos con los que son inicializadas las herramientas.
3. **Pérdida de conciencia de las tareas asociadas a las herramientas en una secuencia de herramientas en actividades de grano grueso.** Automatizar la secuenciación de herramientas en una misma actividad de aprendizaje puede provocar que los participantes pierdan conciencia sobre la tarea a realizar con alguna una herramienta de la secuencia. Las tareas de una actividad de aprendizaje de grano grueso son descritas textualmente en la descripción de actividad, por lo que no pueden ser interpretadas computacionalmente.
4. **Imposibilidad para modificar la situación de aprendizaje en tiempo de ejecución.** Con el método propuesto no se pueden hacer cambios al diseño de aprendizaje, una vez que se ha puesto en marcha una UoLeadFlow4LD.

5 Conclusiones y trabajo futuro

En este capítulo se recogen las principales conclusiones extraídas de la tesis a la vez que se repasan las contribuciones más importantes de la misma. Estas cuestiones y contribuciones son tratadas en la sección 5.1, mientras que en la sección 5.2 se identifican las líneas de trabajo futuro que es posible abordar a partir de esta tesis.

5.1 Conclusiones

El campo de Diseño de Aprendizaje (campo LD) tiene su verdadero impulso a partir del surgimiento de EML usados para representar computacionalmente procesos de aprendizaje en *e-learning*, y no solamente contenidos educativos (Kopper, 2001). Por este tiempo, el campo LD se relacionaba directamente con IMS LD, el cual es considerado su verdadero impulsor. Sin embargo, en la búsqueda de una visión más amplia del campo LD (más allá de IMS LD), algunos investigadores se han alejado de sus orígenes en IMS LD (McLean & Scot, 2007), lo que ha generado que el rápido crecimiento técnico y práctico supere el desarrollo teórico del campo (Goodyear & Dimitriadis, 2013). Ejemplo de ello es la falta de claridad que existe actualmente: el propio campo LD es llamado diseño de aprendizaje, diseño instruccional, diseño curricular, diseño educativo, diseño para aprendizaje y aprendizaje basado en diseño (Mor & Brock, 2012). En esta búsqueda de un nuevo campo LD, se ha llegado al punto en que es necesario definir su marco y sus fundamentos (Mor & Brock, 2012) (Dalziel, 2012) (Cameron, 2009), ya que de lo contrario se corre el riesgo de construir conocimiento sobre bases débiles (Goodyear & Dimitriadis, 2013).

Por tanto, era necesario definir el marco en el que se llevaría a cabo esta investigación, la cual podría ser o bien dentro de un marco existente, o bien proponer un nuevo marco. En este contexto, nos referimos a las aproximaciones de Diseño de Aprendizaje (aproximaciones LD) como los marcos que buscan implementar los objetivos del campo. Ejemplos de estas aproximaciones son la aproximación LD basada en flujo de trabajo aplicado al dominio educativo, que es de hecho el marco de IMS LD; la aproximación LD basada en un marco ontológico (Rius, Conesa, García-Barriocanal & Sicilia, 2014); la aproximación LD basada en un método de planeación de Red Jerárquica de Tareas (HTN) (Challco, Gerosa, Bittencourt & Isotani, 2014); la aproximación LD basada en Redes de Petri (Vidal, Lama, Sánchez & Bugarín, 2008), etc. Entonces, en esta tesis no se propone una nueva aproximación para el campo LD, sino que se busca resolver un problema en un marco existente: la aproximación de LD basada en flujo de trabajo aplicado al dominio educativo.

De esta forma, tomando en cuenta la madurez del campo de Flujo de Trabajo, de sus fundamentos bien establecidos; tomando en cuenta la existencia de un EML estándar de facto (IMS LD), que soporta una variedad de aproximaciones pedagógicas, incluyendo el aprendizaje colaborativo; tomando en cuenta la existencia de una variedad de sistemas y herramientas que dan soporte a toda una comunidad de práctica (la comunidad LD); y

tomando en cuenta que incluso otras aproximaciones LD buscan establecer puentes de interoperabilidad con IMS LD (lo que muestra su importancia y vigencia). Por todo lo anterior, en esta Tesis Doctoral se decidió realizar una investigación en el marco de una aproximación LD basada en flujo de trabajo aplicado al dominio educativo (llamada marco LD o simplemente LD), en lugar de investigar en la dirección de proponer una nueva aproximación para el campo.

En este contexto, en esta Tesis Doctoral se ha estudiado el problema de especificar el flujo de datos dentro del marco LD. Se comprobó que no es posible especificar en LD el flujo de datos a nivel de herramientas, ni en estructuras ni en situaciones de aprendizaje colaborativo. Esto incluye especificar la relación entre datos y herramientas abstractas, entre instancias de datos e instancias de herramientas, así como la secuenciación de invocaciones de herramientas en una misma actividad de aprendizaje. Este análisis de la expresividad de LD para especificar el flujo de datos a nivel de herramientas es una contribución de esta tesis y puede ser encontrada en (Palomino-Ramírez, Bote-Lorenzo, Asensio-Pérez, Dimitriadis & de la Fuente-Valentín, 2008). El alcance de esta contribución puede impactar en la comunidad investigadora en tener un mejor entendimiento del problema.

En esta Tesis Doctoral, se buscó que la solución al problema cumpla con tres importantes objetivos de un proceso basado en flujo de trabajo: automatización del proceso, reutilización de la definición del proceso, e interoperabilidad con sistemas y herramientas. Por un lado, la automatización del flujo de datos a nivel de herramientas de un diseño de aprendizaje es importante, ya que sistemas soportados por ordenador pueden guiar a los participantes en la secuencia de invocaciones de herramientas, así como para gestionar automáticamente el intercambio de datos entre herramientas. Esto permite que los distintos participantes de una actividad de aprendizaje no tengan que preocuparse de localizar y recuperar los distintos artefactos y herramientas que requieren utilizar en una actividad de aprendizaje, ya que el sistema se encargará de entregárselos automáticamente. Por otro lado, una solución que permita la reutilización de un diseño de aprendizaje es importante, ya que diseñadores instruccionales y/o educadores puedan reutilizar la estructura del flujo de aprendizaje y/o del flujo de datos para derivar nuevas situaciones de aprendizaje, de forma que facilite al educador su reutilización en distintas situaciones y contextos. Por último, una solución interoperable con estándares actuales es importante, ya que permite conseguir interoperabilidad con la comunidad de práctica, esto es, permite que diseños de aprendizaje puedan ser editados e intercambios entre educadores (con el apoyo de herramientas de autoría), o ejecutados por sistemas de puesta en marcha que sean conformes con dichos estándares.

Se propuso entonces LeadFlow4LD, el cual es un método para representar computacionalmente procesos de aprendizaje colaborativo a través de guiones colaborativos soportados por ordenador llamados diseños de aprendizaje. Un diseño de aprendizaje especificado conforme a LeadFlow4LD describe tanto la estructura como una situación de aprendizaje colaborativo. LeadFlow4LD empaqueta un diseño de aprendizaje junto con todos

los recursos educativos en una unidad de aprendizaje llamada UoLeadFlow4LD. Una UoLeadFlow4LD se compone de cinco documentos: los primeros tres describen distintos aspectos de la estructura de aprendizaje colaborativo (documentos del flujo de aprendizaje, de flujo de datos y de coordinación entre flujos) y los otros dos describen una situación de aprendizaje particular (documentos de instanciación y de población de grupos).

LeadFlow4LD es la principal contribución de esta tesis y puede ser encontrada en (Palomino-Ramírez, Bote-Lorenzo, Asensio-Pérez, Vignollet & Dimitriadis, 2013). El alcance de esta contribución podría ser la de influir en la comunidad LD en futuras revisiones del estándar IMS LD. Por ejemplo, en extender la expresividad de IMS LD para especificar el flujo de datos a nivel de herramientas (actualmente IMS LD sólo soporta expresividad a nivel de actividades). O bien, extender la expresividad de IMS LD para especificar la instanciación de un diseño de aprendizaje, a través de especificar la asignación de participantes a las actividades de aprendizaje (actualmente esta asignación se lleva a cabo a través de especificaciones propietarias de las herramientas). LeadFlow4LD podría influir también en el campo de Flujo de Trabajo. Por ejemplo, en extender la definición de un proceso de flujo de trabajo para que incluya no sólo la definición de la estructura del proceso (como ocurre actualmente), sino que defina también instancias de dicha estructura, incluyendo la definición de la asignación de recursos concretos a las actividades de flujo de trabajo (actualmente esta asignación de recursos se lleva a cabo a través de especificaciones propietarias de las herramientas).

LeadFlow4LD es un método basado en la composición del flujo de aprendizaje, especificado en el EML estándar de facto de LD (IMS LD), y el flujo de datos a nivel de herramientas, especificado en un lenguaje estándar de flujo de trabajo (XPDL). El primer objetivo de automatización del flujo de datos a nivel de herramientas, se consiguió a través de la propuesta de usar un lenguaje estándar de flujo de trabajo (XPDL). A través de especificar con XPDL, tanto la estructura del flujo de datos como instancias de dicha estructura, se consiguió automatizar la secuencia de invocaciones de herramientas, así como automatizar los datos de entrada y salida de dichas herramientas. El segundo objetivo de reutilización del diseño de aprendizaje se consiguió a través de separar los aspectos que describen una estructura de aprendizaje colaborativo, de los aspectos que describen una situación de aprendizaje particular. Con esta aproximación es posible alcanzar distintos niveles de reutilización para un diseño de aprendizaje. Primero está la reutilización a nivel estructural, la cual se consigue al separar la estructura del flujo de datos de una instancia de dicha estructura, lo que permite crear situaciones de aprendizaje estructuralmente similares, pero que son distintas entre sí. Después está la reutilización a nivel situacional, en la que distintas situaciones de aprendizaje comparten la misma estructura del flujo de aprendizaje y del flujo de datos, pero los participantes se distribuyen de distinta manera en las actividades de aprendizaje y/o en las instancias de herramientas. Y por último, está la reutilización a nivel contextual, ya sea porque intervienen nuevos participantes (manteniéndose la misma estructura y la misma distribución de participantes), o simplemente porque se trata de

distintos contextos (química en lugar de biología), lo cual facilita la reutilización de situaciones de aprendizaje en contextos distintos.

Por último, si bien es cierto que LeadFlow4LD no consigue alcanzar interoperabilidad completa con sistemas y herramientas, sin embargo sí se consiguen distintos niveles de interoperabilidad. En un primer nivel, está la interoperabilidad a nivel del flujo de aprendizaje, la cual se consiguió a través de la propuesta de especificar el flujo de aprendizaje en el lenguaje estándar de LD (IMS LD). Así, la estructura del flujo de aprendizaje de un diseño de aprendizaje puede ser intercambiada entre educadores, editada y ejecutada por herramientas y sistemas de puesta en marcha que sean conformes con el estándar IMS LD. En un segundo nivel, está la interoperabilidad a nivel del flujo de datos. La estructura del flujo de datos, e instancias de dicha estructura, pueden ser intercambiadas entre educadores, editadas y ejecutadas por herramientas y sistemas de puesta en marcha que sean conformes con el estándar XPDL. Finalmente, en un tercer nivel, está la interoperabilidad de intercambio de unidades de aprendizaje que son conformes a LeadFlow4LD (llamadas UoLeadFlow4LD). Diseños de aprendizaje especificados conforme al método propuesto y empaquetados en una UoLeadFlow4LD pueden ser intercambiados entre educadores y editados por herramientas que sean conformes con el estándar IMS CP. Desafortunadamente LeadFlow4LD no consigue interoperabilidad a nivel de instanciación del diseño de aprendizaje. Esto se debe al hecho de que la instanciación de procesos de flujo de trabajo en realidad no se especifica en absoluto, ni en el campo LD ni en el campo de Flujo de Trabajo, por tanto la asignación de participantes tanto a las actividades de aprendizaje, como a las instancias de herramientas de un diseño de aprendizaje, se lleva a cabo a través de especificaciones propietarias.

La evaluación del método propuesto se realizó a través de una metodología basada en ejemplos, que es reconocida como una buena práctica en el campo de la ingeniería de software. La evaluación se realizó con respecto a los objetivos de automatización del flujo de datos, reusabilidad del diseño de aprendizaje, e interoperabilidad con estándares actuales. Distintas situaciones de aprendizaje pertenecientes a diferentes patrones de procesos de aprendizaje colaborativo fueron seleccionadas, y especificadas desde el punto de vista del diseñador instruccional y el educador. Las primeras cuatro situaciones de aprendizaje seleccionadas son simples pero representativas, y están basadas en la técnica de aprendizaje colaborativo de revisión entre pares (*peer-review*). Su propósito fue la de ilustrar la expresividad del método propuesto para automatizar el flujo de datos, la reutilización del diseño de aprendizaje, así como de facilitar el entendimiento de la propuesta. La siguiente situación de aprendizaje seleccionada está basada en la técnica de convergencia del conocimiento, la cual es en realidad una generalización de la técnica de revisión entre pares. Su propósito fue la de ilustrar la expresividad del método propuesto para automatizar el flujo de datos, en particular la automatización de la invocación de una secuencia de herramientas en una misma actividad de aprendizaje. Finalmente, la última situación de aprendizaje seleccionada corresponde al caso MOSAIC, la cual es una situación de aprendizaje compuesta por una jerarquía de tres técnicas de aprendizaje colaborativo bien conocidas:

5. Conclusiones y trabajo futuro

pirámide de 3 niveles, *jigsaw* y revisión entre pares. Su propósito fue la de ilustrar la expresividad del método propuesto para automatizar el flujo de datos, pero también para mostrar que LeadFlow4LD no está limitado a situaciones triviales (*toy examples*), sino a situaciones de aprendizaje complejas compuestas quizás por distintas técnicas de aprendizaje colaborativo, así como de numerosas y complejas interacciones del flujo de datos.

Con el objetivo de poner en marcha estas situaciones de aprendizaje, se propuso e implementó un sistema prototipo que se usó como prueba de concepto para ilustrar que la puesta en marcha de una UoLeadFlow4LD es posible. La puesta en marcha de las distintas situaciones de aprendizaje se llevó a cabo en un entorno controlado con participantes ficticios (simulados), mientras que su puesta en marcha en escenarios auténticos con participantes reales, se deja como un trabajo futuro. Los resultados observados mostraron, por un lado, que artefactos y herramientas fueron entregados automáticamente a los participantes (ficticios) cuando realizaban las actividades de aprendizaje, lo cual ilustra la expresividad de LeadFlow4LD para automatizar el flujo de datos. Por otro lado, a través de la puesta en marcha de distintas situaciones de aprendizaje, se mostró también que con sólo modificar parte de los documentos de la UoLeadFlow4LD, se consiguió la reutilización del diseño de aprendizaje, lo cual ilustró los distintos niveles de reutilización de una UoLeadFlow4LD. Finalmente, la capacidad del flujo de aprendizaje para ser editado y ejecutado por herramientas y sistemas de puesta en marcha conformes al estándar IMS LD; la capacidad del flujo de datos para ser editado y ejecutado por herramientas y sistemas de puesta en marcha conformes al estándar XPDL; y la capacidad de una UoLeadFlow4LD para ser interpretada por herramientas de autoría conformes al estándar IMS CP, ilustraron los distintos niveles de interoperabilidad del método propuesto.

Sin embargo, distintas limitaciones también fueron identificadas durante la puesta en marcha de las distintas situaciones de aprendizaje seleccionadas. La primera limitación, es que los participantes pueden perder conciencia de las tareas asociadas a una secuencia de invocaciones de herramientas en una actividad de aprendizaje, ya que las tareas no son especificadas en actividades de aprendizaje de grano grueso (actividades que se componen de varias tareas). Una solución a este problema puede ser la especificación de las tareas directamente en el flujo de datos, por ejemplo, en la descripción de las actividades de flujo de trabajo que invocan las herramientas. La segunda limitación tiene que ver con la gran cantidad de instancias de datos y herramientas, relaciones entre instancias de datos y herramientas, así como la asignación de participantes a las actividades de aprendizaje y a las instancias de herramientas, que se deben especificar. Esto es de hecho un problema para el diseñador instruccional o el educador en escenarios de aprendizaje colaborativo complejos. Este problema se puede resolver con el uso de herramientas de autoría que ayuden a la automatización de la instanciación del diseño de aprendizaje. Una tercera limitación es la necesidad de especificar los documentos de coordinación de flujos y de población de grupos, ya que al tratarse de especificaciones propietarias, no existen por ahora herramientas de autoría para este propósito. Sin embargo, se tiene la ventaja de que estos documentos no tienen que ser interpretados por motores de LD o de flujo de trabajo. Además, el hecho de

que LeadFlow4LD está principalmente basado en los estándares IMS LD y XPDL, permite, por un lado, una reducción significativa en el esfuerzo de desarrollo de herramientas de autoría (ya que las herramientas pueden ser reutilizadas), y por el otro, una reducción de la curva de aprendizaje mientras se usan herramientas basadas en dichos estándares.

Entre los efectos colaterales identificados durante la puesta en marcha de las distintas situaciones de aprendizaje, está la inflexibilidad para que los participantes puedan elegir sus propias herramientas, ya que con el método propuesto el sistema entrega automáticamente a los participantes las herramientas que deben de usar en las actividades de aprendizaje, en lugar de que ellos puedan usar sus propias herramientas. Una solución a este problema puede ser permitir a los participantes la asignación de sus propias herramientas en tiempo de instanciación. Otro efecto colateral identificado es la falta de conciencia de los participantes sobre el origen o propiedad de los artefactos que son inicializados automáticamente por las herramientas. Una solución a este problema podría ser la imposición de requisitos a herramientas de terceros (Alario-Hoyos, 2012), de tal forma que la herramienta incluya la información de autoría en los artefactos. En cualquier caso, ninguno de estos efectos colaterales identificados afecta a alguno de los tres objetivos de evaluación considerados.

Es notable que LeadFlow4LD pueda contribuir al campo LD no solo como una forma de especificar diseños de aprendizaje dentro del marco LD, sino que pueda ser empleado dentro de la propuesta realizada por otros investigadores con el fin de habilitar el despliegue de diseños de aprendizaje basados en IMS LD dentro de VLE (Alario-Hoyos, Bote-Lorenzo, Gómez-Sánchez, Asensio-Pérez, Vega-Gorgojo & Ruiz-Calleja, 2013). De esta forma, LeadFlow4LD pudiera ser usado para mejorar el soporte proporcionado para la realización de escenarios colaborativos tanto en productos abiertos como Moodle o LAMS, como también en productos comerciales tales como Blackboard. Estas contribuciones serían consideradas para una revisión futura de la especificación IMS LD, pero también por otros EML.

Otra forma de medir la contribución global del presente trabajo de tesis, es a través de las citas que otros investigadores han realizado a los distintos artículos publicados y que están relacionados directamente con esta tesis: Palomino-Ramírez et al. (2007), Palomino-Ramírez et al. (2008), Palomino-Ramírez et al. (2008B) y Palomino-Ramírez et al. (2013). Ejemplos de 9 citas conocidas hasta el momento, realizadas por investigadores ajenos a nuestro grupo de investigación, son: Sobreira & Tchounikine (2015), Sobreira (2014), Magnisalis & Demetriadis (2014), Magnisalis & Demetriadis (2012), Mulholland et al. (2012), Sharples et al. (2011), Caeiro-Rodríguez et al. (2010), Vidal et al. (2009) y Vidal et al. (2008). Adicionalmente, un total de 11 citas han sido realizadas hasta el momento por investigadores cercanos a nuestro grupo de investigación.

Por último, si bien algunos autores centran actualmente sus esfuerzos de investigación en el nuevo campo LD, investigando nuevas aproximaciones LD (más allá de IMS LD) e intentando establecer los fundamentos del nuevo campo. Sin embargo, un último punto a favor en esta tesis es la de contribuir a ampliar la expresividad de LD sin necesidad de crear una nueva aproximación del campo, ya que lo que se ha conseguido es extender LD dentro del mismo campo de Flujo de Trabajo. Por tanto, por un lado, el hecho de que nuevas aproximaciones LD busquen establecer puentes de interoperabilidad con IMS LD, nos muestra la importancia y vigencia del actual marco LD. Por otro lado, la enorme complejidad de poner de acuerdo a toda una comunidad de práctica para definir un único marco para el nuevo campo LD, nos lleva a pensar que LD, y con ello IMS LD, pueden mantenerse vigente por mucho tiempo más.

5.2 Trabajo futuro

A partir del trabajo presentado en esta tesis es posible identificar varias tareas que merece la pena abordar en el futuro, tanto mejoras a la propuesta como nuevas líneas de investigación que surgen de la propuesta.

Mejoras a la Propuesta

- **Diseño de aprendizaje.** Mejoras en la especificación del diseño de aprendizaje con el método propuesto pueden realizarse como, por ejemplo, la flexibilidad para que los participantes elijan sus propias herramientas, en lugar de que estas sean elegidas por el educador y sean las mismas para todos los participantes; conocer a los propietarios de artefactos creados por las herramientas; y conocer las tareas asociadas a una secuencia de herramientas en actividades de grano grueso. Por un lado, actualmente con el método propuesto los participantes no pueden elegir usar sus propias herramientas en una actividad de aprendizaje, ya que el sistema se las entrega automáticamente. LeadFlow4LD podría mejorar la flexibilidad en la selección de herramientas a los participantes permitiendo su asignación en tiempo de instanciación, o bien, a través de *learning buckets*, usados para la flexibilidad en el manejo de artefactos (Muñoz-Cristóbal, 2005). Por otro lado, actualmente con el método propuesto los participantes desconocen la autoría de los artefactos creados por las herramientas. LeadFlow4LD podría mejorar la conciencia de los participantes sobre los propietarios de artefactos a través de imponer requisitos a herramientas de terceros, como se discute en (Alario-Hoyos, 2012). Por último, actualmente con el método propuesto los participantes pueden perder conciencia sobre las tareas a realizar por las herramientas (en una secuencia de herramientas) en actividades de grano grueso (cuando una actividad se componen de múltiples tareas). En el prototipo actual, esto se resolvió parcialmente especificando la tarea a realizar por la herramienta en la descripción de las actividades de flujo de trabajo que representan invocaciones de herramientas, tarea que sin embargo debe ser realizada por el educador en tiempo de instanciación. Una solución definitiva a este problema puede

estar en la dirección de descomponer la especificación de una actividad de grano grueso en sub-actividades, cada una de las cuales describiendo la tarea a realizar por una herramienta en la secuencia de invocaciones de herramientas.

- **Infraestructura de preparación de puesta en marcha.** Se requiere facilitar al diseñador instruccional y al educador la especificación de tres de los cinco documentos que componen una UoLeadFlow4LD. Se requiere invertir en el desarrollo de herramientas de autoría para especificar los documentos del flujo de datos, de coordinación de flujos, y de instanciación y población de grupos. Por ejemplo, en la misma dirección que iCollage (Hernández-Gonzalo, Villasclaras-Fernández, Hernández-Leo, Asencio-Pérez & Diminitriadis, 2008), la cual es una herramienta de autoría usada para especificar la instanciación de grupos, en el contexto del flujo de aprendizaje, pero no para el flujo de datos. Adicionalmente, se requiere de una forma automática de gestionar instancias de herramientas (creación, modificación, y eliminación de instancias de herramientas). En el sistema prototipo actual, esta tarea es realizada manualmente por el administrador del sistema, e implica un incremento significativo en la carga de especificación, cuando se trate de un número considerablemente grande de instancias de herramientas. Actualmente se está planeando enriquecer esta infraestructura de puesta en marcha de LeadFlow4LD con el sistema de integración Glue! (Alario-Hoyos, Bote-Lorenzo, Gómez-Sánchez, Asencio-Pérez, Vega-Gorgojo & Ruiz-Callejas, 2013), de forma que sea factible la gestión automática de instancias de herramientas de terceros.
- **Sistema de puesta en marcha.** Es necesario mejorar la implementación del sistema prototipo de puesta en marcha de una UoLeadFlow4LD de acuerdo a la arquitectura propuesta en la Figura 3.25 (ver sección 3.6.1). Para ello, se requiere implementar el componente analizador LeadFlow4LD, el cual es el responsable de crear automáticamente los documentos del flujo de aprendizaje y flujo de trabajo sincronizados, necesarios para la coordinación entre flujos. En el sistema prototipo actual esta tarea es realizada manualmente por el desarrollador, por lo que para el uso en producción del método propuesto, es imperativo la implementación de dicho componente de acuerdo al procedimiento descrito en la sección 3.6.4. Adicionalmente, se requiere mejorar también la capa de presentación del sistema, de forma que los participantes interactúen con el sistema a través de una única interfaz, en lugar de usar distintas interfaces (a motores de LD y flujo de trabajo), como ocurre actualmente en el sistema prototipo.
- **Evaluación de LeadFlow4LD.** La evaluación del método propuesto debe ser ampliada para tomar en cuenta la puesta en marcha de escenarios de aprendizaje colaborativo en el que intervengan participantes reales. De esta forma, se podría demostrar que los participantes no sufren una carga cognitiva adicional ni situaciones propensas de error cuando el sistema se encarga de gestionar el flujo de datos. Esto permitiría identificar otras ventajas y desventajas del método propuesto. Además, sería interesante también poner en marcha otras estructuras y situaciones de aprendizaje colaborativo que involucren distinto número de participantes.

- **Incorporar LeadFlow4LD en VLE y en procesos de *e-learning*.** Se planea trabajar en el uso LeadFlow4LD para soportar el aprendizaje colaborativo (especificación del flujo de aprendizaje y el flujo de datos) en VLE, por ejemplo la plataforma Moodle. Además, LeadFlow4LD podría ser útil en la especificación del flujo de datos en procesos de evaluación educativa (*assessment*), así como también en la especificación del flujo de resultados de aprendizaje en procesos de *e-Portfolio*.

Nuevas Líneas de Investigación

- **Modificación del diseño de aprendizaje en tiempo de ejecución.** La puesta en marcha de una UoLeadFlow4LD con el método propuesto no permite modificar la UoLeadFlow4LD una vez que esta se ha puesto en marcha. Debería permitirse una cierta flexibilidad de modificar una UoLeadFlow4LD en tiempo de ejecución. Sin embargo, este no es un problema exclusivo de LeadFlow4LD, ni de LD, sino del paradigma de flujo de trabajo en general, por lo que su solución debe estar en la misma dirección que la flexibilidad de procesos de flujo de trabajo (Heinl, Horn, Jablonski, Neeb, Stein & Teschke, 1999).
- **Extensión de IMS LD.** Como una nueva línea de investigación podría ser la propuesta de una extensión a IMS LD para especificar el flujo de aprendizaje y el flujo de datos en estructuras y situaciones de aprendizaje colaborativo. Esto es, se buscaría mapear las distintas ideas de LeadFlow4LD en una propuesta de extensión de IMS LD. Esto permitiría que en una futura revisión del lenguaje las ideas del método propuesto pueden ser más fácilmente tomadas en cuenta.
- **Mapeo de ideas de LeadFlow4LD al campo de Flujo de Trabajo.** Una línea de investigación futura podría ser estudiar cómo mapear las ideas propuestas en LeadFlow4LD en el campo de Flujo de Trabajo. Por un lado, identificar en el campo de Flujo de Trabajo, si realmente existe la necesidad de especificar juntos (en un mismo proceso) la estructura del flujo de datos de una instancia de dicha estructura, como es imperativo en escenarios de aprendizaje colaborativo en el campo LD. Por otro lado, para especificar situaciones de aprendizaje colaborativo, está la necesidad en LeadFlow4LD de especificar la instanciación de la asignación de participantes a las actividades de aprendizaje y a las instancias de herramientas. Sería interesante investigar en el campo de Flujo de Trabajo en qué casos sería necesario especificar la instanciación del proceso, o responder por qué no es necesaria especificar dicha instanciación.
- **Especificación de la instanciación de un diseño de aprendizaje.** Para conseguir interoperabilidad a nivel de instanciación, con el método propuesto, se requiere la estandarización de la especificación de la instanciación de un diseño de aprendizaje. Para ello, se requiere convencer a la comunidad LD de la importancia que tiene especificar la instanciación del proceso en la puesta en marcha de un diseño de aprendizaje. Esto se puede conseguir a través de divulgar la especificación, con el método propuesto, de nuevos y más diversos escenarios de aprendizaje colaborativo, donde se muestre la importancia que tiene la especificación de la asignación de

5. Conclusiones y trabajo futuro

participantes tanto a las actividades de aprendizaje, como a las instancias de herramientas. En este sentido, los esquemas XML propuestos en este documento de tesis pueden ser un punto de partida.

Apéndice A Caso de estudio MOSAIC

En este apéndice se presentan los resultados obtenidos del caso de estudio MOSAIC, incluyendo también los cuestionarios aplicados a los estudiantes, al administrador y al diseñador instruccional.

A.1 Resultados

En este apéndice se presenta un resumen de los resultados del caso de estudio MOSAIC, los cuales fueron usados en esta tesis para soportar el análisis del problema. Además, se presentan también parte de los cuestionarios aplicados a los distintos actores del escenario de aprendizaje: estudiantes, administrador y diseñador instruccional. Por cuestión de espacio, la totalidad de cuestionarios y sus respuestas, son incluidos en el disco anexo y se pueden consultar en las rutas: `.\problema\Mosaic\cuestionarios` y `.\problema\Mosaic\resultados`, respectivamente.

Las Tabla A.1 resume las distintas fuentes de datos utilizadas en el análisis del caso de estudio MOSAIC.

Fuente de Datos	Identificador de la Fuente	Momento de la Colección de Datos
E-mail	[e-mail]	Durante la realización de la experiencia
Análisis de UoL	[uol]	Después de realizada la experiencia
Cuestionario final general	[fgq]	Después de realizada la experiencia
Cuestionario final específico	[fsq]	Después de realizada la experiencia

Tabla A.1: Fuentes de datos utilizadas en el análisis del caso de estudio MOSAIC.

De la Tabla A.2 a la Tabla A.5 se resumen los resultados obtenidos de cada una de estas fuentes. La Tabla A.2 resume los resultados obtenidos al analizar la fuente de datos [e-mail]; la Tabla A.3 resume los resultados obtenidos al analizar la UoL del caso de estudio MOSAIC; la Tabla A.4 y la Tabla A.5 resumen los resultados obtenidos al analizar las respuestas de los estudiantes a los cuestionarios finales específicos. En la Figura A.1 se muestra parte del cuestionario final general aplicado a los estudiantes participantes en el caso de estudio; en la Figura A.2 se muestra parte del cuestionario final específico aplicado a estos mismos estudiantes; en la Figura A.3 se muestra parte del cuestionario final específico aplicado al diseñador instruccional; y por último, en la Figura A.4 se muestra parte del cuestionario final específico aplicado al administrador del curso.

Apéndice A

Source Identifier	Issue Description	Relation with the IMS-LD Data Flow Problem
[eml-uyr-d1]	Al iniciar la primera actividad, el usuario rectángulo amarillo hace la siguiente pregunta al tutor del curso sobre la incompatibilidad en la asignación de papers: "Me surge ahora otra duda: Yo soy "rectángulo amarillo", por lo que acorde con el documento resumen de la experiencia, pensaba que tenía que leer el artículo de "Grid y Servicios". Sin embargo, en la actividad se me indica que tengo que leer el artículo " Arquitectura Grid"	El problema es importante, ya que si el usuario no advierte esta incompatibilidad, y lee por error el paper que no le corresponde, entonces los objetivos de aprendizaje no se cumplirán.
[eml-t1-d1]	Esta es la respuesta del tutor (que también es el autor) al usuario rectángulo amarillo sobre la incompatibilidad en la asignación de papers: "Primero pedir disculpas por el error acerca de la asignación de los Papers. Se me escapó el detalle... He actualizado el recurso, de modo que la asignación es ahora la misma en el documento Word y en la actividad."	El tutor/autor del curso hace referencia a un error suyo en la especificación del flujo de artefactos. El tutor también se refiere a la necesidad de mantener compatibilidad entre el flujo de artefactos especificado y el documento que resume el flujo de artefactos. Sin este documento, el usuario no habría advertido del error del autor en la especificación.
[eml-a1-d1]	Esta es la respuesta del autor del curso, cuando se le pregunta sobre la causa del error en la asignación de papers: "Si te refieres a la confusión en la asignación de papers, se trató de un error de dedo. Se me escapó el corta-pegar una línea más arriba de lo que debía. Eso es todo."	Es un indicio de que la aproximación usada en la especificación del flujo de datos es propensa de errores para el autor.
[eml-a1-d2]	A continuación, la respuesta del autor al preguntarle sobre la especificación en la asignación de artefactos: "He creado un único recurso (puedes acceder a él en el file storage: acto1/estudiante/individual_study.xml) y he puesto varios <div class="..."> para mostrar u ocultar dependiendo de las propiedades. En vez de usar grupos, he utilizado propiedades para los términos "figura" y "color". El uso de grupos está definido demasiado difusamente en IMS-LD. Me equivoqué, como bien dices, por exceso de carga cognitiva."	Es un indicio de que la experiencia del autor afecta directamente a la aproximación usada para especificar el flujo de artefactos. El autor confirma además que el exceso de carga cognitiva ocasionó el error al especificar el flujo de artefactos.

Tabla A.2: Resultados obtenidos de la fuente: [e-mail].

Source Identifier	Issue Description	Relation with the IMS-LD Data Flow Problem
[uol-d1]	Al disectar la UoL se identificó que existe especificación a nivel de instancia en el manifiesto. Valores de propiedades que representan distintos grupos, tales como "yellow", "green", "red" y blue; así como valores de propiedades que representan expertos como: "triangle", "rectangle", y "cruz", no es congruente con el resto de la especificación que tiene lugar a nivel declarativo.	Aproximación para especificar el flujo de datos colaborativo, requiere especificación a nivel de instancia que afecta la reusabilidad de la UoL.
[uol-d1]	Al disectar la UoL se identificó que en la especificación del flujo de artefactos, el autor requiere mantener compatibilidad entre dos partes distintas de la UoL: en expresiones condicionales dentro del manifiesto, y en la propia descripción de actividad .	Aproximación para especificar el flujo de datos colaborativo que es propensa a errores por parte del profesor.

Tabla A.3: Resultados obtenidos de la fuente: [uol].

Data Source Identifier	Issue Description	Relation with the IMS-LD Data Flow Problem
[fgq-u-de7-1]	Opinión de los usuarios sobre la utilidad del repositorio de ficheros. Average: 4,57. Std. Dev.:1,99 Rango: 0-Nada útil ... 6-Muy útil	Indicativo de la relevancia del flujo de artefactos en aprendizaje colaborativo.
[fsq-u-de-7-4]	Opinión de los usuarios sobre la frecuencia de uso del repositorio de ficheros. Average: 4,43 Std. Dev: 1,99	Indicativo de la relevancia del flujo de artefactos en aprendizaje colaborativo.
[fgq-ugt-de7-2]	Comentario del usuario triángulo-verde sobre la utilidad del repositorio de ficheros: "Es indispensable intercambiar ficheros en una experiencia como esta"	Indicativo de la relevancia del flujo de artefactos en experiencias de aprendizaje colaborativo.
[fgq-ugr-de7-2]	Comentarios de rectángulo-verde sobre la utilidad del repositorio de ficheros: "Necesario para cargar y descargar los artefactos producidos a través del curso"	Indicativo de la relevancia del flujo de artefactos en experiencias de aprendizaje colaborativo.
[fgq-urt-de7-2]	Comentarios de triángulo-rojo sobre la utilidad del repositorio de ficheros: "Lo ocupé mucho para buscar los archivos del resto"	Indicativo de la relevancia del flujo de artefactos en experiencias de aprendizaje colaborativo.
[fsq-ubc-de7-5]	Comentario del usuario "cruz-azul" sobre el uso de la herramienta para gestionar el repositorio de ficheros "Problemas en manejo de rutas y tipos de ficheros"	Indicativo de problemas en la aproximación usada para gestionar ficheros

Tabla A.4: Algunos resultados del análisis de cuestionarios aplicados a los estudiantes obtenidos de la fuente: [fgq]. Descripción del identificador de la fuente de datos: [fgq-u-de7-1]: cuestionario final general, datos de todos los participantes, sección 7, subsección 1; [fgq-u-de7-4]: cuestionario final general, datos de todos los participantes, sección 7, subsección 1; [fgq-ugt-de7-2]: cuestionario final general, datos de los participantes del grupo triángulo verde; sección 7, subsección 2; [fgq-ugr-de7-2]: cuestionario final general, datos de los participantes del grupo rectángulo verde; sección 7, subsección 2; [fgq-urt-de7-2]: cuestionario final general, datos de los participantes del grupo triángulo rojo; sección 7, subsección 2; [fgq-urt-de7-5]: cuestionario final general, datos de los participantes del grupo cruz azul; sección 7, subsección 5.

Selected Answers		
Data Source Identifier	Issue Description	Relation with the IMS-LD Data Flow Problem
	Generales	
[fdq-u-d5]	Opinión de los usuarios sobre si la gestión de artefactos es propensa a errores. Average: 3,88. Std. Dev.:1,26 Rango: 0-poco ... 6-mucho	Indicativo de que el usuario es propenso a cometer errores cuando gestiona el flujo de datos
[fdq-u-d6]	Opinión de los usuarios sobre la carga cognitiva adicional requerida para gestionar los artefactos. Average: 3,66 Std. Dev: 1,73 Rango: 0-nada ... 6-mucha	Indicativo de la existencia de una carga cognitiva en el usuario para gestionar el flujo de artefactos.
[fdq-u-d7]	Opinión de los usuarios sobre la necesidad de automatizar la gestión de artefactos en actividades colaborativas. Average: 4,66 Std. Dev: 1,22 Rango: 0-poco ... 6-mucho	Indicativo de la relevancia del problema de automatizar la gestión de artefactos en actividades colaborativas.
[fdq-u-d8]	Opinión de los usuarios sobre la aproximación usada para gestionar el flujo de artefactos. Average: 3,77 Std. Dev: 1,20 Rango: 0-pobre ... 6-buena	Indicativo de que la aproximación usada funciona.

Tabla A.5: Algunos resultados del análisis de cuestionarios finales específicos aplicados a los estudiantes obtenidos de la fuente [fsq]. Descripción del identificador de la fuente: [fdq-u-de5]: cuestionario final específico, datos de todos los usuarios, sección 5; [fdq-u-de6]: cuestionario final específico, datos de todos los participantes, sección 6; [fdq-u-de7]: cuestionario final específico, datos de todos los participantes, sección 7; [fdq-u-de8]: cuestionario final específico, datos de todos los participantes, sección 8.

Apéndice A

A. Identificación

A.1. Introduce tu nombre y apellidos

B. Experiencia previa y opinión

B.1. ¿Has encontrado diferencias en la forma de trabajar en grupo en esta actividad en relación a otras asignaturas/cursos en los que hayas participado?

B.2. ¿Qué beneficios crees que tiene el aprendizaje colaborativo (la realización de actividades como ésta) en la formación académica de grado o de doctorado?

C. Sobre la experiencia...

C.1. Valora cuánto has colaborado con tus compañeros durante la realización de las tareas propuestas a lo largo del curso sobre Grid Computing

C.2. Valora la dificultad de la realización de las tareas propuestas a lo largo del curso sobre Grid Computing

C.3. ¿Qué aspecto del curso sobre Grid Computing ha sido el más complicado?

C.4. ¿Cómo de adecuada a los objetivos del curso fue la estructuración de las tareas propuestas?

C.5. ¿Alguna de las actividades del curso ha sido especialmente útil?

C.6. ¿Alguna de las actividades del curso ha sido especialmente poco útil?

C.7. A tu juicio, ¿la duración del curso ha sido adecuada?

C.8. ¿La colaboración con tus compañeros ha sido positiva?

D. Sobre el sistema de "guiado"

D.1. ¿Cómo de útil consideras el hecho de que el sistema te indicara paso a paso qué debías hacer y qué herramientas debías utilizar?

D.2. ¿Cómo valoras la forma en que las herramientas se proporcionaban en las diferentes actividades?

D.3. Aquí puedes explicar la respuesta a la pregunta anterior

Figura A.1: Parte del cuestionario final general aplicado a los estudiantes participantes en el caso de estudio MOSAIC.

3. ¿Tuvistes alguna vez problemas para localizar los ficheros de alguno de tus compañeros durante una actividad peer-review?
Respuesta [Sí, No]:
Comentarios:

4. Contesta a esta pregunta si has contestado "Sí" a la pregunta anterior. Si en una actividad peer-review, el fichero de tu compañero no se encontraba en la carpeta correspondiente, ¿eras consciente de que posiblemente tu compañero lo hubiera subido a otra carpeta o lo hubiera salvado con otro nombre?
Respuesta [Sí, No]:
Comentarios:

5. ¿Crees que esta aproximación para almacenar y acceder a los ficheros es propensa de errores por parte de los usuarios?, ¿en qué medida?
Respuesta 1 [Sí, No]:
Respuesta 2 [0-poco ... 6-mucho]:
Comentarios:

6. ¿Crees que te producía una carga cognitiva adicional el tener que seguir las instrucciones de la actividad para saber donde almacenar tu fichero y con qué nombre, o donde acceder a los ficheros de tus compañeros y con qué nombres?, ¿en qué medida?,
Respuesta 1 [Sí, No]:
Respuesta 2 [0-nada ... 6-mucha]:
Comentarios:

Figura A.2: Parte del cuestionario final específico aplicado a los estudiantes participantes en el caso de estudio MOSAIC.

Como autor del diseño de la UoL del caso Mosaic:

1. ¿Cuál era tu experiencia previa en el diseño de UoL's con IMS-LD al momento de realizar este diseño?
Respuesta [0-nada ...6-mucha]:
2. ¿Qué problema(s) tuviste, si hubo alguno, al momento de especificar el flujo de artefactos?
Respuesta Abierta:
3. ¿Crees que la aproximación que usaste para especificar el flujo de artefactos es propensa de errores para el autor del diseño? Explica.
Respuesta Abierta:

Figura A.3: Parte del cuestionario final específico aplicado al diseñador instruccional del caso de estudio MOSAIC.

Como tutor y-o administrador del caso Mosaic:

1. ¿En qué grado los usuarios cumplían las instrucciones de la actividad para almacenar sus ficheros? Explica
Respuesta [0-no cumplían ...6-cumplían]:
Respuesta Abierta:
2. ¿Podías distinguir entre un usuario que no subía su fichero del que lo subía a otra carpeta o con otro nombre? Explica.
Respuesta Abierta:
3. ¿Crees que esta aproximación usada para gestionar ficheros es propensa de errores por parte del usuario? Explica.
Respuesta Abierta:

Figura A.4: Parte del cuestionario final específico aplicado al tutor/administrador del curso.

Apéndice B Esquemas XML Propuestos

En este apéndice se presenta en la sección B.1 los esquemas XML propuestos para especificar los documentos de coordinación de flujos y de población de grupos. En la sección B.2 se describe la instancia de la estructura del flujo de datos del caso MOSAIC. En la sección B.3 se resumen los documentos que especifican las distintas estructuras y situaciones de aprendizaje colaborativo que han sido utilizados en el proceso de evaluación del método propuesto.

B.1 Esquema del documento de coordinación entre flujos

La Figura B.1 muestra el esquema XML propuesto usado para representar la coordinación entre el flujo de aprendizaje y el flujo de datos: la coordinación entre flujos.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema elementFormDefault="qualified"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<xsd:element name="flows-coordination-manifest"
type="FlowsCoordinationManifestType">
</xsd:element>
<xsd:complexType name="FlowsCoordinationManifestType">
<xsd:sequence>
<xsd:element name="uoLeadFlow4LD-info"
type="UoLeadFlow4LDInfoType" maxOccurs="1" minOccurs="1">
</xsd:element>
<xsd:element name="activities" type="ActivitiesType" maxOccurs="1"
minOccurs="1"></xsd:element>
<xsd:element name="transitions" type="TransitionsType" maxOccurs="1"
minOccurs="1"></xsd:element>
<xsd:element name="tools" type="ToolsType" maxOccurs="1"
minOccurs="1"></xsd:element>
</xsd:sequence>
<xsd:attribute name="id" type="xsd:string"></xsd:attribute>
</xsd:complexType>
<xsd:complexType name="UoLeadFlow4LDInfoType">
<xsd:attribute name="id" type="xsd:string"></xsd:attribute>
<xsd:attribute name="learning-flow-uri" type="xsd:string"></xsd:attribute>
<xsd:attribute name="data-flow-uri" type="xsd:string"></xsd:attribute>
</xsd:complexType>
<xsd:complexType name="TransitionsType">
<xsd:sequence>
<xsd:element name="master-transition"
type="MasterTransitionType" maxOccurs="unbounded"
minOccurs="1">
</xsd:element>
<xsd:element name="slave-transition"
type="SlaveTransitionType" maxOccurs="unbounded" minOccurs="1">
</xsd:element>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ToolsType">
<xsd:sequence>
<xsd:element name="tool" type="ToolType" maxOccurs="unbounded"
minOccurs="1"></xsd:element>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ActivitiesType">
<xsd:sequence>
<xsd:element name="learning-activity"
type="LearningActivityType" maxOccurs="unbounded">
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:schema>
```

```

minOccurs="1">
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="LearningActivityType">
  <xsd:attribute name="id" type="xsd:string"></xsd:attribute>
</xsd:complexType>
<xsd:complexType name="MasterTransitionType">
  <xsd:sequence>
    <xsd:element name="from" type="ActivityRefType"></xsd:element>
    <xsd:element name="to" type="ToolRefType"></xsd:element>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SlaveTransitionType">
  <xsd:sequence>
    <xsd:element name="from" type="ToolRefType"></xsd:element>
    <xsd:element name="to" type="ActivityRefType"></xsd:element>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ActivityRefType">
  <xsd:attribute name="activity-ref" type="xsd:string"></xsd:attribute>
</xsd:complexType>
<xsd:complexType name="ToolRefType">
  <xsd:attribute name="tool-ref" type="xsd:string"></xsd:attribute>
</xsd:complexType>
<xsd:complexType name="ToolType">
  <xsd:attribute name="id" type="xsd:string"></xsd:attribute>
</xsd:complexType>
</xsd:schema>

```

Figura B.1: Esquema XML propuesto para especificar la coordinación de flujos.

B.2 Esquema del documento de población de grupos

La Figura B.2 muestra el esquema XML propuesto usado para representar la asignación de usuarios a las actividades de aprendizaje y a las instancias de herramientas.

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema elementFormDefault="qualified"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<xsd:attribute name="NewAttribute" type="xsd:string"></xsd:attribute>
<xsd:element name="iLeadFlow4LD-manifest"
type="iLeadFlow4LDManifestType">
</xsd:element>
<xsd:complexType name="iLeadFlow4LDManifestType">
  <xsd:sequence>
    <xsd:element name="uoLeadFlow4LD-info"
type="UoLeadFlow4LDInfoType" maxOccurs="1" minOccurs="1">
</xsd:element>
    <xsd:element name="role-root" type="RoleRootType" maxOccurs="1"
minOccurs="1"></xsd:element>
    <xsd:element name="tool-instances"
type="ToolInstancesType" maxOccurs="1" minOccurs="1">
</xsd:element>
    <xsd:element name="data-instances"
type="DataInstancesType" maxOccurs="1" minOccurs="1">
</xsd:element>
    <xsd:element name="group-population"
type="GroupPopulationType" maxOccurs="1" minOccurs="1">
</xsd:element>
  </xsd:sequence>
  <xsd:attribute name="id" type="xsd:string"></xsd:attribute>
</xsd:complexType>
<xsd:complexType name="RoleRootType">

```

```

    <xsd:sequence>
      <xsd:element name="role" type="RoleType" maxOccurs="unbounded"
minOccurs="1"></xsd:element>
    </xsd:sequence>
  </xsd:complexType>
<xsd:complexType name="ToolInstancesType">
  <xsd:sequence>
    <xsd:element name="tool-instance" type="ToolInstanceType"
maxOccurs="unbounded" minOccurs="1"></xsd:element>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GroupPopulationType">
  <xsd:sequence>
    <xsd:element name="user" type="UserType" maxOccurs="unbounded"
minOccurs="1"></xsd:element>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="UoLeadFlow4LDInfoType">
  <xsd:attribute name="id" type="xsd:string"></xsd:attribute>
  <xsd:attribute name="learning-flow-uri" type="xsd:string"></xsd:attribute>
  <xsd:attribute name="data-flow-instance-uri"
type="xsd:string"></xsd:attribute>
</xsd:complexType>
<xsd:complexType name="RoleType">
  <xsd:attribute name="id" type="xsd:string"></xsd:attribute>
  <xsd:attribute name="occurrence" type="xsd:string"></xsd:attribute>
</xsd:complexType>
<xsd:complexType name="ToolInstanceType">
  <xsd:attribute name="id" type="xsd:string"></xsd:attribute>
</xsd:complexType>
<xsd:complexType name="UserType">
  <xsd:sequence>
    <xsd:element name="role-occurrence-ref"
type="RoleOccurrenceRefType" maxOccurs="unbounded"
minOccurs="1">
      </xsd:element>
    <xsd:element name="role-active-occurrence-ref"
type="RoleActiveOccurrenceRefType" maxOccurs="1"
minOccurs="1">
      </xsd:element>
    <xsd:element name="tool-instance-id-ref"
type="ToolInstanceOccurrenceRefType" maxOccurs="unbounded"
minOccurs="1">
      </xsd:element>
  </xsd:sequence>
  <xsd:attribute name="id" type="xsd:string"></xsd:attribute>
  <xsd:attribute name="name" type="xsd:string"></xsd:attribute>
  <xsd:attribute name="email" type="xsd:string"></xsd:attribute>
</xsd:complexType>
<xsd:complexType name="RoleOccurrenceRefType">
  <xsd:attribute name="ref" type="xsd:string"></xsd:attribute>
</xsd:complexType>
<xsd:complexType name="RoleActiveOccurrenceRefType">
  <xsd:attribute name="ref" type="xsd:string"></xsd:attribute>
</xsd:complexType>
<xsd:complexType name="ToolInstanceOccurrenceRefType">
  <xsd:attribute name="ref" type="xsd:string"></xsd:attribute>
</xsd:complexType>
<xsd:complexType name="DataInstancesType">
  <xsd:sequence>
    <xsd:element name="data-instance" type="DataInstanceType"
maxOccurs="unbounded" minOccurs="1"></xsd:element>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DataInstanceType">
  <xsd:attribute name="id" type="xsd:string"></xsd:attribute>
  <xsd:attribute name="url" type="xsd:string"></xsd:attribute>
</xsd:complexType>

```

```
</xsd:schema>
```

Figura B.2: Esquema XML del documento de población de grupos.

Apéndice C Especificación de situaciones de aprendizaje de los ejemplos

En este apéndice, en la sección C.1 se resumen los documentos que especifican las distintas estructuras y situaciones de aprendizaje colaborativo que han sido utilizados en el proceso de evaluación del método propuesto. En la sección C.2 se describe la instancia de la estructura del flujo de datos del caso MOSAIC.

C.1 Instancia de la Estructura del Flujo de Datos del Caso MOSAIC

La relación entre instancias de datos y herramientas de la situación de aprendizaje derivada del caso MOSAIC se resume de la Tabla C.2 a la Tabla C.4. La asignación de participantes a las actividades de aprendizaje se muestra en la Tabla C.5, mientras que la asignación de participantes a las instancias de herramientas se muestra de la Tabla C.6 a la Tabla C.17.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H1	12	H1-1	D1-1	-	Visor del artículo 1 por el participante 1
		H1-2	D1-2	-	Visor del artículo 2 por el participante 2
		H1-3	D1-3	-	Visor del artículo 3 por el participante 3
		H1-4	D1-1	-	Visor del artículo 1 por el participante 4
		H1-5	D1-2	-	Visor del artículo 2 por el participante 5
		H1-6	D1-3	-	Visor del artículo 3 por el participante 6
		H1-7	D1-1	-	Visor del artículo 1 por el participante 7
		H1-8	D1-2	-	Visor del artículo 2 por el participante 8
		H1-9	D1-3	-	Visor del artículo 3 por el participante 9
		H1-10	D1-1	-	Visor del artículo 1 por el participante 10
		H1-11	D1-2	-	Visor del artículo 2 por el participante 11
		H1-12	D1-3	-	Visor del artículo 3 por el participante 12

Tabla C.2: Instancias y argumentos de la herramientas H1 (visor de documentos) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H2	12	H2-1	-	D2-1	Editor del mapa conceptual del

				participante 1	
		H2-2	-	D2-2	Editor del mapa conceptual del participante 2
		H2-3	-	D2-3	Editor del mapa conceptual del participante 3
		H2-4	-	D2-4	Editor del mapa conceptual del participante 4
		H2-5	-	D2-5	Editor del mapa conceptual del participante 5
		H2-6	-	D2-6	Editor del mapa conceptual del participante 6
		H2-7	-	D2-7	Editor del mapa conceptual del participante 7
		H2-8	-	D2-8	Editor del mapa conceptual del participante 8
		H2-9	-	D2-9	Editor del mapa conceptual del participante 9
		H2-10	-	D2-10	Editor del mapa conceptual del participante 10
		H2-11	-	D2-11	Editor del mapa conceptual del participante 11
		H2-12	-	D2-12	Editor del mapa conceptual del participante 12

Tabla C.3: Instancias y argumentos de la herramienta H2 (editor de mapa conceptual) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H3	12	H3-1	-	D3-1	Editor de preguntas del participante 1
		H3-2	-	D3-2	Editor de preguntas del participante 2
		H3-3	-	D3-3	Editor de preguntas del participante 3
		H3-4	-	D3-4	Editor de preguntas del participante 4
		H3-5	-	D3-5	Editor de preguntas del participante 5
		H3-6	-	D3-6	Editor de preguntas del participante 6
		H3-7	-	D3-7	Editor de preguntas del participante 7
		H3-8	-	D3-8	Editor de preguntas del participante 8
		H3-9	-	D3-9	Editor de preguntas del participante 9
		H3-10	-	D3-10	Editor de preguntas del participante 10
		H3-11	-	D3-11	Editor de preguntas del participante 11
		H3-12	-	D3-12	Editor de preguntas del participante 12

Tabla C.4: Instancias y argumentos de la herramienta H3 (editor de preguntas) de la situación de aprendizaje 3.1.

Hta.	Inst.	Id	Entrada	Salida	Descripción
------	-------	----	---------	--------	-------------

	Cant.				
<i>H4</i>	12	<i>H4-1</i>	<i>D1-1</i>	-	Visor del artículo 1 por el participante 1
		<i>H4-2</i>	<i>D1-2</i>	-	Visor del artículo 2 por el participante 2
		<i>H4-3</i>	<i>D1-3</i>	-	Visor del artículo 3 por el participante 3
		<i>H4-4</i>	<i>D1-1</i>	-	Visor del artículo 1 por el participante 4
		<i>H4-5</i>	<i>D1-2</i>	-	Visor del artículo 2 por el participante 5
		<i>H4-6</i>	<i>D1-3</i>	-	Visor del artículo 3 por el participante 6
		<i>H4-7</i>	<i>D1-1</i>	-	Visor del artículo 1 por el participante 7
		<i>H4-8</i>	<i>D1-2</i>	-	Visor del artículo 2 por el participante 8
		<i>H4-9</i>	<i>D1-3</i>	-	Visor del artículo 3 por el participante 9
		<i>H4-10</i>	<i>D1-1</i>	-	Visor del artículo 1 por el participante 10
		<i>H4-11</i>	<i>D1-2</i>	-	Visor del artículo 2 por el participante 11
		<i>H4-12</i>	<i>D1-3</i>	-	Visor del artículo 3 por el participante 12

Tabla C.5: Instancias y argumentos de la herramienta H4 (visor de documentos) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
<i>H5</i>	48	<i>H5-1</i>	<i>D2-1</i>	<i>D2-1</i>	Editor del mapa conceptual del participante 1 por el participante 1
		<i>H5-2</i>	<i>D2-4</i>	<i>D2-4</i>	Editor del mapa conceptual del participante 4 por el participante 1
		<i>H5-3</i>	<i>D2-7</i>	<i>D2-7</i>	Editor del mapa conceptual del participante 7 por el participante 1
		<i>H5-4</i>	<i>D2-10</i>	<i>D2-10</i>	Editor del mapa conceptual del participante 10 por el participante 1
		<i>H5-5</i>	<i>D2-2</i>	<i>D2-2</i>	Editor del mapa conceptual del participante 2 por el participante 2
		<i>H5-6</i>	<i>D2-5</i>	<i>D2-5</i>	Editor del mapa conceptual del participante 5 por el participante 2
		<i>H5-7</i>	<i>D2-8</i>	<i>D2-8</i>	Editor del mapa conceptual del participante 8 por el participante 2
		<i>H5-8</i>	<i>D2-11</i>	<i>D2-11</i>	Editor del mapa conceptual del participante 11 por el participante 2
		<i>H5-9</i>	<i>D2-3</i>	<i>D2-3</i>	Editor del mapa conceptual del participante 3 por el participante 3
		<i>H5-10</i>	<i>D2-6</i>	<i>D2-6</i>	Editor del mapa conceptual del participante 6 por el participante 3
		<i>H5-11</i>	<i>D2-9</i>	<i>D2-9</i>	Editor del mapa conceptual del participante 9 por el participante 3
		<i>H5-12</i>	<i>D2-12</i>	<i>D2-12</i>	Editor del mapa conceptual del participante 12 por el participante 3
		<i>H5-13</i>	<i>D2-1</i>	<i>D2-1</i>	Editor del mapa conceptual del participante 1 por el participante 4
		<i>H5-14</i>	<i>D2-4</i>	<i>D2-4</i>	Editor del mapa conceptual del participante 4 por el

				participante 4	
		H5-15	D2-7	D2-7	Editor del mapa conceptual del participante 7 por el participante 4
		H5-16	D2-10	D2-10	Editor del mapa conceptual del participante 10 por el participante 4
		H5-17	D2-2	D2-2	Editor del mapa conceptual del participante 2 por el participante 5
		H5-18	D2-5	D2-5	Editor del mapa conceptual del participante 5 por el participante 5
		H5-19	D2-8	D2-8	Editor del mapa conceptual del participante 8 por el participante 5
		H5-20	D2-11	D2-11	Editor del mapa conceptual del participante 11 por el participante 5
		H5-21	D2-3	D2-3	Editor del mapa conceptual del participante 3 por el participante 6
		H5-22	D2-6	D2-6	Editor del mapa conceptual del participante 6 por el participante 6
		H5-23	D2-9	D2-9	Editor del mapa conceptual del participante 9 por el participante 6
		H5-24	D2-12	D2-12	Editor del mapa conceptual del participante 12 por el participante 6

Tabla C.6: Instancias de la 1 a la 24 y argumentos de la herramienta H5 (editor de mapa conceptual) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H5	48	H5-25	D2-1	D2-1	Editor del mapa conceptual del participante 1 por el participante 7
		H5-26	D2-4	D2-4	Editor del mapa conceptual del participante 4 por el participante 7
		H5-27	D2-7	D2-7	Editor del mapa conceptual del participante 7 por el participante 7
		H5-28	D2-10	D2-10	Editor del mapa conceptual del participante 10 por el participante 7
		H5-29	D2-2	D2-2	Editor del mapa conceptual del participante 2 por el participante 8
		H5-30	D2-5	D2-5	Editor del mapa conceptual del participante 5 por el participante 8
		H5-31	D2-8	D2-8	Editor del mapa conceptual del participante 8 por el participante 8
		H5-32	D2-11	D2-11	Editor del mapa conceptual del participante 11 por el participante 8
		H5-33	D2-3	D2-3	Editor del mapa conceptual del participante 3 por el participante 9
		H5-34	D2-6	D2-6	Editor del mapa conceptual del participante 6 por el participante 9
		H5-35	D2-9	D2-9	Editor del mapa conceptual del participante 9 por el participante 9
		H5-36	D2-12	D2-12	Editor del mapa conceptual del participante 12 por el participante 9
		H5-37	D2-1	D2-1	Editor del mapa conceptual del participante 1 por el participante 10
		H5-38	D2-4	D2-4	Editor del mapa conceptual del participante 4 por el participante 10
		H5-39	D2-7	D2-7	Editor del mapa conceptual del participante 7 por el participante 10
		H5-40	D2-10	D2-10	Editor del mapa conceptual del participante 10 por el

Apéndice C

					participante 10
		H5-41	D2-2	D2-2	Editor del mapa conceptual del participante 2 por el participante 11
		H5-42	D2-5	D2-5	Editor del mapa conceptual del participante 5 por el participante 11
		H5-43	D2-8	D2-8	Editor del mapa conceptual del participante 8 por el participante 11
		H5-44	D2-11	D2-11	Editor del mapa conceptual del participante 11 por el participante 11
		H5-45	D2-3	D2-3	Editor del mapa conceptual del participante 3 por el participante 12
		H5-46	D2-6	D2-6	Editor del mapa conceptual del participante 6 por el participante 12
		H5-47	D2-9	D2-9	Editor del mapa conceptual del participante 9 por el participante 12
		H5-48	D2-12	D2-12	Editor del mapa conceptual del participante 12 por el participante 12

Tabla C.7: Instancias de la 25 a la 48 y argumentos de la herramienta H5 (editor de mapa conceptual) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H6	48	H6-1	D3-1	D3-1	Editor de preguntas del participante 1 por el participante 1
		H6-2	D3-4	D3-4	Editor de preguntas del participante 4 por el participante 1
		H6-3	D3-7	D3-7	Editor de preguntas del participante 7 por el participante 1
		H6-4	D3-10	D3-10	Editor de preguntas del participante 10 por el participante 1
		H6-5	D3-2	D3-2	Editor de preguntas del participante 2 por el participante 2
		H6-6	D3-5	D3-5	Editor de preguntas del participante 5 por el participante 2
		H6-7	D3-8	D3-8	Editor de preguntas del participante 8 por el participante 2
		H6-8	D3-11	D3-11	Editor de preguntas del participante 11 por el participante 2
		H6-9	D3-3	D3-3	Editor de preguntas del participante 3 por el participante 3
		H6-10	D3-6	D3-6	Editor de preguntas del participante 6 por el participante 3
		H6-11	D3-9	D3-9	Editor de preguntas del participante 9 por el participante 3
		H6-12	D3-12	D3-12	Editor de preguntas del participante 12 por el participante 3
		H6-13	D3-1	D3-1	Editor de preguntas del participante 1 por el participante 4
		H6-14	D3-4	D3-4	Editor de preguntas del participante 4 por el participante 4
		H6-15	D3-7	D3-7	Editor de preguntas del participante 7 por el participante 4
		H6-16	D3-10	D3-10	Editor de preguntas del participante 10 por el participante 4
		H6-17	D3-2	D3-2	Editor de preguntas del participante 2 por el participante 5
		H6-18	D3-5	D3-5	Editor de preguntas del participante 5 por el participante 5
		H6-19	D3-8	D3-8	Editor de preguntas del participante 8 por el participante 5
		H6-20	D3-11	D3-11	Editor de preguntas del participante 11 por el participante 5
		H6-21	D3-3	D3-3	Editor de preguntas del participante 3 por el participante 6
		H6-22	D3-6	D3-6	Editor de preguntas del participante 6 por el participante 6
		H6-23	D3-9	D3-9	Editor de preguntas del participante 9 por el participante 6
		H6-24	D3-12	D3-12	Editor de preguntas del participante 12 por el participante 6

Tabla C.8: Instancias de la 1 a la 24 y argumentos de la herramienta H6 (editor de preguntas) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H6	48	H6-25	D3-1	D3-1	Editor de preguntas del participante 1 por el participante 7
		H6-26	D3-4	D3-4	Editor de preguntas del participante 4 por el participante 7
		H6-27	D3-7	D3-7	Editor de preguntas del participante 7 por el participante 7
		H6-28	D3-10	D3-10	Editor de preguntas del participante 10 por el participante 7

Apéndice C

		H6-29	D3-2	D3-2	Editor de preguntas del participante 2 por el participante 8
		H6-30	D3-5	D3-5	Editor de preguntas del participante 5 por el participante 8
		H6-31	D3-8	D3-8	Editor de preguntas del participante 8 por el participante 8
		H6-32	D3-11	D3-11	Editor de preguntas del participante 11 por el participante 8
		H6-33	D3-3	D3-3	Editor de preguntas del participante 3 por el participante 9
		H6-34	D3-6	D3-6	Editor de preguntas del participante 6 por el participante 9
		H6-35	D3-9	D3-9	Editor de preguntas del participante 9 por el participante 9
		H6-36	D3-12	D3-12	Editor de preguntas del participante 12 por el participante 9
		H6-37	D3-1	D3-1	Editor de preguntas del participante 1 por el participante 10
		H6-38	D3-4	D3-4	Editor de preguntas del participante 4 por el participante 10
		H6-39	D3-7	D3-7	Editor de preguntas del participante 7 por el participante 10
		H6-40	D3-10	D3-10	Editor de preguntas del participante 10 por el participante 10
		H6-41	D3-2	D3-2	Editor de preguntas del participante 2 por el participante 11
		H6-42	D3-5	D3-5	Editor de preguntas del participante 5 por el participante 11
		H6-43	D3-8	D3-8	Editor de preguntas del participante 8 por el participante 11
		H6-44	D3-11	D3-11	Editor de preguntas del participante 11 por el participante 11
		H6-45	D3-3	D3-3	Editor de preguntas del participante 3 por el participante 12
		H6-46	D3-6	D3-6	Editor de preguntas del participante 6 por el participante 12
		H6-47	D3-9	D3-9	Editor de preguntas del participante 9 por el participante 12
		H6-48	D3-12	D3-12	Editor de preguntas del participante 12 por el participante 12

Tabla C.9: Instancias de la 25 a la 48 y argumentos de la herramienta H6 (editor de preguntas) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H7	3	H7-1	D1-1	-	Visor del artículo 1 por el grupo de expertos 1
		H7-2	D1-2	-	Visor del artículo 2 por el grupo de expertos 2
		H7-3	D1-3	-	Visor del artículo 3 por el grupo de expertos 3

Tabla C.10: Instancias y argumentos de la herramienta H7 (visor de documentos) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H8	12	H8-1	D2-1	-	Editor del mapa conceptual del participante 1 por el grupo de expertos 1
		H8-2	D2-2	-	Editor del mapa conceptual del participante 2 por el grupo de expertos 2
		H8-3	D2-3	-	Editor del mapa conceptual del participante 3 por el grupo de expertos 3
		H8-4	D2-4	-	Editor del mapa conceptual del participante 4 por el grupo de expertos 1
		H8-5	D2-5	-	Editor del mapa conceptual del participante 5 por el grupo de expertos 2
		H8-6	D2-6	-	Editor del mapa conceptual del participante 6 por el grupo de expertos 3
		H8-7	D2-7	-	Editor del mapa conceptual del participante 7 por el grupo de expertos 1
		H8-8	D2-8	-	Editor del mapa conceptual del participante 8 por el grupo de expertos 2
		H8-9	D2-9	-	Editor del mapa conceptual del participante 9 por el grupo de expertos 3
		H8-10	D2-10	-	Editor del mapa conceptual del participante 10 por el grupo de expertos 3

Apéndice C

					de expertos 1
		H8-11	D2-11	-	Editor del mapa conceptual del participante 11 por el grupo de expertos 2
		H8-12	D2-12	-	Editor del mapa conceptual del participante 12 por el grupo de expertos 3

Tabla C.11: Instancias y argumentos de la herramienta H8 (editor de mapa conceptual) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H9	12	H9-1	D3-1	-	Editor de preguntas del participante 1 por el grupo de expertos 1
		H9-2	D3-2	-	Editor de preguntas del participante 2 por el grupo de expertos 2
		H9-3	D3-3	-	Editor de preguntas del participante 3 por el grupo de expertos 3
		H9-4	D3-4	-	Editor de preguntas del participante 4 por el grupo de expertos 1
		H9-5	D3-5	-	Editor de preguntas del participante 5 por el grupo de expertos 2
		H9-6	D3-6	-	Editor de preguntas del participante 6 por el grupo de expertos 3
		H9-7	D3-7	-	Editor de preguntas del participante 7 por el grupo de expertos 1
		H9-8	D3-8	-	Editor de preguntas del participante 8 por el grupo de expertos 2
		H9-9	D3-9	-	Editor de preguntas del participante 9 por el grupo de expertos 3
		H9-10	D3-10	-	Editor de preguntas del participante 10 por el grupo de expertos 1
		H9-11	D3-11	-	Editor de preguntas del participante 11 por el grupo de expertos 2
		H9-12	D3-12	-	Editor de preguntas del participante 12 por el grupo de expertos 3

Tabla C.12: Instancias y argumentos de la herramienta H9 (editor de preguntas) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H10	3	H10-1	-	D4-1	Editor colaborativo de mapa conceptual del grupo de expertos 1
		H10-2	-	D4-2	Editor colaborativo de mapa conceptual del grupo de expertos 2
		H10-3	-	D4-3	Editor colaborativo de mapa conceptual del grupo de expertos 3

Tabla C.13: Instancias y argumentos de la herramienta H10 (editor colaborativo del mapa conceptual) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H11	3	H11-1	-	D5-1	Editor colaborativo de preguntas del grupo de expertos 1
		H11-2	-	D5-2	Editor colaborativo de preguntas del grupo de expertos 2
		H11-3	-	D5-3	Editor colaborativo de preguntas del grupo de expertos 3

Tabla C.14: Instancias y argumentos de la herramienta H11 (editor colaborativo de preguntas) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H12	12	H12-1	D1-1	-	Visor del artículo 1 del grupo de <i>jigsaw</i> 1
		H12-2	D1-1	-	Visor del artículo 1 del grupo de <i>jigsaw</i> 2
		H12-3	D1-1	-	Visor del artículo 1 del grupo de <i>jigsaw</i> 3
		H12-4	D1-1	-	Visor del artículo 1 del grupo de <i>jigsaw</i> 4
		H12-5	D1-2	-	Visor del artículo 2 del grupo de <i>jigsaw</i> 1
		H12-6	D1-2	-	Visor del artículo 2 del grupo de <i>jigsaw</i> 2
		H12-7	D1-2	-	Visor del artículo 2 del grupo de <i>jigsaw</i> 3
		H12-8	D1-2	-	Visor del artículo 2 del grupo de <i>jigsaw</i> 4
		H12-9	D1-3	-	Visor del artículo 3 del grupo de <i>jigsaw</i> 1
		H12-10	D1-3	-	Visor del artículo 3 del grupo de <i>jigsaw</i> 2
		H12-11	D1-3	-	Visor del artículo 3 del grupo de <i>jigsaw</i> 3
		H12-12	D1-3	-	Visor del artículo 3 del grupo de <i>jigsaw</i> 4

Tabla C.15: Instancias y argumentos de la herramienta H12 (visor de documentos) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H13	12	H13-1	D4-1	D4-1	Editor colaborativo del mapa conceptual del grupo de expertos 1 por el grupo de <i>jigsaw</i> 1
		H13-2	D4-1	D4-1	Editor colaborativo del mapa conceptual del grupo de expertos 1 por el grupo de <i>jigsaw</i> 2
		H13-3	D4-1	D4-1	Editor colaborativo del mapa conceptual del grupo de expertos 1 por el grupo de <i>jigsaw</i> 3
		H13-4	D4-1	D4-1	Editor colaborativo del mapa conceptual del grupo de expertos 1 por el grupo de <i>jigsaw</i> 4
		H13-5	D4-2	D4-2	Editor colaborativo del mapa conceptual del grupo de expertos 2 por el grupo de <i>jigsaw</i> 1
		H13-6	D4-2	D4-2	Editor colaborativo del mapa conceptual del grupo de expertos 2 por el grupo de <i>jigsaw</i> 2
		H13-7	D4-2	D4-2	Editor colaborativo del mapa conceptual del grupo de expertos 2 por el grupo de <i>jigsaw</i> 3
		H13-8	D4-2	D4-2	Editor colaborativo del mapa conceptual del grupo de expertos 2 por el grupo de <i>jigsaw</i> 4
		H13-9	D4-3	D4-3	Editor colaborativo del mapa conceptual del grupo de expertos 3 por el grupo de <i>jigsaw</i> 1
		H13-10	D4-3	D4-3	Editor colaborativo del mapa conceptual del grupo de expertos 3 por el grupo de <i>jigsaw</i> 2
		H13-11	D4-3	D4-3	Editor colaborativo del mapa conceptual del grupo de expertos 3 por el grupo de <i>jigsaw</i> 3
		H13-12	D4-3	D4-3	Editor colaborativo del mapa conceptual del grupo de expertos 3 por el grupo de <i>jigsaw</i> 4

Tabla C.16: Instancias y argumentos de la herramienta H13 (editor colaborativo de mapa conceptual) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H14	12	H14-1	D5-1	D5-1	Editor colaborativo de preguntas del grupo de expertos 1 por el grupo de <i>jigsaw</i> 1
		H14-2	D5-1	D5-1	Editor colaborativo de preguntas del grupo de expertos 1 por el grupo de <i>jigsaw</i> 2

		H14-3	D5-1	D5-1	Editor colaborativo de preguntas del grupo de expertos 1 por el grupo de <i>jigsaw</i> 3
		H14-4	D5-1	D5-1	Editor colaborativo de preguntas del grupo de expertos 1 por el grupo de <i>jigsaw</i> 4
		H14-5	D5-2	D5-2	Editor colaborativo de preguntas del grupo de expertos 2 por el grupo de <i>jigsaw</i> 1
		H14-6	D5-2	D5-2	Editor colaborativo de preguntas del grupo de expertos 2 por el grupo de <i>jigsaw</i> 2
		H14-7	D5-2	D5-2	Editor colaborativo de preguntas del grupo de expertos 2 por el grupo de <i>jigsaw</i> 3
		H14-8	D5-2	D5-2	Editor colaborativo de preguntas del grupo de expertos 2 por el grupo de <i>jigsaw</i> 4
		H14-9	D5-3	D5-3	Editor colaborativo de preguntas del grupo de expertos 3 por el grupo de <i>jigsaw</i> 1
		H14-10	D5-3	D5-3	Editor colaborativo de preguntas del grupo de expertos 3 por el grupo de <i>jigsaw</i> 2
		H14-11	D5-3	D5-3	Editor colaborativo de preguntas del grupo de expertos 3 por el grupo de <i>jigsaw</i> 3
		H14-12	D5-3	D5-3	Editor colaborativo de preguntas del grupo de expertos 3 por el grupo de <i>jigsaw</i> 4

Tabla C.17: Instancias y argumentos de la herramienta H14 (editor colaborativo de preguntas) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H15	12	H15-1	D1-1	-	Visor del artículo 1 del grupo de <i>jigsaw</i> 1
		H15-2	D1-1	-	Visor del artículo 1 del grupo de <i>jigsaw</i> 2
		H15-3	D1-1	-	Visor del artículo 1 del grupo de <i>jigsaw</i> 3
		H15-4	D1-1	-	Visor del artículo 1 del grupo de <i>jigsaw</i> 4
		H15-5	D1-2	-	Visor del artículo 2 del grupo de <i>jigsaw</i> 1
		H15-6	D1-2	-	Visor del artículo 2 del grupo de <i>jigsaw</i> 2
		H15-7	D1-2	-	Visor del artículo 2 del grupo de <i>jigsaw</i> 3
		H15-8	D1-2	-	Visor del artículo 2 del grupo de <i>jigsaw</i> 4
		H15-9	D1-3	-	Visor del artículo 3 del grupo de <i>jigsaw</i> 1
		H15-10	D1-3	-	Visor del artículo 3 del grupo de <i>jigsaw</i> 2
		H15-11	D1-3	-	Visor del artículo 3 del grupo de <i>jigsaw</i> 3
		H15-12	D1-3	-	Visor del artículo 3 del grupo de <i>jigsaw</i> 4

Tabla C.18: Instancias y argumentos de la herramienta H15 (visor de documentos) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H16	12	H16-1	D4-1	-	Editor colaborativo del mapa conceptual del grupo de expertos 1 por el grupo de <i>jigsaw</i> 1
		H16-2	D4-1	-	Editor colaborativo del mapa conceptual del grupo de expertos 1 por el grupo de <i>jigsaw</i> 2
		H16-3	D4-1	-	Editor colaborativo del mapa conceptual del grupo de expertos 1 por el grupo de <i>jigsaw</i> 3
		H16-4	D4-1	-	Editor colaborativo del mapa conceptual del grupo de expertos 1 por el grupo de <i>jigsaw</i> 4
		H16-5	D4-2	-	Editor colaborativo del mapa conceptual del grupo de expertos 2 por el grupo de <i>jigsaw</i> 1
		H16-6	D4-2	-	Editor colaborativo del mapa conceptual del grupo de expertos 2 por el grupo de <i>jigsaw</i> 2
		H16-7	D4-2	-	Editor colaborativo del mapa conceptual del grupo de expertos 2 por el grupo de <i>jigsaw</i> 3

		H16-8	D4-2	-	Editor colaborativo del mapa conceptual del grupo de expertos 2 por el grupo de <i>jigsaw</i> 4
		H16-9	D4-3	-	Editor colaborativo del mapa conceptual del grupo de expertos 3 por el grupo de <i>jigsaw</i> 1
		H16-10	D4-3	-	Editor colaborativo del mapa conceptual del grupo de expertos 3 por el grupo de <i>jigsaw</i> 2
		H16-11	D4-3	-	Editor colaborativo del mapa conceptual del grupo de expertos 3 por el grupo de <i>jigsaw</i> 3
		H16-12	D4-3	-	Editor colaborativo del mapa conceptual del grupo de expertos 3 por el grupo de <i>jigsaw</i> 4

Tabla C.19: Instancias y argumentos de la herramienta H16 (editor colaborativo de mapa conceptual) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H17	12	H17-1	D5-1	-	Editor colaborativo de preguntas del grupo de expertos 1 por el grupo de <i>jigsaw</i> 1
		H17-2	D5-1	-	Editor colaborativo de preguntas del grupo de expertos 1 por el grupo de <i>jigsaw</i> 2
		H17-3	D5-1	-	Editor colaborativo de preguntas del grupo de expertos 1 por el grupo de <i>jigsaw</i> 3
		H17-4	D5-1	-	Editor colaborativo de preguntas del grupo de expertos 1 por el grupo de <i>jigsaw</i> 4
		H17-5	D5-2	-	Editor colaborativo de preguntas del grupo de expertos 2 por el grupo de <i>jigsaw</i> 1
		H17-6	D5-2	-	Editor colaborativo de preguntas del grupo de expertos 2 por el grupo de <i>jigsaw</i> 2
		H17-7	D5-2	-	Editor colaborativo de preguntas del grupo de expertos 2 por el grupo de <i>jigsaw</i> 3
		H17-8	D5-2	-	Editor colaborativo de preguntas del grupo de expertos 2 por el grupo de <i>jigsaw</i> 4
		H17-9	D5-3	-	Editor colaborativo de preguntas del grupo de expertos 3 por el grupo de <i>jigsaw</i> 1
		H17-10	D5-3	-	Editor colaborativo de preguntas del grupo de expertos 3 por el grupo de <i>jigsaw</i> 2
		H17-11	D5-3	-	Editor colaborativo de preguntas del grupo de expertos 3 por el grupo de <i>jigsaw</i> 3
		H17-12	D5-3	-	Editor colaborativo de preguntas del grupo de expertos 3 por el grupo de <i>jigsaw</i> 4

Tabla C.20: Instancias y argumentos de la herramienta H17 (editor colaborativo de preguntas) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H18	4	H18-1	-	D6-1	Editor colaborativo de mapa conceptual del grupo de <i>jigsaw</i> 1
		H18-2	-	D6-2	Editor colaborativo de mapa conceptual del grupo de <i>jigsaw</i> 2
		H18-3	-	D6-3	Editor colaborativo de mapa conceptual del grupo de <i>jigsaw</i> 3
		H18-4	-	D6-4	Editor colaborativo de mapa conceptual del grupo de <i>jigsaw</i> 4
H19	4	H19-1	-	D7-1	Editor colaborativo de preguntas del grupo de <i>jigsaw</i> 1
		H19-2	-	D7-2	Editor colaborativo de preguntas del grupo de <i>jigsaw</i> 2
		H19-3	-	D7-3	Editor colaborativo de preguntas del grupo de <i>jigsaw</i> 3
		H19-4	-	D7-4	Editor colaborativo de preguntas del grupo de <i>jigsaw</i> 4

Tabla C.21: Instancias y argumentos de las herramientas H18 (editor colaborativo de mapa conceptual) y H19 (editor de preguntas) de la situación de aprendizaje 3.1.

Apéndice C

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H20	6	H20-1	D1-1	-	Visor del artículo 1 por el grupo 1 del nivel 2 de la pirámide
		H20-2	D1-1	-	Visor del artículo 1 por el grupo 2 del nivel 2 de la pirámide
		H20-3	D1-2	-	Visor del artículo 2 por el grupo 1 del nivel 2 de la pirámide
		H20-4	D1-2	-	Visor del artículo 2 por el grupo 2 del nivel 2 de la pirámide
		H20-5	D1-3	-	Visor del artículo 3 por el grupo 1 del nivel 2 de la pirámide
		H20-6	D1-3	-	Visor del artículo 3 por el grupo 2 del nivel 2 de la pirámide
H21	4	H21-1	D6-1	D6-1	Editor colaborativo del mapa conceptual del grupo de <i>jigsaw</i> 1 por el grupo 1 del nivel 2 de la pirámide
		H21-2	D6-2	D6-2	Editor colaborativo del mapa conceptual del grupo de <i>jigsaw</i> 2 por el grupo 1 del nivel 2 de la pirámide
		H21-3	D6-3	D6-3	Editor colaborativo del mapa conceptual del grupo de <i>jigsaw</i> 3 por el grupo 2 del nivel 2 de la pirámide
		H21-4	D6-4	D6-4	Editor colaborativo del mapa conceptual del grupo de <i>jigsaw</i> 4 por el grupo 2 del nivel 2 de la pirámide
H22	4	H22-1	D7-1	D7-1	Editor colaborativo de preguntas del grupo de <i>jigsaw</i> 1 por el grupo 1 del nivel 2 de la pirámide
		H22-2	D7-2	D7-2	Editor colaborativo de preguntas del grupo de <i>jigsaw</i> 2 por el grupo 1 del nivel 2 de la pirámide
		H22-3	D7-3	D7-3	Editor colaborativo de preguntas del grupo de <i>jigsaw</i> 3 por el grupo 2 del nivel 2 de la pirámide
		H22-4	D7-4	D7-4	Editor colaborativo de preguntas del grupo de <i>jigsaw</i> 4 por el grupo 2 del nivel 2 de la pirámide

Tabla C.22: Instancias y argumentos de las herramientas H20 (visor de documentos), H21 (editor colaborativo de mapa conceptual) y H22 (editor colaborativo de preguntas) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H23	6	H23-1	D1-1	-	Visor del artículo 1 por el grupo 1 del nivel 2 de la pirámide
		H23-2	D1-1	-	Visor del artículo 1 por el grupo 2 del nivel 2 de la pirámide
		H23-3	D1-2	-	Visor del artículo 2 por el grupo 1 del nivel 2 de la pirámide
		H23-4	D1-2	-	Visor del artículo 2 por el grupo 2 del nivel 2 de la pirámide
		H23-5	D1-3	-	Visor del artículo 3 por el grupo 1 del nivel 2 de la pirámide
		H23-6	D1-3	-	Visor del artículo 3 por el grupo 2 del nivel 2 de la pirámide
H24	4	H24-1	D6-1	-	Editor colaborativo del mapa conceptual del grupo de <i>jigsaw</i> 1 por el grupo 1 del nivel 2 de la pirámide
		H24-2	D6-2	-	Editor colaborativo del mapa conceptual del grupo de <i>jigsaw</i> 2 por el grupo 1 del nivel 2 de la pirámide
		H24-3	D6-3	-	Editor colaborativo del mapa conceptual del grupo de <i>jigsaw</i> 3 por el grupo 2 del nivel 2 de la pirámide
		H24-4	D6-4	-	Editor colaborativo del mapa conceptual del grupo de <i>jigsaw</i> 4 por el grupo 2 del nivel 2 de la pirámide
H25	4	H25-1	D7-1	-	Editor colaborativo de preguntas del grupo de <i>jigsaw</i> 1 por el grupo 1 del nivel 2 de la pirámide
		H25-2	D7-2	-	Editor colaborativo de preguntas del grupo de <i>jigsaw</i> 2 por el grupo 1 del nivel 2 de la pirámide
		H25-3	D7-3	-	Editor colaborativo de preguntas del grupo de <i>jigsaw</i> 3 por el grupo 2 del nivel 2 de la pirámide
		H25-4	D7-4	-	Editor colaborativo de preguntas del grupo de <i>jigsaw</i> 4 por el grupo 2 del nivel 2 de la pirámide
H26	2	H26-1	-	D8-1	Editor colaborativo del mapa conceptual del grupo 1 del nivel 2 de la pirámide
		H26-2	-	D8-2	Editor colaborativo del mapa conceptual del grupo 2 del nivel 2 de la pirámide
H27	2	H27-1	-	D9-1	Editor colaborativo de preguntas del grupo 1 del nivel 2 de la

Apéndice C

					pirámide
		H27-2	-	D9-2	Editor colaborativo de preguntas del grupo 2 del nivel 2 de la pirámide

Tabla C.23: Instancias y argumentos de la herramienta H23 (visor de documentos), H24 (editor colaborativo de mapa conceptual), H25 (editor colaborativo de preguntas), H26 (editor colaborativo de mapa conceptual) y H27 (editor colaborativo de preguntas) de la situación de aprendizaje 3.1.

Hta.	Inst. Cant.	Id	Entrada	Salida	Descripción
H28	3	H28-1	D1-1	-	Visor del artículo 1 para toda la clase
		H28-2	D1-2	-	Visor del artículo 2 para toda la clase
		H28-3	D1-3	-	Visor del artículo 3 para toda la clase
H29	2	H29-1	D8-1	-	Editor colaborativo del mapa conceptual del grupo 1 del nivel 2 de la pirámide
		H29-2	D8-2	-	Editor colaborativo del mapa conceptual del grupo 2 del nivel 2 de la pirámide
H30	2	H30-1	D9-1	-	Editor colaborativo de preguntas del grupo 1 del nivel 2 de la pirámide
		H30-2	D9-2	-	Editor colaborativo de preguntas del grupo 2 del nivel 2 de la pirámide
H31	1	H31-1	-	D10-1	Editor colaborativo del mapa conceptual de toda la clase
H32	1	H32-1	-	D11-1	Editor colaborativo de las conclusiones de toda la clase

Tabla C.24: Instancias y argumentos de la herramienta H28 (visor de documentos), H29 (editor colaborativo de mapa conceptual), H30 (editor colaborativo de preguntas), H31 (editor colaborativo de mapa conceptual) y H32 (editor de conclusiones) de la situación de aprendizaje 3.1.

Situación ID	Roles del Flujo de Aprendizaje				
	Estudiante	Experto	Jigsaw	Pirámide Nivel 2	Pirámide Nivel 3
3.1	p1,p2,p3,p4, p5,p6,p7,p8, p9,p10,p11,p12	p1,p2,p3,p4, p5,p6,p7,p8, p9,p10,p11,p12	p1,p2,p3,p4, p5,p6,p7,p8, p9,p10,p11,p12	p1,p2,p3,p4, p5,p6,p7,p8, p9,p10,p11,p12	p1,p2,p3,p4, p5,p6,p7,p8, p9,p10,p11,p12

Tabla C.25: Asignación de participantes a roles de aprendizaje correspondiente a la situación de aprendizaje 3.1.

Hta.	Id	Particip.									
H1	H1-1	p1	H2	H2-1	p1	H3	H3-1	p1	H4	H4-1	p1
	H1-2	p2		H2-2	p2		H3-2	p2		H4-2	p2
	H1-3	p3		H2-3	p3		H3-3	p3		H4-3	p3
	H1-4	p4		H2-4	p4		H3-4	p4		H4-4	p4
	H1-5	p5		H2-5	p5		H3-5	p5		H4-5	p5
	H1-6	p6		H2-6	p6		H3-6	p6		H4-6	p6
	H1-7	p7		H2-7	p7		H3-7	p7		H4-7	p7
	H1-8	p8		H2-8	p8		H3-8	p8		H4-8	p8
	H1-9	p9		H2-9	p9		H3-9	p9		H4-9	p9
	H1-10	p10		H2-10	p10		H3-10	p10		H4-10	p10
	H1-11	p11		H2-11	p11		H3-11	p11		H4-11	p11
	H1-12	p12		H2-12	p12		H3-12	p12		H4-12	p12

Tabla C.26: Asignación de participantes a las instancias de herramientas H1-H4 de la situación de aprendizaje 3.1.

Hta.	Id	Particip.	Id	Particip.	Id	Particip.	Id	Particip.
H5	H5-1	p1	H5-13	p4	H5-25	p7	H5-37	p10
	H5-2	p1	H5-14	p4	H5-26	p7	H5-38	p10

Apéndice C

H5-3	p1	H5-15	p4	H5-27	p7	H5-39	p10
H5-4	p1	H5-16	p4	H5-28	p7	H5-40	p10
H5-5	p2	H5-17	p5	H5-29	p8	H5-41	p11
H5-6	p2	H5-18	p5	H5-30	p8	H5-42	p11
H5-7	p2	H5-19	p5	H5-31	p8	H5-43	p11
H5-8	p2	H5-20	p5	H5-32	p8	H5-44	p11
H5-9	p3	H5-21	p6	H5-33	p9	H5-45	p12
H5-10	p3	H5-22	p6	H5-34	p9	H5-46	p12
H5-11	p3	H5-23	p6	H5-35	p9	H5-47	p12
H5-12	p3	H5-24	p6	H5-36	p9	H5-48	p12

Tabla C.27: Asignación de participantes a las instancias de la herramienta H5 de la situación de aprendizaje 3.1.

Hta.	Id	Particip.	Id	Particip.	Id	Particip.	Id	Particip.
H6	H6-1	p1	H6-13	p4	H6-25	p7	H6-37	p10
	H6-2	p1	H6-14	p4	H6-26	p7	H6-38	p10
	H6-3	p1	H6-15	p4	H6-27	p7	H6-39	p10
	H6-4	p1	H6-16	p4	H6-28	p7	H6-40	p10
	H6-5	p2	H6-17	p5	H6-29	p8	H6-41	p11
	H6-6	p2	H6-18	p5	H6-30	p8	H6-42	p11
	H6-7	p2	H6-19	p5	H6-31	p8	H6-43	p11
	H6-8	p2	H6-20	p5	H6-32	p8	H6-44	p11
	H6-9	p3	H6-21	p6	H6-33	p9	H6-45	p12
	H6-10	p3	H6-22	p6	H6-34	p9	H6-46	p12
	H6-11	p3	H6-23	p6	H6-35	p9	H6-47	p12
	H6-12	p3	H6-24	p6	H6-36	p9	H6-48	p12

Tabla C.28: Asignación de participantes a las instancias de la herramienta H6 de la situación de aprendizaje 3.1.

Hta.	Id	Participantes
H7	H7-1	p1,p4,p7,p10
	H7-2	p2,p5,p8,p11
	H7-3	p3,p6,p9,p12

Tabla C.29: Asignación de participantes a las instancias de la herramienta H7 de la situación de aprendizaje 3.1.

Hta.	Id	Participantes	Hta.	Id	Participantes
H8	H8-1	p1,p4,p7,p10	H9	H9-1	p1,p4,p7,p10
	H8-2	p2,p5,p8,p11		H9-2	p2,p5,p8,p11
	H8-3	p3,p6,p9,p12		H9-3	p3,p6,p9,p12
	H8-4	p1,p4,p7,p10		H9-4	p1,p4,p7,p10
	H8-5	p2,p5,p8,p11		H9-5	p2,p5,p8,p11
	H8-6	p3,p6,p9,p12		H9-6	p3,p6,p9,p12
	H8-7	p1,p4,p7,p10		H9-7	p1,p4,p7,p10
	H8-8	p2,p5,p8,p11		H9-8	p2,p5,p8,p11
	H8-9	p3,p6,p9,p12		H9-9	p3,p6,p9,p12
	H8-10	p1,p4,p7,p10		H9-10	p1,p4,p7,p10
	H8-11	p2,p5,p8,p11		H9-11	p2,p5,p8,p11
	H8-12	p3,p6,p9,p12		H9-12	p3,p6,p9,p12

Tabla C.30: Asignación de participantes a las instancias de herramientas H8 y H9 de la situación de aprendizaje 3.1.

Hta.	Id	Participantes	Hta.	Id	Participantes
H10	H10-1	p1,p4,p7,p10	H11	H11-1	p1,p4,p7,p10
	H10-2	p2,p5,p8,p11		H11-2	p2,p5,p8,p11
	H10-3	p3,p6,p9,p12		H11-3	p3,p6,p9,p12

Tabla C.31: Asignación de participantes a las instancias de herramientas H10 y H11 de la situación de aprendizaje 3.1.

Hta.	Id	Participantes	Hta.	Id	Participantes	Hta.	Id	Participantes
H12	H12-1	p1,p2,p3	H13	H13-1	p1,p2,p3	H14	H14-1	p1,p2,p3
	H12-2	p4,p5,p6		H13-2	p4,p5,p6		H14-2	p4,p5,p6
	H12-3	p7,p8,p9		H13-3	p7,p8,p9		H14-3	p7,p8,p9
	H12-4	p10,p11,p12		H13-4	p10,p11,p12		H14-4	p10,p11,p12
	H12-5	p1,p2,p3		H13-5	p1,p2,p3		H14-5	p1,p2,p3
	H12-6	p4,p5,p6		H13-6	p4,p5,p6		H14-6	p4,p5,p6
	H12-7	p7,p8,p9		H13-7	p7,p8,p9		H14-7	p7,p8,p9
	H12-8	p10,p11,p12		H13-8	p10,p11,p12		H14-8	p10,p11,p12
	H12-9	p1,p2,p3		H13-9	p1,p2,p3		H14-9	p1,p2,p3
	H12-10	p4,p5,p6		H13-10	p4,p5,p6		H14-10	p4,p5,p6
	H12-11	p7,p8,p9		H13-11	p7,p8,p9		H14-11	p7,p8,p9
	H12-12	p10,p11,p12		H13-12	p10,p11,p12		H14-12	p10,p11,p12

Tabla C.32: Asignación de participantes a las instancias de herramientas H12-H14 de la situación de aprendizaje 3.1.

Hta.	Id	Participantes	Hta.	Id	Participantes	Hta.	Id	Participantes
H15	H15-1	p1,p2,p3	H16	H16-1	p1,p2,p3	H17	H17-1	p1,p2,p3
	H15-2	p4,p5,p6		H16-2	p4,p5,p6		H17-2	p4,p5,p6
	H15-3	p7,p8,p9		H16-3	p7,p8,p9		H17-3	p7,p8,p9
	H15-4	p10,p11,p12		H16-4	p10,p11,p12		H17-4	p10,p11,p12
	H15-5	p1,p2,p3		H16-5	p1,p2,p3		H17-5	p1,p2,p3
	H15-6	p4,p5,p6		H16-6	p4,p5,p6		H17-6	p4,p5,p6
	H15-7	p7,p8,p9		H16-7	p7,p8,p9		H17-7	p7,p8,p9
	H15-8	p10,p11,p12		H16-8	p10,p11,p12		H17-8	p10,p11,p12
	H15-9	p1,p2,p3		H16-9	p1,p2,p3		H17-9	p1,p2,p3
	H15-10	p4,p5,p6		H16-10	p4,p5,p6		H17-10	p4,p5,p6
	H15-11	p7,p8,p9		H16-11	p7,p8,p9		H17-11	p7,p8,p9
	H15-12	p10,p11,p12		H16-12	p10,p11,p12		H17-12	p10,p11,p12

Tabla C.33: Asignación de participantes a las instancias de herramientas H15-H17 de la situación de aprendizaje 3.1.

Hta.	Id	Participantes	Hta.	Id	Participantes
H18	H18-1	p1,p2,p3	H19	H19-1	p1,p2,p3
	H18-2	p4,p5,p6		H19-2	p4,p5,p6
	H18-3	p7,p8,p9		H19-3	p7,p8,p9
	H18-4	p10,p11,p12		H19-4	p10,p11,p12

Tabla C.34: Asignación de participantes a las instancias de herramientas H18 y H19 de la situación de aprendizaje 3.1.

Apéndice C

Hta	Id	Participantes	Hta	Id	Participantes	Hta	Id	Participantes
H20	H20-1	p1,p2,p3,p4,p5,p6	H21	H21-1	p1,p2,p3,p4,p5,p6	H22	H22-1	p1,p2,p3,p4,p5,p6
	H20-2	p7,p8,p9,p10,p11, p12		H21-2	p1,p2,p3,p4,p5,p6		H22-2	p1,p2,p3,p4,p5,p6
	H20-3	p1,p2,p3,p4,p5,p6		H21-3	p7,p8,p9,p10,p11, p12		H22-3	p7,p8,p9,p10,p11, p12
	H20-4	p7,p8,p9,p10,p11, p12		H21-4	p7,p8,p9,p10,p11, p12		H22-4	p7,p8,p9,p10,p11, p12
	H20-5	p1,p2,p3,p4,p5,p6						
	H20-6	p7,p8,p9,p10,p11, p12						

Tabla C.35: Asignación de participantes a las instancias de herramientas H20-H22 de la situación de aprendizaje 3.1.

Hta	Id	Participantes	Hta	Id	Participantes	Hta	Id	Participantes
H23	H23-1	p1,p2,p3,p4,p5,p6	H24	H24-1	p1,p2,p3,p4,p5,p6	H25	H25-1	p1,p2,p3,p4,p5,p6
	H23-2	p7,p8,p9,p10,p11, p12		H24-2	p1,p2,p3,p4,p5,p6		H25-2	p1,p2,p3,p4,p5,p6
	H23-3	p1,p2,p3,p4,p5,p6		H24-3	p7,p8,p9,p10,p11, p12		H25-3	p7,p8,p9,p10,p11, p12
	H23-4	p7,p8,p9,p10,p11, p12		H24-4	p7,p8,p9,p10,p11, p12		H25-4	p7,p8,p9,p10,p11, p12
	H23-5	p1,p2,p3,p4,p5,p6						
	H23-6	p7,p8,p9,p10,p11, p12						

Tabla C.36: Asignación de participantes a las instancias de herramientas H23-H25 de la situación de aprendizaje 3.1.

Hta.	Id	Participantes	Hta.	Id	Participantes	Hta.	Id	Participantes
H26	H26-1	p1,p2,p3,p4,p5,p6	H27	H27-1	p1,p2,p3,p4,p5,p6	H28	H28-1	p1,p2,p3,p4,p5,p6, p7,p8,p9,p10,p11,p12
	H26-2	p7,p8,p9,p10,p11,p12		H27-2	p7,p8,p9,p10,p11,p12			

Tabla C.37: Asignación de participantes a las instancias de herramientas H26-H28 de la situación de aprendizaje 3.1.

Apéndice D Puesta en marcha de una UoLeadFlow4LD

El componente analizador LeadFlow4LD requerido para la coordinación entre flujos no se ha conseguido implementar en esta Tesis. Sin embargo, en este apéndice se describe el procedimiento “manual” que el desarrollador ha usado para conseguir poner en marcha una UoLeadFlow4LD. Todos los documentos y ficheros relacionados con este propósito son incluidos en el disco anexo en la ruta: `.\prototipo\`. En la sección D.1 se lista el software utilizado para este propósito. En la sección D.2 se describe el procedimiento usado por el desarrollador para preparar la puesta en marcha de una UoLeadFlow4LD. En las secciones D.3 y D.4 se describe el procedimiento de despliegue de los motores de flujo de aprendizaje y flujo de trabajo, respectivamente. En las secciones D.5 y D.6 se dan detalles de los adaptadores usados para invocar a los motores de flujo de aprendizaje y flujo de trabajo, respectivamente. En la sección D.7 se dan detalles del adaptador usado para invocar herramientas Web. Por último, en la sección D.8 se describe el procedimiento de publicación e instanciación del documento del flujo de aprendizaje sincronizado.

D.1 Documentos de estructuras y situaciones

Debido a la gran cantidad de documentos que describen todas las estructuras y situaciones de aprendizaje colaborativo, y que fueron usados para validar el método propuesto, todos estos documentos son incluidos en el disco anexo en la ruta: `.\ejemplos`. Un resumen de tales documentos es mostrado en la *Tabla D.1*.

Apéndice D

		Documentos						
Técnica de AC	Sit.	Flujo de Aprendizaje	Flujo de Datos	Coordinación entre Flujos	Instanciación de Grupos	Población de Grupos	Flujo de Aprendizaje Sincronizado	Flujo de Datos Sincronizado
Revisión entre Pares	1.1	lf-Peer-Review.zip	df-Peer-Review.xpdl	fc-Peer-Review.xml	idf-Peer-Review-1.xpdl	sit-Peer-Review-1.xml	sync-lf-Peer-Review-1.zip	sync_df_Peer_Review_1—1.0.bar
	1.2					sit-Peer-Review-2.xml	sync-lf-Peer-Review-2.zip	sync_df_Peer_Review_1—2.0.bar
	1.3				idf-Peer-Review-2.xpdl	sit-Peer-Review-3.xml	sync-lf-Peer-Review-3.zip	sync_df_Peer_Review_1—3.0.bar
	1.4					sit-Peer-Review-4.xml	sync-lf-Peer-Review-4.zip	sync_df_Peer_Review_1—4.0.bar
Convergencia del Conocimiento	2.1	lf-Knowledge-Convergence.zip	df-Knowledge-Convergence.zip	fc-Knowledge-Convergence.zip	idf-Knowledge-Convergence-1.zip	sit-Knowledge-Convergence-1.zip	sync-lf-Knowledge-Convergence-1.zip	sync_df_Knowledge_Convergence_1—1.0.bar
MOSAIC	3.1	lf-Mosaic.zip	df-Mosaic.xpdl	fc-Mosaic.xml	idf-Mosaic-1.xpdl	sit-Mosaic-1.xpdl	sync-lf-Mosaic-1.zip	sync_df_Mosaic_1—1.0.bar

Tabla D.1: Listado de documentos usados para validar el método propuesto y localizados en el disco anexo.

Apéndice D

D.2 Listado de software

El software que ha sido utilizado en esta tesis para poner en marcha una UoLeadFlow4LD se resume en la *Tabla D.2*, mientras que una lista de herramientas de autoría usadas para especificar estructuras y situaciones de aprendizaje colaborativo se muestra en la *Tabla D.3*. El software usado para poner en marcha una UoL, incluyendo herramientas de autoría para especificar estructuras y situaciones de aprendizaje colaborativo es incluido en el disco anexo en la ruta: `.\prototipo\software\`.

Software	Nombre	Versión	Fichero	Observaciones
Compilador de Java	Java JDK	1.6.0_25	jdk-6p25-windows-i586.exe	
Servidor de Aplicaciones	Apache Tomcat	6	apache-tomcat-6.0.32-windows-x64.zip	+Instalar en C:\Tomcat6 +Usar el puerto 8086 (el 8080 es usado por CopperCore) +Al instalar definir el directorio raíz del JDK (C:\jdk1.6.0_25) en lugar del JRE
Motor de Flujo de Aprendizaje	CopperCore	3.0	coppercore_ccrt_3.0-Eloy.rar	+Definir la variable de entorno JAVA_HOME = C:\jdk1.6.0_25\bin
Motor de flujo de trabajo	Bonita Studio	5.8	BOS-5.8-win-setup.exe	
Cliente del Motor CopperCore	clicc	3.0	coppercore_ccrt_3.0-Eloy.rar	

Tabla D.2: Listado del software usado por el sistema prototipo propuesto para poner en marcha una UoLeadFlow4LD.

Tipo	Nombre	Versión	Fichero
Editor Java	Eclipse IDE	Helios	eclipse-SDK-3.7-win32-x86_64.rar
Editor de Diseño de Aprendizaje	<i>Reload Learning Design Editor</i>	2.1.3	reload-ldeditor-win-213.rar
Editor de XPDL	<i>Together Workflow Editor (Jawe)</i>	3.1.2	twe-3.1-2.x86.exe
Editor XML	XML Notepad Editor	2.5.2	XmlNotepad.msi

Tabla D.3: Listado de las herramientas de autoría usadas para especificar una UoLeadFlow4LD conforme al método propuesto.

D.3 Procedimiento de preparación de la puesta en marcha de una UoLeadFlow4LD

La *Figura D.1* describe el procedimiento propuesto para poner en marcha una UoLeadFlow4LD que ha sido especificada conforme al método propuesto. Primero, el diseñador instruccional y el educador son los responsables de definir una UoLeadFlow4LD que representa la estructura y situación de aprendizaje colaborativo. Después, el componente

analizador LeadFlow4LD es el responsable de generar el documento del flujo de aprendizaje sincronizado “incompleto” (más adelante se explica por qué tal nombre), así como el documento del flujo de trabajo sincronizado. A continuación, el administrador del sistema es el responsable de preparar y desplegar los entornos de ejecución, así como de instanciar e inicializar ambos procesos (lo que incluye completar el documento del flujo de aprendizaje sincronizado). Por último, los usuarios son quienes participan durante la puesta en marcha de la situación de aprendizaje. A continuación se describen con más detalle los pasos descritos y se ejemplifica con el caso ilustrativo de revisión entre dos estudiantes.

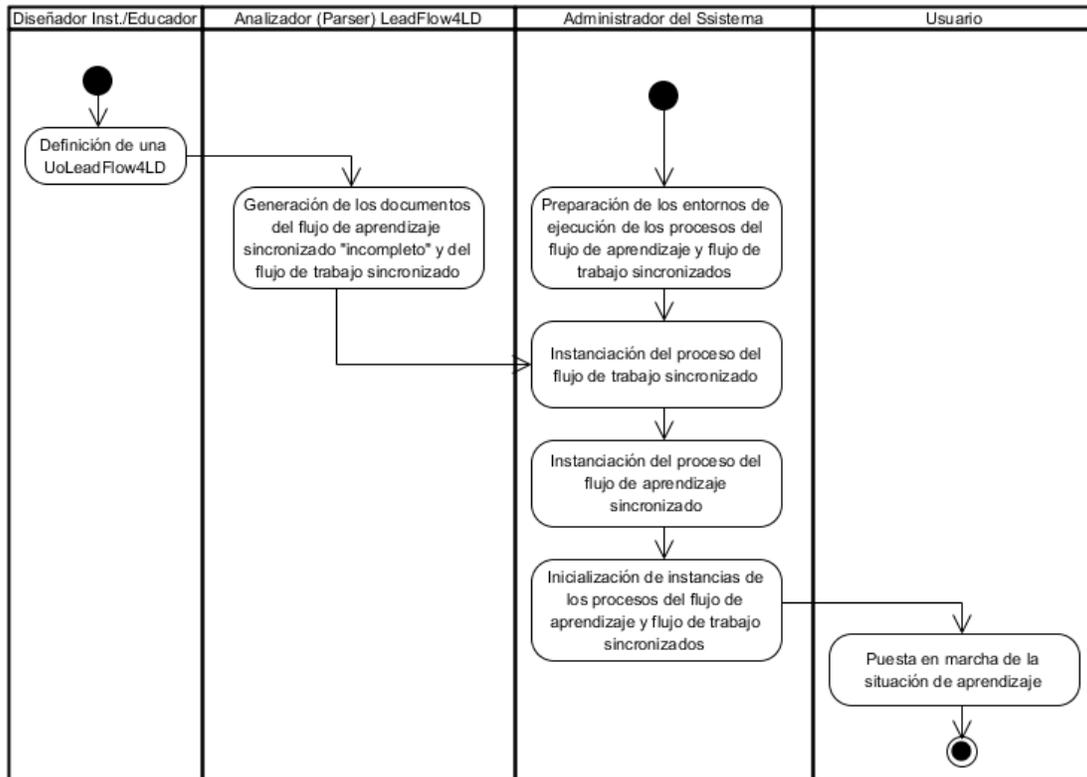


Figura D.1: Procedimiento propuesto para la puesta en marcha de una UoLeadFlow4LD.

De acuerdo a la *Figura D.1*, para poner en marcha una UoLeadFlow4LD, el administrador del sistema debe preparar los entornos de ejecución de ambos flujos. Un entorno de ejecución es un sistema que se compone de:

- *Motores estándares de flujo de aprendizaje y flujo de trabajo.* Son componentes *software* usados para interpretar las definiciones de procesos (flujo de aprendizaje y flujo de trabajo sincronizados) y crear un modelo computacional (máquina de estados) que representa a cada proceso. El motor del flujo de aprendizaje debe ser conforme a IMS LD, mientras que el motor del flujo de trabajo debe ser conforme a XPDL.

- *Herramientas de despliegue e instanciación de procesos.* Son aplicaciones *software* usadas para instalar procesos en un servidor de aplicaciones, así como para crear instancias de los procesos desplegados.
- *Herramientas de asignación de recursos.* Son aplicaciones *software* usadas para mapear participantes concretos a roles abstractos en el proceso del flujo de aprendizaje, y para mapear participantes concretos a actividades de flujo de trabajo humanas en el proceso del flujo de trabajo.
- *Herramientas de gestión de instancias de procesos.* Son aplicaciones *software* usadas para monitorear instancias de procesos, así como para gestionar su estado en tiempo de ejecución.
- *Aplicaciones de invocación de motores de flujo.* Son aplicaciones *software* usadas para comunicación con los motores de flujo. En nuestro caso se requieren para habilitar la coordinación entre flujos, por ejemplo, a través de asignar en tiempo de ejecución, valores a variables de flujo de trabajo (o propiedades IMS LD) de instancias de procesos.

De acuerdo a la *Figura D.1*, una vez preparados los entornos de ejecución, el siguiente paso para poner en marcha una UoLeadFlow4LD es la instanciación de cada proceso. Sin embargo, uno de los problemas a resolver es la cuestión de qué proceso se debe instanciar primero: ¿el proceso del flujo de aprendizaje sincronizado o el proceso del flujo de trabajo sincronizado? El problema radica en que las aplicaciones de invocación de motores usadas para la coordinación entre instancias de procesos dan lugar a un bloqueo mutuo (*deadlock*). Ambas aplicaciones referenciadas dentro de recursos en la definición del proceso, ya sea del flujo de aprendizaje o del flujo de trabajo sincronizados, requieren conocer la instancia del proceso con la que se van a comunicar. Sin embargo, para poder instanciar cualquiera de los procesos, primero deben estar definidos éstos.

Para resolver este bloqueo mutuo, se deben configurar dichos recursos en tiempo de instanciación. Sin embargo, puesto que los recursos de contenido no se pueden modificar una vez que la UoL ha sido desplegada e instanciada, lo que se propone es instanciar primero el proceso del flujo de trabajo sincronizado. Una vez conocida la instancia de este proceso, el administrador del sistema la utiliza para completar el documento del flujo de aprendizaje sincronizado que se había generado “incompleto”. A continuación, el administrador del sistema crea una instancia del flujo de aprendizaje sincronizado (crea una *run* de la UoL), recupera el identificador de la UoL y de la *run*, y los utiliza para ajustar las variables globales de flujo de trabajo que representan la comunicación entre instancias de procesos.

Instanciación del Proceso del Flujo de Trabajo Sincronizado

De acuerdo al procedimiento propuesto, el administrador del sistema debe crear primero una instancia del proceso de flujo de trabajo sincronizado, siguiendo los siguientes pasos:

1. Asignación de recursos

Se refiere al mapeo de recursos concretos a las actividades de flujo de trabajo abstractas definidas en el proceso, tales como participantes concretos, herramientas concretas, formularios de terminación de herramientas y recursos para delegar el control al flujo de aprendizaje. La forma en que esto se lleva a cabo depende del entorno de ejecución seleccionado, en particular, de las herramientas de asignación de recursos integradas al entorno de ejecución.

Por ejemplo, para el caso del ejemplo ilustrativo y el entorno de ejecución Bonita *Open Solution* (BOS), el mapeo de participantes reales a participantes abstractos definidos en el proceso del flujo de trabajo sincronizado se lleva a cabo a través de una herramienta que viene integrada en el sistema BOS. La *Figura D.2* muestra el momento en que el administrador del sistema realiza la asignación del participante concreto *p1* a la actividad de flujo de trabajo que invoca la instancia de la herramienta de edición *H1-1*. Esta tarea debe ser realizada por el administrador del sistema en todas las actividades de flujo de trabajo en donde participan personas. Note que aunque esta tarea la realiza el administrador del sistema, la información la recoge del documento de población de grupos incluida en la *UoLeadFlow4LD*, la cual ha sido definida por el diseñador instruccional y/o el educador.

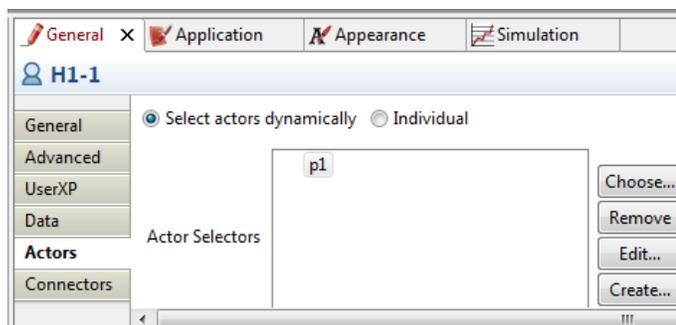


Figura D.2: Configuración en BOS de la asignación del participante concreto p1 a la actividad de flujo de trabajo H1-1, la cual invoca una instancia de la herramienta de edición en el ejemplo ilustrativo.

Similarmente, el mapeo de herramientas concretas a actividades de flujo de trabajo que representan invocaciones de herramientas, se lleva a cabo en el sistema BOS, definiendo y asignando conectores a todas las actividades de flujo de trabajo que representan invocaciones de herramientas. Un conector es una aplicación Java que comunica el motor de BOS (Bonita) con otras aplicaciones. La idea es invocar un conector a la entrada de una actividad de flujo de trabajo que representa la invocación de herramienta. Este conector tiene el propósito de lanzar un navegador Web inicializado en la URL del servicio que será usado como herramienta por el participante. La *Figura D.3* muestra el momento en que el administrador del sistema realiza la asignación del conector *Conn-H1-1* a la actividad de flujo de trabajo *H1-1*, la cual representa una invocación de una instancia de la herramienta de edición. En la *Figura D.4a* se muestra la configuración en BOS del conector *Conn-H1-1*, el cual es invocado a la entrada de la actividad de flujo de trabajo *H1-1*, mientras que en la *Figura D.4b*

se muestra la configuración del único parámetro del conector (el URL de la herramienta). Amabas tareas (asignación y configuración de conectores) son realizadas por el administrador del sistema en todas las actividades de flujo de trabajo que representan invocaciones de herramientas.

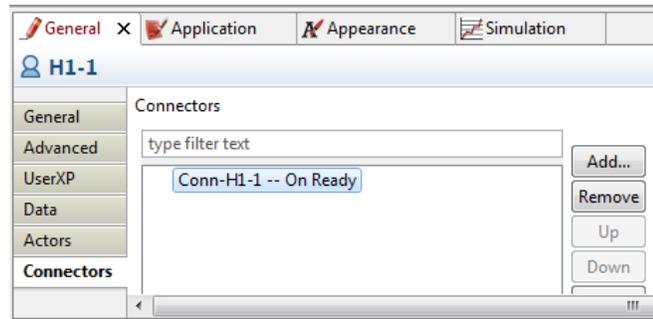
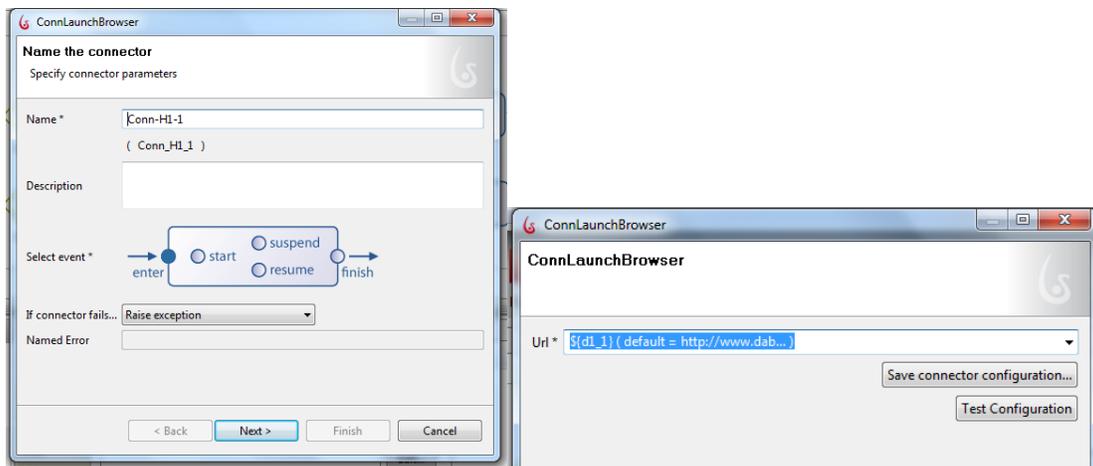


Figura D.3: Configuración en BOS de la asignación del conector *Conn-H1-1* a la actividad de flujo de trabajo que representa una invocación a la herramienta *H1-1* en el ejemplo ilustrativo.



(a)

(b)

Figura D.4: Configuración en BOS del conector usado para invocar la instancia de la herramienta de edición *H1-1*.

Otro tipo de asignación de recursos es la creación y asignación de formularios a las actividades de flujo de trabajo que representan invocaciones de herramientas. En el caso del sistema BOS, los formularios son recursos que comunican a los participantes con Bonita (a través de una interfaz de usuario). La idea es que los participantes indiquen a través del formulario el momento en que han terminado de usar una herramienta. La *Figura D.5* muestra el momento en que el administrador del sistema edita el formulario de flujo de trabajo usado en la actividad *H1-1*, y que le permite al participante de la actividad indicar la terminación de la herramienta (a través del *checkbox* incluido en el formulario). Ambas tareas

(creación y asignación de formularios) es realizada por el administrador del sistema en todas las actividades de flujo de trabajo que representan invocaciones de herramientas.

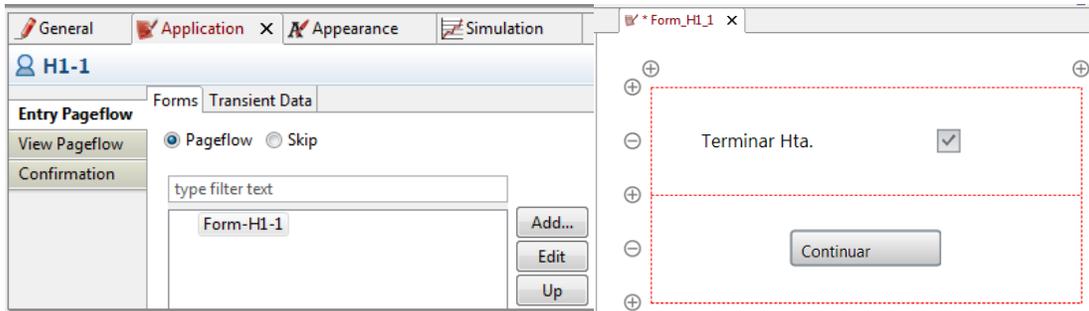


Figura D.5: Configuración en BOS del formulario de flujo de trabajo usado para cambiar el estado de terminación de la instancia de herramienta e edición H1-1.

Por último, se lleva a cabo la asignación de recursos de invocación del flujo de aprendizaje. En el caso del sistema BOS, este tipo de recursos son conectores asignados a las actividades de flujo de trabajo que representan la terminación del flujo de datos. La idea es usar dichos conectores para comunicarse con el motor del flujo de aprendizaje y cambiar el valor de las propiedades IMS LD que representan la terminación de una actividad de aprendizaje. La Figura D.6 muestra el momento en que el administrador del sistema realiza la asignación del conector *Conn-HI-1-fin* a la actividad de flujo de trabajo *HI-1-fin* que representa la terminación del flujo de datos de la actividad de edición. En la Figura D.7 se muestra el *template* usado en el sistema BOS para la configuración de los distintos parámetros del conector. La idea es que al entrar a la actividad de flujo de trabajo que representa el fin del flujo de datos se invoque un adaptador del motor del flujo de aprendizaje (CopperCore), a través del punto de acceso representado por la variable *Endpoint*. Entonces, se cambie el valor de la propiedad IMS LD representado por la variable *Property* para cada uno de los participantes de la actividad, los cuales están representados por la variable *UserList*. La instancia del proceso del flujo de aprendizaje sincronizado es representada por los identificadores de la UoL y de la *run*, esto es, las variables *UoLID* y *RunID*, respectivamente. Ambas tareas (asignar y configurar el conector) son realizadas por el administrador del sistema en todas las actividades de flujo de trabajo que representan la terminación del flujo de datos, mientras que la configuración del conector (inicialización de las variables *UoLID* y *RunID*) se realiza más adelante en tiempo de inicialización.

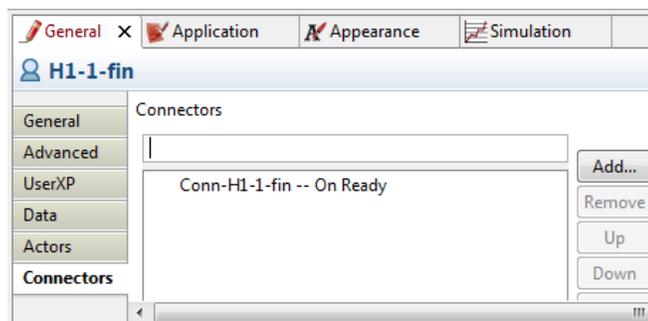


Figura D.6: Configuración en BOS de la asignación del participante concreto p1a la actividad de flujo de trabajo H1-1, la cual invocación una instancia de la herramienta de edición.

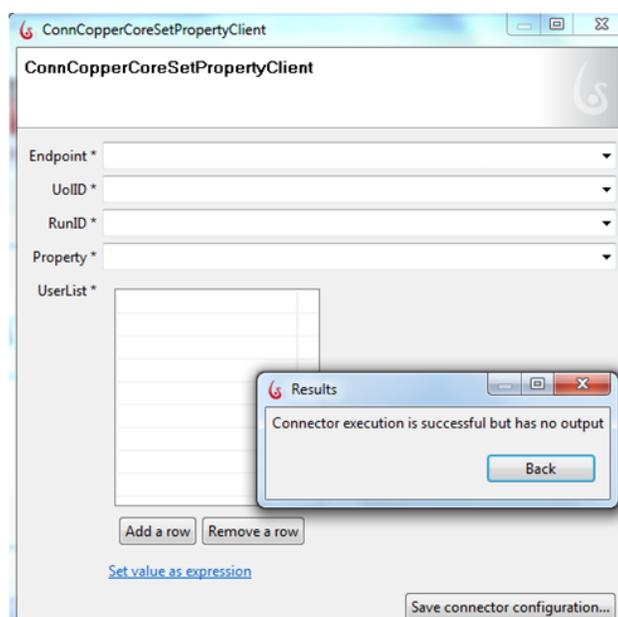


Figura D.7: Plantilla generada por Bonita Studio del conector Java usado para invocar el motor CopperCore con el propósito de modificar el valor de propiedades IMS LD que representan el fin de una actividad de aprendizaje.

2. Exportación del proceso

La exportación se refiere al empaquetamiento del proceso junto con todos sus recursos para su distribución, despliegue e instanciación. La exportación del proceso del flujo de trabajo sincronizado la lleva a cabo el administrador del sistema desde Bonita Studio, que es la herramienta de autoría de flujo de trabajo del sistema BOS. El fichero exportado desde Bonita Studio es un fichero de extensión .bar (*Business ARchive*). La Figura D.8 muestra el momento de exportar el fichero del proceso del flujo de trabajo sincronizado desde Bonita Studio para el caso del ejemplo ilustrativo.

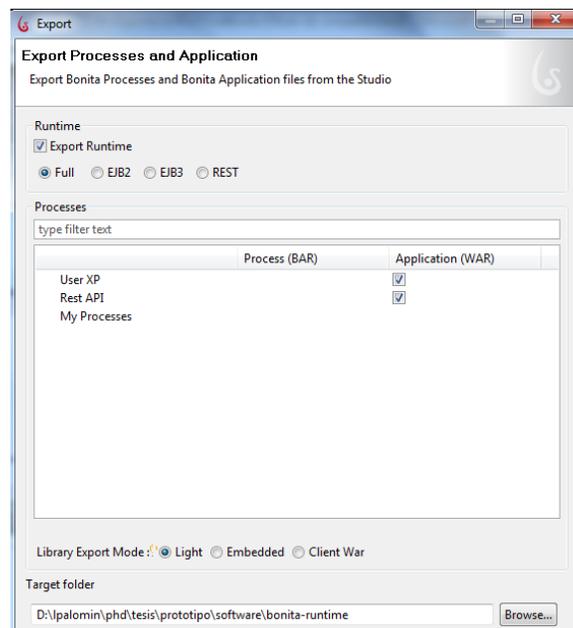


Figura D.8: Exportación desde Bonita Studio del proceso del flujo de trabajo sincronizado.

3. Despliegue del proceso

El despliegue del proceso se refiere a la instalación del proceso dentro del entorno de ejecución. Esta tarea la lleva a cabo el administrador del sistema, que para el caso de la infraestructura tecnológica seleccionada la realiza desde Bonita *User eXPerience* (Bonita *User XP*), que es una herramienta integrada al sistema BOS. Primero, el administrador del sistema se autentica en el sistema BOS (Figura D.9) y a continuación procede al despliegue del proceso (Figura D.10), para el caso del ejemplo ilustrativo.

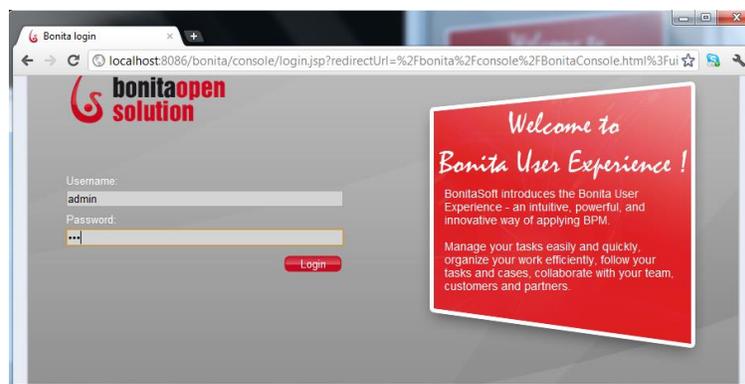


Figura D.9: Autenticación del administrador de sistema en el entorno de ejecución de Bonita Open Solution (BOS).

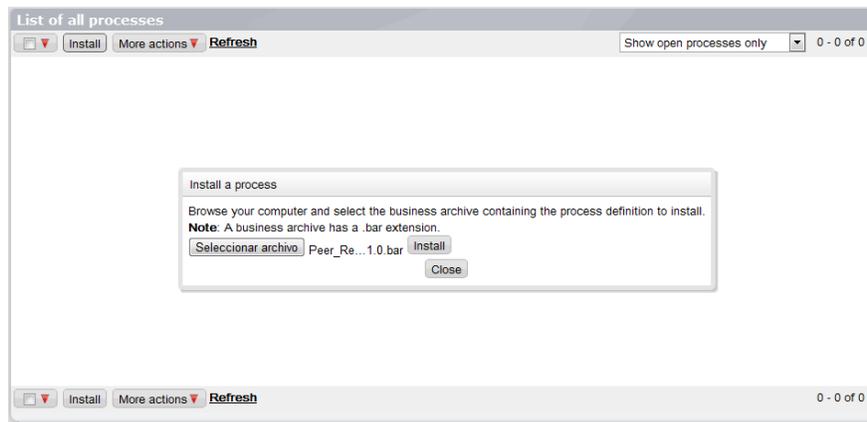


Figura D.10: Despliegue del proceso *sync-Peer-Reviewing-1—1.0.bar* dentro del entorno de ejecución de flujo de trabajo (Bonita Open Solution), correspondiente al ejemplo ilustrativo de revisión entre dos estudiantes.

4. Instanciación del proceso

La instanciación del proceso se refiere a interpretar la definición del proceso (por el motor de flujo) y después crear una copia de los estados que caracterizan el proceso, de tal forma que cada instancia del proceso mantenga su propio estado (por ejemplo, a través de un conjunto de máquinas de estado finito). La instanciación de un proceso es conocida como caso (*case*) en terminología de flujo de trabajo. A cada instancia del proceso de flujo de trabajo sincronizado le corresponde un identificador (*id* de la instancia del proceso), el cual es necesario conocerlo para la coordinación entre procesos. Al igual que la tarea de despliegue, la tarea de instanciación del proceso del flujo de trabajo sincronizado la realiza el administrador del sistema a través de la herramienta Bonita *User XP*, como se muestra en la Figura D.11 para el caso del ejemplo ilustrativo y para la infraestructura tecnológica seleccionada.

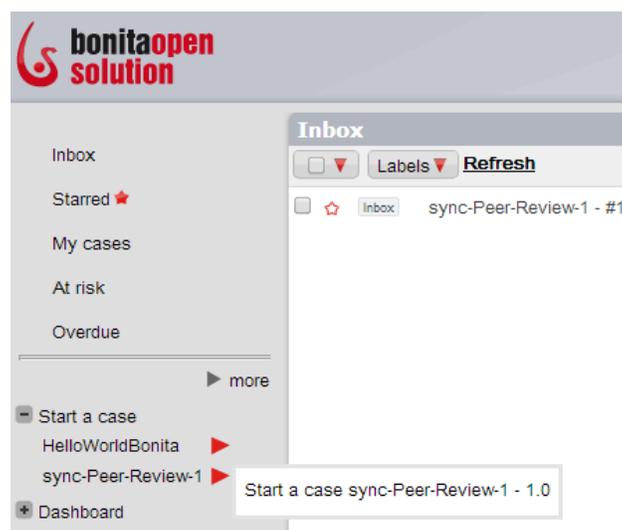


Figura D.11: Despliegue del proceso sync-Peer-Reviewing-1—1.0.bar dentro del entorno de ejecución de flujo de trabajo (Bonita Open Solution), correspondiente al ejemplo ilustrativo de revisión entre dos estudiantes.

Instanciación del Proceso del Flujo de Aprendizaje Sincronizado

De acuerdo al procedimiento propuesto en la *Figura D.1*, el administrador del sistema debe crear a continuación una instancia del proceso del flujo de aprendizaje sincronizado, siguiendo los siguientes pasos:

1. *Completar el documento del flujo de aprendizaje sincronizado*

Completar el documento del flujo de aprendizaje sincronizado se refiere a completar los recursos de invocación del motor de flujo de trabajo (necesarios para la coordinación), los cuales se habían generado “incompletos” por el componente analizador LeadFlow4LD, ya que en tiempo de definición se desconocía el *id* de la instancia del proceso con quien se deben comunicar. Entonces, primero se debe recuperar el *id* del proceso del flujo de trabajo sincronizado para lo que se utiliza una herramienta de gestión de instancias de procesos (que típicamente viene integrada al entorno de ejecución seleccionado), y con esta información se completan los recursos de invocación del motor de flujo de trabajo. La tarea de recuperar el *id* de la instancia del proceso de flujo de trabajo sincronizado la realiza el administrador del sistema utilizando una herramienta que monitorea las instancias creadas en el entorno de ejecución. Para el caso de la infraestructura tecnológica seleccionada, se utilizó una herramienta desarrollada especialmente para éste propósito y que es llamada “Cliente Bonita”. La *Figura D.12* muestra el momento en que “Cliente Bonita” monitorea y muestra los *ids* de los procesos instanciados en el sistema BOS, que corresponden al ejemplo ilustrativo.

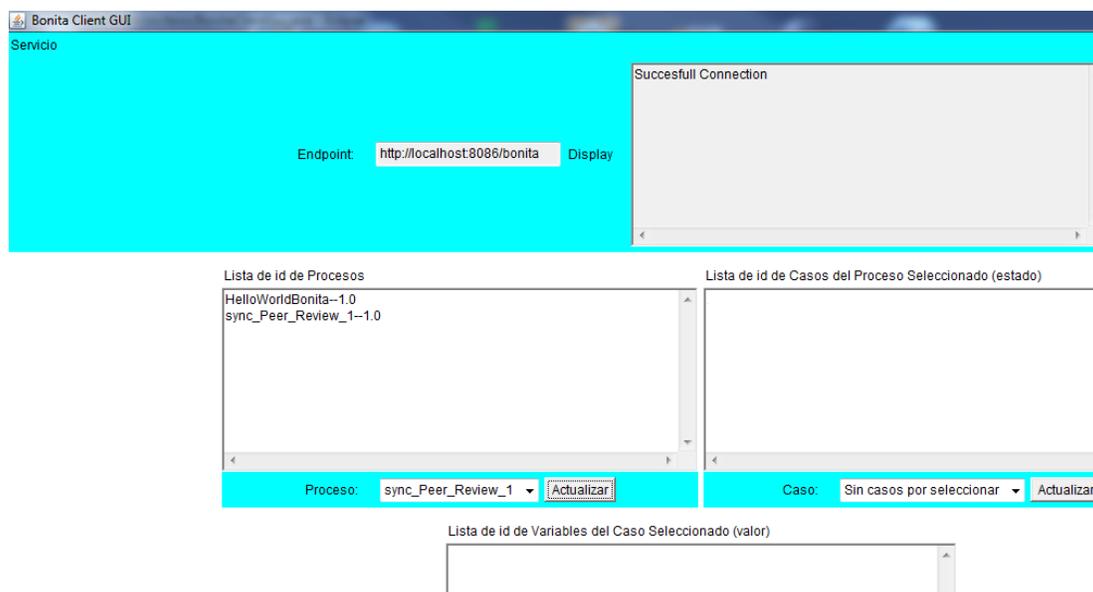


Figura D.12: Cliente Bonita: herramienta usada para monitorear instancias de procesos desplegados en el entorno de ejecución de Bonita Open Solution.

2. Exportación del proceso

Una vez completado el documento del flujo de aprendizaje sincronizado, el siguiente paso es exportarlo (empaquetarlo junto con todos los recursos asociados) para su distribución, publicación (despliegue) y eventual instanciación (creación de una *run*). La exportación del proceso del flujo de aprendizaje sincronizado la lleva a cabo el administrador del sistema utilizando una herramienta de autoría de diseño de aprendizaje que sea conforme con IMS LD, como puede ser *Reload LD Editor*. El fichero exportado es en realidad una UoL (fichero de extensión .zip) conforme con IMS LD. La Figura D.13 muestra el momento de exportar desde *Reload LD Editor*, la UoL que describe el proceso del flujo de aprendizaje sincronizado para el caso del ejemplo ilustrativo.

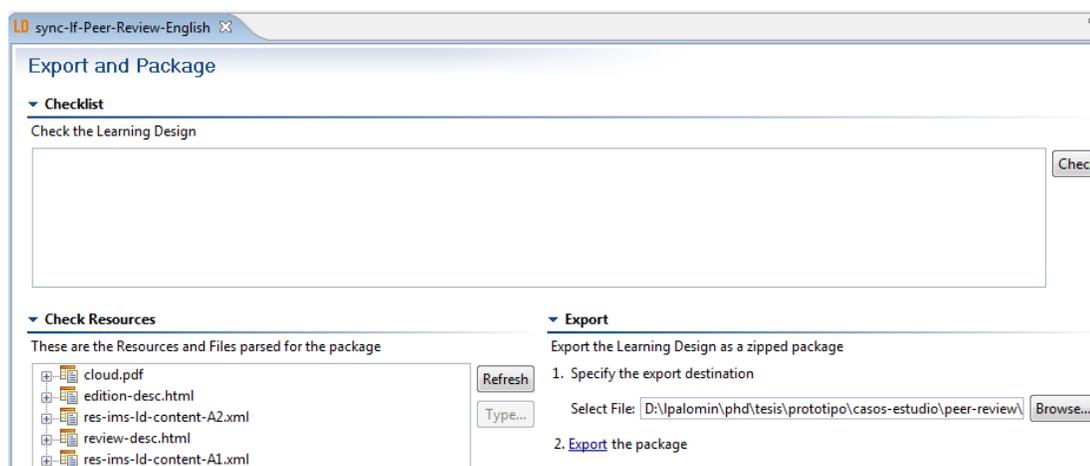


Figura D.13: Cliente Bonita: herramienta usada para monitorear instancias de procesos desplegados en el entorno de ejecución de Bonita Open Solution.

3. Validación y publicación del proceso

Validar el proceso del flujo de aprendizaje sincronizado se refiere a verificar la validez de la definición de la UoL. La publicación del proceso se refiere al despliegue o instalación de la UoL en el entorno de ejecución del flujo de aprendizaje. Ambas tareas son realizadas por el administrador del sistema, que para el caso de la infraestructura tecnológica seleccionada (*CopperCore Run Time Environment*, CCRT), es a través de la herramienta *Publisher* que viene integrada con *CopperCore* y que puede ser invocada desde la URL: <http://localhost:8080/Publisher/publication.html>. La *Figura D.14* muestra la validación y publicación de la UoL correspondiente al ejemplo ilustrativo.

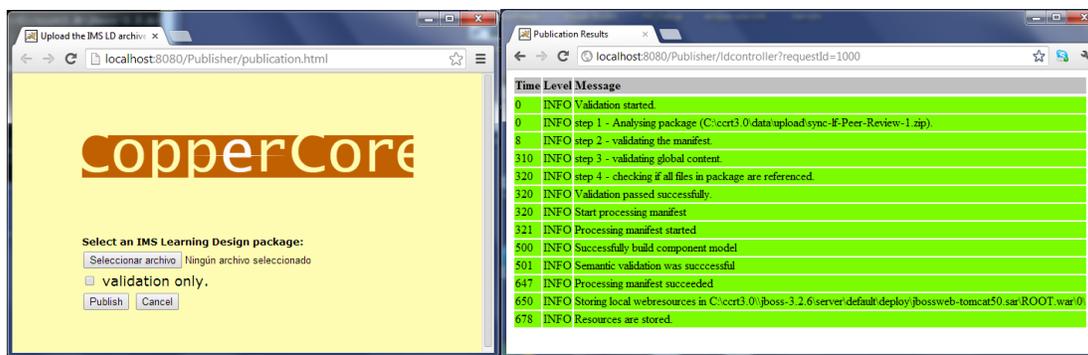


Figura D.14: Validación y publicación de una UoL que describe el proceso del flujo de aprendizaje sincronizado, correspondiente al ejemplo ilustrativo.

4. Instanciación del proceso

Análogo a la instanciación de procesos de flujo de trabajo, la instanciación de proceso del flujo de aprendizaje se refiere a interpretar la definición de la UoL y después crear una copia de los estados que caracterizan el proceso. La instanciación del proceso del flujo de aprendizaje es conocida como *run* en terminología de LD. A cada UoL desplegada y a cada *run* le corresponde un identificador (*id*), que son necesarios conocer para la coordinación entre procesos. Al igual que la tarea de despliegue, la tarea de instanciación del proceso del flujo de aprendizaje sincronizado la realiza el administrador del sistema a través de la herramienta *Command Line Interface CopperCore* (Clicc), como se muestra en la *Figura D.15* para el caso del ejemplo ilustrativo y para la infraestructura tecnológica seleccionada. Note que se utiliza Clicc para crear una *run* de la UoL publicada.



```

Clicc v3.0 - CopperCore Management Console
Clicc:/>ls
UoLid=0,title="Estructura de Aprendizaje Colaborativo de Revisión entre Pares",
uri="http://www.reload.ac.uk/uri/id-665c1953-12ac-de4e-00a0-32971422fc0c",conten
tUri="http://localhost:8080/0/"
Clicc:/>cd 0
Clicc:/uol=0>createuser 0
0
Clicc:/uol=0>cd 0
Clicc:/uol=0/run=0>_

```

Figura D.15: Instanciación del proceso del flujo de aprendizaje sincronizado. En terminología LD representa la creación de una run de la UoL identificada con el id 0 en la Figura.

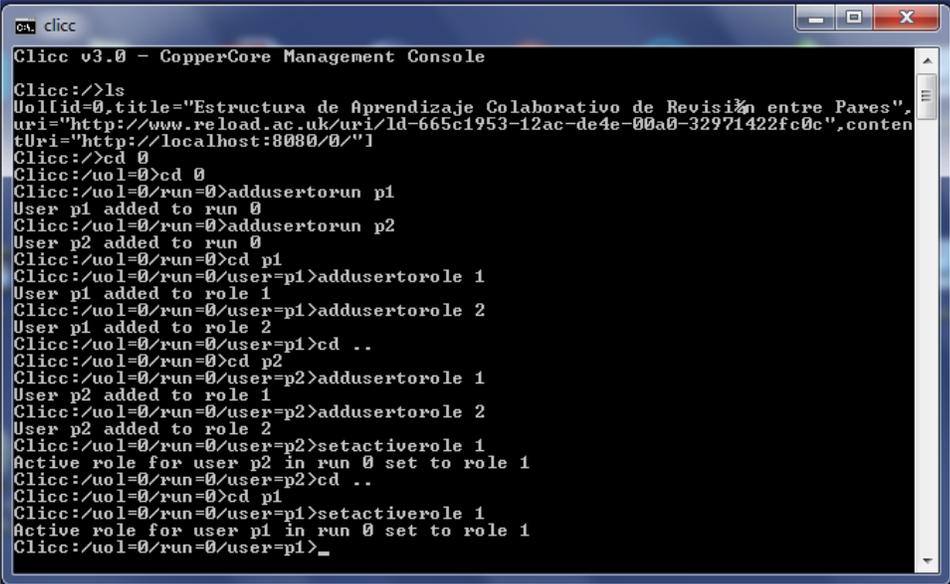
5. Asignación de participantes

Se refiere al mapeo de participantes concretos a las actividades de aprendizaje abstractas definidas en el documento del flujo de aprendizaje sincronizado. La forma en que esto se lleva a cabo depende del entorno de ejecución seleccionado. Por ejemplo, para el caso del entorno de ejecución CopperCore, el mapeo de participantes a las actividades de aprendizaje se puede realizar a través de la herramienta Clicc, la cual se comunica con el motor CopperCore. A través de Clicc, el administrador del sistema crea primero a todos los participantes (usando el comando *createuser*), si es que no han sido creados con anterioridad, luego se asignan éstos a una run (*addusertorun*), luego se asignan a los roles abstractos definidos en el proceso del flujo de aprendizaje sincronizado (*addusertorole*). Para esto, se requiere conocer el *id* de cada rol (para lo que se usa el comando *listroles*). Finalmente se establece el rol activo o inicial de cada participante (*setactiverole*). La *Tabla D.4* describe la secuencia de comandos usados en la herramienta Clicc para el ejemplo ilustrativo, mientras que la *Figura D.16* captura el momento de tal asignación. Note que aunque esta tarea la realiza el administrador del sistema, la información la recoge del documento de población de grupos incluida en la UoLeadFlow4LD, el cual ha sido definido por el diseñador instruccional y/o el educador.

Comando Clicc	Descripción
<i>createuser p1</i>	Crea el participante concreto <i>p1</i>
<i>createuser p2</i>	Crea el participante concreto <i>p2</i>
<i>addusertorun p1</i>	Agrega el participante <i>p1</i> al run 0
<i>addusertorun p2</i>	Agrega el participante <i>p2</i> al run 0
<i>cd p1</i>	Activa al participante <i>p1</i>
<i>addusertorole 1</i>	Agrega el participante activo (<i>p1</i>) al rol abstracto 1 (rol Editor)
<i>addusertorole 2</i>	Agrega el participante activo (<i>p1</i>) al rol abstracto 2 (rol Revisor)
<i>setactiverole 1</i>	Fija el rol 1 (rol Editor) del participante activo (<i>p1</i>) como rol inicial
<i>cd ..</i>	Muévete atrás
<i>cd p2</i>	Activa al participante <i>p2</i>
<i>addusertorole 1</i>	Agrega el participante activo (<i>p2</i>) al rol abstracto 1 (rol Editor)

<code>addusertorole 2</code>	Agrega el participante activo (<i>p2</i>) al rol abstracto 2 (rol Revisor)
<code>setactiverole 1</code>	Fija el rol 1 (rol Editor) del participante activo (<i>p2</i>) como rol inicial

Tabla D.4: Lista de comandos usados en la herramienta *Clicc* para crear una *run* de la *UoLeadFlow4LD* correspondiente al ejemplo ilustrativo.



```

clicc
Clicc v3.0 - CopperCore Management Console
Clicc: />ls
UoId=0,title="Estructura de Aprendizaje Colaborativo de Revisión entre Pares",
uri="http://www.reload.ac.uk/uri/ld-665c1953-12ac-de4e-00a0-32971422fc0c",conten
tUri="http://localhost:8080/0/"
Clicc: />cd 0
Clicc: /uol=0/>cd 0
Clicc: /uol=0/run=0>addusertorun p1
User p1 added to run 0
Clicc: /uol=0/run=0>addusertorun p2
User p2 added to run 0
Clicc: /uol=0/run=0/>cd p1
Clicc: /uol=0/run=0/user=p1>addusertorole 1
User p1 added to role 1
Clicc: /uol=0/run=0/user=p1>addusertorole 2
User p1 added to role 2
Clicc: /uol=0/run=0/user=p1>cd ..
Clicc: /uol=0/run=0/>cd p2
Clicc: /uol=0/run=0/user=p2>addusertorole 1
User p2 added to role 1
Clicc: /uol=0/run=0/user=p2>addusertorole 2
User p2 added to role 2
Clicc: /uol=0/run=0/user=p2>setactiverole 1
Active role for user p2 in run 0 set to role 1
Clicc: /uol=0/run=0/user=p2>cd ..
Clicc: /uol=0/run=0/>cd p1
Clicc: /uol=0/run=0/user=p1>setactiverole 1
Active role for user p1 in run 0 set to role 1
Clicc: /uol=0/run=0/user=p1>_

```

Figura D.16: Instanciación del proceso del flujo de aprendizaje sincronizado utilizando el cliente *Coppercore* (*clicc*), correspondiente al ejemplo ilustrativo.

Inicialización de la Instancia del Proceso del Flujo de Aprendizaje Sincronizado

El siguiente paso para poner en marcha una *UoLeadFlow4LD* es la inicialización de la instancia del proceso del flujo de aprendizaje sincronizado. Esta tarea también es realizada por el administrador del sistema utilizando una herramienta del entorno de ejecución. El objetivo es inicializar el valor de la propiedad IMS LD local personal *P-User*, usada para representar a los distintos participantes. Para el caso de la infraestructura tecnológica seleccionada, se utilizó una herramienta desarrollada especialmente para éste propósito y que es llamada “Cliente CopperCore”. Esta herramienta permite ajustar el valor de una propiedad IMS LD de una instancia del proceso al valor deseado. Para el caso del ejemplo ilustrativo, la *Figura D.17* muestra el momento en que “Cliente CopperCore” ajusta el valor de la propiedad IMS LD *P-User* correspondiente al participante *p1*, al valor *p1*.

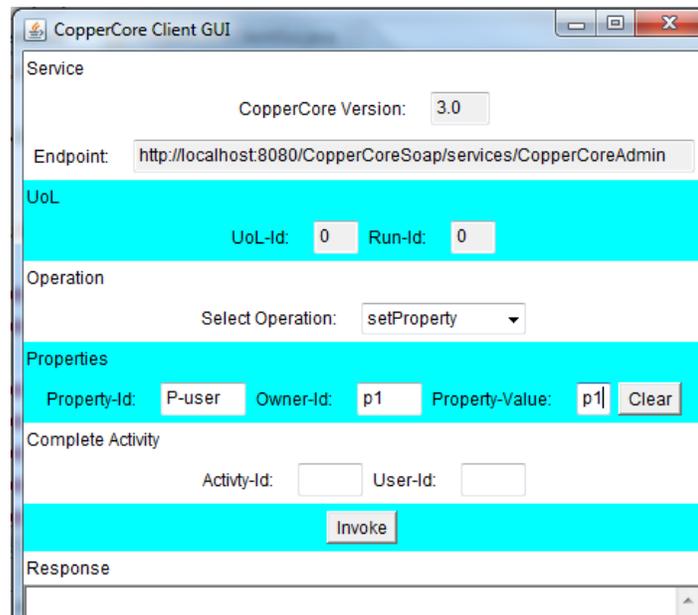


Figura D.17: Cliente del motor CopperCore usado por el administrador del sistema para asignar valores a propiedades de los participantes (*CopperCoreClientGui.java*).

Inicialización de la Instancia del Proceso del Flujo de Trabajo Sincronizado

Similarmente, también se requiere inicializar variables de la instancia del proceso del flujo de trabajo sincronizado. Esta tarea también es realizada por el administrador del sistema utilizando una herramienta que monitorea las instancias creadas en el entorno de ejecución de flujo de trabajo. El objetivo es inicializar el valor de variables globales de flujo de trabajo que representan a la instancia del proceso del flujo de aprendizaje sincronizado (requerida para la coordinación). Para el caso de la infraestructura tecnológica seleccionada, se utilizó “Cliente Bonita”. Como se indicó con anterioridad, esta herramienta mostrada en la *Figura D.12*, no solo permite monitorear instancias de procesos de flujo de trabajo existentes en el sistema BOS, sino que también permite modificar el valor de variables de flujo de trabajo de cualquiera de las instancias existentes. Para el caso del ejemplo ilustrativo, la *Tabla D.5* muestra el ajuste de variables de flujo de trabajo que debe llevar a cabo el administrador del sistema.

Variable de Flujo de Trabajo	Valor
<i>endPoint</i>	<i>http://localhost:8080/CopperCoreSoap/services/CopperCoreAdmin</i>
<i>UoIID</i>	<i>0</i>
<i>RunID</i>	<i>0</i>

Tabla D.5: Inicialización de variables del proceso del flujo de trabajo sincronizado que representan la instancia del proceso del flujo de aprendizaje con la que se van a comunicar.

Puesta en marcha

Finalmente, el último paso es la puesta en marcha por parte de los usuarios participantes. Aquí, los participantes utilizan una aplicación para navegar en el flujo de aprendizaje (donde se describen las actividades de aprendizaje, los contenidos educativos, etc.) y formularios de flujo de trabajo para interactuar con el sistema e indicar la terminación de herramientas. A continuación, se ejemplifica la puesta en marcha de una UoLeadFlow4LD para el caso del ejemplo ilustrativo.

El Flujo de Aprendizaje

La puesta en marcha de la situación de aprendizaje correspondiente al ejemplo ilustrativo inicia cuando el sistema presenta automáticamente a cada uno de los dos participantes (*p1* y *p2*) la interfaz del flujo de aprendizaje que se muestra en la *Figura D.18*. Esta interfaz se compone de tres marcos o ventanas. En la parte superior izquierda se encuentra la venta del flujo de aprendizaje, que contiene la secuencia de actividades de aprendizaje correspondientes al rol activo. En la parte inferior izquierda se encuentra la ventana de contexto, que contiene los recursos y servicios que dan soporte a la actividad de aprendizaje. Típicamente consiste de enlaces a contenidos educativos y al flujo de datos (inicio de la secuencia de herramientas correspondiente a la actividad de aprendizaje). Por último, está la ventana principal que despliega el detalle del ítem seleccionado en cualquiera de las dos ventanas anteriores. Por ejemplo, la ventana principal de la *Figura D.18* despliega la descripción de la actividad de edición que ha sido seleccionada en la ventana del flujo de aprendizaje.



Figura D.18: Descripción de la interfaz del flujo de aprendizaje presentada por el sistema a cada uno de los participantes de la actividad de aprendizaje.

Bloqueo del Flujo de Aprendizaje

En este momento, cuando alguno de los participantes intenta continuar a la siguiente actividad de aprendizaje (la actividad de revisión), ninguna actividad es mostrada al participante (ver *Figura D.19*), ya que el flujo de aprendizaje se encuentra bloqueado. Esto se debe a que en la actividad de edición existe un flujo de datos que debe ser iniciado, realizado y terminado por ambos participantes antes de que estos puedan continuar con la siguiente actividad de aprendizaje.

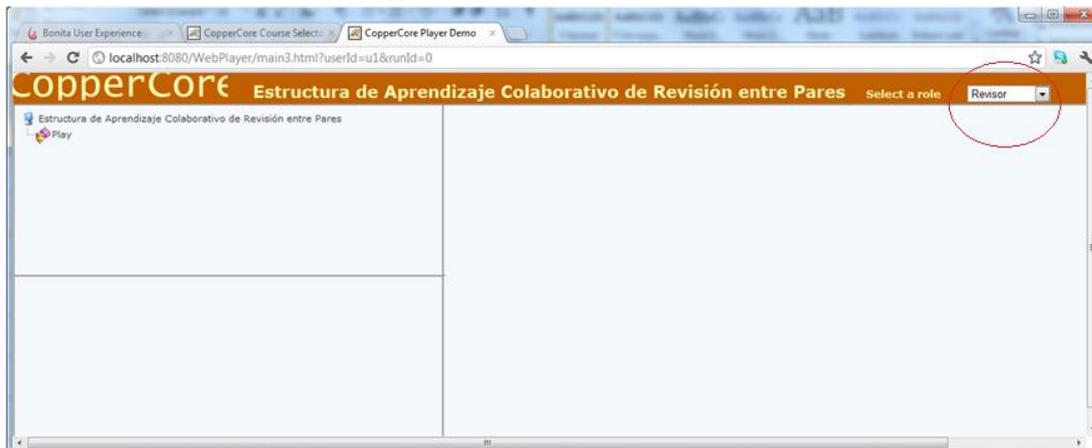


Figura D.19: Momento en que el participante p1 intenta acceder (sin éxito) a la actividad de revisión (en el rol revisor), cuando aún ninguno de los participantes han terminado el flujo de datos de la actividad de edición.

Invocación Automática de Herramientas

La *Figura D.20* muestra el momento en que el participante *p1* da inicio el flujo de datos de la actividad de edición. Es en este momento, se entrega el control al flujo de trabajo, quien lanza automáticamente una instancia de la herramienta de edición (una pizarra individual de mapas conceptuales) y la presenta al participante *p1*, como se muestra en la *Figura D.21*. Similarmente, una instancia de la pizarra individual es lanzada automáticamente por el sistema y presentada al participante *p2*, una vez de que éste da inicio al flujo de datos desde la actividad de aprendizaje.



Figura D.20: Momento en que el participante p1 inicia el flujo de datos desde la actividad edición, correspondiente a la situación de aprendizaje 1.1.

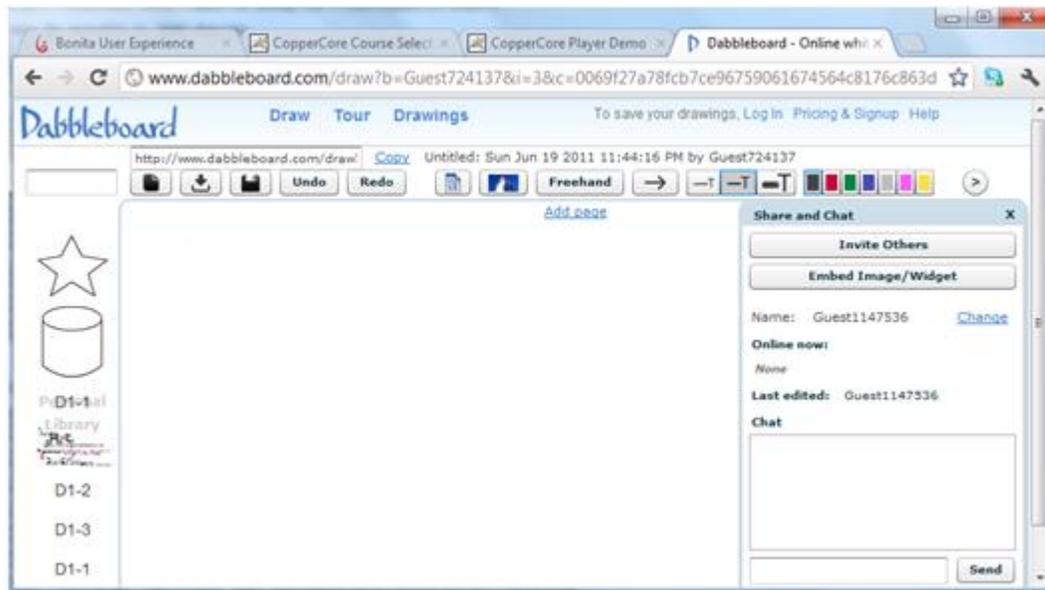


Figura D.21: Instancia de la herramienta de edición de mapas conceptuales (pizarra individual) lanzada automáticamente por el sistema y presentada al participante p1 inmediatamente después de éste da inicio el flujo de datos de la actividad de edición.

Finalización del Flujo de Datos

Una vez lanzada la herramienta de edición de mapas conceptuales, el sistema le permite al usuario usarla y terminarla. Cuando un participante termina una herramienta, el sistema lanza la siguiente herramienta (si existe una secuencia de herramientas), o bien se termina el flujo de datos de la actividad de aprendizaje. Esta funcionalidad se ilustra en el formulario de la *Figura D.22*, que el sistema presenta automáticamente a cada uno de los participantes. A través del formulario mostrado en la *Figura D.22*, el participante puede decidir terminar la herramienta (seleccionando el *checkbox* que aparece en el formulario de la *Figura D.22*, y oprimir después el botón de continuar).

Note que para el ejemplo ilustrativo, el flujo de datos en la actividad de edición se compone solamente de dos instancias simultáneas de la pizarra individual (una para cada participante) y no existen más herramientas en la secuencia. Entonces, cuando ambos participantes terminan sus respectivas herramientas de edición, en ese momento termina el flujo de datos de la actividad de edición y el flujo de aprendizaje queda liberado para continuar con la siguiente actividad de aprendizaje.

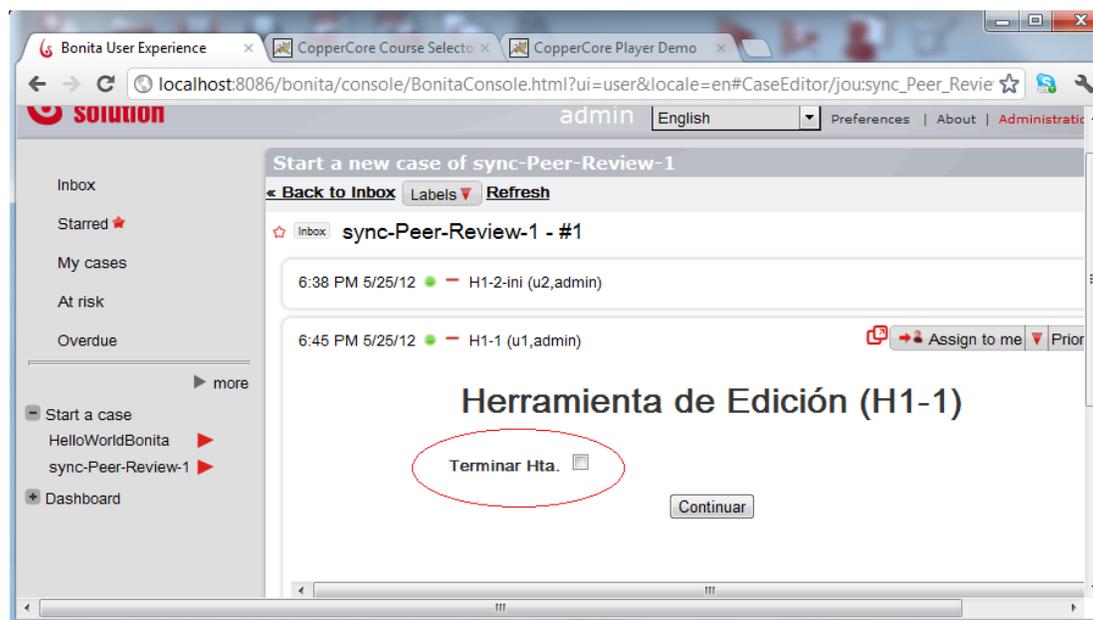


Figura D.22: Formulario usado por el sistema y entregado automáticamente a cada participante al lanzar una herramienta. A través de este formulario, el participante puede decidir si continúa usando la herramienta, o si decide terminarla para continuar con el flujo de datos.

Bloqueo del Flujo de Datos

Un bloqueo similar al descrito para el flujo de aprendizaje ocurre también para el flujo de datos. Una vez que ambos participantes han finalizado sus herramientas de edición, entonces el flujo de datos permanece bloqueado hasta que alguno de los participantes lo reinicie desde la siguiente actividad de aprendizaje: la actividad de revisión. La *Figura D.23* muestra el momento en que el participante *p1* intenta, sin éxito, reiniciar el flujo de datos

desde el entorno de ejecución del flujo de trabajo, cuando aún no ha iniciado el flujo de datos desde la actividad de aprendizaje de revisión.

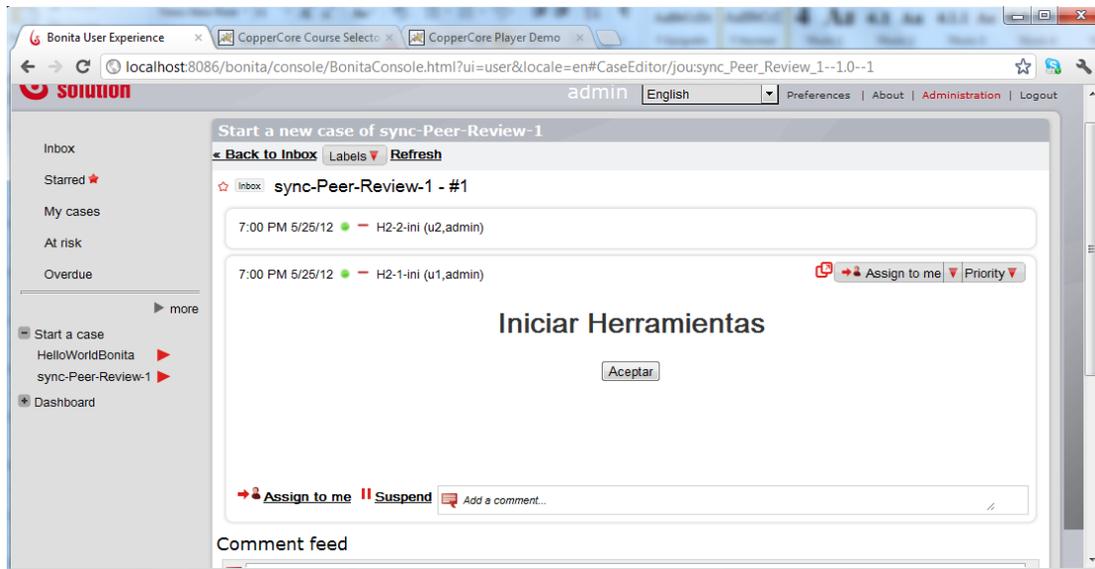


Figura D.23: Momento en que el participante $p1$ intenta iniciar (sin éxito) el flujo de datos desde el entorno de flujo de trabajo, sin haberlo iniciado antes desde la actividad de revisión.

Automatización de los Datos de Entrada y Salida de las Herramientas

Una vez que los participantes inician el flujo de datos desde la actividad de revisión, el sistema, vía el motor de flujo de trabajo, lanza automáticamente las herramientas que soportan la actividad de revisión. Tal es el caso de la pizarra individual presentada al participante $p1$ y mostrada en la *Figura D.24a*, pero que además está inicializada con el mapa conceptual del participante $p2$ creado en la actividad anterior. Similarmente, la *Figura D.24b* muestra otra instancia de la pizarra individual que el sistema entrega automáticamente al participante $p2$, pero que está inicializada con el mapa conceptual del participante $p1$ creado en la actividad anterior.

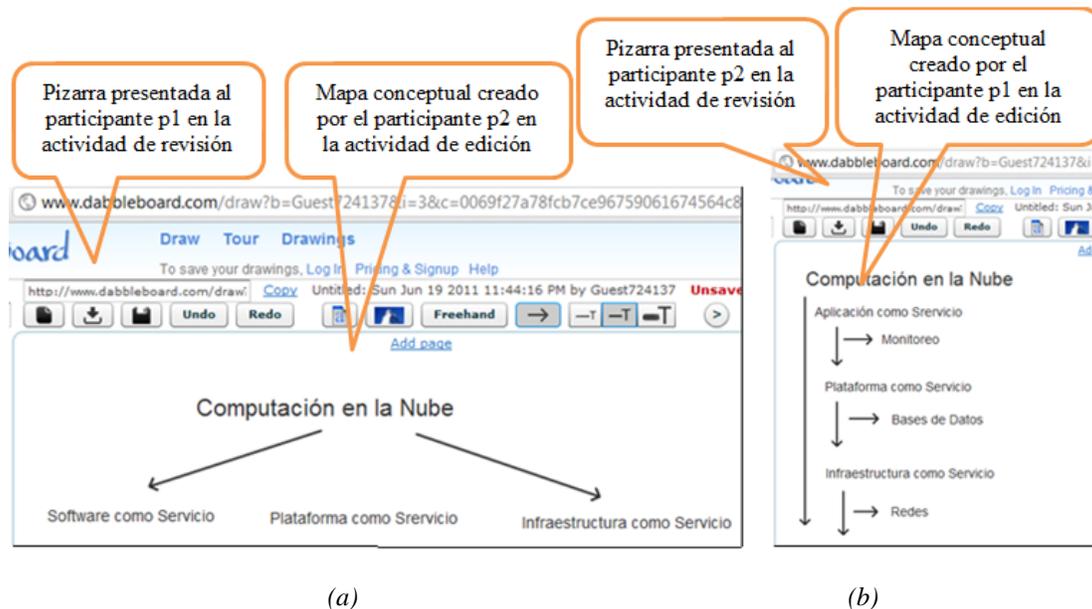


Figura D.24: Instancias de la pizarra individual entregadas automáticamente por el sistema a (a) el participante p1, inicializada con el mapa conceptual del participante p2 e (b) el participante p2, inicializada con el mapa conceptual del participante p1.

D.4 Despliegue del motor de ejecución del flujo de aprendizaje

Para desplegar el motor de ejecución de flujo de aprendizaje (CopperCore), el primer paso es instalar el servidor de aplicaciones (Tomcat versión 6) y cambiar la configuración de los puertos 8080 y 8009 que usa por defecto Tomcat, ya que ambos puertos serán usados por el motor de ejecución de flujo de aprendizaje (CooperCore). Por tanto, se debe ir al fichero `server.xml` (localizado en el directorio `<tomcat_home>/conf/`) y modificar los puertos como se muestra en la *Tabla D.6*.

#Puerto (valor inicial)	#Puerto (valor modificado)
8080	8086
8009	8019

Tabla D.6: Modificación de puertos en el fichero `<tomcat_home>/conf/server.xml`.

Una vez modificados los puertos de Tomcat, el siguiente paso es iniciar el servidor de aplicaciones directamente desde el fichero: `<tomcat_home>\bin\startup.bat`, como se ilustra en la *Figura D.25*.

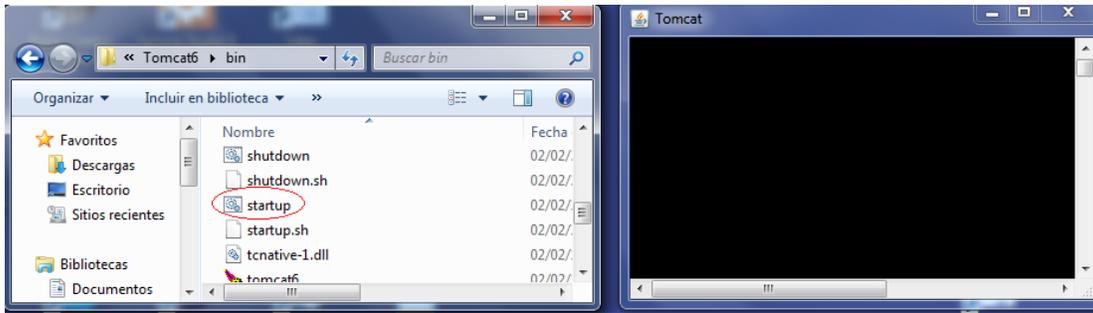


Figura D.25: Iniciación del servidor de aplicaciones Tomcat.

A continuación se instala CopperCore en el directorio: <coppercore_home>, y se despliega directamente desde el fichero: <coppercore_home>\coppercore.bat, como se ilustra en la Figura D.26. Entonces, el motor de ejecución de flujo de aprendizaje está listo para instanciar unidades de aprendizaje (UoLs).

```

JBoss Application Server Running CopperCore
state: INIT_WAITING_DEPLOYER
watch: file:/C:/ccrt3.0/jboss-3.2.6/server/default/deploy/coppercore.ear_origi
nal
lastDeployed: 1333155624964
lastModified: 1333155624964
mbeans:

org.jboss.deployment.DeploymentInfo02fdeef1f < url=file:/C:/ccrt3.0/jboss-3.2.6/
server/default/deploy/web.xml >
  deployer: null
  status: null
  state: INIT_WAITING_DEPLOYER
  watch: file:/C:/ccrt3.0/jboss-3.2.6/server/default/deploy/web.xml
  lastDeployed: 1333155625012
  lastModified: 1333155625012
  mbeans:

19:00:25,218 INFO [Http11Protocol] Arrancando Coyote HTTP/1.1 en puerto http-0.
0.0.0-8080
19:00:25,424 INFO [ChannelSocket] JK2: ajp13 listening on /0.0.0.0:8080
19:00:25,424 INFO [JkMain] Jk running ID=0 time=0/32 config=null
19:00:25,455 INFO [Server] JBoss (MX MicroKernel) [3.2.6 <build: CUSTag=JBoss_3
2.6 date=200410140106>] Started in 24s:571ms

```

Figura D.26: Despliegue del motor de flujo de aprendizaje CopperCore.

D.5 Despliegue del motor de ejecución de flujo de trabajo

Para desplegar el motor de ejecución del de flujo de trabajo (Bonita), el primer paso es instalar Bonita *Open Solution* (BOS). Desde la herramienta de autoría de flujo de trabajo: Bonita *Studio*, se debe exportar el entorno de ejecución de Bonita (Bonita *runtime*), de la siguiente manera: dentro del entorno de Bonita *Studio*, ir a *Process Advanced Export*; seleccionar “yes” a ambas preguntas; seleccionar *userXP*; seleccionar *Export Runtime*, opción *Full*, *Application WAR* de *UserXP* y *Rest API*. Además, seleccionar exportación “*Light*”, así como la ubicación del directorio donde serán guardados los ficheros exportados (<*export_folder*>), como se ilustra en la Figura D.27.

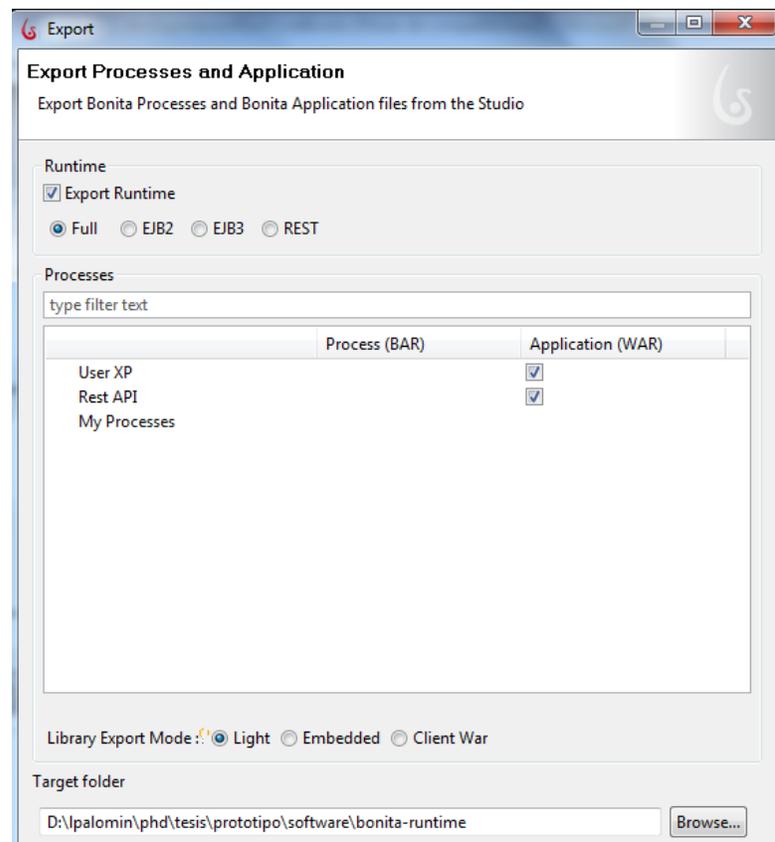


Figura D.27: Exportación del entorno de ejecución de Bonita Open Solution, desde Bonita Studio.

La Figura D.28 muestra el directorio de exportación (<export_folder>) del entorno de ejecución de Bonita, una vez que éste ha sido exportado desde Bonita Studio.

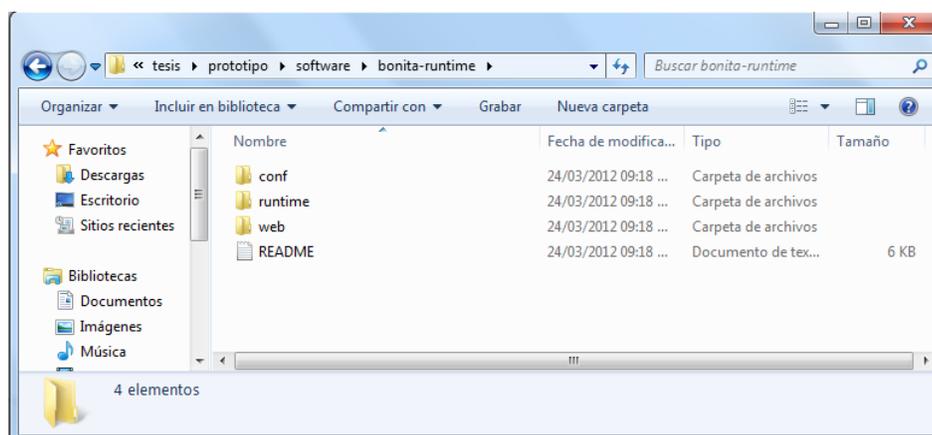


Figura D.28: Directorio de exportación <export_folder> de Bonita runtime.

Una vez exportado el entorno de ejecución de Bonita, el siguiente paso es desplegarlo. Para este propósito, se puede consultar el procedimiento en la siguiente dirección URL:

http://www.bonitasoft.org/wiki/doku.php?id=install_bos5.4_in_tomcat. Un resumen de dicho procedimiento, se muestra en la *Figura D.29*.

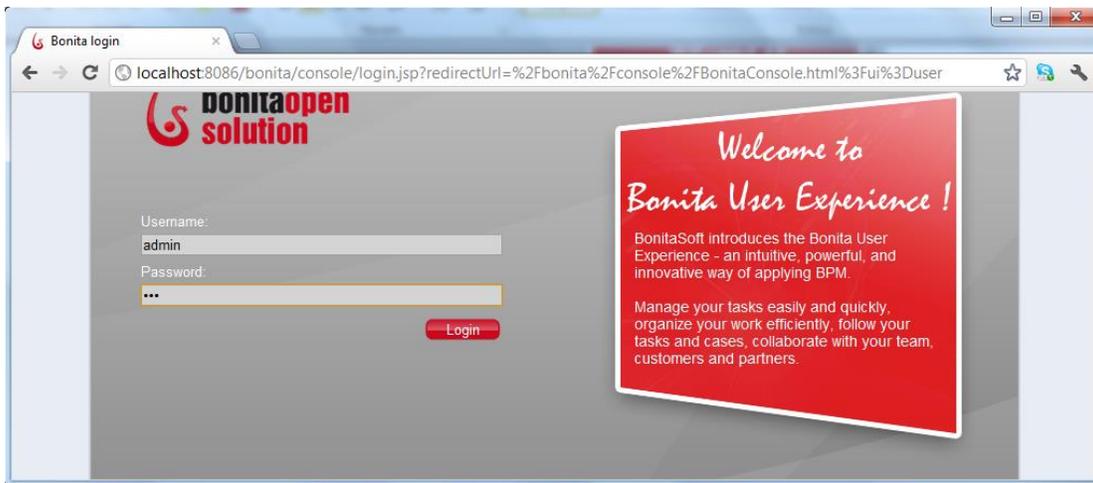
1. Copiar <export_folder>\web\bonita.war al directorio: <tomcat_home>\webapps
2. Copiar <export_folder>\web\bonita.war al directorio: <tomcat_home>\webapps
3. Crear el directorio: <tomcat_home>\lib\bonita
4. Copiar todos los ficheros *.jar de: <export_folder>\web\lib\ al directorio: <tomcat_home>\lib\bonita
5. Ir al directorio <export_folder>\conf y copiar los directorios bonita y external al directorio: <tomcat_home>
6. Abrir el fichero: <export_folder>\runtime\war\bonita-server-rest.war y borrar todos los ficheros *.jar del subdirectorio: WEB-INF/lib. Después, copiar este fichero bonita-server-rest.war en el directorio <tomcat_home>\webapps
7. Modificar el fichero <tomcat_home>\conf\catalina.properties para añadir a la propiedad common.loader:,{ catalina.home }\lib\bonita/*.jar
Para obtener:
common.loader=\${ catalina.base }\lib,{ catalina.base }\lib/*.jar,{ catalina.home }\lib,{ catalina.home }\lib/*.jar,{ catalina.home }\lib\bonita/*.jar
8. Crear un fichero setenv.bat (Windows) en <tomcat_home>\bin
Ejemplo de un fichero setenv.bat:


```
@echo on
rem Sets some variables
set BONITA_HOME="-DBONITA_HOME=%CATALINA_HOME%\bonita"
set LOG_OPTS="-Djava.util.logging.config.file=%CATALINA_HOME%\external\logging\logging.properties"
set SECURITY_OPTS="-Djava.security.auth.login.config=%CATALINA_HOME%\external\security\jaas-tomcat.cfg"
set MEMORY_OPTS="-Xshare:auto -Xms512m -Xmx1024m -XX:MaxPermSize=256m -XX:+HeapDumpOnOutOfMemoryError"
set CATALINA_OPTS=%JAVA_OPTS% %LOG_OPTS% %SECURITY_OPTS% %BONITA_HOME% %MEMORY_OPTS%
```

Figura D.29: Procedimiento para desplegar el entorno de ejecución de Bonita.

Autenticación del Administrador del Sistema

Una vez desplegando el entorno de ejecución de Bonita, el siguiente paso es para el administrador del sistema quien debe autenticarse (ver *Figura D.30*), usando la aplicación Web llamada Bonita *User Xperience (XP)* (ver *Figura D.31*). Para ello, el administrador puede dirigirse a la URL: <http://localhost:8086/bonita/>, y autenticarse con la cuenta y contraseña creadas por defecto y que se muestran en la *Tabla D.7*.



Figur D.30: Autenticación del administrador del sistema en la aplicación Web del entorno de ejecución de Bonita.

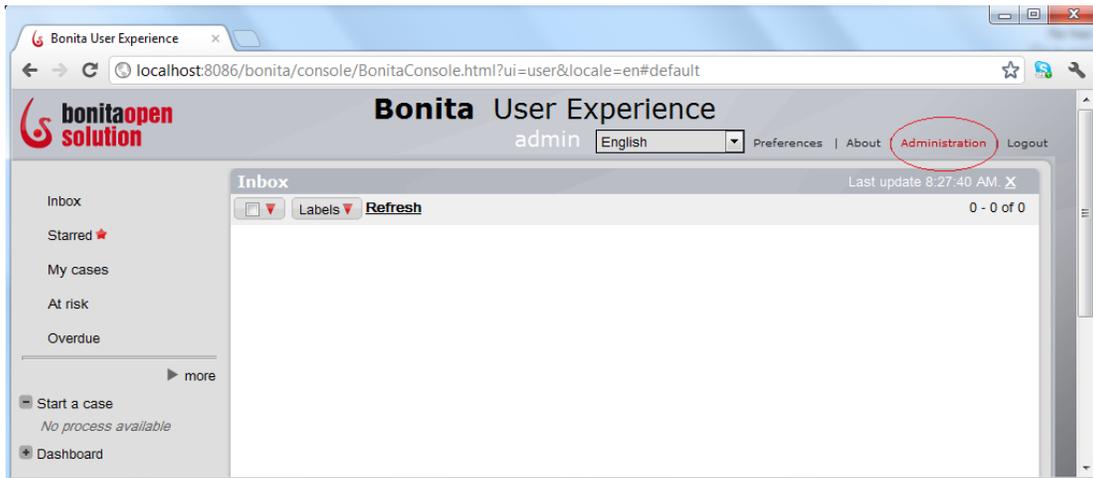


Figura D.31: Vista de participante de la aplicación Web Bonita User XP.

	Aplicación Web	Servidor REST
Cuenta	admin	Restuser
Contraseña	bpm	Restbpm

Tabla D.7: Cuentas y contraseñas creadas por defecto en el entorno de ejecución de Bonita.

D.6 Adaptador del motor de flujo de aprendizaje

De acuerdo al método propuesto, estas aplicaciones son usadas con varios propósitos: para inicializar propiedades IMS LD de los participantes y para modificar el valor de propiedades IMS LD de terminación de actividad. La primera es una tarea que es realizada por el administrador del sistema en tiempo de preparación de la puesta en marcha de una UoLeadFlow4LD, mientras que la segunda es una tarea que debe ser realizada

automáticamente por el componente analizador LeadFlow4LD cuando genera el documento del de flujo de trabajo sincronizado.

En la *Tabla D.8* se listan las aplicaciones de invocación del motor del flujo de aprendizaje (CopperCore), cuya implementación está incluida en el disco anexo en la ruta: `.\prototipo\software\apps\`. La *Figura D.32* muestra el cliente (CopperCoreClientGui.java) usado por el administrador del sistema para inicializar el valor de propiedades locales personales IMS LD correspondientes a cada participante.

Nombre de la Aplicación	Descripción
CopperCoreClientSetProperty.java	Usada por conectores de flujo de trabajo para modificar el valor de propiedades locales personales IMS LD (de fin de actividad) al valor verdadero.
CopperCoreClientGui.java	Usada por el administrador del sistema para asignar el valor inicial de propiedades locales personales IMS LD al valor de cada participante.

Tabla D.8: Adaptadores del motor de ejecución de flujo de aprendizaje (CopperCore).

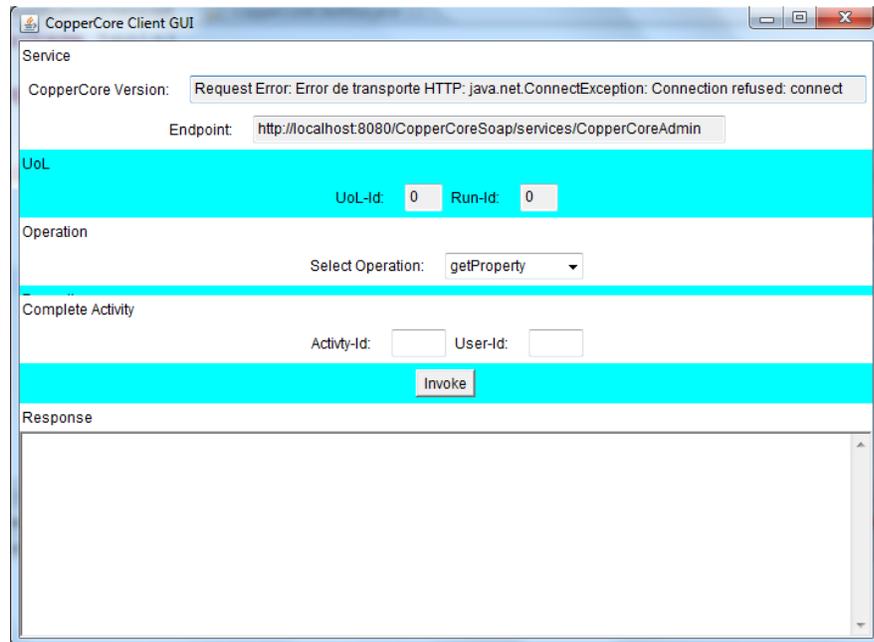


Figura D.32: Cliente del motor CopperCore usado por el administrador del sistema para asignar valores a propiedades de los participantes (CopperCoreClientGui.java).

En la *Figura D.33* se ilustra la configuración de requerida por Eclipse para compilar la aplicación cliente: CopperCoreClient.java. Las librerías necesarias son incluidas en el disco anexo en la ruta: `.\prototipo\software\lib\`.

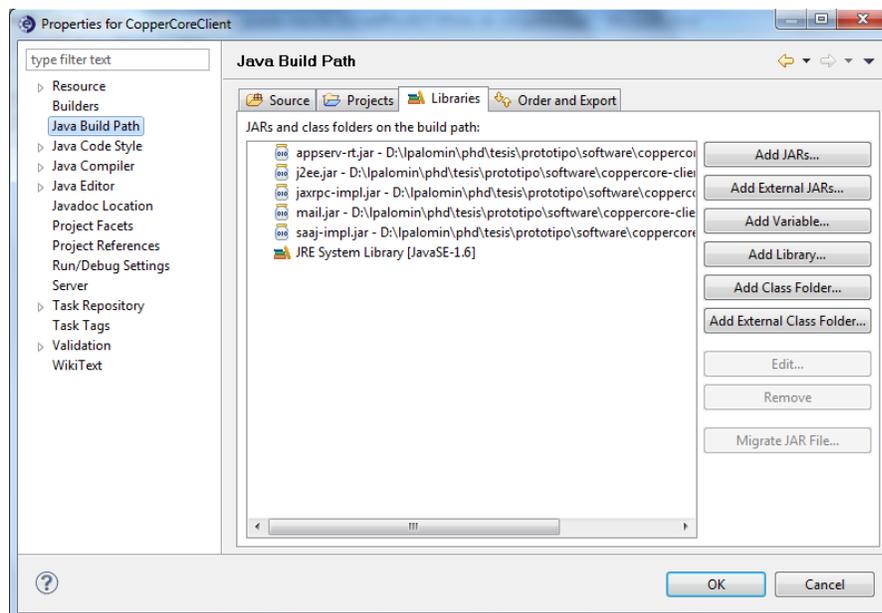


Figura D.33: Configuración del Java Build Path de Eclipse usado para compilar el cliente: *CopperCoreClient.java*.

Conector Java del Cliente CopperCore

Con el objetivo delegar el control al motor del flujo de aprendizaje (CopperCore), el componente coordinador de LeadFlow4LD agrega automáticamente conectores Java a las actividades de fin de herramientas. Dichos conectores son recursos de invocación del motor CopperCore cuyo propósito es modificar el valor de propiedades IMS LD que representan la terminación de una actividad de aprendizaje.

La Figura D.34 muestra una vista de la plantilla generada por Bonita Studio del conector Java usado para este propósito. La aplicación usada como para implementar dicho conector se llama *CopperCoreSetPropertyClient.java* y es incluido en el disco anexo en la ruta: `.\prototipo\software\apps\.`

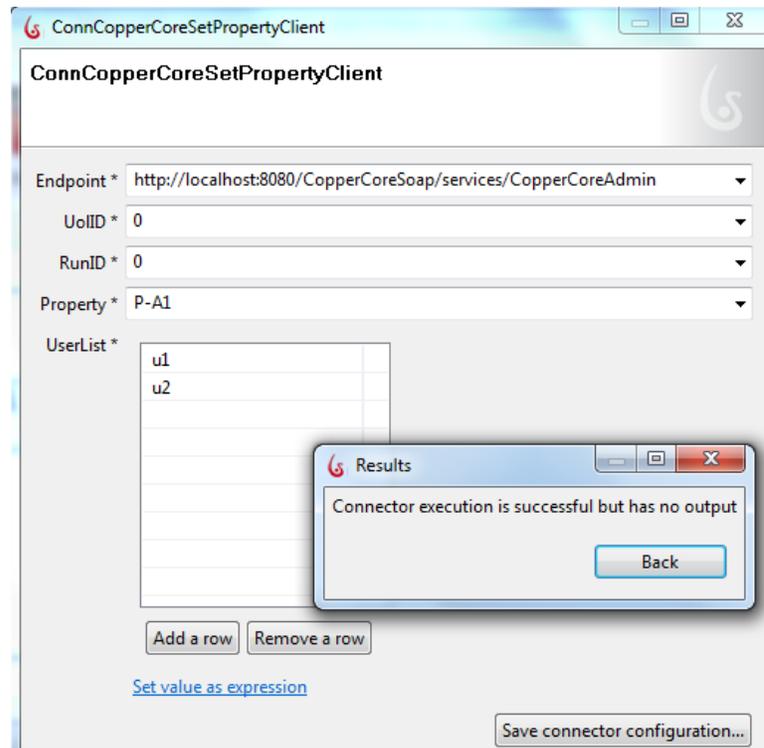


Figura D.34: Plantilla generada por Bonita Studio del conector Java usado para invocar el el motor CopperCore con el propósito de modificar el valor de propiedades IMS LD que representan el fin de una actividad de aprendizaje.

D.7 Adaptador del motor de flujo de trabajo

De acuerdo con el método propuesto, dicha aplicación es usada con dos propósitos: para recuperar los identificadores de instancias de procesos de flujo de trabajo (casos) y para inicializar variables de flujo de trabajo. La primera es una tarea que es realizada por el administrador del sistema en tiempo de preparación de la puesta en marcha de una UoLeadFlow4LD, mientras que la segunda es una tarea que debe ser realizada automáticamente por el componente analizador LeadFlow4LD cuando genera el documento del flujo de aprendizaje sincronizado.

Para crear instancias de procesos de flujo de trabajo (casos) en el entorno de ejecución de Bonita, el administrador del sistema puede utilizar la aplicación cliente *User XP* que viene integrado con el entorno de ejecución, como se ilustra en la *Figura D.35*.



Figura D.35: Cliente de Bonita User XP usado para monitorear instancias de procesos de flujo de trabajo (casos).

Sin embargo, para recuperar los identificadores de instancias de procesos y para inicializar variables de flujo de trabajo, el desarrollador del sistema utiliza la aplicación cliente de Bonita que el desarrollador del sistema ha creado para este propósito. La Figura D.36 ilustra dicho cliente (BonitaClientGui.java), mientras que su implementación está incluida en el disco anexo en la ruta: `\prototipo\software\apps\`.

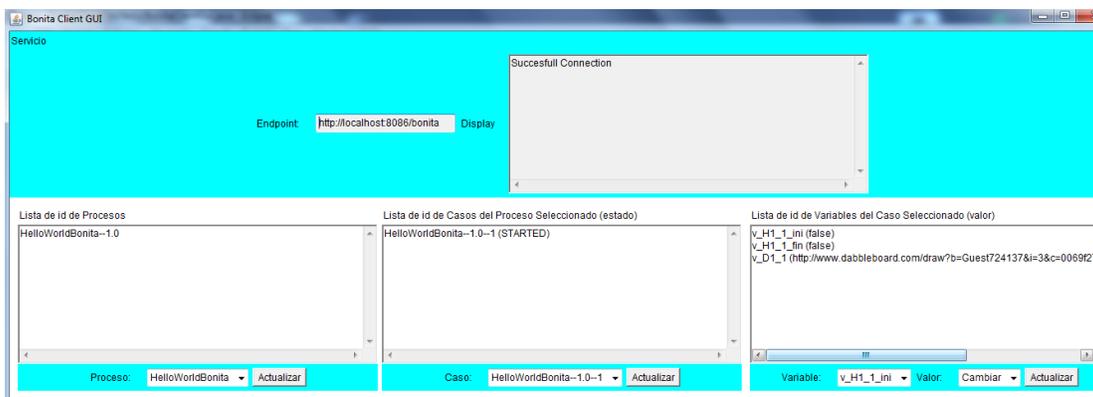


Figura D.36: Cliente de Bonita usado para recuperar instancias de procesos (casos) y para inicializar variables de flujo de trabajo.

Para la compilación y ejecución de la aplicación, es necesario configurar lo siguiente:

- Agregar el fichero “jaas-tomcat.cfg” localizado en el directorio: `<export_folder>\conf\external\security` al directorio src de Eclipse, como se ilustra en la Figura D.37.

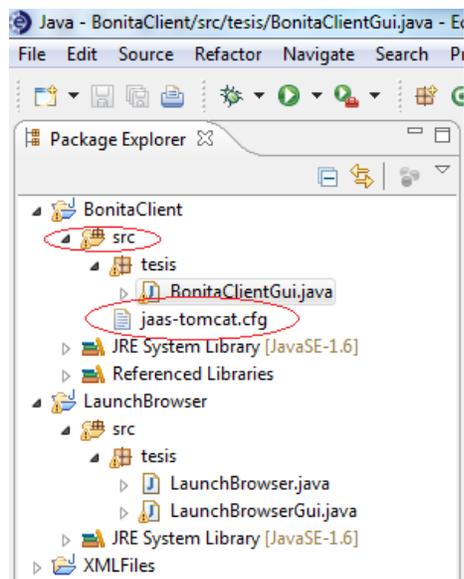


Figura D.37: Fichero `jaas-tomcat.cfg` agregado al directorio `src` de Eclipse.

- Agregar al *Path* de compilación de Eclipse todas las bibliotecas (ficheros `.jar`) localizadas en los directorios *client* y *rest* del *runtime* exportado de Bonita con la opción *Full*, como se ilustra en la Figura D.38:
 - `<export_folder>\runtime\lib\client`
 - `<export_folder>\runtime\lib\rest`

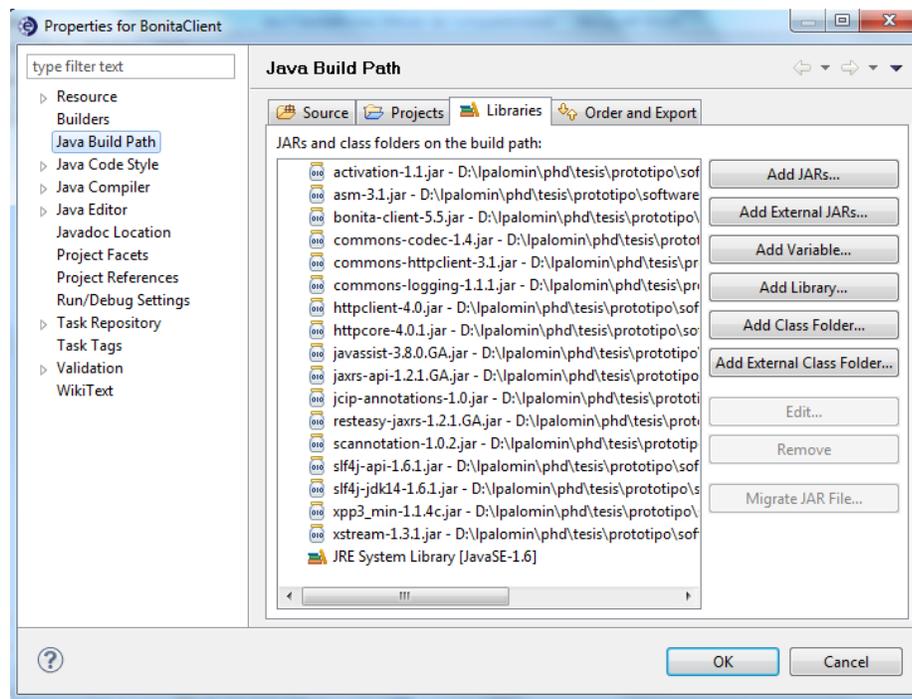


Figura D.38: Configuración del Java Build Path de Eclipse usado para compilar el cliente: *BonitaClientGui.java*.

- Por último, en la configuración de ejecución (*Run configuration*), en la pestaña de argumentos (*Arguments*), en el directorio de trabajo (*Working directory*), la opción seleccionada es “*Other*” (no *Default*), con el texto: $\${workspace_loc:nombre_del_proyecto_en_Eclipse/bin}$, como se ilustra en la Figura D.39.

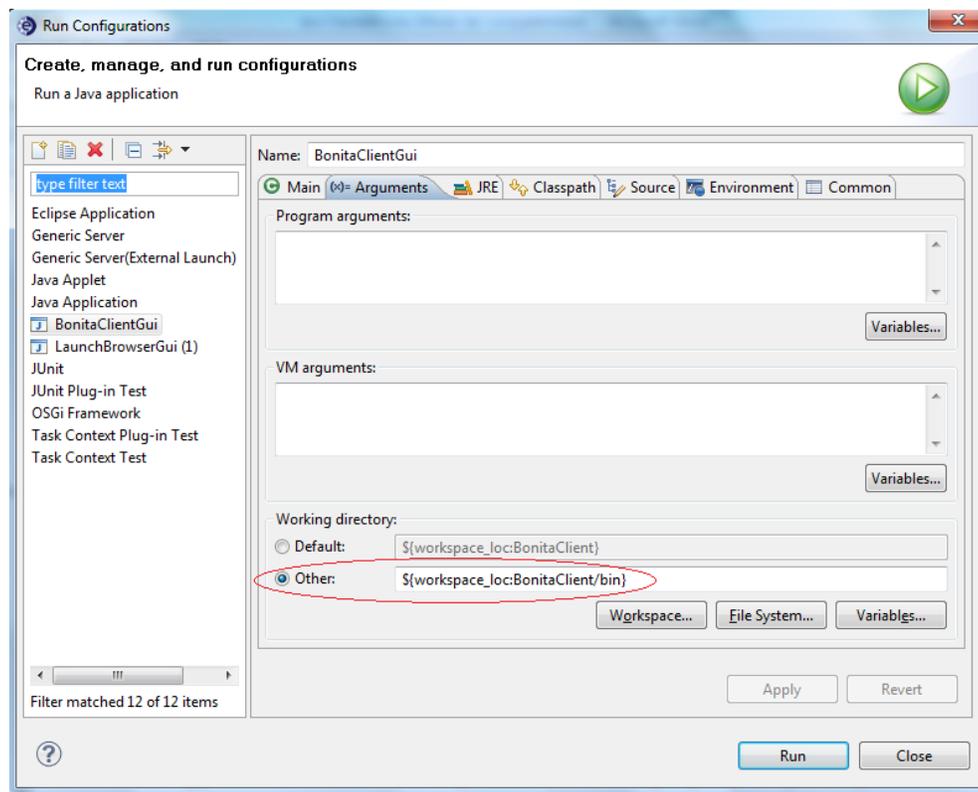


Figura D.39: Configuración de ejecución en Eclipse de la aplicación *BonitaClientGui.java*.

Recursos de Invocación del Motor de flujo de trabajo

Por último, con el objetivo delegar automáticamente el control al motor de ejecución de flujo de trabajo, se utilizan recursos de invocación que usan tecnología REST. Dichos recursos de invocación son agregados como recursos *imsldcontent* en el documento del flujo de aprendizaje sincronizado, el cual debe ser generado automáticamente por el componente analizador *LeadFlow4LD*.

El ejemplo ilustrado en la *Figura D.40*, tiene como objetivo modificar la variable *V-H1-1-ini* al valor verdadero, correspondiente a la instancia del proceso de flujo de trabajo identificada como: *HelloWorldBonita--1.0--1*.

```
<html>
<header>
<title>Set Variable </title>
</header>
<body>
<h3>Iniciar Herramientas</h3>
<form name="setV1False" action="http://localhost:8086/bonita-server-
rest/API/runtimeAPI/setProcessInstanceVariable/HelloWorldBonita--1.0--1" method="post">
<input type="hidden" name="variableId" value="v_H1_1_ini" />
<input type="hidden" name="variableValue" value="true"/>
<input type="hidden" name="options" value="user:admin"/>
<input type="submit" value="Aceptar" />
```

```

</form>
</body>
</html>

```

Figura D.40: Recurso de invocación (usando tecnología REST) de Bonita (HelloWorldBonitaRESTClient.html).

En tiempo de ejecución, el recurso es llamado cuando el participante oprime el botón aceptar para iniciar su servicio de herramientas, como se ilustra en la *Figura D.41*.

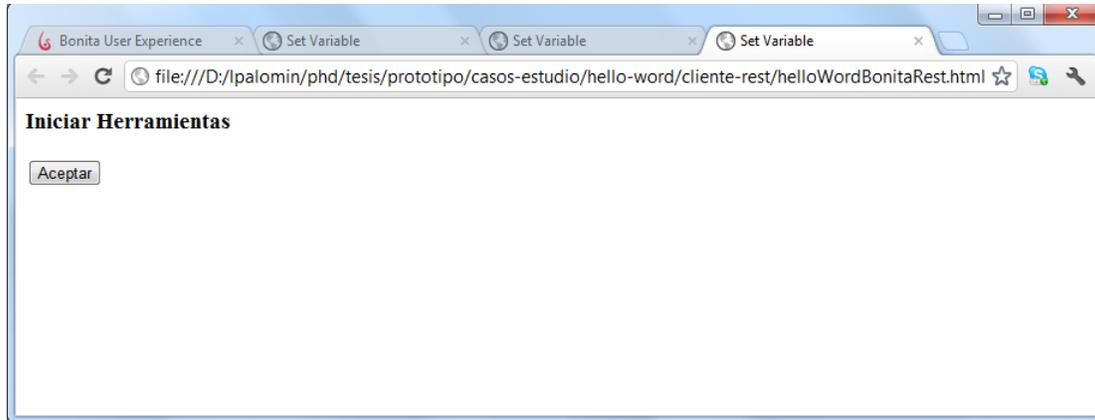


Figura D.41: Formulario usado para acceder al recurso de invocación de Bonita.

Entonces, el servidor REST solicitará la primera vez una petición de autenticación. Por tanto, se debe introducir una sola vez el siguiente participante y contraseña: participante (*login*): *restuser* y la contraseña (*password*): *restbpm*.

D.8 Adaptador de herramientas Web

Con el objetivo de lanzar navegadores Web que sean usados como clientes de herramientas en línea, Bonita hace uso de conectores Java (adaptadores) para este propósito. Dichos conectores o adaptadores son agregados automáticamente por el componente analizador LeadFlow4LD a las actividades de flujo de trabajo que representan invocaciones de herramientas (en el documento del de flujo de trabajo sincronizado). La *Figura D.42* muestra la plantilla generada por Bonita del conector Java que se usa para lanzar un navegador Web. La aplicación usada como base para implementar este conector se llama *LaunchBrowser.java* y es incluida en el disco anexo en la ruta: `.\prototipo\software\conectores\`.

```

1 package tesis;
2
3 import java.util.List;
4
5 public class ConnLaunchBrowser extends ProcessConnector {
6
7     // DO NOT REMOVE NOR RENAME THIS FIELD
8     private java.lang.String url;
9
10    @Override
11    protected void executeConnector() throws Exception {
12        // TODO Auto-generated method stub
13    }
14
15    @Override
16    protected List<ConnectorError> validateValues() {
17        // TODO Auto-generated method stub
18        return null;
19    }
20
21    /**
22     * Setter for input argument 'url'
23     * DO NOT REMOVE NOR RENAME THIS SETTER, unless you also change the related entry in the XML descriptor file
24     */
25    public void setUrl(java.lang.String url) {
26        this.url = url;
27    }
28 }

```

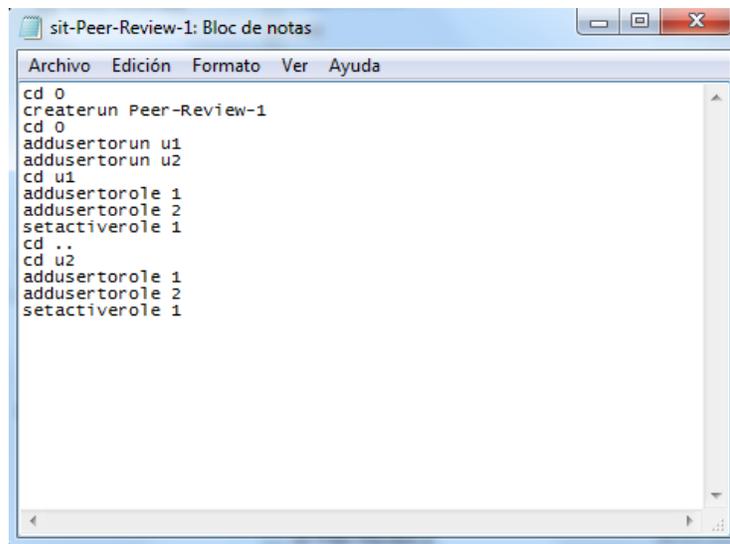
Figura D.42: Plantilla generada por Bonita Studio para implementar el conector Java usado para invocar herramientas en línea.

D.9 Publicación e instanciación del documento del flujo de aprendizaje sincronizado

Una vez que ha sido desplegado el motor del de flujo de trabajo (CopperCore), se utiliza la herramienta de validación y publicación de una UoL llamada *Publisher*, la cual es un *servlet* que proporciona una interfaz para la validación y publicación de UoLs que son conformes a IMS LD. Se puede acceder a dicha herramienta en la siguiente dirección URL: <http://localhost:8080/Publisher/publication.html>.

Una vez validada y publicada una UoL, el siguiente paso es instanciarla. Con este propósito, se utiliza el cliente CopperCore llamado “*clic*”, el cual es una interfaz de línea de comandos que permite la ejecución de varios tipos de tareas administrativas, incluyendo la creación de *runs* de una UoL, la creación de participantes, la asignación de participantes a *runs*, la asignación de participantes a roles, fijar el rol activo de cada participante, etc. De hecho, en esta tesis, se utiliza una versión modificada de esta herramienta, llamada “*clicc2*”, y que permite editar todos los comandos dentro de un fichero de texto. En la *Figura D.43*, se muestra un ejemplo de cómo editar este archivo. La primera línea indica que se activa la UoL cuyo identificador es 0; a continuación se indica que se crea una *run* de la UoL y se le llama “Peer-Review-1”; después se activa dicha *run* (cuyo identificador es cero); después se añaden a la *run* dos participantes (*p1* y *p2*); después, se activa el participante *p1* y se asigna a los roles editor y revisor, identificados como roles 1 y 2, respectivamente; finalmente, se activa el rol editor o rol 1 como el rol activo del participantep1; a continuación se hace lo propio para el participante *p2*. Para llamar la aplicación *clicc2.bat*, simplemente se invoca la aplicación seguida del nombre del archivo de texto. La aplicación clic modificada (*clicc2.bat*)

es incluida en el disco anexo en la ruta: `.\prototipo\software\motores\` dentro del fichero: `coppercore_ccrt_3.0-Eloy.zip`.



```
cd 0
createrun Peer-Review-1
cd 0
addusertorun u1
addusertorun u2
cd u1
addusertorole 1
addusertorole 2
setactiverole 1
cd ..
cd u2
addusertorole 1
addusertorole 2
setactiverole 1
```

Figura D.43: Archivo de texto usado para instanciar una UoL usando clicc2.

Inicialización de Propiedades IMS LD

El siguiente paso de la preparación de la puesta en marcha de una UoLeadFlow4LD es la inicialización de propiedades IMS LD de los participantes. Esta tarea es realizada por el administrador del sistema utilizando el cliente CopperCore ilustrado en la *Figura D.32*. Básicamente el administrador del sistema debe asignar a la propiedad local personal *P-user*, que representa a los participantes, el valor que le corresponde a cada participante, es decir, su propio identificador: *p1*, *p2*, etc.

Apéndice E Contenido del disco anexo

Toda la información (documentos y ficheros) relacionada a este documento de tesis es incluida en el disco anexo. En este apéndice se proporciona un índice general de la estructura del contenido de este disco.

E.1 Estructura del contenido

De la Tabla E.1 a la Tabla E.4 se describe la estructura de directorios del contenido del disco anexo.

Directorio	Descripción
\ejemplos	Incluye: - UoLeadFlow4LDs de cada uno de los ejemplos (revisión entre pares, convergencia del conocimiento y caso MOSAIC). - Versión de puesta en marcha (documentos sincronizados del flujo de aprendizaje y flujo de trabajo) de cada una de las situaciones de aprendizaje usadas en la evaluación de LeadFlow4LD.
\memoria	Incluye el documento de memoria de tesis.
\mosaic	Incluye: - Los cuestionarios del caso de estudio MOSAIC. - Los resultados del análisis - La UoL
\software	Incluye el software utilizado en la implementación del sistema prototipo para poner en marcha una UoLeadFlow4LD: -Herramientas -Motores -Librerías -Adaptadores de motores de flujo

Tabla E.1: Primer nivel de la estructura de directorios contenidos en el disco anexo.

Directorio	Descripción
\ejemplos\prep-puesta-en.marcha	Incluye los documentos sincronizados de cada una de las situaciones de aprendizaje: -Revisión entre pares: situaciones: 1.1, 1.2, 1.3 y 1.4 -Convergencia del conocimiento: situación 2.1 -Caso MOSAIC: situación 3.1
\ejemplos\UoLeadFlow4LD-IMS-CP	Incluye las UoLeadFlow4LD de cada uno de los ejemplos (revisión entre pares, convergencia del conocimiento y caso MOSAIC)

Tabla E.2: Estructura del directorio ejemplos.

Directorio	Descripción
\problema\Mosaic\cuestionarios	Incluye los cuestionarios aplicados a los estudiantes y sus respuestas sobre el caso de estudio MOSAIC.
\problema\Mosaic\resultados	Incluye los resultados del análisis del problema obtenidos del caso de estudio MOSAIC
\problema\Mosaic\uol	Incluye los resultados del análisis de la UoL correspondiente al caso MOSAIC.

Tabla E.3: Estructura del directorio mosaic

Directorio	Descripción
\software\apps	Incluye los adaptadores de los motores de flujo de aprendizaje y de flujo de trabajo.
\software\conectores	Incluye los conectores Java usados por Bonita Open Solution
\software\lib	Incluye las bibliotecas usadas para configurar los adaptadores.
\software\motores	Incluye el software para instalar los motores y clientes del flujo de aprendizaje (CopperCore y Clicc) y de flujo de trabajo (Bonita Studio y Bonita Open Solution)
\software\tools	Incluye las herramientas usadas para especificar los distintos documentos de LeadFlow4LD.

Tabla E.4: Estructura del directorio software

Referencias

Abiteboul, S., Bourhis, P., Vianu, V. (2012). Comparing workflow specification languages: a matter of views. *ACM Transactions on Database Systems (TODS)*, 37(2), 10.

Adrion, W. R. (1993). Research methodology in software engineering. Summary of the Dagstuhl Workshop on Future Directions in Software Engineering SIGSoft Software Engineering Notes. 18(1), 36-37.

Agostino, S. (2011). The use of a visual learning design representation to support the design process of teaching in higher education. *Australasian Journal of Educational Technology*, 27(6), 961-978.

Alario-Hoyos, C. (2012). GLUE!: An Architecture for the Integration of External Tools in Virtual Learning Environments, PhD Thesis, School of Telecommunication Engineering, University of Valladolid, Spain.

Alario-Hoyos, C., Bote-Lorenzo, M. L., Gómez-Sánchez, E., Asensio-Pérez, J. I., Vega-Gorgojo, G., Ruiz-Calleja, A. (2013). GLUE!: An Architecture for the Integration of External Tools in Virtual Learning Environments. *Computers & Education*, 600(1), 122-137.

Anderson, T. (2003). Modes of interaction in distance education. (I. M. Anderson, Ed.) *Handbook of distance education*, 129-144.

Andrews, T., Curbeta, F., Dholakia, H., Goland, Y., Klein, Y., Leymann, F., Liu, K., Roller, D., Smith, D., Thatte, S., Trickovic, I., Weerawarana, S. (2003). Business Process Execution Language for Web services, Version 1.1. BEA Systems, IBM Corp., Microsoft Corp., SAP AG, Siebel Systems.

Baeyens, T. (2004). The State of Workflow. Available at (last retrieved Feb. 2011): <http://www.theserverside.com/news/1365159/The-State-of-Workflow>

Baggetun, R., Barros, B., Fesakis, G., Girardin, F., Hoeksema, K., Hämäläinen, R., Miao, Y., Vantroys, T. (2004). A review of schemes for modelling scripts. Kaleidoscope Network of Excellence. Deliverable 23.2.1.

Barkley, E. F., Cross, K. P., Major, C. H. (2005). Collaborative Learning Techniques. John Wiley & Sons, Inc.

Bartels, N. (2003). Written Peer Response in L2 Writing. In *English Teaching Forum*, 41(1), 34-36.

- Barthelmess, P., Wainer, J. (1995). WorkFlow systems: a few definitions and a few suggestions. In Proceedings of conference on Organizational computing systems, 138-147. ACM.
- Beetham, H., Sharpe, R. (Eds.) (2013). Rethinking Pedagogy for a Digital Age: Designing for 21st Century Learning (1). Routledge, London.
- Berggren, A., Burgos, D., Fontana, J. M., Hinkelman, D., Hung, V., Hursh, A. Tielemans, G. (2005). Practical and pedagogical issues for teacher adoption of IMS learning design standards in Moodle LMS. Journal of Interactive Media in Education (1).
- Billington, J., Christensen, S., van Hee, K. E., Kindler, E., Kummer, O., Petrucci, L., Weber, M. (2003). The Petri net markup language: concepts, technology, and tools. In Applications and Theory of Petri Nets 2003, 483-505, Springer Berlin Heidelberg.
- Bohl, O., Scheuhase, J., Sengler, R., Winand, U. (2002). The sharable content object reference model (SCORM)-a critical review. In Computers in Education, 2002. Proceedings. International Conference on, 950-951, IEEE.
- Bordies, O., Villasclaras-Fernández, E. D., Dimitriadis, Y., Ruiz-Calleja, A. (2012). Reusability of data flow designs in complex CSCL scripts: Evaluation results from a case study. Proceedings of the 18th International Conference on Collaboration and Technology, CRIWG 2012.
- Bote-Lorenzo, M. (2005). Gridcole, un sistema maleable basado en servicios grid para el apoyo del aprendizaje colaborativo guiado. Valladolid: Tesis Doctoral, Universidad de Valladolid.
- Bote-Lorenzo, M. L., Gómez-Sánchez, E., Vega-Gorgojo, G., Dimitriadis, Y., Asensio-Pérez, J. I., Jorrín-Abellán, I. M. (2008). Gridcole: a tailorable grid service based system that supports scripted collaborative learning. Computers & Education, 51(1), 155-172.
- Botturi, L. (2003). E2ML-A modeling language for technology-dependent educational environments. EDMEDIA 2003. Honolulu, Hawaii, USA.
- Brasher, A., Conole, G., Cross, S., Weller, M., Clark, P., White, J. (2008). CompendiumLD: a tool for effective, efficient and creative learning design. Proceedings of the 2008 European LAMS Conference: Practical Benefits of Learning Design, 25-27. Cadiz, Spain.
- Britain, S. (2004). A Review of Learning Design: Concept, Specifications and Tools. A Report for the JISC E-learning Pedagogy Programme. Available at (last retrieved Feb. 2011): <http://www.jisc.ac.uk/media/documents/programmes/elearningpedagogy/learningdesigntoolsfinalreport.pdf>

Britain, S. (2007). Learning design systems: current and future developments. In H. Beetham, & R. Sharpe, *Rethinking Pedagogy for a Digital Age: Designing and delivering e-learning*. London: Routledge.

Brooks, F. P. (1987). No Silver Bullet. Essence and Accidents of Software Engineering. *IEEE computer*, 20(4), 10-19.

Brooks, F. P. (1988). Grasping reality through illusion—interactive graphics serving science. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems (CHI'88)*, 1-11.

Brouns, G. A. (2000). *Featuring Workflow Management. An overview of the distinctive features of Workflow Processes and their consequences for Workflow Management.*

Eindhoven University of Technology, Department of Mathematics and Computing Science, Eindhoven, The Netherlands.

Buendía, F. (2003). *Xedu: Un modelo para la especificación de las componentes estática y dinámica de una aplicación instructiva. Tesis Doctoral, Universidad Politécnica de Valencia. Valencia, España.*

Cabrera, F., Donoso, T., Marín, M. (1993). *Manual de formación pedagógica básica para formadores. Promociones y Publicaciones Universitarias PPU. Barcelona, España.*

Caeiro-Rodríguez, M. (2007). *Contribuciones a los Lenguajes de Modelado Educativo. Tesis Doctoral. Departamento de Ingeniería Telemática. Universidad de Vigo.*

Caeiro, M., Anido, L., Llamas, M. (2003). A Critical Analysis of IMS Learning Design. In *Proceedings of Computer Supported Collaborative Learning Conference (CSCL 2003)*, 363-367.

Caeiro-Rodríguez, M. (2008). PoEML: A Separation of Concerns proposal to Instructional Design. *Handbook of Visual Languages for Instructional Design: Theories and Practices*, 185-209.

Caeiro-Rodríguez, M., Llamas-Nistal, M., Anido-Rifón, L. (2006). The PoEML proposal to model services in educational modeling languages. In *Groupware: Design, Implementation, and Use, Lecture Notes in Computer Science*, 187-202, Springer Berlin Heidelberg.

Caeiro-Rodríguez, M., Anido-Rifón, L., Llamas-Nistal, M. (2010). Challenges in Educational Modelling: Expressiveness of IMS Learning Design. *Educational Technology & Society*, 13(4), 215-226.

Cameron, L. (2009). How learning design can illuminate teaching practice. In *The Future of Learning Design Conference*, paper 3, Wollongong, Australia: Available at (last retrieved Nov. 2013): <http://ro.uow.edu.au/fld/09/Program/3/>.

- Cesarini, M., Monga, M., Tedesco, R. (2004). Carrying on the e-learning process with a workflow management engine. In Proceedings of the 2004 ACM symposium on Applied computing, 940-945, ACM.
- Challco, G. C., Gerosa, M. A., Bittencourt, I. I., Isotani, S. (2014). Automated instructional design for CSCL: A hierarchical task network planning approach. *Expert Systems with Applications*, 41(8), 3777-3798.
- Chan, T. W., Roschelle, J., Hsi, S., Kinshuk, Sharples, M., Brown, T., Hoppe, U. (2006). One-to-one technology-enhanced learning: An opportunity for global research collaboration. *Research and Practice in Technology Enhanced Learning*, 1(01), 3-29.
- Cichocki, A., Helal, A., Rusinkiewicz, M., Woelk, D. (1998). *Workflow and Process Automation: Concepts and Technology*. Springer.
- Clark, J. (1999). XSL Transformations (XSLT). World Wide Web Consortium (W3C). URL <http://www.w3.org/TR/xslt>.
- Clark, J., Casanave, C., Kanaskie, K., Harvey, B., Smith, N., Yunker, J. Riemer, K. (2001). ebXML Business Process Specification Schema. Version 1.01. UN/CEFACT and OASIS.
- Collin, M., Beauvoir, P., Sharples, P. (2005). The Reload learning design tools. *Journal of Interactive Media in Education* (1).
- Conole, G. (2008). The role of mediating artefacts in learning design. In S. Bennett, & B. Harper, *Handbook of research on learning design and learning objects: Issues, applications and technologies* (108-208). New York: Information Science Reference.
- Conole, G. (2013). *Designing for learning in an open world*, Vol. 4, Springer.
- Dalziel, J. (2003). Implementing learning design: the Learning Activity Management System (LAMS). Proceedings of the ASCILITE 2003 Conference, Adelaide, Australia.
- Dalziel, J. (2009). Prospects for learning design research and LAMS. *Teaching English with Technology-Special Issue on LAMS and Learning Design*, 9(2), i-iv.
- Dalziel, J. (2012). Larnaca Declaration on Learning Design. Available at (last retrieved: Nov. 2013): <http://www.larnacadeclaration.org/>.
- Dalziel, J. R. (2006). Lessons from LAMS for IMS Learning Design. *Learning Design of Collaborative Learning Activities Languages, models and Tools*. The 6th IEEE International Conference on Advanced Learning Technologies (ICALT 2006), 1101-1102, Kerkrade, The Netherlands.

- Dan, Y., XinMeng, C. (2007). Supporting Collaborative Learning Activities with IMS LD. In *Advanced Communication Technology, The 9th International Conference*, Vol 1, 316-320, IEEE.
- Dayal, U., Hsu, M., Ladin, R. (2001). Business Process Coordination: State of the Art, Trends, and Open Issues. *Proceedings of the 27th Very Large Databases Conference (VLDB 2001)*, Vol. 1, 3-11, Rome, Italy.
- de la Fuente, L., Pardo, A., Delgado, C. (2007). Experiences with GRAIL: Learning Design support in .LRN. In *TENCompetence Open workshop on current research in IMS Learning Design and lifelong competence development infrastructures*.
- Dobozy, E. (2011). Typologies of learning design and the introduction of a "LD-Type 2" case example. *eLearning Papers*, 27(27), 1-11.
- Decker, G., Mendling, J. (2008). Instantiation semantics for process models. In *Proceedings of the 6th International Conference on Business Process Management, BPM 2008*, 164-179, Springer Berlin Heidelberg.
- Derntl, M., Neumann, S., Oberhuemer, P. (2011). Community support for authoring, sharing, and reusing instructional models: the Open Graphical Learning Modeller (OpenGLM). In *Proceedings of IEEE ICALT 2011*, 431-435, Athens, Georgia, USA.
- Derntl, M., Neumann, S., Griffiths, D., Oberhuemer, P. (2012). The conceptual Structure of IMS Learning Designs Does Not Impede Its Use for Authoring. In *IEEE Transactions on Learning Technologies*, IEEE Transactions on, 5(1), 74-86.
- Dillenbourg, P. (1999). What do you mean by collaborative learning. In I. P. (Ed.), *Collaborative-learning: cognitive and computational approaches* (1-19). Elsevier.
- Dillenbourg, P. (2002). Over-scripting CSCL: the risks of blending collaborative learning with instructional design. In *Three worlds of CSCL. Can we support CSCL?*, P.A, Kirscher (Ed.), 61-91, Heerlen, Open Universiteit Netherlands.
- Dillenbourg, P., Hong, F. (2008). The Mechanics of CSCL Macro Scripts. *International Journal of Computer-Supported Collaborative Learning*, 3(1), 5-23.
- Dillenbourg, P., Schneider, D. (1995). Collaborative learning and the Internet. In *Proceedings of the 4th International Conference on Computer Assisted Instruction, ICCAI*.
- Dillenbourg, P., Jermann, P. (2007). Designing integrative scripts. In Fisher, F., Kollar, I., Mandl, H., Haake, J., *Scripting computer-supported collaborative learning: Cognitive, computational and educational perspectives*, 275-301, Springer, US.

Dimitracopoulou, A., Petrou, A. (2003). Advanced collaborative distance learning systems for young students: design issues and current trends on new cognitive and metacognitive tools. *Themes in Education International Journal*.

Dodl, N. (1990). Instructional groupware: Design considerations. In *Proceedings of the 32nd Annual International Conference of the Association for the Development of Computer-Based Instructional Systems*, 344-352, Computerized Conferencing and Communications Center, Columbus, Ohio State University, USA.

Donald, C., Blake, A., Girault, I., Datt, A., Ramsay, E. (2009). Approaches to learning design: past the head and the hands to the HEART of the matter. *Distance Education*, 30(2), 179-199.

Dougiamas, M., Taylor, P. (2003). Moodle: Using learning communities to create an open source course management system. In *World conference on educational multimedia, hypermedia and telecommunications*, 2003(1), 171-178.

Ellis, C., Nutt, G. (1980). Office Information Systems and Computer Science, *ACM Computing Surveys (CSUR)*, 12(1), 27-60.

Ellis, C., Gibs, S., Rein, G. (1991). Groupware: some issues and experiences. *Communications of the ACM*, 34(1), 39-58.

Emin, V., Pernin, J. P., Aguirre, J. L. (2010). SceneEdit: an intention-oriented authoring environment to design learning scenarios. *Proceedings of the European Conference on Technology-Enhanced Learning (EC-TEL 2009)*, 626-631, Barcelona, Spain.

Escobedo del Cid, J.P., de la Fuente Valentín, L, Gutiérrez, S, Pardo, A & Delgado Kloos, C (2007): Implementation of a Learning Design Run-Time Environment for the .LRN Learning Management System. *Journal of Interactive Media in Education* 2007(1):7, DOI: <http://dx.doi.org/10.5334/2007-7>

Fielding, R. T., Taylor, R., N. (2002). Principled design of the modern Web architecture. *ACM Transactions on Internet Technology (TOIT)* 2.2: 115-150.

Fisher, F., Kollar, I., Mandl, H., Haake, J. (2007). *Scripting computer-supported collaborative learning: Cognitive, computational and educational perspectives*. Springer.

Georgakopoulos, D., Hornick, M., Sheth, A. (1995). An Overview of Workflow Management: From Process Modeling to Workflow Automation Infrastructure. *Distributed and Parallel Databases*, 3(2), 119-153.

Ghiglione, E., Dalziel, J. (2007). Design principles for LAMS version 2 and the LAMS "Tools Contract". *Current Research on IMS Learning Design and Lifelong Competence Development Infrastructures*, 13.

- Glass, R.L. (1995). A Structure-Based Critique of Contemporary Computing Research. *Journal of Systems and Software*, 28(1), 3-7.
- Goodyear, P., Dimitriadis, Y. (2013): In medias res: Reframing design for learning. *Research in Learning Technology*, 21. doi: 10.3402/rlt.v21i0.19909
- Griffiths, D., Liber, O. (2008). Opportunities, Achievements, and Prospects for Use of IMS LD. *Handbook of Research on Learning Design and Learning Objects: Issues, Applications, and Technologies*, 87-112.
- Heinl, P., Horn, S., Jablonski, S., Neeb, J., Stein, K., Teschke, M. (1999): A comprehensive approach to flexibility in workflow management systems. In *ACM SIGSOFT Software Engineering Notes*, Vol. 24, No. 2, 79-88, ACM.
- Hermans, H., Janssen, J., Koper, R. (2014). Flexible authoring and delivery of online courses using IMS LD. *Interactive Learning Environments*.
- Hernández-Gonzalo, J.A., Villasclaras-Fernández, E.D., Hernández-Leo, D., Asensio-Pérez, J.I., Dimitriadis, Y (2008). InstanceCollage: a Graphical Tool for the Particularization of Role/Group Structures in Pattern-based IMS-LD Collaborative Scripts *Proceedings of the 8th IEEE International Conference on Advanced Learning Technologies, ICALT 08*, 506-510, Santander, Spain.
- Hernández-Leo, D. (2007). A pattern-based design process for the creation of CSCL macro-scripts computationally represented with IMS LD. Tesis Doctoral. Departamento de Ingeniería Telemática. Universidad de Valladolid, España.
- Hernández-Leo, D., Asensio-Peréz, J. I., Dimitriadis, Y. (2004). IMS Learning Design Support for the Formalization of Collaborative Learning Patterns (extended version in J. ET&S), *Proceedings of the 4th International Conference on Advanced Learning Technologies (Best Paper Award), ICALT 04*, 350-354.
- Hernández-Leo, D., Villasclaras-Fernández, E., Jorrín-Abellán, I., Asensio-Pérez, J., Dimitriadis, Y., Ruiz-Requies, I., Rubia-Avi, B. (2006). Collage, a Collaborative Learning Design Editor Based on Patterns. *Special Issue on Learning Design. Educational Technology & Society*, 9(1), 58-71.
- Hernández-Leo, D., Burgos, D., Tattersall, C., Koper, R. (2007). Representing Computer-Supported Collaborative Learning macro-scripts using IMS Learning Design. In *Proceedings of the Second European Conference on Technology Enhanced Learning, CEUR Workshop Proceedings*, ISSN 1613-0073, online CEUR-WS.org/Vol-280, EC-TEL'07, Crete, Greece.
- Hernández-Leo, D. H., Fernández, E. D. V., Pérez, J. I. A., Dimitriadis, Y., Lorenzo, M. L. B., & Abellán, I. M. J. (2008). The added value of implementing the Planet Game scenario with Collage and Gridcole. *Journal of Interactive Media in Education*, 2008(2), Art-20.

Hernández-Leo, D., Moreno, P., Chacón, J., & Blat, J. (2014): LdShake support for team-based learning design. *Computers in Human Behavior*, 37, 402-412.

Hilera, J. R., Palomar, D. (2005). Modelado de procesos de enseñanza-aprendizaje reutilizables con XML, UML e IMS-LD. *RED: Revista de Educación a Distancia*, (3) 6, available at (last retrieved: Nov. 2013) <http://www.um.es/ead/red/M2/>.

Hollingsworth, D., Hampshire, U. K. (January de 1994). Workflow Management Coalition. The Workflow Reference Model. Document Number TC00-1003. Workflow Management Coalition, available at (last retrieved Nov. 2013): <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.198.5206&rep=rep1&type=pdf>.

Holmberg, B. (1986). *Growth and Structure of Distance Education*. London: Croom-Helm.

Hornung, T., Koschmider, A., Mendling, J. (2006). Integration of heterogeneous BPM Schemas: The Case of XPDL and BPEL. 18th Conference on Advanced Information Systems Engineering (CAiSE), 23-36.

Hummel, H., Manderveld, J., Tattersall, C., Koper, R. (2004). Educational Modelling Language and learning design: new opportunities for instructional reusability and personalized learning. *International Journal of Learning Technology*, 1(1), 111-126.

IMS LD (2002). IMS Learning Design Specification. Available at: <http://www.imsglobal.org/learningdesign/>.

IMS LD (2003). IMS Learning Design Information Model v1.0, Technical Specification. Available at: <http://www.imsglobal.org/learningdesign/>.

IMS LD Best-Practices (2003). IMS Global Learning Consortium. IMS Learning Design Best Practice and Implementation Guide, version 1.0 Final Specification. Available at: <http://www.imsglobal.org/learningdesign/>.

IMS CP (2004). IMS Content Packaging Specification, v1.1.4. Available at: <http://www.imsglobal.org/learningdesign/>.

IMS QTI (2010). IMS Question & Test Interoperability Specification. Available at: <http://www.imsglobal.org/question/>

Johnson, R. T., Johnson, D. W. (1988). Cooperative Learning: two heads learn better than one. *Transforming Education*(18), 34-36.

Johnson-Lenz, P., Johnson-Lenz, T. (1981). Consider the Groupware: Design and Group Process Impacts on Communication in the Electronic Medium. *Studies of Computer-Mediated Communications Systems: A Synthesis of the Findings*. New jersey, USA.: In Hiltz, S. y Kerr, E. editors, Research Report No. 16, Computerized Conferencing and Communications Center, New Jersey Institute of Technology.

Jorrín-Abellán, I. M., Stake, R.E., Martínez-Monés, A (2009). The Needlework in evaluating a CSCL system: The Evaluand oriented Responsive Evaluation Model. In Proceedings of the 9th international conference on Computer-Supported Collaborative Learning, CSCL 2009, 68-72, Rhodes, Greece, ISLS.

Katsamani, Mary, Symeon Retalis. (2011). Making Learning Design in Layers: The CADMOS Approach. In Proceedings of the IADIS Multi Conference on Computer Science and Information Systems, 305-312.

Kavantzias, N., Burdett, D., Ritzinger, G. (2004). Web Services Choreography Description Language. Version 1.01, World Wide Web Consortium.

Kitchenham, B., Linkman, S., Law, D. (1997). "DESMET: a methodology for evaluating software engineering methods and tools." *Computing & Control Engineering Journal* 8.3, 120-126.

Knight, C., Gasevic, D., Richards, G. (2006). An ontology-based framework for bridging learning design and learning content, *Educational Technology & Society*, 9(1), 23-37.

Kobbe, L., Weinberger, A., Dillenbourg, P., Harrer, A., Hämäläinen, R., Häkkinen, P., Fischer, F. (2007). Specifying computer-supported collaboration scripts. *International Journal of Computer-Supported Collaborative Learning*, 2(2-3), 211-224.

Kollar, I., Fischer, F., Slotta, J. D. (2007). Internal and external scripts in computer-supported collaborative inquiry learning. *Learning and Instruction*, 17(6), 708-721.

König, F., & Paramythis, A. (2010). Towards Improved Support for Adaptive Collaboration Scripting in IMS LD. In Wolpers, M., Kirschner, P., Scheffel, M., Lindstaedt, S., Dimitrova, V. (eds), *Sustaining TEL: From Innovation to Learning and Practice*, 197-212. Springer Berlin Heidelberg.

Kopper, R. (2001). Modeling Units of Study from a Pedagogical Perspective: the pedagogical meta-model behind EML. Technical Report. Educational Technology Expertise Technology Centre, Open University of the Netherlands.

Koper, R., Manderveld, J. (2004). Educational modeling language: Modelling reusable, interoperable, rich and personalised units of learning. *British Journal of Educational Technology*, 35(5), 537-551.

Koper, R., Oliver, B. (2004). Representing the learning design of units of learning. *Educational Technology & Society*, 7(3), 97-111.

Koper, R., Tattersall, C. (2005). *Learning Design: A Handbook on Modelling and Delivering Networked Education and training*. Springer.

Koper, R., Miao, Y. (2007). Using the IMS LD Standard to Describe Learning Designs. Submitted book chapter. Available at (last retrieved: Nov. 2007): <http://dspace.ou.nl/bitstream/1820/927/1/UsingIMSLD.pdf>

Koper, R., Miao, Y. (2008). Using the IMS LD Standard to Describe Learning Designs. In Handbook of Research on Learning Design and Learning Objects: Issues, Applications and Technologies, Lockyer, L., Bennet, S., Agostinho, S., Harper, B. (Eds.). 41-86.

Koper, R. (2009). IMS Learning Design: State-of-the-art, Keynote Speech at the 4th European Conference on Technology Enhanced Learning (ECTEL 2009) Workshop: Relating IMS Learning Design to Web 2.0 Technology.

Koschmann, T. (1996). Paradigm shifts and instructional technology. In CSCL: theory and practice of an emerging paradigm, T. Koschmann (Ed.), Mahwah, NJ, USA: Lawrence Erlbaum, 1-23.

Laurillard, D. (2006). Learning design futures: What are our ambitions? Presentation at Innovating e-Learning 2006: Transforming Learning Experiences, Online Conference, 27-31.

Laurillard, D. (2012). Teaching as a Design Science: Building Pedagogical Patterns for Learning and Technology. New York, Routledge.

Laurillard, D., Charlton, P., Craft, B., Dimakopoulos, D., Ljubojevic, D., Magoulas, G., Whittlestone, K. (2013). A constructionist learning environment for teachers to model learning designs. Journal of computer assisted learning, 29(1), 15-30.

Learning Design Editor Web Site. (s.f.). Available at: <http://www.reload.ac.uk/ldeditor.html>

Lin, J., Ho, C., Sadiq, W., Orlowska, M. E. (2002). Using workflow technology to manage flexible e-learning services. Educational Technology & Society, 5(4), 116-123.

LTSC (2000). Learning Technology Standards Committee Website. (2000). Available at (last retrieved Feb. 2006): <http://ieeeltsc.org/>.

MacLean, P., Scot, B. (2007). Learning design: requirements, practice and prospects. Campus-Wide Information Systems, 24(3), 187-198.

Magnisalis, I. D., Demetriadis, S. N. (2014). Case Studies on the Orchestration of Technology-Enhanced Collaboration Scripts through the MAPIS3 Architecture. In Intelligent Networking and Collaborative Systems (INCoS), 2014 International Conference on, 665-670, IEEE.

Magnisalis, I. D., Demetriadis, S. N. (2012). Orchestrating adaptive and complex CSCL scenarios through a choreography among IMS-LD and external services. In Intelligent Networking and Collaborative Systems (INCoS), 2012 4th International Conference on, 89-96, IEEE.

- Mariño, O., Casallas, R., Villalobos, J., Correal, D., Contamines, J. (2007). Bridging the Gap between e-learning Modeling and Delivery through the Transformation of Learnflows into Workflows. In *E-Learning Networked Environments and Architectures*, 27-59, Springer London.
- Marjanovic, O. (2005). Towards A Web-Based Handbook of Generic, Process-Oriented Learning Designs. *Educational Technology & Society*, 8(2), 66-82.
- Martel, C., Vignollet, L, Ferraris, C., Durand, G. (2006). LDL: a Language to Model Collaborative Learning Activities. In *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications (EDMEDIA 2006)*, 838-844.
- Martel, C., Vignollet, L. (2008). Using the Learning Design Language to model activities supported by services. *International Journal of Learning Technology*, 3(4), 368-387.
- Martínez-Monés, A., Dimitriadis, Y., Rubia-Avi, B., Gómez-Sánchez, E., de la Fuente-Redondo, P. (2003). Combining qualitative evaluation and social network analysis for the study of classroom social interactions. *Computers and Education*, 41(4), 353-368.
- Mendling, J., Müller, M. (2003). A Comparison of BPEL4WS and BPML. In Tolksdorf, R. and Eckstein, R. (Eds.), *Proceedings of Berliner XML-Tage*, 305–316.
- Mendling, J., Neuman, G., Nüttgens, M. (2005). A Comparison of XML Interchange Formats for Business Process Modelling. *Workflow handbook*, 185-198.
- Menling, J., Pérez de Laborda, C., Zdun, U. (2005). Towards Semantic Integration of XML-based Business Process Models. In Althoff, K.D.; Dengel, A.; Bergmann, R.; Nick, M.; Roth-Berghofer, T., *Professional Knowledge Management-Experiences and Visions. Semantic Model Integration Workshop (SMI 2005) as part of the 3rd Conference Professional Knowledge Management (WM 2005)*, 513-517, Kaiserslautern, Germany.
- Miao, Y. D., Burgos, Griffiths, D., Koper, R. (2008). Representation of Coordination Mechanisms in IMS LD. *Handbook of Research on Learning Design and Learning Objects: Issues, Applications and Technologies*, 330.
- Miao, Y., Hoeksema, K., Hoppe, H. U., & Harrer, A. (2005). CSCL scripts: modelling features and potential use. In *Proceedings of th 2005 conference on Computer support for collaborative learning: the next 10 years!*, 423-432, International Society of the Learning Sciences.
- Miers, D. (2006). Best practice (BPM). *Queue* 4.2: 40-48.
- Miers, D. (2006b). The keys to bpm project success. *BPTrends*, Jan.
- Moore, M. (1898). Editorial: Three types of interaction. *The American Journal of Distance Education*, 3(2), 1-7.

Mor, Y., Craft, B. (2012). Learning design: reflections upon the current landscape. *Research in Learning Technology*, 20.

Mor, Y., Craft, B., Hernández-Leo, D. (2013). Editorial: The art and science of learning design. *Research In Learning Technology*, v.21, <http://dx.doi.org/10.3402/rlt.v21i0.22513>.

Müller, D. (Ed.) (2010): ICOPER Deliverable D5.3: ISURE: Usage of IMS-LD for Communication and Collaboration Services. http://www.icoper.org/deliverables/ICOPER_D5.3.pdf

Mulholland, P., Anastopoulou, S., Collins, T., Feisst, M., Gaved, M., Kerawalla, L., Wright, M. (2012). nQuire: technological support for personal inquiry learning. *Learning Technologies, IEEE Transactions on*, 5(2), 157-169.

Muñoz-Cristóbal, Juan, A. (2005). Supporting teacher orchestration of across-spaces learning situations. PhD Thesis, School of Telecommunication Engineering, University of Valladolid, Spain.

Myers, K. L., Berry, P. (1999). *Workflow Management Systems: An AI Perspective*. Artificial Intelligence Center, SRI International. Menlo Park.

National Institute for Science Education (1997). Doing collaborative learning: collaborative learning structures. Available at (last retrieved: Nov. 2013): <http://www.wcer.wisc.edu/archive/cl1/cl/doingcl/clstruc.htm>

Neumann, S., Klebl, M., Griffiths, D., Hernández-Leo, D., de la Fuente-Valentín, L., Hummel, H., Oberhuemer, P. (2009). Report of the Results of an IMS Learning Design Expert Workshop. *International Journal of Emerging International Journal of Emerging Technologies In Learning (IJET)*, 5(1): 58-72.

Neumann, S., Oberhuemer, P., Derntl, M. (2009B). Visualizing learning designs using IMS Learning Design: the position of the Graphical Learning Modeller. In *Advanced Learning Technologies, 2009, ICAIT 2009, Ninth IEEE International Conference on*, IEEE, 732-733.

Nodenot, T., Laforcade, P. (2006). CPM: A UML Profile to Design Cooperative PBL Situations at Didactical Level. *Sixth International Conference on Advanced Learning Technologies*, 1113-1114, IEEE, Kerkrade, The Netherlands.

Norin, R. (2002). *Workflow Process Definition Interface–XML Process Definition Language*. Document Number WfMC. TC-1025, Lighthouse Point (FL): Workflow Management Coalition.

Nutt, G. J. (1996). The Evolution Towards Flexible Workflow Systems. *Distributed Systems Engineering*, 3(4), 276-294.

OMG (2004). *Unified Modeling Language Version 2.0*, Object Management Group.

Osuna-Gómez, C. (1999). DELFOS: un marco telemático educativo basado en niveles orientado a situaciones de aprendizaje cooperativo. Tesis Doctoral, Universidad de Valladolid, España.

OUNL EML (2003). OUNL Educative Modeling Language. Available at (last retrieved Feb. 2006): <http://eml.ou.nl>.

Pagola, B. (2006). Procesamiento de Imágenes. Filtros. Available at (last retrieved Feb. 2006): http://www.ayc.unavarra.es/miguel.pagola/P_Filtros.htm

Palomino-Ramírez, L., Martínez-Monés, A., Bote-Lorenzo, M., Asensio-Pérez, J., Dimitriadis, Y. (2007). Data Flow between Tools: Towards a Composition-Based Solution for Learning Design . Proceedings of the 7th International Conference on Advanced Learning Technologies (ICALT 2007), Niigata, Japan.

Palomino-Ramírez, L., Bote-Lorenzo, M., Asensio-Pérez, J., Dimitriadis, Y., de la Fuente-Valentín, L. (2008). The Data Flow Problem in Learning Design: A Case Study. 6th International Conference on Networked Learning (NLC 2008), Halkidiki, Greece.

Palomino-Ramírez, L., Bote-Lorenzo, M., Asensio-Pérez, J., Dimitriadis, Y. (2008b). LeadFlow4LD: A Composition-based Solution for Learning Design in CSCL. 14th Collaboration Researchers' International Workshop on Groupware (CRIWG 2008), Omaha, Nebraska, USA.

Palomino-Ramírez, L., Bote-Lorenzo, M.L., Asensio-Pérez, J.I., Vignollet, L., Dimitriadis, Y. (2013). LeadFlow4LD: A Method for the Computational Representation of the Learning Flow and Data Flow in Collaborative Learning. Journal of Universal Computer Science, 19(6), 805-830.

O'Donnell, A. M., Dansereau, D. F. (1992). Scripted cooperation in student dyads: A method for analyzing and enhancing academic learning and performance. In R. Hertz-Lazarowitz and N. Miller (Eds.), Interaction in cooperative groups: The theoretical anatomy of group learning, 120-141, London: Cambridge University Press.

Paquette, G. (2004). Educational Modeling Languages, from an instructional engineering perspective. Online education using learning objects, 331-346.

Paquette, G. (2006). Discussion of Collaborative Learning Designs, Languages, Models and Tools-LICEF-Télé université's contribution. In Proceedings of the ICALT 2006 conference, (1105-1106). Kerkrade, The Netherlands.

Paquette, G. (2009). Ontology-based educational modelling. Extending the IMS LD Specification (submitted).

Paquette, G., Aubin, C., Crevier, F. (1999). MISA, A knowledge-based method for the engineering of learning systems. Journal of Courseware Engineering, 2, 63-78.

Paramythis, A. (2008). Adaptive support for collaborative learning with ims learning design: Are we there yet. In Proceedings of the Workshop on Adaptive Collaboration Support, held in conjunction with the 5th International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, 17-29, Hannover, Germany.

Perisco, D., Pozzi, F., Anastopoulou, S., Conole, G., Craft, B., Dimitriadis, Y., Walmsley, H. (2013). Learning design Rashomon I-supporting the design of one lesson through different approaches. *Research in Learning Technology*, 21 (In Press).

Peter, Y., Vantroys, T. (2005). Platform Support for Pedagogical Scenarios. *Journal of Educational Technology & Society*, 8, 122-137.

Prieto, L.P., Dimitriadis, Y., Craft, B., Derntl, M., Émin, V., Katsamani, M., Laurillard, D., Masterman, E., Retalis, S., Villasclaras, E.. Learning design Rashomon II: exploring one lesson through multiple tools. *Research in Learning Technology* 21 (2013).

Prieto-Santos, L.P., Asensio-Pérez, J.I., Dimitriadis, Y., Gómez-Sánchez, E., Muñoz-Cristóbal, J.A. (2011). GLUE!-PS: A multi-language architecture and data model to deploy TEL designs to multiple learning environments. Proceedings of the 6th European Conference on Technology Enhanced Learning, EC-TEL 2011, Palermo, Italy.

Rawlings, A., van Rosmalen, P., Koper, R., Rodríguez-Atacho, M., Lefrere, P. (2002). Survey of Educational Modeling Languages (EMLs). Technical Report, CEN/ISSS WS/LT.

Redwine, S., Becker, L. G., Marmor-Squires, A.B., Martin, R.J., Nash, S.H., Riddle, W. (1984). DoD Related Software Technology Requirements, Practices, and Prospects for the Future. IDA Paper P-1788.

Reiser, R. A. (2001). A history of instructional design and technology: Part II: A history of instructional design. *Educational technology research and development*, 49(2), 57-67.

Rius, À., Conesa, J., García-Barriocanal, E., Sicilia, M. A. (2014). Automating educational processes implementation by means of an ontological framework. *Computer Standards & Interfaces*, 36(2), 335-348.

Rodríguez-Artacho, M., Maillo, M. (2004). Modeling Educational Content: The Cognitive Approach of the PALO Language. *Journal of Educational Technology and Society*, 7(3), 124-137.

Ronen, M., Kohen-Vacs, D., Raz-Fogel, N. (2006). Adopt & adapt: structuring, sharing and reusing asynchronous collaborative pedagogy. In Proceedings of the 7th International Conference on Learning Sciences, International Society of the Learning Sciences, 599-605.

Roschelle, J., Kaput, J., Stroup, W., Kahn, T. (1998). Scalable Integration of Educational Software: Exploring The Promise of Component Architectures. *Journal of Interactive Media in Education*, 6, 1-31.

Rossi, D., Turrini, E. (2006). What your next workflow language should look like. In 2nd International Workshop on Coordination and Organization.

Russell, N., ter Hofstede, A., Edmond, D., van der Aalst, W. (2005). Workflow data patterns: Identification, representation and tool support. In Delcambre, Lois M. L., Kop, C., Mayr, H. C., Mylopoulos, J., Pastor, O. (Eds.), 24th International Conference on Conceptual Modeling, 24-28, Klagenfurt, Austria.

Russell, N., ter Hofstede, A.H.M., van der Aalst, W.M.P., Mulyar, N. (2006). Workflow Control-Flow. Patterns: A Revised View. BPM Center Report BPM-06-22, BPMcenter.org.

Sadiq, S., Sadiq, W., Orlowska, M. (2002). Workflow Driven e-Learning–Beyond Collaborative Environments. Networked Learning in a Global Environment. Challenges and Solutions for Virtual Education, 1-4.

Sampson, D., Karampiperis, P., Zervas, P. (2005). ASK-LDT: A Web-Based Learning Scenarios Authoring Environment based on IMS Learning Design. International Journal on Advanced Technology for Learning (ATL), 2(4), 207-215.

Schmuck, R. A., Schmuck, P. A. (1988). Group Process in the Classroom. Wm. C. Brown, Dubuque, IA, 5th edition.

Shapiro, R. (2008). XPDL 2.1: Integrating Process Interchange & BPMN. Workflow Management Coalition.

Sharples, M., Collins, T., Feißt, M., Gaved, M., Mulholland, P., Paxton, M., Wright, M. (2011). A “Laboratory of Knowledge-Making” for Personal Inquiry Learning. In Artificial Intelligence in Education, 312-319, Springer Berlin Heidelberg.

Shaw, M. (2002). What makes good research in software engineering? International Journal on Software Tools for Technology Transfer, 4(1), 1-7.

Sheth, A., Georgakopoulos, D., Joosten, S., Rusinkiewicz, M., Scacchi, W., Wileden, J., Wolf, A. (1997). Report from the NSF Workshop on Workflow and Process Automation in Information Systems. ACM SIGSOFT Software Engineering Notes, 22(1), 28-38.

Sicilia, M., Miltiadis, L. (2005). Editorial, Learning objects & Learning designs, 1(1).

Sife, A., Edda, L., Camilius, S. (2007). New Technologies for teaching and learning: Challenges for higher learning institutions in developing countries. International Journal of Education and Development using ICT, 3(2).

Simon, C., Reckmann, N., Swatman, P. M. (2003). Initiating eLearning via Workflow Management, 319-333, Springer US.

- Slavin, R. (1987). Cooperative Learning and Students Achievement. Technical Report EORIG-86-006, Office of Educational Research, Washington DC, USA.
- Smith, P. L., Ragan, T. J. (2005). Instructional design. Hoboken, NJ, USA, Wiley.
- Sobreira, P., Tchounikine, P. (2015). Table-based representations can be used to offer easy-to-use, flexible, and adaptable learning scenario editors. *Computers & Education*, 80, 15-27.
- Sobreira, P. (2014). T2/ediT2—a flexible and easy-to-use model/system for editing and operationalizing learning scenarios (Doctoral dissertation, Université de Grenoble).
- Stahl, G., Koschmann, T., Suthers, D. (2006). Computer-supported collaborative learning: An historical perspective. *Cambridge handbook of the learning sciences*, 409-426.
- Stahl, G., Spada, H., Miyake, N., Law, N. (2011). Introduction to the proceedings of CSCL 2011. Proceedings of the 9th international conference on Computer-Supported Collaborative Learning (CSCL 2011), vii-xi, In G. Stahl, H. Spada, N. Miyake, and N. Law (Eds.).
- Stylianakis, G., Moumoutzis, N., Arapi, P., Mylonakis, M., Christodoulakis, S. (2014). COLearn and open discovery space portal alignment: A case of enriching open learning infrastructures with collaborative learning capabilities. In *Interactive Mobile Communication Technologies and Learning (IMCL)*, 2014 International Conference on, 252-256, IEEE.
- Surhome, L., Tennoe, M., Henssonow, S. (2010). Bonita Open Solution, <http://www.bonitasoft.com/>.
- Swenson, K. D., Irwin, K. (1995). Workflow technology: trade-offs for business process re-engineering. In *Proceedings of conference on Organizational computing systems*, 22-29. ACM.
- Tattersall, C., Vogten, H., Brouns, F., Koper, R., Van Rosmalen, P., Sloep, P., van Bruggen, J. (2003). Delivering courses modeled using IMS Learning Design.
- The Workflow Reference Model (1994). Workflow Management Coalition, TC00-1003.
- Tichy, W. F., Lukowicz, P., Prechelt, L., Heinz, E. A. (1995). Experimental evaluation in computer science: A quantitative study. *Journal of Systems and Software*, 28(1), 9-18.
- Tichy, W. F. (1998). Should computer scientists experiment more? 16 reasons to avoid experimentation. *IEEE Computer*, 31(5), 32-40.
- van der Aalst, W. (2004). Business Process Management Demystified: A Tutorial on Models, Systems and standards for Workflow Management. In J. Desel, W. Reisig, and G. Rozenberg (Eds.), *Lectures on Concurrency and Petri Nets*, 3098, 1-65.
- van der Aalst, W., van Hee, K. M. (2002). *Workflow Management: Models, Methods, and Systems*, MIT press.

- van der Aalst, W., Hofstede, t., Kiepuszewski, B., Barros, A. (2003). Workflow Patterns. *Distributed and Parallel Databases*, 14(1), 5-51.
- Vantroys, T., Peter, Y. (2003). COW, a Flexible Platform for the Enactment of Learning Scenarios. In *Groupware: Design, Implementation, and Use*, 168-182, Springer Berlin Heidelberg.
- Vantroys, P., Vantrouys, T. (2005). Platform Support for Pedagogical Scenarios. *International Forum of Educational Technology & Society*, 8(3), 122-137.
- Vasko, M., Dustdar, S. (2006). A View Based Analysis of Workflow Modeling Languages. *Proceedings of the 14th Euromicro International Conference on Parallel, Distributed, and Networked-Based processing (PDP)*, IEEE computer Society, 293-300.
- Vidal, J., Lama, M., Sánchez, E., Bugarín, A. (2008). Application of Petri Nets on the Execution of IMS Learning Design documents. *Proceedings of the 3rd European Conference on Technology Enhanced Learning (EC-TEL 2008)*, 461-466, Maastricht, The Netherlands.
- Vidal, J., Lama, M., Sánchez, E., Bugarí, A., Novegil, A. (2009). OPENET LD: An Ontology-based Petri Net Engine to Execute IMS LD Units of Learning. In *Proceedings of the 9th IEEE International Conference on Advanced Learning Technologies (ICALT 2009)*, 499-503, Riga, Latvia.
- Vignollet, L., Bote Lorenzo, M., Asensio Pérez, J. (2010). Work Flow Management and Learn Flow Management: commonalities and differences. *IEEE Learning Technology*, 12(3), 35-37.
- Vignollet, L., Bote-Lorenzo, M., Asensio-Pérez, J., Dimitriadis, Y. (2009). A generic specification of the data-flow issue in the learning design field. In *Proceedings of the 9th IEEE International Conference on Advanced Learning Technologies (ICALT 2009)*, 56-58, Riga, Latvia.
- Vignollet, L., Ferraris, C., Martel, C., Burgos, D. (2008). A Transversal Analysis of Different Learning Design Approaches. *Journal of Interactive Media in Education* (2).
- Villasclaras-Fernández, E., Hernández-Leo, D., Asensio-Pérez, J., Dimitriadis, Y. (2013). Web Collage: An implementation of support for assessment design in CSCL macro-scripts. *Computers & Education*, 67, 79-97.
- Vogten, H., Martens, H. (2005). CopperCore 2.2.4. Heerlen: Open University of the Netherlands, <http://coppercore.sourceforge.net/>.
- Vogten, H., Tattersall, C., Koper, R., van Rosmalen, P., Brouns, F., van Bruggen, J., Sloep, P., Martens, H. (2004). Designing a Learning Design Engine as a collection of Finite State Machines. *International Journal on E-Learning*.

Vogten, H., Koper, R. (2014). Towards a new generation of Learning Management Systems. In Proceedings of the 6th International Conference on Computer Supported Education. Barcelona.

Wagner, E. D. (1994). In Support of a functional definition of interaction. The American Journal of Distance Education, 8(2), 6-26.

Weinberger, A., Fischer, F., Mandl, H. (2004). Knowledge convergence in computer-mediated learning environments: Effects of collaboration scripts. Proceedings of the 85th Annual Meeting of the American Educational Research Association (AERA).

Weinberger, A., Kollar, I., Dimitriadis, Y., Mäkitalo-Siegl, K., Fischer, F. (2009). Computer-supported collaboration scripts. In Technology-enhanced learning, 155-173, Springer Netherlands.

Weinberger, A., Stegmann, K., Fischer, F. (2005). Computer-supported collaborative learning in higher education: Scripts for argumentative Knowledge construction in distributed groups. In Proceedings of Computer Supported Collaborative Learning (CSCL 2005): The next 10 Years!, 717-726.

WfMC Terminology & Glossary (1999). The Workflow Management Coalition, Terminology & Glossary, Document No. WfMC-TC-1011, http://www.wfmc.org/standards/docs/TC-1011_term_glossary_v3.pdf.

White, S. A. (2004). Business Process Modeling Notation. Specification. BPMI.org.

Wijen, W. (2000). Towards design-based learning. OGO brochure, No 2. Educational Service Centre. Eindhoven, NL: Technische Universiteit. Available at (last retrieved Sept. 2013): http://w3.tue.nl/fileadmin/stu/stu_oo/doc/OGO_brochure_1_EN.pdf.

Wiley, D. A. (2000). Connecting Learning Objects to Instructional Design Theory: A Definition, a Metaphor, and a Taxonomy. The Instructional Use of Learning Objects, 2830(435), 1-35.

Wilson, S. (2005). CETIS. Workflow and web services. Available at (last retrieved Feb. 2006): <http://www.e-framework.org/resources/SOAandWorkflow2.pdf>

Wilson, S. (2005). Workflow and web services. CETIS White paper. Available at (Last retrieved Jan. 2007.): <http://www.e-framework.org/resources/SOAandWorkflow2.pdf>.

XPDL (2008). Process Definition Interface -- XML Process Definition Language. Document Number WfMC-TC-1025. Workflow Management Coalition. http://www.xpdl.org/standards/xpdl-2.0/TC-1025_xpdl_2_2005-10-03.pdf.

XPDL Editor Web Site. (s.f.). <http://www.together.at/prod/workflow/twe>

Young, M. (1993). Alternatives to Quantitative Evaluation. Summary of the Dagstuhl Workshop on Future Directions in Software Engineering SIGSoft Software Engineering Notes. 18(1), 36-37.

Zelkowitz, M. V., Wallace, D. R. (1997). Experimental validation in software engineering. Information and Software Technology, 39(11), 735-743.

Zelkowitz, M. V., Wallace, D. R. (1998). Experimental models for validating technology. Computer, 31(5), 23-31.

zur Muehlen, M., Becker, J. (1999). Workflow Process Definition Language--Development and Directions of a Meta-Language for Workflow Processes.

zur Muehlen, M. (2007). Business Process Management Standards. Origen, Overview and Directions. Available at (last retrieved: Nov. 2013): <http://bpm07.fit.qut.edu.au/program/slides/Thursday/Thursday-Tutorials/Muehlen.pdf>