

Aspectos esenciales de condicionales para los contratos sensibles al contexto

Alejandro Sartorio ^{1,2}
Guillermo Rodríguez ^{1,2}
Marcelo Vaquero ²

¹ Centro Internacional Franco Argentino de Ciencias de la Información y de Sistemas, CIFASIS (CONICET-UNR-UPCAM), Bv. 27 de febrero 210 bis, 2000 Rosario, Argentina

² Centro de Altos Estudios en Tecnología Informática, CAETI, Universidad Abierta Interamericana, Sede Rosario, Ov. Lagos 944, 2000 Rosario, Argentina

sartorio@cifasis-conicet.gov.ar, guille@fceia.unr.edu.ar, marcelo.vaquero@vaneduc.edu.ar

Resumen. Teniendo en cuenta los aspectos tecnológicos y de diseño de los mecanismos de inyección de contratos sensibles al contexto en el Dispositivo Hipermedial Dinámico (DHD), se desarrollan tres tipos de condicionales para las reglas de los contratos que representan información de contexto. A su vez se presenta un modelo conceptual de condicionales donde se capturan aspectos esenciales para el *framework* colaborativo Sakai utilizado para la implementación tecnológica del DHD en el agregado de nuevas propiedades de adaptación y dinamismo.

Palabras Clave: Dispositivo Hipermedial Dinámico – Coordinación de Contratos – Sistemas sensibles al contexto – TIC.

1 Introducción

Se conceptualiza como Dispositivo Hipermedial Dinámico (DHD) [1] a la red heterogénea conformada por la conjunción de tecnologías y aspectos sociales que posibilitan a los sujetos realizar acciones en interacción responsable con el otro para investigar, aprender, dialogar, confrontar, componer, evaluar, bajo la modalidad de taller físico-virtual, utilizando la potencialidad comunicacional, transformadora y abierta de lo hipermedial, regulados según el caso, por una “coordinación de contratos” [2].

En la implementación y optimización del DHD para la producción y disseminación de conocimiento se utilizan diferentes mecanismos de mediciones que condicionarán el uso y configuración de los servicios del DHD. En este trabajo se entiende como servicio a las implementaciones que brindan funcionalidades a una

herramienta. Por ejemplo, el servicio *editar* de la herramienta *foro*.

Para lograr este uso y reconfiguración el DHD está constituido con el agregado de una pieza de software que posibilita la inyección de coordinación de contratos sensibles al contexto. Esta propiedad se logra a través de la implementación de contratos [3] con mecanismos de coordinación y componentes de sistemas sensibles al contexto [4].

La utilización de reglas es parte esencial en la implementación de las acciones de los contratos y las tareas de coordinación. A su vez, las reglas están compuestas por condicionales donde se centra parte de los mecanismos de adaptación que se requiere en los DHD. Algunos de los condicionales implementados requieren de mecanismos externos que colaboren en la composición de sus valores de verdad [5].

En este trabajo se definen y analizan tres tipos de condicionales como piezas de primera clase, brindando la posibilidad de construir sus valores de verdad a partir de resultados obtenidos por sistemas externos. Esto hace que desde el punto de vista de la definición, se obtenga un nuevo grado de libertad en la representación e implementación de los DHD para la adaptabilidad al contexto. A su vez, este análisis determina cuatro aspectos esenciales que permiten establecer un diseño conceptual de los condicionales para los contratos sensibles al contexto del DHD.

Tras esta introducción, en la sección 2 se identifican los elementos tecnológicos del DHD teniendo en cuenta su relación con los condicionales. Luego, en las secciones 3, 4 y 5, se presentan los tres tipos de condicionales con sus principales características y modelos de integración dentro del *framework* Sakai (para la justificación de dicha elección ver [6]). Por último, se presentan las principales conclusiones y consideraciones generales.

2. Aspecto tecnológico del DHD

En esta sección se describen aspectos tecnológicos y componentes del DHD que intervienen en la integración con el *framework* Sakai (www.sakaiproject.org). Como resultado de esto, se lo dota al *framework* de un mecanismo de adaptación dinámica mediante la utilización de contratos orientados a la implementación de servicios sensibles al contexto.

El uso de contratos parte de la noción de Programación por Contrato (*Programming by Contract*) de Meyer [3] basada en la metáfora de que un elemento de un sistema de *software* colabora con otro, manteniendo obligaciones y beneficios mutuos. En nuestro dominio de aplicación consideraremos que un objeto cliente y un objeto servidor “acuerdan” a través de un contrato, representado con un nuevo objeto, que el objeto servidor satisfaga el pedido del cliente, y al mismo tiempo que el cliente cumpla con las condiciones impuestas por el proveedor. De esta manera, las decisiones de comportamiento de los servicios se verán influenciadas por el valor de verdad de las instancias de los condicionales.

Como ejemplo de la aplicación de la idea de Meyer en nuestro dominio de sistemas DHD planteamos la situación en la que el pedido de un usuario del sistema ejecuta un comando a través de la interfaz de una herramienta (representado por un objeto cliente) utiliza un servicio de edición de mensajes (representado por un objeto

servidor) a través de un contrato que garantizará las siguientes condiciones: el usuario debe poder editar aquellos mensajes que tiene autorización según su perfil (obligación del proveedor y beneficio del cliente); el proveedor debe tener acceso a la información del perfil del usuario (obligación del cliente y beneficio del proveedor).

A partir de la conceptualización de contratos se propone una extensión por medio del agregado de nuevas componentes para instrumentar mecanismos que permitan ejecutar acciones dependiendo del contexto. En aplicaciones sensibles al contexto [4], el contexto (o información de contexto) es definido como la información que puede ser usada para caracterizar la situación de una entidad más allá de los atributos que la definen. En nuestro caso, una entidad es un usuario (alumno, docente, etc.), lugar (aula, biblioteca, sala de consulta, etc.), recurso (impresora, fax, etc.), u objeto (examen, trabajo práctico, etc.) que se comunica con otra entidad a través del contrato.

En este trabajo consideramos una especificación del concepto de contexto [1] partiendo de las nociones de Dourish [7] y adaptadas al dominio *e-learning*, que será la que consideraremos en este trabajo. Entonces, contexto es todo tipo de información que pueda ser sensada y procesada, a través de la aplicación *e-learning*, que caracteriza a un usuario o entorno. Por ejemplo: intervenciones en los foros, promedios de notas, habilidades, niveles de conocimientos, máquinas (direcciones ip) conectadas, nivel de intervención en los foros, cantidad de usuarios conectados, fechas y horarios, estadísticas sobre cursos, etc.

A través de un diagrama UML [8] se definen las clases utilizadas en la implementación de los condicionales dentro de las reglas de los contratos. La figura 1 describe los elementos y relaciones relevantes en la creación de condicionales.

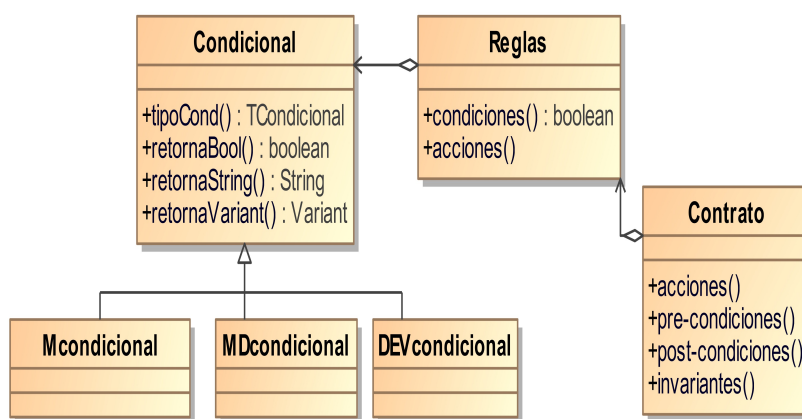


Fig. 1. Elementos y relaciones relevantes en la creación de los condicionales.

Dentro del mecanismo de tratamiento de los condicionales, el contrato presenta en su interfaz métodos para la configuración de acciones y reglas; además de las que cubren las propiedades inherentes a la definición original de contrato sobre la configuración de *pre-condiciones*, *post-condiciones* e *invariantes*. De esta manera se representa a las reglas anteriormente mencionadas como una clase de agregación del

contrato. Del mismo modo se determina una nueva clase para la representación de los condicionales como relación referencial de las reglas. Las reglas contienen referencias a las acciones de los contratos por medio de la interfaz *acciones*.

Se representa a los condicionales como objetos de primera clase con el propósito de establecer un nuevo grado de abstracción que permite conectar a los contratos a subsistemas externos que le proporcionen nuevos mecanismos de adaptabilidad, dinamismo e interpretación. De esta manera, teniendo en cuenta las experiencias de diseño e implementación del uso de condicionales [9] se extienden al objeto condicional en tres tipos diferentes. Cada uno de ellos hereda las interfaces de *condicional*, encargada de establecer las presentaciones para el tipo de dato necesario en las reglas.

El primer paso, es lograr la construcción de las reglas del contrato y que los condicionales representen criterios de decisión sobre aspectos relevantes de los procesos didácticos, investigativos, de producción y/o de gestión; por ejemplo, un estudiante puede adquirir un servicio determinado de una herramienta, previo a la evaluación de una condición representada como condicional de una regla.

En general, a estas reglas debemos diseñarlas con el cuidado de no incorporar redundancias, ambigüedades o incoherencias; tanto entre las propias reglas de un contrato como con otras reglas implícitas que se desprenden de los servicios. Entonces, definimos a las reglas del contrato como un conjunto de condiciones, acciones y prioridades.

De esta manera, las reglas forman parte de un mecanismo de agregación encargado de la composición de diferentes tipos de condicionales: Mcondicionales, MDcondicionales y DEVScondicionales, que se comportan de manera similar teniendo en cuenta diversos modelos de integración que explicitaremos en las secciones siguientes.

A través de los elementos utilizados en el diseño de la figura para la creación de condicionales se identifica como un primer aspecto esencial (denominado por AE #1) a la estructura que define los diferentes tipos de condicionales. Esto quiere decir que para cualquier diseño o implementación de condicionales en el DHD se sugiere que se mantenga esta estructura.

De esta manera se comienza con las descripciones de los tres tipos de condicionales de los cuales se extraerán nuevos aspectos esenciales. Por cuestiones de espacio se brindará una breve introducción que ayude a interpretar el propósito funcional de cada uno de ellos. También se muestra un diseño representativo que identifica los módulos y sus correspondientes métodos a tener en cuenta en la implementación.

3. Los Mcondicionales

Al incorporar las métricas en el modelo original de contratos sensible al contexto, perseguimos que el sistema sea más adaptable a los cambios del contexto de los usuarios. Esto implica que las reglas que se puedan establecer para el contrato se formulan en el marco de la complejidad de los procesos anteriormente enunciado (educación, investigación, etc.) entendiendo que los mismos siempre exceden la

posibilidad de ser absolutamente expresados por un conjunto de reglas. Por lo tanto, las mencionadas reglas estarán singularizadas en función de la dinámica de dichos procesos debiendo ser explícitas, de clara implementación y adaptables para los contextos específicos de los usuarios.

En esta sección utilizaremos un modelo particular de métrica y una propuesta de integración de dicho modelo al *framework* Sakai a través de los contratos. Para implementar la invocación de métricas mediante métodos correctos, propusimos desde la perspectiva del rediseño e implementación computacional, un modelo de integración de muy bajo costo, sin cambios sustanciales en la arquitectura original y en el código de la implementación.

El modelo conceptual de métrica está basado en el Modelo INCAMI (*Information Need, Concept model, Attribute, Metric and Indicator*: Información relevante, Modelo Conceptual, Atributos, Métricas e Indicadores) [10]. INCAMI es un *framework* organizacional, orientado a la medición y evaluación que permite economizar consistentemente valores mensurables en contextos físicos.

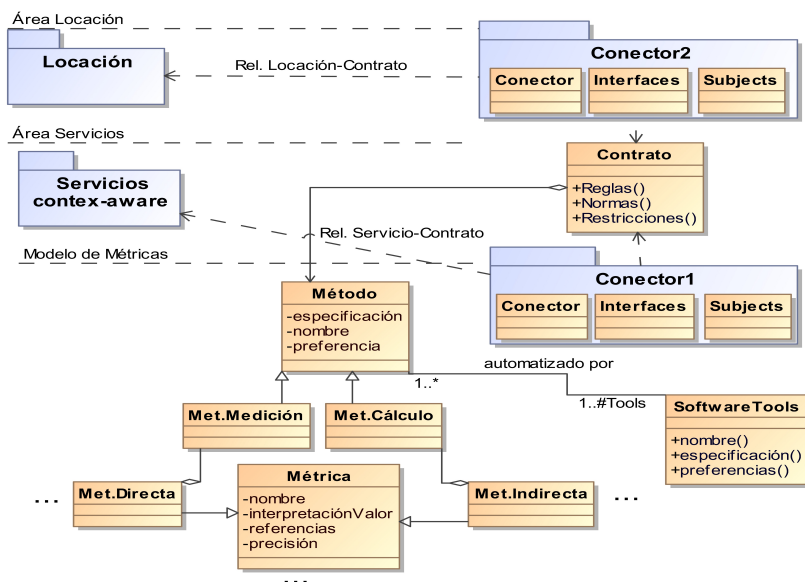


Fig. 2. Modelo de integración para los Mcondicionales.

A través del diagrama de la Figura 2 se representa un modelo global de integración. La misma contempla el modelo de coordinación de contrato y el modelo de métrica propuesto por Olsina et al. [11].

La figura también nos muestra, que la principal componente para lograr la integración está representada por la incorporación de una relación de agregación entre la componente *Contrato*, representado por una clase en el modelo UML y la entidad *Método*. Los condicionales de las reglas de los contratos son invocados (mediante un método explícito relacionado con la noción de los Mcondicionales, por ejemplo, *getForum_theme*) por medio de un mecanismo de "callback" que permite la correcta

invocación de la métrica.

El modelo de métrica proporciona una implementación y diseño de adecuadas métricas para suplir los requerimientos de interacción referidos al registro de actividades [1]. Si tenemos en cuenta el proceso de definición de métricas debemos crear métricas por cada atributo. Cabe aclarar que este proceso de confección de métricas posee una metodología propia con fines específicos de los procesos evaluativos.

Como reflexión sobre la implementación de un modelo de métrica a través de los contratos sensibles al contexto, podemos observar que en el *framework* de investigación y desarrollo, la incorporación de un modelo de métrica, en este caso INCAMI, enriquece la captura de información de contexto; una tarea tan dificultosa y esencial en el campo de las aplicaciones sensibles al contexto.

Siguiendo con el lineamiento sobre la detección de los aspectos esenciales, en este caso se identifica a la conexión que se establece entre subsistema de métrica y el contrato. La conexión está a cargo del objeto *método* de la métrica, esto implica que se tenga que efectuar modificaciones dentro de un subsistema que se trata como externo para los condicionales.

De esta manera surge la necesidad de contemplar la posibilidad de abstraer este tipo de conexiones por medio de una clase de conexión. A esta nueva pieza y su correspondiente estructura que la contiene la definimos como un segundo aspecto esencial (denominada por AE #2).

4. MDcondicionales

Las técnicas de clasificación y agrupamiento o *clustering* [12] consisten en la habilidad intelectual para ordenar o dividir fenómenos complejos (descritos por conjuntos de objetos con datos altamente dimensionales) en pequeños y comprensibles unidades o clases que permiten un mejor control o comprensión de la información. Su aplicación a sistemas *e-learning* [13] [14] [15] [16] permite agrupar a los usuarios por su comportamiento de navegación, agrupar a las páginas por su contenido, tipo o acceso, y agrupar similares comportamientos de navegación.

Las reglas de clasificación tienen como objetivo almacenar conocimiento encaminado a la construcción de un clasificador preciso. En su antecedente contienen una serie de requisitos (en forma de condiciones) que debe cumplir un objeto determinado para que pueda considerarse perteneciente a la clase identificada con el consecuente de la regla. Desde el punto de vista sintáctico, la principal diferencia con las reglas de asociación es que presentan una sola condición en el consecuente, que además pertenece a un identificador de clase.

La estructura de una regla puede ser caracterizada (en términos generales) mediante un conjunto de condicionales lógicos en donde se fija un criterio de ejecución de la acción. Interesa destacar la estructura de la regla Si-Entonces (*If-Then*) y los condicionales de los cuales se desprende la ejecución de un algoritmo de minería de datos. De esta manera, por medio de los MDcondicionales, se capturan los resultados de la inferencia de un árbol de decisión.

Una de las formas más simples de representar conocimiento es mediante árboles

de decisión. El algoritmo de construcción de árboles de decisión más popular es el algoritmo ID3 [17]. Este algoritmo está basado en la división sucesiva, y construye el árbol de decisión desde la raíz hacia las hojas, incrementando en cada paso la complejidad del árbol.

Para lograr la incorporación de la componente *Data Mining*, volvemos a proponer una integración de muy bajo costo. En la figura 3, al contrato lo representamos con un diagrama de clase, y a su vez, las normas, reglas y restricciones están representadas con tres métodos independientes en el diagrama de la clase contrato.

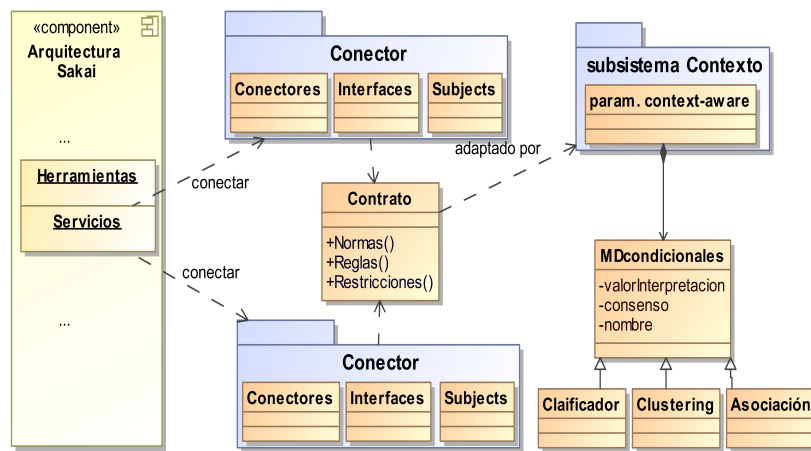


Fig. 3. Modelo de integración para los MD condicionales.

Teniendo en cuenta la analogía planteada entre las reglas del contrato y la estructura de los árboles es posible plantear un nuevo mecanismo total o parcial de construcción de reglas dentro de los contratos.

Cuando, dentro en una regla se utiliza un método para implementar algunas de las técnicas de *minería de datos*, queda conformada una regla donde el valor de su condicional (el cual fue denominado MD condicionales) responde conceptualmente a una estructura, en este caso: un árbol. El vínculo entre las reglas del contrato y la efectiva representación del mismo se concreta a través de una relación de agregación entre los parámetros del contrato (representado por la clase parámetros *context-aware*), que se encuentra dentro del subsistema que engloba al contexto, similar a la propuesta de Schmidt [18] y la clase MD condicionales (representada como una generalización de subclases donde se distinguen las posibles técnicas de minería de datos).

Volviendo a la figura 3, se muestra las componentes significativas para la integración de los modelos. La primera, pertenece al área de coordinación de contrato y mantiene la estructura original del modelo propuesto. La segunda componente, se encuentra representada por una clase conceptual que simboliza las características y funcionalidades de minería de datos, articuladas para cumplir los objetivos y

requerimientos propuestos.

En resumen, la figura es una reelaboración de integración arquitectónica entre dos modelos (parte del *framework* Sakai y el modelo de Minería de Datos) donde se muestran, además, relaciones salientes de dependencias entre clases de las distintas áreas. Por tanto, el análisis sobre el diseño de implementación de la figura 3 persigue dos enfoques relacionados. Por un lado se identifica como una componente de conexión a la denominada *parámetros context-aware*, este mismo caso se corresponde con los enmarcados anteriormente como AE #2. Por otro lado, las diferentes tipos de estructuras para la representación y toma de datos, en este caso clasificación, agrupamientos y asociación; define un tercer aspecto esencial (AE #3) compuesto por la estructuras de diseño que identifican un objeto para la toma y representación de los datos que se extienden en otros tipos especiales de estructuras de datos.

5. DEVScondicionales

El desarrollo del presente condicional surge de la particularidad de que las interacciones en el DHD pueden ser consideradas como eventos asincrónicos en una base de tiempo continuo. Por esto, se ha modelado al DHD como sistema complejo con el formalismo DEVS, Discrete EVent System specification [19], considerándose además la adaptación del mismo para modelizar sistemas dinámicos, y su simplicidad y eficiencia en la implementación de simulaciones.

A continuación se describen los aspectos principales que se tuvieron en cuenta para la aplicación de un nuevo sistema de cálculo de niveles de interactividad para cada participación de los usuarios del sistema mediante un modelo de simulación DEVS. Aquí también la integración se produce mediante la conexión de las reglas a través de sus condicionales. A su vez, la métrica [20] para el cálculo de los niveles de interactividad de las participaciones de los usuarios es interpretada por un modelo DEVS diseñado para devolver resultados de simulación [21].

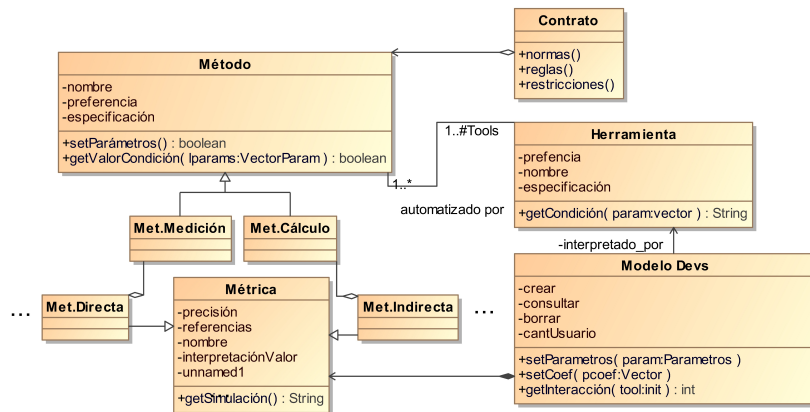


Fig. 4. Modelo de integración para los DEVScondicionales.

Al igual que en los casos anteriores, se produce la integración por la incorporación de una relación de agregación entre la componente *Contrato* y la entidad *Método* (Figura 4). La interpretación y manipulación de los resultados de interacciones resueltos en el *Modelo DEVS*, es manipulado por una herramienta representada por la clase *Herramienta*. A su vez, la herramienta es la encargada de brindar la información necesaria sobre los parámetros que necesita la clase *Método* que es utilizada como argumento de la función *setParámetro*. El método *getValorCondición* representa los valores de verdad del condicional que formará parte de la regla explícita representada por el método *reglas* de la clase *Contrato*.

Técnicamente la *Herramienta* es una aplicación que respeta la arquitectura del *framework* colaborativo Sakai, utilizando los servicios base para el acceso datos. Por otro lado, permite la aplicación de una función transferencia que transforma dichos datos teniendo en cuenta un archivo de parametrización.

De esta manera la *Herramienta* constituye una nueva entidad de conexión con un grado de sofisticación mayor a los componentes utilizados en los anteriores tipos de condicionales. Puede ser vista como un propio subsistema de conexión que para este caso se lo denomina *Integrador*. Un integrador puede ser visto como una representación de varias partes de componente que reúne a todas los aspectos esenciales anteriormente mencionados, lo que lo hace a su vez el cuarto aspecto esencial (denominado por AE #4) de condicionales para contratos sensibles al contexto del DHD.

6. Hacia un modelo conceptual de condicionales

En esta sección se presenta un modelo conceptual y la información necesaria para la creación de condicionales para el DHD. Partiendo de las decisiones de diseño e implementación que se presentaron se establecen estructuras conceptuales a tener en cuenta como instrumento de adaptación a los contratos sensibles al contextos.

En este caso, se propone un modelo de integración para conectar un subsistema soporte de configuración (cálculo) de valores de verdad de las reglas de contratos. Las componentes y relaciones que se presentan, resuelven aspectos esenciales anteriormente definidos. La figura 5 representa la idea de la propuesta basada en los módulos a tener en cuenta en las decisiones para concretar un eficiente diseño de condicionales. En este caso, a partir de un módulo de integración se concentra el control de las partes intervinientes. De esta manera se define un módulo para efectuar los cálculos finales que determinan el valor de verdad del condicional (*Cálculo*). Otro módulo es el encargado de la recolección y toma de datos (*Tomar Datos*), extendiéndose para los casos particulares donde es necesario contar con estructuras de datos (*Estructuras*) conteniendo métodos que implementan cada una de ellas.

Además, un módulo aparte se configura para describir todas las restricciones que deben cumplir el condicional (*Restricciones*), teniendo en cuenta su utilización dentro de las reglas de los contratos, con el propósito de no incurrir en contradicciones o inconsistencias en relación a las pre y post condiciones e invariantes. Las conexiones con otros subsistema, por ejemplo, el subsistema sensible al contexto representado en la figura 3, se encuentran encapsuladas en otro módulo de conexión (*Conexión*).

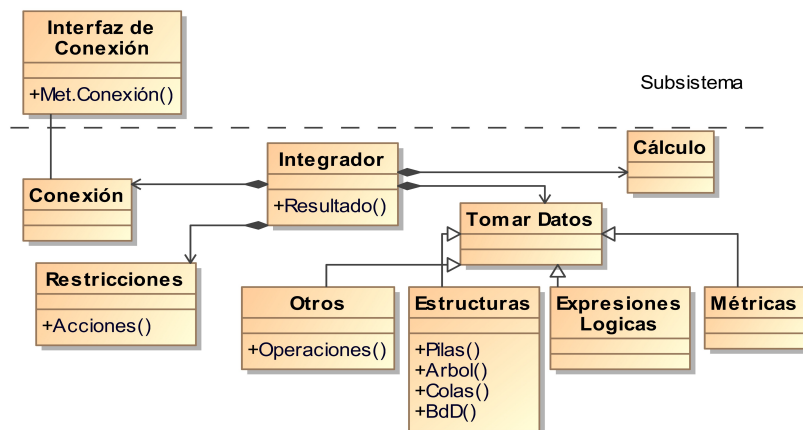


Fig. 5. Modelo conceptual de condicionales para los contratos sensibles al contexto.

7. Conclusiones

En este trabajo se desarrollaron las posibilidades de extender las propiedades expresivas de las reglas de los contratos de los DHD mediante la participación de mecanismos externos representados como subsistemas.

Cada subsistema está relacionado con un tipo de condicional. Los *Mcondicionales* permiten aumentar el grado de expresión de las reglas, lo que implica una mayor flexibilidad. Los *MDcondicionales* a través de la implementación de técnicas de minería de datos potencian la clasificación de los valores de contexto. Por último y en relación a los *DEVSccondicionales* el resultado de este análisis a partir de la simulación en un modelo externo resignifica una característica del comportamiento de los participantes. De esta manera, es posible establecer un lazo de retroalimentación entre las prácticas efectuadas en los entornos colaborativos y las acciones que devengan de los contratos.

A partir de todo lo expuesto podemos señalar la independencia entre los modelos y la concreción de la vinculación a muy bajo costo de implementación. Las propuestas de integración siguen respetando las líneas de diseño e implementación establecidas por el modelo de los DHD.

Partiendo del diseño e implementación planteado para cada tipo de condicional, se fueron estableciendo aspectos esenciales e información necesaria construyendo un modelo conceptual general para la creación de condicionales en el DHD.

Referencias

1. San Martín, P. (2008) Hacia un dispositivo hipermedial dinámico. Educación e investigación para el campo audiovisual interactivo. Universidad Nacional de Quilmes

- Editorial, Bernal.
2. Sartorio, A. y Cristiá, M. (2008) Primera aproximación al diseño e implementación de los DHD. XXXIV Congreso Latinoamericano de Informática, CLEI. Santa Fe.
 3. Meyer, B. (1992) Applying Design by Contract, IEEE Computer, 40-51.
 4. Dey, A.K., Salber, D., Abowd, G.: A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications, anchor article of a special issue on Context-Aware Computing. Human-Computer Interaction (HCI) Journal, Vol. 16 (2-4), pp. 97-166. (2001).
 5. Sartorio, A.; Cristiá, M. (2009) First Approximation to DHD Design and Implementation. Clei electronic journal, Vol.12 N. 1.
 6. Sartorio, A. (2007) Un modelo comprensivo para el diseño de procesos en una Aplicación E-Learning. XIII Congreso Argentino de Ciencias de la Computación. CACIC 2007. ISBN 978-950-656-109-3.
 7. Dourish, P. (2004) What we talk about when we talk about context. Personal and Ubiquitous Computing, vol. 8, No 1, Roma, pp. 19-30, <http://www.springerlink.com/content/y8h8l9me8yabycl3/>.
 8. Rumbaugh, J.; Jacobson, I. and Booch, G. (1999) The Unified Modeling Language Reference Manual. Massachusetts: Addison Wesley Logman, Inc.
 9. Sartorio, A.; Rodríguez, G. y Vaquero, M. (2010) Condicionales DEVS en la coordinación de contratos sensibles al contexto para los DHD. XVI Congreso Argentino de Ciencias de la Computación. Morón.
 10. Rivera, M. B.; Molina, H. y Olsina, L. (2007) Sistema Colaborativo de Revisión para el soporte de información de contexto en el marco C-INCAMI. XIII Congreso Argentino de Ciencias de la Computación, CACIC 2007. Corrientes y Resistencia.
 11. Olsina, L., Rossi, G. (2002) Measuring Web Application Quality with WebQEM, IEEE Multimedia, 9(4), pp. 20-29.
 12. Arabie, P., Hubert, J., De Soete, G. (1996) Clustering and Classification, World Scientific Publishers. Singapore.
 13. Tang, T., McCalla, G. (2004) Evaluating A Smart Recommender for an Evolving E-Learning System, Proceedings of Canadian Artificial Intelligence Conference. Canada.
 14. Romero, C. (2003) Aplicación de técnicas de adquisición de conocimiento para la mejora de cursos hipermedia adaptivos basados en web, Tesis Doctoral, disponible en: <http://www.isys.ucl.ac.be/bchi/publications/Ph.D.Theses/L%F3pez-PhD2005.pdf>.
 15. Mor, E., Minguillón, J. (2004) E-learning Personalization based on Itineraries and Longterm Navigational Behavior. World Wide Web Conference.
 16. Sutinen, E., Hämäläinen W., Suhonen, J., Toivonen, H. (2004) Data Mining in Personalizing Distance Education Courses. Conference on Open Learning and Distance Education. Hong Kong.
 17. Quinlan, J. (1987) Generating Production Rules from Decision Trees. Proceedings of the Tenth International Joint Conference on Artificial Intelligence (pp. 304–307). Milan.
 18. Schmidt, A. (2005) Bridging the Gap Between E-Learning and Knowledge Management with Context-Aware Corporate Learning Solutions. Proceedings WM '05, Springer LNCS 3782.
 19. Zeigler, B.; King, Tan Gon; Praehofer, H. (2000) Theory of modeling and Simulation. Second edition, Academic Press, New York.
 20. Rodríguez, G. (2010) Desarrollo e implementación de métricas para el análisis de las interacciones del Dispositivo Hipermedial Dinámico. Jornadas Argentinas de Informática. JAIIO 2010, Ciudad Autónoma de Buenos Aires.
 21. PowerDEVS 2.0 Integrated Tool for Edition and Simulation of Discrete Event Systems. Desarrollado por: Esteban Pagliero, Marcelo Lapadula, Federico Bergero. Dirigido por Ernesto Kofman. (<http://www.fceia.unr.edu.ar/lsd/powerdevs/index.html>).