

Middleware para CSCL: Marco de Componentes Software y Apoyo de Tecnología Grid

**Yannis Dimitriadis¹, Juan I. Asensio Pérez¹, Eduardo Gómez Sánchez¹,
Alejandra Martínez Monés², Miguel L. Bote Lorenzo¹ y Guillermo Vega Gorgojo¹**

¹ETSI de Telecomunicación, Universidad de Valladolid

²ETSI de Informática, Universidad de Valladolid

Camino Viejo del Cementerio s/n, 47011 Valladolid, Spain

{yannis@tel, juaase@tel, edugom@tel, amartine@infor, migbot@tel, guiveg@tel}.uva.es

Resumen

El Aprendizaje Colaborativo Apoyado por Ordenador (CSCL) viene siendo en los últimos años un campo de investigación y práctica educativa muy activo. Sin embargo, siguen existiendo serios problemas que impiden el uso eficiente de las tecnologías emergentes y el establecimiento de un canal fluido de comunicación entre los distintos actores de la comunidad (tecnólogos y educadores). Este artículo presenta la trayectoria del grupo interdisciplinar EMIC (Educación, Medios, Informática y Cultura) de la Universidad de Valladolid en un intento por remediar estos problemas. En este sentido, se plantean los esfuerzos y lecciones aprendidas en el proceso de conseguir un marco de componentes software para CSCL así como el uso eficiente de tecnologías Grid. El proceso se analiza también a través de varias aplicaciones cuya puesta en marcha y evaluación permiten la validación de las propuestas.

Palabras clave: CSCL. Middleware, componentes software, diseño participativo, Grid, P2P, marco, grupo interdisciplinar

1. Introducción

Los esfuerzos por aplicar las innovaciones tecnológicas a la mejora de la educación se han sucedido a lo largo de la historia. En el caso de las TICs, han dado lugar a diversos paradigmas de informática educativa: enseñanza asistida por ordenador, sistemas tutores inteligentes, simulaciones o micromundos y, más recientemente, con la generalización de las redes de ordenadores, al *e-learning* y al CSCL [20].

Esta diversidad refleja, muchas veces implícitamente, tanto la evolución de las tecnologías como de las tendencias educativas. Así, el término de *e-learning*, que representa la corriente dominante actual, incorpora principalmente las tecnologías Web y realiza un mayor énfasis en un aprendizaje centrado en el alumno, autónomo y flexible (típicamente a distancia).

El CSCL, surgido en parte como evolución del CSCW (*Computer-Supported Collaborative Work* o Trabajo Colaborativo Apoyado por Ordenador) refleja un nuevo paradigma de investigación y práctica educativa fuertemente interdisciplinar. Éste se caracteriza por realzar la importancia de las interacciones sociales (colaboración) como

elemento esencial del aprendizaje, la preferencia por un enfoque interpretativo frente al positivista tradicional a la hora de evaluar el aprendizaje, así como el papel del análisis y diseño participativo de toda la comunidad en la creación de nuevos entornos tecnológicos [13]. Por otro lado, el CSCL ha recurrido a las tecnologías de sistemas distribuidos para dar soporte a las tres características fundamentales para este tipo de sistemas: comunicación, colaboración y coordinación.

El carácter interdisciplinar del CSCL, así como la gran cantidad de actores implicados en su puesta en práctica, hacen que éste sea un campo de investigación muy complejo, con muchas perspectivas, y a la vez un lugar de investigación e innovación estratégico por las muchas cuestiones que se plantean en el mismo [7].

Siendo un área con muchas facetas, el acercamiento de los investigadores al CSCL puede realizarse por distintas vías. En el caso de los autores, la evolución surgió de un interés inicial por los aspectos tecnológicos de los sistemas CSCW, que pronto evolucionó al ámbito del CSCL, al ir aplicando los problemas al entorno educativo y adoptando una visión más global del problema, en la que ha sido fundamental la experiencia de trabajo

interdisciplinar en el grupo EMIC (Educación, Medios, Informática y Cultura) de la Universidad de Valladolid.

Este artículo presenta los pasos del grupo EMIC desde 1994 hasta la actualidad, centrándose en la visión del subgrupo tecnológico GSIC (Grupo de Sistemas Inteligentes y Colaborativos) como testigo de la evolución de I+D+i en CSCL. La perspectiva elegida ha sido la de exponer las aproximaciones a los problemas que han ido surgiendo, tanto a nivel teórico como de las aplicaciones desarrolladas, con especial énfasis en las motivaciones de dichos intentos, las lecciones aprendidas y los principales logros conseguidos. En concreto, la discusión se centra en el problema de la elaboración de un *lenguaje común* entre educadores y tecnólogos que permita la comprensión mutua, requisito imprescindible para el avance en este campo, y las *posibilidades del CSCL como dominio significativo* para el estudio de ciertas tecnologías, como la ingeniería del software basada en componentes (ISBC), aplicaciones distribuidas, y tecnologías emergentes como grid y P2P.

El artículo se estructura como sigue: en primer lugar se presenta la trayectoria del grupo EMIC, así como algunos de los proyectos llevados a cabo por el mismo como respuesta a los problemas encontrados durante su evolución; a continuación se presentan dos de las principales líneas de investigación del grupo en los últimos años: así, la sección 3 presenta los esfuerzos realizados para el desarrollo de un marco de componentes para CSCL, y la sección 4, una discusión sobre potenciales posibilidades de las tecnologías de computación grid en este dominio. La sección 5 finaliza el artículo con una discusión sobre las principales ideas aportadas, y presenta las líneas de investigación en las que el grupo pretende trabajar a corto plazo.

2. El Grupo EMIC y el Dominio de CSCL

La introducción de las TIC en el campo de la educación ha fascinado a un cierto sector de la sociedad, así como a múltiples profesionales de los campos educativo y tecnológico. Sin embargo, en un ámbito complejo como el que se acaba de exponer, ¿quién decide cuáles son las tecnologías a aplicar? ¿los tecnólogos, que conocen los últimos avances de las TIC, los educadores y pedagogos que conocen la práctica y la teoría de la educación, o los políticos, que tienen la última palabra sobre las decisiones y los presupuestos? Es evidente que cualquiera de estos grupos sólo es capaz de ofrecer una visión parcial del problema, y por tanto, más propensa al fracaso. Ello lleva a pensar en *equipos*

interdisciplinares, donde profesionales de disciplinas diversas colaboran ofreciendo la experiencia y conocimientos de los distintos campos. Con esta idea en mente surgió en 1994 el grupo EMIC formado, en el ámbito universitario, por profesores pertenecientes a distintas áreas de conocimiento dentro de la Pedagogía, Telemática e Informática de la Universidad de Valladolid, en colaboración con grupos de otras universidades como la UOC y la UPC. En EMIC trabajan también personas de otros centros educativos (Asociación Familiar Rondilla de Valladolid), empresas (Edebé, Espáis y Tecsidel), e instituciones (Centro de Profesorado y Recursos de la Junta de Castilla y León).

El grupo tuvo su origen en el interés por el estudio de la escritura por parte de los tecnólogos los cuales, al conocer que había profesores en la Facultad de Educación de Valladolid que trabajaban sobre esta cuestión, se pusieron en contacto e iniciaron así las conversaciones. Pronto se observó la gran diferencia entre las perspectivas de ambos grupos, pero aún así se planteó la colaboración, y empezaron los pasos hacia el establecimiento de un marco más amplio de trabajo. Y pronto también empezaron las dificultades para llevar a cabo una buena colaboración. La primera de ellas consistió en las diferencias en las motivaciones entre tecnólogos y educadores. La segunda, en el establecimiento de una comprensión mutua que permita trabajar de una forma realmente interdisciplinar.

En lo que respecta a la primera dificultad, la *diferencia de motivaciones*, ésta se manifestó principalmente al comienzo de la vida del grupo. Mientras los tecnólogos buscaban un dominio interesante donde probar un cierto tipo de tecnología (sistemas de reconocimiento de escritura y sistemas distribuidos), los educadores buscaban aplicaciones finales que les ayudaran a llevar a cabo sus clases. Evidentemente, estas diferencias causaron desajustes, que sólo se vieron superadas cuando se puso por encima de las motivaciones particulares el objetivo común de la mejora de la enseñanza. Éste obligaba a tecnólogos y a educadores a acercarse al otro ámbito de forma crítica, y a buscar soluciones comunes que no dieran nada por supuesto. Todo ello acompañado de una fuerte dosis de persistencia, siempre necesaria cuando los resultados no aparecen fácilmente.

Como medio para afianzar este objetivo común, se vio necesario hacer explícitos los principios del grupo. Para ello, se inició un proceso de introspección que dio lugar al enunciado de una serie de principios básicos, que rigen desde entonces la vida del grupo. Éstos son:

- perspectiva crítica ante la educación y la utilización de las TIC.
- enfoque constructivista de los procesos de enseñanza y aprendizaje.
- valoración positiva de la colaboración en dichos procesos.
- fomento de la interdisciplinariedad y el trabajo colaborativo entre profesores.
- vinculación entre la enseñanza y la investigación en el área profesional de cada actor implicado.

De acuerdo con estos principios, la tarea del grupo ha supuesto, por un lado, un esfuerzo por repensar la propia práctica educativa bajo los principios de la perspectiva crítica y del enfoque constructivista, y por otro, la búsqueda de nuevas formas en las cuales las TIC puedan servir a estos principios. Un elemento central en ambas direcciones ha sido la preocupación por el fomento de la colaboración en las aulas, lo que ha llevado a situar la mayor parte de la investigación del grupo dentro del CSCL. Este fomento de la colaboración es también aplicado a la propia tarea del grupo, y constituye la estrategia para seguir avanzando. En este sentido, se puede decir que, en lo tocante al desarrollo de nuevos proyectos y aplicaciones, los principios del análisis y diseño participativos son los que dominan la acción del grupo, y los que desde la experiencia adquirida, se propone como base para el trabajo de todo grupo interdisciplinar.

El segundo de los problemas mencionados ha sido más importante, y consiste en las dificultades existentes para que los miembros del grupo se entiendan mutuamente. Esta situación es típica en los ámbitos interdisciplinares. El problema se plantea porque los diferentes orígenes de los componentes del grupo hacen que éstos tengan marcos de referencia diferentes, lo que incluye el uso de términos propios y no comprensibles para el otro, pero sobre todo, y mucho más problemático, de términos que adoptan diferentes significados según el área de conocimiento de referencia. Para lograr superar este problema es necesaria la creación de un marco común, con conceptos y herramientas entendibles por ambas partes, que sirvan de puente para la elaboración de proyectos comunes.

Ante esta cuestión se planteó la definición de un marco para el desarrollo de aplicaciones CSCL que sirviera de puente entre todos los actores implicados. Este objetivo se concretó en el marco telemático-educativo DELFOS (*A Description of Educational Layer Framework Oriented to learning Situations*, Una descripción de un marco basado en capas orientado a situaciones de aprendizaje) [17], compuesto fundamentalmente por tres elementos:

un *modelo del aprendizaje* basado en las ideas del constructivismo social, una *arquitectura híbrida* basada en capas y objetos para el desarrollo de las aplicaciones, y una *metodología de desarrollo* fundamentada en las ideas del análisis y diseño participativos. DELFOS propone el concepto de *situación de aprendizaje*, que modela un determinado ambiente de aprendizaje, y que incluye aspectos fundamentales para un educador, como son el contexto y los objetivos pedagógicos perseguidos. Tanto la situación de aprendizaje como sus principales elementos (actividades, roles, objetos e interacciones) se modelan con la ayuda de unas plantillas que fueron definidas dentro de la propuesta.

El marco fue validado mediante el desarrollo de tres aplicaciones de características diferentes. Estas experiencias mostraron que DELFOS proporciona herramientas útiles en la etapa de análisis de las necesidades, permitiendo así estructurar el dominio. De esta forma, se hace más sencilla la comunicación entre usuarios finales y desarrolladores de software, lo que constituyó la principal aportación del marco. Además, DELFOS posee otros objetivos relacionados con la reutilización de las aplicaciones desarrolladas, que serán comentados en la siguiente sección.

El método de desarrollo propuesto en DELFOS requería una evaluación adaptada a las necesidades del CSCL, que permitiera aprender de la práctica e ir mejorando de forma continua las aplicaciones. En respuesta a este requisito, en los últimos años se ha estado trabajado en la definición de un método de evaluación formativa de situaciones CSCL centrado en el estudio de los aspectos participativos del aprendizaje. El resultado ha sido la propuesta de un *método mixto*, que combina tres tipos de análisis: cualitativo, cuantitativo y análisis de redes sociales [15]. Un aspecto fundamental del método es la definición de herramientas de análisis de la interacción, que se fundamentan en una representación genérica de la acción colaborativa, sobre la cual es posible modelar la interacción y extraer la información que se considere significativa para el evaluador o los propios estudiantes [14]. Método y modelo han sido desarrollados en paralelo a uno de los proyectos centrales del grupo en los últimos años, en los que se ha aplicado el aprendizaje por proyectos y casos de estudio apoyado por herramientas CSCL a una asignatura de Arquitectura de Ordenadores de los estudios de Ingeniería de Telecomunicación. Este proyecto, (denominado proyecto LAO por las siglas de “Laboratorio de Arquitectura de Ordenadores”) ha constituido uno de los hitos del grupo EMIC, puesto que en él se ha experimentado una sinergia

verdaderamente positiva entre un conjunto reducido de miembros del mismo procedentes de diferentes disciplinas (Telecomunicaciones, Informática y Pedagogía). Una de las lecciones aprendidas con este proyecto es la conveniencia de que dentro del grupo “grande” se puedan dar configuraciones flexibles, donde unos pocos miembros del mismo, aún procedentes de distintas disciplinas, puedan colaborar de forma operativa en proyectos donde todos los participantes encuentren una motivación fuerte hacia la consecución del objetivo.

A lo largo de esta sección se ha presentado una visión retrospectiva de la vida del grupo, así como de algunos de los proyectos más directamente ligados a la propia historia del mismo. Sin embargo, no todos los proyectos llevados a cabo por los autores han surgido de esta historia, sino que se ha buscado dar respuesta a problemas generales y abiertos dentro del dominio, como la mejora en la reutilización de los desarrollos CSCL y el estudio de cómo ciertas tecnologías novedosas, como la computación grid, pueden apoyar y ampliar las posibilidades del CSCL. Estos aspectos son abordados, respectivamente, en las dos secciones siguientes.

3. Hacia un Marco de Componentes Software

El software educativo se ha enfrentado tradicionalmente a las necesidades de adaptación y personalización requeridas por los educadores para poder ser utilizado en diferentes contextos educativos y sociales e, incluso, con diferentes estilos pedagógicos. Esta necesidad ha conducido generalmente al desarrollo de un gran número de aplicaciones específicas. Éstas suelen ser aplicaciones monolíticas, dependientes de tecnologías particulares, e incompatibles entre sí, por lo que los profesores encuentran grandes dificultades para integrar varias de ellas en sus clases [18]. Estos proyectos poseen un alto índice de fracaso, debido a su falta de capacidad para adaptarse a nuevas situaciones educativas y para asumir las innovaciones tecnológicas que continuamente se producen.

Este problema se hace patente en la propia evolución del grupo EMIC en lo relacionado con el desarrollo de software educativo de apoyo al aprendizaje colaborativo. De hecho, la primera aplicación del grupo, denominada Pencacolas (una aplicación para la escritura colaborativa de documentos), fue desarrollada sin partir de ningún tipo de aplicación o herramienta existente y,

además, sin ninguna pretensión de que, posteriormente, pudiera ser, a su vez, reutilizada o integrada con otras aplicaciones. El resultado final fue una aplicación difícil de mantener y poco robusta. Para la segunda versión de esta herramienta, Pencacolas II, se decidió utilizar las facilidades, en lo referente al soporte a la colaboración, del *toolkit* JSDT (*Java Shared Data Toolkit*) con lo que se consiguió reducir el tiempo de desarrollo y el coste de mantenimiento de la aplicación. Desarrollos posteriores, como ATOIDI (una aplicación colaborativa para la enseñanza de habilidades sociales) o CECI (una herramienta para la elaboración colaborativa de cuentos en inglés), además de seguir haciendo uso de JSDT, se orientaron a la identificación de módulos funcionales que pudieran ser intercambiados para, de esa forma, disponer de aplicaciones adaptables a un mayor número de contextos.

Ese avance hacia la modularidad y la reutilización, junto con la necesidad de disponer de un marco conceptual en el que poder situar las capacidades de todas las aplicaciones desarrolladas hasta entonces, guiaron al grupo EMIC a la definición del ya descrito marco DELFOS. DELFOS supuso un primer intento de sistematizar la identificación de elementos comunes y puntos de integración de herramientas software de soporte a diferentes *situaciones* de aprendizaje.

Sin embargo, a pesar de las contribuciones de DELFOS en el campo conceptual, la consecución de la reutilización, adaptación y flexibilidad de las aplicaciones CSCL dependía todavía de un adecuado soporte tecnológico: de poco sirve, por ejemplo, identificar módulos funcionales comunes a un número elevado de aplicaciones si estos no se pueden reutilizar en el desarrollo de las mismas.

Por ello, el grupo EMIC se planteó, en el año 2000, el empleo de los principios y tecnologías asociadas a la denominada Ingeniería Software Basada en Componentes (ISBC). La ISBC [19] ofrece la promesa de la composición de herramientas procedentes de diversos proveedores y, por tanto, aparece como una solución potencial a los problemas aquí señalados proporcionando la capacidad de reutilización y adaptación de aplicaciones tan necesaria en el dominio educativo, en general, y en el CSCL en particular.

A la hora de tratar el problema de la reutilización en la ISBC es fundamental tener en cuenta el concepto de marco de componentes [16]: un conjunto extensible de componentes software reutilizables en un dominio de aplicación particular junto con una serie de patrones de diseño software que

documentan su utilización. Los componentes contenidos en un marco se pueden reutilizar, particularizar y ensamblar con componentes adicionales proporcionados por los desarrolladores para obtener aplicaciones concretas más rápidamente y con un menor coste.

De hecho, la ISBC ha dado lugar a varios proyectos en los que se ha aplicado con éxito la idea de marco de componentes al desarrollo de aplicaciones educativas [18]. No obstante, la cuestión del soporte a la colaboración, tan inherente al dominio particular del CSCL, no fue considerada en estos proyectos.

Por todo ello, el grupo EMIC inició un proyecto, denominado COSACO (Componentes Software para las Aplicaciones de aprendizaje COLaborativo) destinado a la obtención de un marco de componentes software para el dominio del CSCL.

Las ventajas de un Marco de Componentes software para aplicaciones CSCL no se limita a la ya comentada de facilitar a los desarrolladores software la obtención de aplicaciones reutilizando y particularizando software existente. Supóngase una biblioteca de componentes software suficientemente surtida y una adecuada traducción de conceptos propios del dominio del aprendizaje colaborativo a requisitos funcionales y no funcionales de dichos componentes. En ese caso sería posible disponer de herramientas capaces de permitir a educadores y pedagogos generar, sin conocimientos de ingeniería software, aplicaciones CSCL particularizadas a las necesidades del entorno de aprendizaje para el que van a ser destinadas.

Ahora bien, el desarrollo de un marco de componentes no es una tarea trivial. El desarrollador de un marco debe afrontar diversos problemas relacionados tanto con las particularidades del dominio de interés como con las tecnologías empleadas para dar soporte a los componentes obtenidos [16]. Uno de los problemas más importantes a tener en cuenta en este contexto es la identificación y dimensionamiento (en otras palabras, el nivel de granularidad) de los componentes. La consecución de esta tarea depende, en gran medida, del grado en que los conceptos y principios fundamentales del dominio de interés son comprendidos por los desarrolladores de software [5]. En el dominio CSCL este problema es especialmente difícil de resolver debido a las grandes diferencias existentes entre las abstracciones utilizadas por los expertos en el aprendizaje colaborativo (pedagogos, psicólogos, didactas...) y las propias de los desarrolladores de software.

En ese sentido, el grupo EMIC contaba con la experiencia previa obtenida gracias al desarrollo del marco DELFOS. Se puede decir que DELFOS supone una aproximación arriba-abajo para acercar los dominios del aprendizaje colaborativo y la ingeniería software. Sin embargo, se pudo apreciar que la idea de *situación* de aprendizaje en la que DELFOS basa la definición de aplicaciones CSCL era muy difícil de reutilizar, puesto que, por definición, es altamente dependiente del contexto en el que se va a desarrollar la actividad educativa. Es decir, una vez definida una situación educativa mediante las herramientas de DELFOS, ésta no podría ser aplicada sin grandes cambios a otros entornos educativos [17]. En este sentido, se vio la necesidad de identificar aspectos independientes del contexto, que pudieran adaptarse sin grandes cambios a distintas situaciones educativas.

Ante la falta de resultados de la aproximación arriba-abajo, el grupo EMIC intentó una segunda vía (abajo-arriba) consistente en el desarrollo de aplicaciones concretas CSCL basadas en componentes con el objetivo de extraer elementos significativos del marco. Los autores, consecuentemente, trataron de determinar un tipo de aplicación CSCL cuyo nivel de complejidad fuera lo suficientemente bajo como para permitir el desarrollo rápido de prototipos y, al mismo tiempo, lo suficientemente elevado como para compartir la problemática de las aplicaciones CSCL más elaboradas de forma que los componentes identificados y desarrollados pudieran, potencialmente, ser reutilizados en la mayoría de ellas. La decisión, en este sentido, fue la de desarrollar una aplicación de resolución colaborativa síncrona de puzles (denominada *MagicPuzzle*) destinada a alumnos de educación infantil. Este tipo de aplicaciones destaca por su conocido beneficio educativo y socializador además de por su capacidad de reflejar el proceso de construcción del conocimiento: los diversos aprendices pueden compartir piezas de conocimiento, necesarias para la resolución de un problema más complejo, sin tener necesariamente una única y cerrada solución, buscando al mismo tiempo las justificaciones de sus acciones. Todo este proceso se puede entender como una metáfora que incorpora métodos educativos, tales como los orientados al estudio de casos y por proyectos. Las funcionalidades con las que se dotó a *MagicPuzzle* fueron determinadas de acuerdo con las indicaciones de expertos en educación infantil.

Además de la propia aplicación *MagicPuzzle*, y como parte del proyecto COSACO, se desarrolló una primera versión de una herramienta de *Gestión*

de Componentes. Dicha herramienta tenía como objetivo, en función de los requisitos no técnicos especificados por un didacta (sin conocimientos de ingeniería software), generar una variante de la aplicación *MagicPuzzle* particularizada a dichos requisitos. La generación de la variante particular de la aplicación se llevaba a cabo mediante la adecuada traducción de requisitos no técnicos en características deseables de los componentes software a elegir de los disponibles en una biblioteca de componentes existente. Aunque esta primera versión del *Gestor de Componentes* se limitaba a la generación de aplicaciones tipo *MagicPuzzle*, sirvió como primera aproximación al problema general de cómo ensamblar aplicaciones basadas en componentes a partir del contenido de un marco en función de requisitos del dominio del aprendizaje colaborativo.

Una vez finalizados los prototipos de *MagicPuzzle*, el grupo se planteó hasta qué punto los componentes obtenidos y desarrollados podrían ser empleados como parte del desarrollo de otras aplicaciones CSCL diferentes. Para resolver este interrogante, se acometió el desarrollo de otras dos aplicaciones CSCL: una aplicación colaborativa de apoyo para la asignatura Arquitectura de Ordenadores en los estudios de Ingeniería de Telecomunicación (una vez más empleada como escenario de validación), denominada *eLAO*, con la que se esperaba poder reutilizar los componentes de soporte básico a la colaboración; y una aplicación con la que jugar colaborativamente al conocido *Tangram*, con la que, al ser una aplicación basada en piezas al igual que *MagicPuzzle*, se esperaba alcanzar mayor grado de reutilización.

Sin embargo, en las primeras iteraciones del desarrollo de estas dos nuevas aplicaciones se pudo comprobar que la reutilización no iba a ser tan elevada como se esperaba. Las razones hay que buscarlas en el hecho de que el desarrollo de *MagicPuzzle* estuvo muy sesgado hacia el problema de aprendizaje concreto relacionado con la resolución colaborativa de puzles. Este problema, muy documentado en el campo de la ISBC [5], junto con la falta de formalización de los conceptos del dominio de interés, dificulta la identificación y obtención de componentes reutilizables.

La búsqueda de características de aplicaciones CSCL que fuesen generalizables a diversos contextos y que facilitarían la identificación de componentes reutilizables hizo que los autores se

plantearan una tercera aproximación, basada en la definición de los denominados Patrones de Aprendizaje Colaborativo (*Collaborative Learning Pattern, CLP*).

Un CLP [8] se puede entender como un mecanismo de descripción de una *técnica de aprendizaje colaborativo* fácilmente entendible por desarrolladores de software. Estas técnicas de aprendizaje colaborativo hacen referencia a formas comunes de estructurar las interacciones entre los participantes en diferentes actividades de aprendizaje colaborativo, así como la información que se intercambian y los objetos que manipulan. La idea de CLP es una extensión del concepto de “Patrón de Diseño Colaborativo” descrito en [6].

Los didactas del aprendizaje colaborativo (profesores, fundamentalmente) son los encargados de identificar y definir CLP’s mientras que los pedagogos son los encargados de su validación. Los CLP’s están destinados a ser empleados por los desarrolladores de software para obtener requisitos comunes de las aplicaciones CSCL que han de dar soporte a actividades de aprendizaje colaborativo que se basan en una misma técnica. Gracias a estos requisitos comunes, se simplifica la tarea de identificar componentes software de un nivel de granularidad adecuado para facilitar su reutilización. No obstante, es importante destacar que los CLP’s no contienen ningún tipo de información técnica.

Desde una perspectiva más relacionada con la riqueza conceptual, los CLP’s se puede considerar como una aproximación intermedia (ver Figura 1) con respecto a las dos descritas con anterioridad: no parten de aplicaciones concretas pero tampoco intentan abarcar todos los conceptos y principios del campo del aprendizaje colaborativo (como intentaba DELFOS).

El modo de empleo de los CLP’s depende de la metodología de desarrollo software que se utilice. A modo de ejemplo, si se utiliza una metodología de desarrollo basada en el ampliamente aceptado Proceso Unificado (UP, *Unified Process*), la información proporcionada por los CLP’s se podría utilizar como base para la identificación de los actores y casos de uso, el modelo conceptual (también conocido como modelo de dominio) y el análisis de los casos de uso durante las iteraciones de la denominada “Fase de Arranque” (*Inception Phase*).

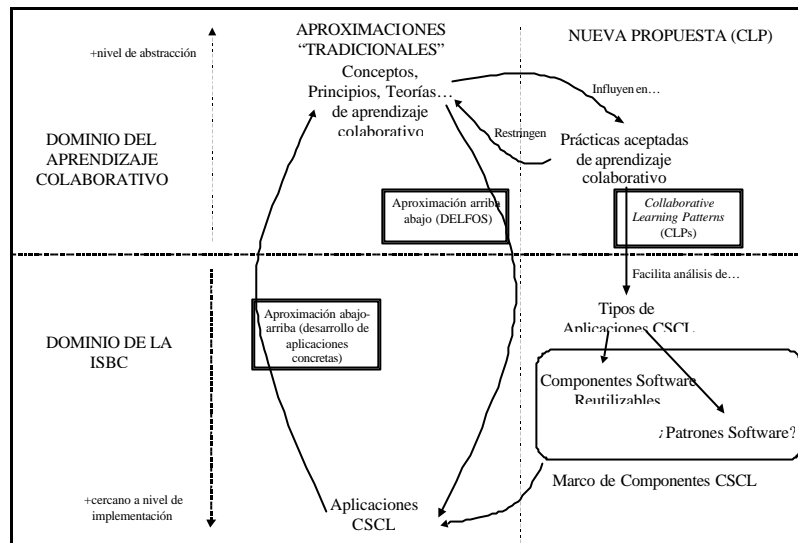


Figura 1 Tres aproximaciones para acercar los dominios del aprendizaje colaborativo y la ISBC

Tras completar la Fase de Arranque UP utilizando la información proporcionada por los CLP's, y empleando técnicas habituales de desarrollo software prescritas, en este ejemplo, por UP, es posible obtener el diseño arquitectónico para una aplicación del tipo indicado por el CLP. Por supuesto, los autores reconocen que es difícil prescribir una forma única de llegar a un diseño concreto partiendo de la definición de un CLP. Los CLP's constituyen una ayuda al desarrollo software pero no pretenden proporcionar una metodología de desarrollo completa.

En lo referente a la reutilización del software, las implicaciones de esta aproximación son muy importantes: el uso de CLP's ayuda a los desarrolladores a entender los requisitos y los conceptos involucrados en el soporte de las diferentes técnicas CSCL. De esa manera se facilita la identificación de componentes software comunes para aplicaciones CSCL que se basen en las mismas técnicas. Estos componentes comunes son, potencialmente, más reutilizables que los obtenidos del desarrollo de una aplicación CSCL particular.

Los CLP's han sido aplicados con éxito en el desarrollo de la aplicación *eLAO*, ya mencionada con anterioridad. La reutilización de los componentes software de *eLAO* se está evaluando mediante el desarrollo de nuevas aplicaciones CSCL basadas en los mismos CLP's.

A pesar de estos intentos por avanzar en la identificación y dimensionamiento de componentes software CSCL, es importante no perder de vista

tampoco la aparición de nuevas posibilidades tecnológicas que permitan aumentar la escalabilidad y rendimiento de las aplicaciones CSCL basadas en componentes obtenidas a partir de los contenidos del tan deseado marco CSCL. En este sentido, el grupo EMIC se ha involucrado, en el último año, en un proyecto de investigación destinado a evaluar la potencial aplicabilidad de las tecnologías de computación grid como infraestructura sobre la que desplegar los componentes software CSCL obtenidos en el ámbito del proyecto COSACO. La siguiente sección se centra en esta nueva vía de investigación.

4. La Tecnología *Grid* como Apoyo al CSCL

El *grid* computacional [2] es una infraestructura hardware y software que se caracteriza por ser de gran escala, geográficamente distribuida, y compuesta por recursos heterogéneos pertenecientes a múltiples organizaciones administrativas que los comparten con el objetivo de proporcionar soporte computacional a un amplio rango de aplicaciones de forma transparente, de calidad, ubicua y consistente [4]. A pesar de que el *grid* fue inicialmente concebido para dar soporte a aplicaciones de supercomputación pertenecientes a un ámbito estrictamente científico, el proceso inicial de maduración de esta tecnología demostró pronto la utilidad de la misma en aplicaciones de computación de alto rendimiento, de computación bajo demanda, de computación intensiva de datos y de computación colaborativa [10]. Sin embargo, el uso del soporte *grid* continuó restringido al dominio de las aplicaciones científicas.

En la actualidad, la tecnología *grid* se encuentra en una nueva fase de maduración tras la reciente definición de Arquitectura Abierta de Servicios Grid (OGSA – *Open Grid Services Architecture*) [11]. En este nuevo proceso de maduración particularmente intenso, OGSA supone un fuerte impulso para sacar a la tecnología *grid* del ámbito estrictamente científico y extenderla a otros campos como el empresarial o el educativo. Este hecho sugiere el interés del trabajo, que actualmente está siendo desarrollado por el grupo EMIC, de exploración de las posibilidades de uso del *grid* para dar soporte a aplicaciones CSCL. Dicha exploración trata de establecer la *utilidad* que el soporte *grid* puede proporcionar a las aplicaciones CSCL, la *adecuación* del *grid* como infraestructura para las aplicaciones CSCL así como los principales *puntos de investigación* que deberán ser abordados con el objetivo de hacer realidad el uso del *grid* en el dominio CSCL. Las principales ideas resultantes de este trabajo se resumen en la presente sección.

El *grid* es una infraestructura que puede dar un soporte de gran *utilidad* para las aplicaciones CSCL facilitando su despliegue y potenciando su rendimiento. Entre estos beneficios es posible destacar los siguientes. En primer lugar, la gran escala del *grid* podría permitir la participación en la aplicación CSCL de un elevado número de participantes individuales o en grupo. También la naturaleza heterogénea de los recursos que son compartidos en el *grid* podría permitir a los usuarios participar empleando no sólo ordenadores, sino también otros dispositivos como PDAs, videocámaras o pizarras electrónicas además de recursos software como simuladores. Además, la amplia distribución geográfica de los recursos que forman parte del *grid* facilitaría la colaboración entre usuarios muy distantes entre sí. Por otra parte, el acceso de calidad proporcionado por la infraestructura *grid* a través de un servicio garantizado o de mayor esfuerzo puede mejorar el rendimiento de la aplicación cuando ésta es desplegada *ad hoc*. Este último aspecto es especialmente importante para el dominio CSCL dado que el éxito de la colaboración entre usuarios, y por tanto del aprendizaje como objetivo último, sólo es posible si el rendimiento de la aplicación es bueno. Así, por ejemplo, en el caso de un puzzle que ha de ser resuelto por niños de forma colaborativa, dicha colaboración será imposible si el rendimiento de la aplicación no es lo suficientemente bueno como para distribuir rápidamente a todos los usuarios los cambios que se vayan produciendo en el puzzle.

La existencia de una sinergia entre el *grid* y CSCL a través de la ingeniería de software basada en componentes (ISBC) apunta la *adecuación* del *grid* como infraestructura para las aplicaciones CSCL. Por una parte, la ISBC permite el desarrollo de aplicaciones CSCL reutilizables, configurables y adaptables al contexto en el que se produce el aprendizaje [9]. Por otra, el estándar OGSA especifica los requisitos que deben cumplir las interfaces de los servicios a través de los cuales se debe acceder a las aplicaciones en el *grid*, pero no determina la forma en que se deben implementar dichos servicios [11]. De hecho, la versión 3 del Globus Toolkit (GT3) [1], el *middleware* estándar *de facto* para la construcción de *grids*, proporciona herramientas que permiten exponer las aplicaciones basadas en componentes como servicios *grid*. De este modo, es posible afirmar que OGSA *permite* el uso de la ISBC para el desarrollo de servicios *grid*, mientras que Globus *facilita* la tarea del despliegue de aplicaciones basadas en componentes sobre el *grid*.

Para hacer realidad el uso de infraestructuras *grid* en el soporte de aplicaciones CSCL es necesario abordar una serie de *puntos de investigación*. El primero de ellos es la *definición de un entorno de uso de aplicaciones CSCL sobre un grid* en el que se especifique claramente el papel que juega cada uno de los actores involucrados (profesores, alumnos, administradores, desarrolladores, proveedores de servicio, etc.), cómo se relacionan con el *grid* y cómo interactúan de forma directa o indirecta. Este entorno debería guiar servir de referencia para el proceso de investigación y ser actualizado constantemente durante el mismo. En la actualidad, el grupo EMIC se encuentra trabajando en la definición de un entorno de estas características denominado GRIDCOLE (*Grid-enabled Collaborative Learning Environment*).

Otro punto de investigación es la *integración de aplicaciones CSCL a partir de servicios grid*. La especificación de OGSA precisa que las aplicaciones deben ser ofrecidas como servicios *grid*. Por este motivo es necesario establecer con claridad el paso de un modelo de distribución basado en componentes a un modelo de distribución basado en servicios. Este estudio debería definir tanto la granularidad de los servicios *grid* en los que es posible descomponer las aplicaciones CSCL como el grado de acoplamiento entre dichos servicios.

También sería necesario llevar a cabo la *identificación de servicios grid comunes al dominio CSCL*. De este modo, sería posible evaluar la posibilidad de reutilización y adaptación de

servicios grid ya existentes para otros dominios. Este es el caso de la mensajería, un servicio imprescindible para las aplicaciones CSCL, especialmente las síncronas, para el que se podrían adaptar implementaciones ya desarrolladas para grid como NaradaBrokering [12]. En cambio, otros servicios exclusivos del dominio CSCL o cuya implementación dependa del dominio deberán ser desarrollados.

La *planificación de recursos* es precisamente un servicio dependiente del dominio de aplicación que también debe ser investigado. Para aprovechar el potencial que ofrece el grid para aplicaciones CSCL, es necesario disponer de un planificador de recursos. Éste es el encargado de decidir qué recursos del grid va a utilizar la aplicación de forma que se cumplan unos criterios de calidad de servicio. Un primer acercamiento a este problema se encuentra en [3], donde el grupo EMIC propone un modelo genérico de planificador de recursos grid para aplicaciones CSCL basadas en componentes.

Finalmente, un problema relacionado con la planificación de recursos, que también debe ser estudiado, es el *despliegue ad hoc de servicios CSCL* en el grid. Este tipo de despliegue permitiría al planificador la instalación de los componentes que conforman la aplicación CSCL en los recursos del grid que considere más adecuados para proporcionar el nivel de calidad de servicio comprometido.

5 Conclusiones y Trabajo Futuro

Este artículo ha presentado el trabajo realizado dentro del grupo interdisciplinar EMIC para tratar de resolver diversas preguntas de investigación relacionadas con el dominio del CSCL: cómo obtener un lenguaje común entre educadores y tecnólogos que facilite el diseño participativo de las aplicaciones; cómo conseguir que los desarrolladores de software comprendan los principales conceptos y principios del campo del aprendizaje colaborativo; cómo utilizar los principios y conceptos de la ISBC en el desarrollo de aplicaciones CSCL reutilizables, flexibles y particularizables; cómo conseguir aplicaciones CSCL de gran escala y con mejor rendimiento empleando las tecnologías emergentes de computación grid; y cómo disponer de una metodología de evaluación del proceso de aprendizaje colaborativo, junto con su correspondiente apoyo computacional, para poder evaluar el impacto real de las soluciones propuestas.

En suma, una mirada de cuestiones que involucran el trabajo de investigadores procedentes de diversas

disciplinas y que, además, se enfrentan a la complejidad propia del campo de estudio: múltiples actores en la fase de diseño de soluciones tecnológicas, múltiples actores en la fase de uso de las mismas, múltiples situaciones de uso, múltiples teorías sobre el aprendizaje colaborativo, etc.

En este contexto, el grupo EMIC ha conseguido notables avances: la introducción de los principios del aprendizaje colaborativo en la práctica educativa de sus integrantes; el marco DELFOS como intermediario entre tecnólogos y educadores; la propuesta de una metodología de evaluación formativa de aspectos participativos en entornos CSCL; y el uso de Patrones de Aprendizaje Colaborativo para la identificación y dimensionamiento de componentes software CSCL reutilizables y adaptables.

Sin embargo, no todo han sido aciertos en este recorrido de casi una década: desarrollo de aplicaciones CSCL monolíticas difíciles de mantener y reutilizar; desarrollo de aplicaciones CSCL basadas en componentes muy sesgadas a situaciones de aprendizaje particulares; dificultad en la aplicación del marco DELFOS al proceso de obtención de componentes CSCL reutilizables, etc.

Estos problemas, junto con los objetivos propios de los nuevos proyectos en los que el grupo se ha involucrado, marcan la definición de las futuras líneas de investigación de EMIC: combinación de DELFOS y la aproximación basada en el uso de CLP's; validación de la aproximación basada en CLP's mediante el desarrollo de nuevas aplicaciones CSCL basadas en componentes; incremento de los contenidos del marco CSCL perseguido por el proyecto COSACO (componentes software y patrones de diseño); validación del marco mediante su utilización por terceros; y establecimiento de una infraestructura de computación grid sobre la que desplegar aplicaciones CSCL y poder así validar y cuantificar las ventajas de esta nueva tecnologías.

Agradecimientos

Los autores quieren expresar su agradecimiento a los demás miembros del grupo interdisciplinar EMIC (Educación, Medios, Informática y Cultura) de la Universidad de Valladolid, y de forma muy particular a Teresa Blasco, Rocio Anguita, Lino Barrio, Bartolomé Rubia, Ivan Jorrín (Facultad de Educación), Francisco J. Álvarez, Marta Heredia, Davinia Hernández, Sonia Díez (ETS de Ingenieros de Telecomunicación), Pablo de la Fuente (ETS de Ingeniería Informática), Pablo Orozco (UNAM), César Osuna (Instituto Mexicano del Petróleo). Especiales agradecimientos a los educadores y

alumnos que hacen posible la puesta en marcha y evaluación de todos estos sistemas tecnológicos. Financiación parcial del trabajo descrito en este artículo proviene del Ministerio de Ciencia y Tecnología (proyectos TIC-2000-1054 y TIC-2002-04258-C3-02) y de la Junta de Castilla y León (VA 117/01)

Referencias

- [1] The Globus project. <http://www.globus.org>. 2003.
- [2] Berman, F., Fox, G., and Hey, A. *Grid computing: making the global infrastructure a reality*, Chichester, United Kingdom: John Wiley & Sons, 2003.
- [3] Bote-Lorenzo, M. L., Dimitriadis, Y., Gómez-Sánchez, E., Asensio-Pérez, J. I., Vaquero-González, L. M., and Vega-Gorgojo, G., "Grid resources scheduler for component-based CSCL applications (in spanish)", *Proceedings of the JITEL 2003*, Las Palmas de Gran Canaria, Spain, Sept. 2003.
- [4] Bote-Lorenzo, M. L., Dimitriadis, Y. A., and Gómez-Sánchez, E., "Grid characteristics and uses: a grid definition", *Proceedings of the 1st European Across Grids Conference (CD)*, Santiago de Compostela, Spain, 2003.
- [5] Carey, J. and Carlson, B., Lessons learned becoming a framework developer *Software Practice and Experience*, vol. 43, pp. 789-800, 2002.
- [6] DiGiano, C., Yarnall, L., Patton, C., Roschelle, J., Tatar, D., and Manley, M., "Collaboration design patterns: conceptual tools for planning for the wireless classroom", *Proceedings of the IEEE International Workshop on Wireless and Mobile Technologies in Education (WMTE'02)*, 2002.
- [7] Dillenbourg, P. What do you mean by "Collaborative Learning"? In: *Collaborative Learning, Cognitive and Computational Approaches*, ed. Dillenbourg, P. Elsevier Science Ltd., 1999.
- [8] Dimitriadis, Y., Asensio, J. I., Martínez, A., and Osuna, C., Component Based Software Engineering and CSCL: Component identification and dimensioning *Upgrade (digital journal of European Professional Informatics Societies)*, special issue on e-learning: Boarderless education, vol. 4, no. 5, pp. 21-28, Oct., 2003.
- [9] Dimitriadis, Y. A., Asensio, J. I., Toquero, J., Estébanez, L., Martín, T. A., and Martínez, A., "Hacia un Sistema de Componentes Software para el Dominio del Aprendizaje Colaborativo Apoyado por Ordenador (CSCL)", *Proceedings of the Symposium on Informatics and Telecommunications*, Sevilla, Spain, 2002.
- [10] Foster, I. Computational grids. In: *The Grid: blueprint for a future computing infrastructure*, eds. Foster, I. and Kesselman, C. San Francisco, CA, USA: Morgan Kaufmann Publishers, 1998. pp. 15-52.
- [11] Foster, I., Kesselman, C., Nick, J. M., and Tuecke, S., The physiology of the Grid: an Open Grid Services Architecture for distributed systems integration *Technical Report, Open Grid Service Infrastructure WG, Global Grid Forum*, June, 2002. Notes: Draft
- [12] Fox, G. and Pallickara, S. NaradaBrokering: an event-based infrastructure for building scalable and durable peer-to-peer Grids. In: *Grid computing: making the global infrastructure a reality*, eds. Berman, F., Fox, G., and Hey, A. Chichester, United Kingdom: John Wiley & Sons, 2003. pp. 579-601.
- [13] Koschmann, T. *CSCL: theory and practice of an emerging paradigm*, Malwah, NJ, USA: Lawrence Erlbaum, 1996.
- [14] Martínez, A., Dimitriadis, Y., and de la Fuente, P., "An XML-based representation of collaborative interactions", *Proceedings of the International Conference on Computer Supported Collaborative Learning CSCL 2003*, Bergen, Noruega.
- [15] Martínez, A., Dimitriadis, Y., Gómez, E., Rubia, B., and de la Fuente, P., Combining qualitative and social network analysis for the study of classroom social interactions *Computers and Education, special issue on Documenting Collaborative Interactions: Issues and Approaches*, vol. 41, no. 4, pp. 353-368, 2003.
- [16] Mili, H., Fayad, M., Brugali, D., Hamu, D., and Dori, D., Enterprise frameworks: issues and research directions *Software Practice and Experience*, vol. 32, pp. 801-831, 2002.
- [17] Osuna, C. and Dimitriadis, Y., "A framework for the development of educational collaborative applications based on social constructivism", *Proceedings of the CYTED RITOS International Workshop on Groupware (CRIWG'99)*, 1999.
- [18] Roschelle, J., Kaput, J., Stroup, W., and Kahn, T. M., Scalable integration of educational software: exploring the promise of component architectures *Journal of Interactive Media in Education*, vol. 98, no. 6, Oct., 1998.
- [19] Szypersky, C. *Component software. Beyond object-oriented programming*, NY, USA: Addison Wesley, 1998.
- [20] Vaquero, A., Las TIC para la enseñanza, formación y el aprendizaje *Novática*, vol. 132, pp. 4-14, Mar., 1998.