

Un modelo de hipertexto para la especificación de Requisitos

María Carmen Leonardi*, Gustavo Rossi **, Julio Cesar Sampaio do Prado Leite***

* ISISTAN-UNCPBA & CIC-Bs.As, Rca. Argentina, e-mail: cleonard@exa.unicen.edu.ar

** LIFIA- UNLP, Rca. Argentina, e-mail: grossi@info.unlp.edu.ar

***Dpto. de Informatica, PUC-Rio de Janeiro – RJBrasil e-mail: julio@inf.puc-rio.br

Apoio do CNPq, processo 523894/95-3

Resumen

En este trabajo se presenta un modelo de hipertexto para la Requirements Baseline [Leite'97a] y las tarjetas CRCs derivadas a partir de ella [Leonardi'97][Leite'97b]. Este modelo se basa principalmente en el diseño navegacional de OOHDM [Rossi'97][Schwabe'98], una metodología de diseño Orientada a Objetos para aplicaciones hipermedia. El objetivo de este trabajo es presentar al modelo de hipertexto como un soporte para la etapa de modelización de requisitos. Esta integración trae ventajas en la facilidad de acceso a la información proveyendo diferentes formas de recorrer y analizar los documentos. De esta forma, se facilita la tarea del ingeniero durante la etapa de adquisición de requisitos [Fiorini'97]

Palabras claves: especificación de requisitos, modelo de hipertexto, pre y post-seguimiento

Abstract

In this paper we present the hypertext view belonging to Requirements Baseline [Leite'97a] and the derived CRCs [Leonardi'97][Leite'97b]. This model is based on OOHDM [Rossi'97][Schwabe'98], an Object-Oriented hypermedia design method. The objective of this paper is to present the hypertext technology as a support for the requirements phase. This integration improves the access of the information by allowing different ways of navigation, therefore it facilitates the task of the analyst during the requirements acquisition phase [Fiorini'97]

Key keywords: requirement specification, hypertext model, pre-post traceability

1. Introducción

El proceso de definición de requisitos tiene tres etapas bien diferenciadas pero estrictamente relacionadas [Fiorini'97]: el proceso de "elicitación" de requisitos, en donde se adquiere la información

relacionada al problema en estudio, el proceso de modelado, responsable por describir al problema de una forma sistemática y el proceso de validación de los modelos generados. Dentro de la etapa de modelado no solo se representa los requisitos sino, fundamentalmente todo el contexto en donde estos están insertos. En trabajos anteriores se presentó la Requirements Baseline como una herramienta para la modelización del macrosistema[Leite'97a]. Las relaciones naturales entre las componentes de la Baseline sugieren la formación de un hipertexto para recorrer la información[Leite'95], sin embargo la capacidad de esta vista no se había investigado en profundidad. Por esta razón se estudia OOHDm con el objeto de aplicar sus conceptos al modelo de Baseline. Este trabajo propone un estudio inicial sobre el modelo de hipertexto de la Baseline, el cual permitiría recorrer la información siguiendo diferentes criterios y puntos de vista con el objeto de recolectar el mayor conocimiento posible a la hora de tomar decisiones. Sin bien es un primer estudio consideramos importante la discusión sobre la integración de hipertexto como medio para representar los requisitos y cómo repercute en la etapa de adquisición de los mismos. En la sección dos se presenta el modelo de Requirements Baseline, en la sección 3 se describe brevemente los conceptos principales del modelo de navegación de OOHDm. En la sección 4 se define el modelo de hipertexto para la Requirements Baseline discutiendo sus ventajas y problemas. La sección 5 muestra un ejemplo, y finalmente, en la sección 6 se presentan las conclusiones y futuros trabajos.

2. Requirements Baseline

La Requirements Baseline [Leite'95][Leite'97] es una estructura que incorpora sentencias sobre el sistema deseado. Estas sentencias son escritas en lenguaje natural siguiendo patrones definidos. La idea subyacente a la Baseline es que ésta tiene naturaleza perenne. Se desarrolla durante la etapa de requisitos pero evoluciona a la par de la evolución del proceso de desarrollo de software. Esta Baseline se comporta como un engranaje hacia el macrosistema, es decir, hacia el ambiente del cual el sistema de software formara parte.

La Requirements Baseline se estructura de la siguiente manera:

- * Un modelo de léxico
- * Un modelo básico
- * Un modelo e escenarios
- * Un modelo de reglas
- * Una vista de hipertexto
- * Una vista de configuración

Las vistas de hipertexto y la de configuración soportan la representación y evolución de las otras vistas. La vista de configuraciones[Leite'95] mantiene información de los cambios realizados: persona que los realizó, fecha, hora, razones del cambio, y tipo del mismo. La vista de hipertexto[Leite'95][Leite'97a] permite relacionar todos los modelos obteniendo información complementaria sobre el macrosistema y permitiendo "seguimiento" de la información. Esta vista es la que se estudia con mayor profundidad en este trabajo.

El modelo de léxico está representado por el Léxico Extendido del Lenguaje, que constituye un meta-modelo diseñado para ayudar durante la adquisición del vocabulario usado en el macrosistema. Está formado por un conjunto de símbolos, donde por cada símbolo se define nombre y conjunto de sinónimos, nociones (definiciones del símbolo) y los impactos (describen como repercute el símbolo en el sistema). En la descripción de las nociones e impactos existen dos reglas básicas que se deben cumplir simultáneamente:

- "principio de circularidad": en la noción o impacto de los símbolos se pueden referenciar a otros. Esto tiende a maximizar el uso de símbolos en el significado de otros símbolos.

- "principio del vocabulario mínimo": se minimiza el uso de símbolos externos al lenguaje de la aplicación. Se acota el lenguaje al menor conjunto de símbolos posible.

El modelo básico es una estructura de sentencias centrado en el concepto que la identificación de las acciones de los clientes es un camino indirecto de descubrir cual es la información necesaria para soportar decisiones de negocio [Carvalho'88], un concepto similar a JSD [Jackson'83]. Describe las acciones del sistema a través de Diagramas de Relaciones extendidos. Las entidades que se describen son: cliente(individuo o grupo responsable de acciones), acciones, eventos externos, estímulos temporales y externos, entradas y salidas del sistema. A estas entidades se agrega el concepto de restricciones y diagnóstico. Una restricción es un requerimiento de una determinada entidad. Un diagnóstico es una observación de esa entidad que comete al sistema.

El modelo de escenarios es un meta-modelo para representar escenarios. Los escenarios son descripciones de situaciones del macrosistema cuyo comportamiento se describe[Leite'97]. Los escenarios son descritos en lenguaje natural respetando la siguiente estructura: Título, Objetivo (establece la finalidad del escenario), Contexto (estado inicial del escenario, la ubicación física y temporal), Recursos (entidades pasivas con los cuales los actores trabajan), Actores (entidades que se involucran activamente en el escenario) y un conjunto de Episodios (cada episodio representa una acción realizada por un actor). Un episodio puede referenciar a un escenario, definiendo el concepto de sub-escenario.

El modelo de reglas [Leite'98][Leonardi'98] describe a través de reglas funcionales y no funcionales, las políticas y normas de la organización que afectarán al comportamiento y a los datos de la misma. Las reglas funcionales describen políticas sobre el funcionamiento de la organización. Las reglas no funcionales pueden clasificarse en reglas del macrosistema y reglas de calidad. Las reglas del macrosistema describen políticas que restringen el comportamiento de la organización. Las reglas de calidad son demandas de la organización sobre las características de sus productos, usualmente reflejan políticas generales relacionadas a standard de calidad o expectativas de calidad de la organización.

Los modelos de LEL, escenarios y reglas permiten representar al macrosistema desde tres perspectivas de conocimiento diferentes: datos, comportamiento y reglas[Petrounias'94]. A partir de estos modelos se deriva un modelo conceptual de objetos [Leite'97b] [Leonardi'97] permitiendo al analista visualizar al macrosistema bajo los mismos términos que usará en los diferentes modelos que realizará en un desarrollo orientado a objetos. Este modelo se representa mediante tarjetas CRCs, similares a las tarjetas presentadas en [Wirfs'90]. Una CRC representa a una clase del macrosistema, el conjunto de responsabilidades, servidores y clientes junto con la justificación que le permite ser modelada como una clase. El conjunto de CRCs actúan como una interfaz entre la Requirement Baseline y una especificación Orientada a Objetos, como puede ser el modelo de análisis de OMT [Rumbaugh'91]. De esta manera se logra un "seguimiento" de la información, en particular dada una clase del modelo conceptual orientado a objetos, se puede obtener información a través de las CRCs y su relación con las componentes de la Requirements Baseline, justificando su existencia.

3. Diseño navegacional en OOHDMM

En esta sección se resumen los principales aspectos del modelo conceptual de OOHDMM[Rossi'96] [Schwabe'98], una metodología Orientada a objetos para el desarrollo de aplicaciones de hipermedia. La metodología OOHDMM diferencia cuatro etapas en el desarrollo de las aplicaciones de hipermedia: el Diseño Conceptual, el Diseño Navegacional, el Diseño de Interfaz Abstracto y la etapa de

Implementación. En el Diseño Conceptual se realiza un modelo orientado a objetos que representa al dominio de la aplicación en estudio. Luego se construye sobre este modelo un modelo de navegación. Finalmente se definen los aspectos de la interfaz y de implementación.

Para este trabajo nos concentramos en la etapa de Diseño Navegacional. Este modelo se construye como una vista sobre el modelo conceptual de la aplicación. Cada modelo navegacional provee una vista subjetiva de acuerdo a diferentes perfiles de usuarios. Durante esta fase se definen los objetos navegacionales que reflejarán a los objetos de la aplicación. Los objetos predefinidos para el modelo navegacional son: nodos, links y estructuras de acceso, que representan diferentes formas de acceder a la información. La principal estructura de navegación es el contexto navegacional: un contexto navegacional es un conjunto de nodos, links y otros contextos navegacionales anidados. Puede ser definido enumerando una condición que deben cumplir sus miembros o enumerando cada uno de ellos. La definición incluye el orden de acceso a los elementos que lo componen y relaciones con otras estructuras de acceso relacionadas. Existen cinco formas básicas de definir un contexto:

- * Basados en una clase simple: todos los objetos que pertenecen al contexto son de la misma clase de la aplicación
- * Basados en grupo: es un conjunto de contextos, cada uno es un contexto basado en una clase simple de aplicación.
- * Basados en links: los elementos de esta clase son objetos de la misma clase y son seleccionados cuando pertenecen a relaciones 1-n
- * Basados en grupos de links: es un conjunto de contextos en donde cada contexto es de tipo basado en link
- * enumeración: los elementos de este contexto son explícitamente numerados
- * dinámicos: los elementos son insertados durante la navegación.

Los nodos son enriquecidos con clases especiales que le permiten al nodo mostrarse con diferentes aspectos de acuerdo al contexto navegacional. Los contextos navegacionales organizan el espacio de navegación, ayudando al usuario a recorrer la información.

El objetivo de este trabajo es ampliar las capacidades de navegación de la vista de hipertexto utilizando los conceptos de OOHDM, no sólo para las vistas de la Requirements Baseline(LEL, escenarios, reglas) sino también para su interfaz (en nuestra propuesta las CRCs) con un modelo de análisis orientado a objetos.

4. La Vista de Hipertexto

La vista de hipertexto es ortogonal a los modelos de LEL, escenarios, reglas y CRCs. Definimos esta vista en base a los conceptos presentados en la sección anterior.

Los nodos del hipertexto se derivan a partir de cada una de las instancias de los diferentes modelos, es decir, nodos que representan a los símbolos LEL, nodos escenarios, nodos reglas y nodos CRCs. En principio modelaremos un nodo para cada entidad, luego se verá que esta consideración es ampliada para tener en cuenta aspectos de evolución. Por otro lado, se está estudiando la incorporación de nodos que represente la racionalidad de la información[Conclin'88], esta discusión se incluye en la sección de conclusiones.

A partir de estos nodos y sus relaciones definimos diversos tipos de links:

* Estructurales: son links que unen nodos que tienen una relación de composición. En nuestro modelo estos links son "link sub-escenario", "link composición de términos de LEL", este último no puede derivarse automáticamente, ya que el usuario, analizando los términos de las nociones de cada un símbolo del LEL, puede darse cuenta si se está describiendo una noción de composición. La figura 1 muestra la semántica de estos links y los tipos de nodo que conecta.

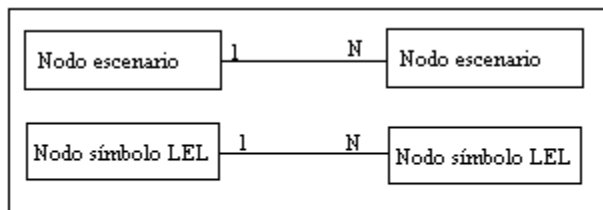


Figura 1: links estructurales

* Intra-link: son links que conectan a nodos de un mismo modelo, "link LEL" que une los términos del LEL respetando el principio de circularidad; "link excepciones" que una a un escenario con otro que ha sido definido como una curso alternativo de acción. Los links de las CRCs son los "link cliente" y "link servidor" que conectan a cada CRCs con sus clientes y servidores.

* Inter-link o Complementarios: estos links permiten relacionar cada entidad del modelo conceptual con otra. Permite al analista navegar a través del hipertexto y adquirir conocimiento complementario sobre el macrosistema en estudio. Ejemplos de estos links son:

- * Desde una regla conectar los términos del LEL que aparece en ella con el nodo que representa la descripción del mismo
- * Desde un escenario conectar los términos del LEL que figuran en él con el nodo que representa la descripción del mismo
- * Desde los episodios de los escenarios conectar los reglas que aparecen en él con los nodos que las describen.

Estos son los links que se derivan automáticamente de los modelos de la Baseline. Existen otras relaciones que se obtendrán a través de los contextos navegacionales. La figura 2 muestra los links complementarios y los tipos de nodos que conectan

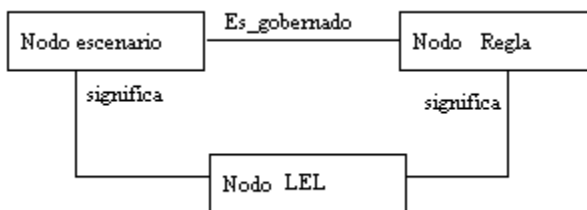


Figura2: links complementarios

* Links de seguimiento: estos links pueden dividirse en pre y post seguimiento, adoptando los conceptos de [Gotel'94]. Los links de post-seguimiento conectan a los elementos de la Baseline con las tarjetas CRCs, de esta manera se relaciona la definición del macrosistema con un primer modelo de una solución

orientada a objetos. Hay tres tipos de estos links. El "link significa" conecta a los términos del LEL que aparecen en las CRCs con los nodos que representan su definición; los "links es_escenario" que conectan a una CRCs con un escenario si el nombre de una responsabilidad de la CRCs es el mismo que el título de un escenario. Finalmente, el nodo "es_regla" conecta a las CRCs con una regla funcional si alguna de las responsabilidades de la CRCs es una regla funcional.

Los links de pre-seguimiento surgen a partir de la vista configuraciones. Esta vista permite tener la historia de cada una de las componentes de la Baseline, por ejemplo podemos saber quien es el autor, en que fecha la creó, así como también su evolución. Un elemento de la Baseline tendrá tantos nodos que lo represente como versiones. Estos nodos estarán conectados por links del tipo "versión siguiente" y "versión anterior" que permitirán recorrer las diferentes versiones de una misma componente. Además se incorpora en los nodos los datos de la vista de configuración para cada entidad[Leite'95], es decir, quién fue el responsable de cada cambio, fecha y justificación. Se obtiene así pre-seguimiento de las componentes, ya que conectamos la información con su fuente de origen.

4.1 Contextos navegacionales

Tomando como base las formas de definir un contexto navegacional presentado en la sección anterior definimos los siguientes contextos:

* Basados en una clase:

Recorrer todos los escenarios del macrosistema

Recorrer todos los elementos de LEL del macrosistema

Recorrer todas las Reglas del macrosistema

Recorrer todos las CRCs del macrosistema

* Basado en un link:

Recorrer todos los sub-escenarios de un determinado escenario

Recorrer todos los escenarios que comparten un determinado símbolo de LEL

Recorrer todos los escenarios en donde aparece una determinada CRC

Recorrer todas las reglas que comparten un determinado símbolo de LEL

Recorrer todas las reglas que aparecen en un escenario

Recorrer todos los términos de LEL que comparten un determinado símbolo de LEL

Recorrer todas las CRCs que aparecen en un determinado escenario

Recorrer todas las CRCs en donde aparece un determinado símbolo del LEL

* Basados en grupo de links:

Recorrer todos los escenarios, LEL, reglas que comparten un mismo símbolo de LEL

Recorrer todos los escenarios, LEL, reglas que comparten una determinada CRC

Existen también contextos que surgen a partir de los links de pre-seguimiento. Es importante destacar que los contextos no sólo definen los elementos que lo componen sino también la forma en que se navega: desde un mismo nodo, el link: "siguiente" puede ir a nodos diferentes dependiendo del contexto por el cual se está navegando.

Este modelo de hipertexto puede implementarse especializando el OOframework [Garrido'96] en el cual se define una jerarquía de nodos y links siguiendo los conceptos de OOHDM. Hasta el momento se ha implementado un ambiente de autoría y navegación de los modelos de LEL, escenarios y CRCs llamada "ReCase" que implementa algunos de estos conceptos de navegación[Antonelli'98]

4.2 Discusiones sobre el modelo

Esta vista de hipertexto permite que el analista recorra la información de una forma organizada, ayudándolo en la tarea de definición de requisitos. En este sentido el modelo definido se puede utilizar como un complemento a la propuesta realizada en [Fiorini'97] donde un analista utiliza un modelo de empresa representado como un hipertexto para extraer la información que definirán los requisitos siguiendo el modelo Serbac[Leite'95]. Nuestra vista de hipertexto servirá como un medio adicional en donde el ingeniero de software encontrará información sobre cómo se comporta el macrosistema, cuales son sus datos y sus reglas, ampliando el conocimiento sobre los procesos, objetivos estratégicos y demás información que le provee el modelo de la empresa.

Algunas de las ventajas de utilizar un modelo de hipertexto para representar al macrosistema son las siguientes:

El acceso de la información a través de un hipertexto relaciona ideas permitiendo obtener nueva información, sacar conclusiones, encontrar inconsistencias y conflictos.

* Los contextos navegacionales permiten concentrarse en determinados aspectos de la Baseline, filtrando la información que se desea analizar del resto.

* Permite relacionar los conceptos del macrosistema, obteniendo información complementaria. También se logra pre-seguimiento[Gotel'94] para las componentes de la Baseline.

Los problemas que ocasiona modelar el macrosistema a través de un modelo de hipertexto provienen de los dos problemas clásicos de la navegación en un hipertexto[Nielsen'90]: sobrecarga cognitiva y la desorientación. La información redundante debe ser utilizada con un buen criterio, y se debe ayudar al usuario a elegir la forma en que navega de una manera controlada y consistente[Schwabe'98]. En nuestro caso, este problema se ha atacado a través de los conceptos del modelo navegacional de OOHDM, en particular con los contextos navegacionales. Los nodos en nuestra vista de hipertexto se derivan de cada una de las instancias de los modelos de Baseline, por lo que responden a un concepto dentro del macrosistema, con una semántica bien definida. Los links derivan a partir de las relaciones que surgen naturalmente del modelo conceptual y como resultado de consultas que definen los contextos navegacionales que agrupan nodos según cumplan algún criterio en común. De esta forma se logra una estructura de navegación fuertemente relacionada al modelo conceptual que representa y el analista comprende donde está, a que nodos puede acceder y como hacerlo.

El problema de la desorientación, puede resolverse mediante la construcción de mapas que ayudan a

orientarse al usuario, en donde muestra los nodos visitados, en donde se encuentra el analista, los posibles nodos siguientes y cual es el criterio por el cual está recorriendo la información actualmente.

5 Caso de Estudio círculo cerrado para la compra de un automotor

Las ideas expuestas en la sección anterior fueron utilizadas parcialmente en los ejemplos extraídos de los casos de estudio desarrollados por alumnos en un curso dictado en la UNCPBA [Leite'97b]. Se tomó del trabajo de los alumnos el modelo de LEL, escenarios y CRCs. Luego se definió un modelo de reglas y se determinaron las relaciones entre las componentes y los contextos navegacionales, simulando el hipertexto. La figura 3 muestra el nodo escenario "Adjudicación" que está siendo navegando como un nodo perteneciente al contexto navegacional "escenarios que comparten el término de LEL: adherente". Esta figura proviene de la herramienta ReCase

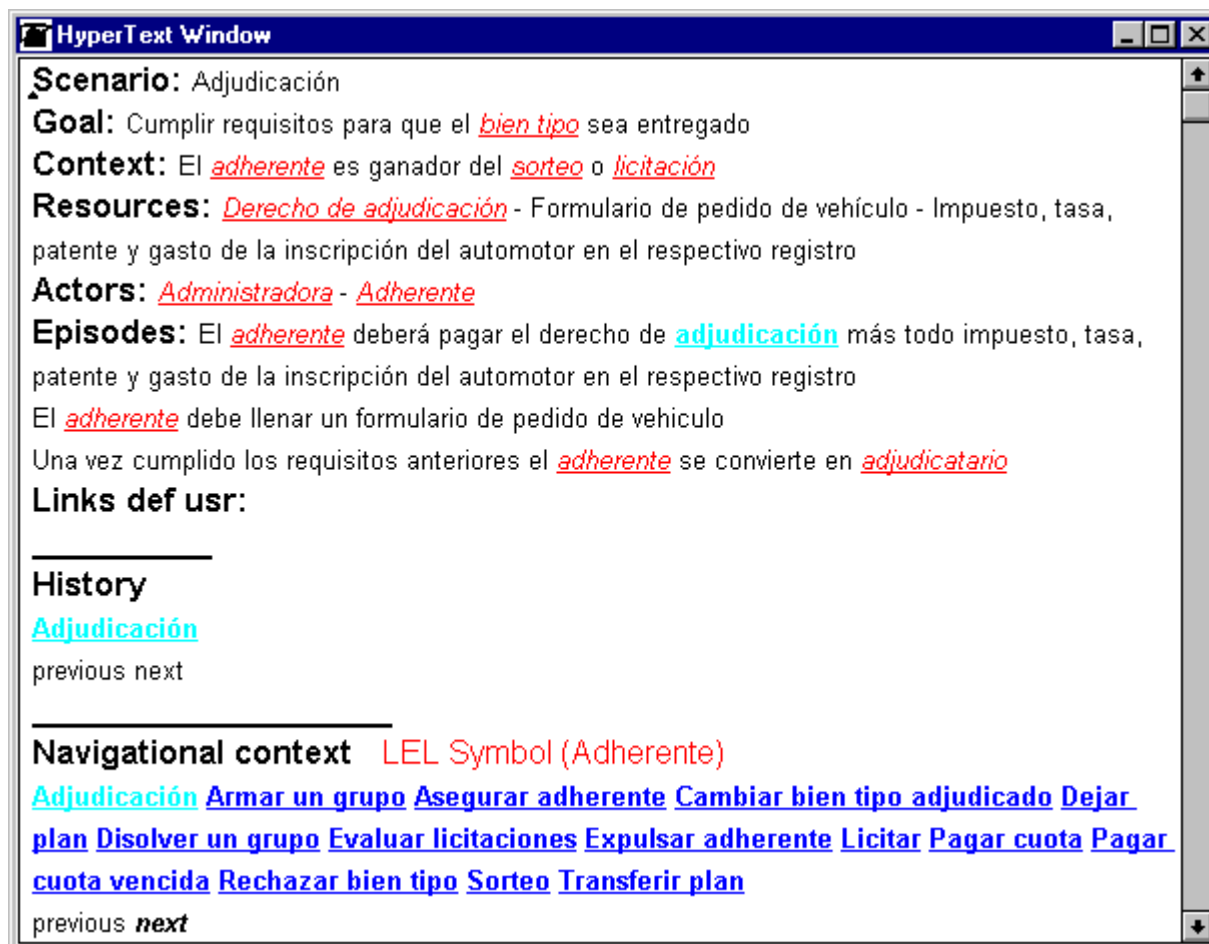


Figura 3: ventana del ambiente ReCase mostrando el escenario Adjudicación

La figura 4 muestra nodos relacionados a través de dos contextos navegacionales. El primero, cuyos

links están marcados por una línea punteada es el contexto navegacional "CRC en donde aparece el símbolo del LEL: adherente". El segundo contexto es "Todas las CRCs que aparecen en el escenario: Expulsar adherente", en donde los links se marcan con línea llena. Observe como desde la CRC Administradora el link "next" conecta con diferentes nodos, dependiendo del contexto.

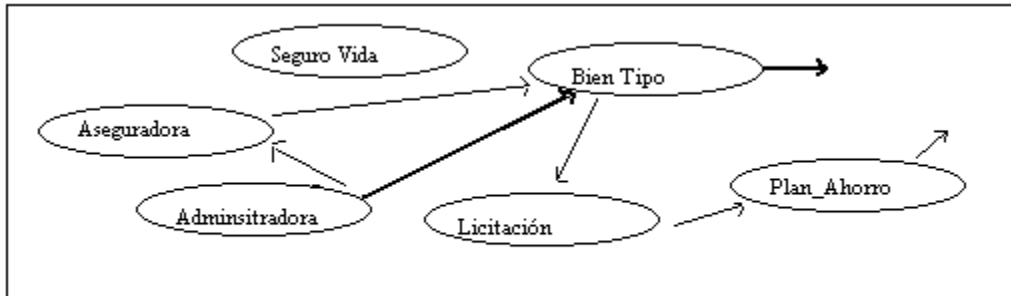


Figura 4: CRCs relacionadas por dos contexto navegacionales

6. Conclusiones y futuros trabajos

En este trabajo se define una vista hipertexto, ampliando la propuesta de [Leite'97] para considerar el modelo de reglas y las tarjetas CRCs. Se define los nodos conceptuales y principalmente las relaciones entre ellos. Se discuten una serie de ventajas y problemas relacionados a modelar los requisitos mediante un modelo de hipertexto.

Un aspecto que ya se está estudiando es la incorporación de racionalidad. El objetivo es documentar las negociaciones asociadas a las reglas y los procedimientos. Esta idea es adoptada de GIBIS [Conklin'88]. Según nuestra propuesta [Leite'98], hay veces en que una regla puede implementarse de diferentes maneras, es decir, tiene diferentes procedimientos asociados. Por ejemplo una regla de una Universidad puede ser "Apoyar económicamente los viajes de estudio". Asociado a esta regla puede haber diferentes procedimientos para llevarla a cabo: dar dinero en efectivo, proveer de tarjetas de crédito especiales para el viaje, entre otras posibilidades. Existirán argumentos a favor y en contra de cada una de las alternativas. Una posibilidad es considerar a los procedimientos como nodos, de una forma similar a los "Positions" de GIBIS y tener nodos que son considerados como argumentos a favor o en contra de las diferentes propuestas de los procedimientos. Es necesario extender este estudio a trabajos relacionados, en especial [Rosca'97] y analizar la posibilidad de incorporar racionalidad a todos los elementos de la Requirements Baseline.

Dentro de los trabajos futuros se quiere estudiar la integración del modelo de Requirements Baseline al modelo de la empresa propuesto en [Fiorini'97] y la forma en que éste se adaptaría a un modelo de hipertexto con las características de OOHD. De esta forma se incrementaría la capacidad de pre-seguimiento conectando la información del macrosistema con todos los procesos y datos de la organización.

Se debe estudiar la forma en que se incorporará los user profiles y viewpoints, que permitirán registrar diferentes aristas de una misma información según sea el punto de vista y restringir las formas de acceso

a la información. Finalmente debe incorporarse todas estas facilidades a la herramienta presentada en [Antonelli'98] que actualmente permite la autoría y algunos aspectos de navegación de los modelos de LEL, escenarios y CRCs.

Referencias

- [Antonelli'98] Antonelli L "ReCase" trabajo final de grado. Fac. de Ciencias Exactas, Dpto. de Informática, UNLP.
- [Conklin'88] Conklin J, Begeman M. "gIBIS: A Hypertext Tool for Exploratory Policy Discussion". ACM Transactions on Office Information Systems, vol. 6, no 4 October 1988, pp 303-331.
- [Fiorini'97] Fiorini S, Leite JCS, Macedo-Soares T. "Integrando Processos de Negocio a Elicitacao de Requisitos" Revista de Informatica Teorica e Aplicada. Instituto de Informatica da Universidade Federal do Rio Grande do Sul, Volumen IV, Numero I, pp 7-48.
- [Garrido'96] Garrido A, Rossi G "A framework for extending Object oriented Applications with hypermedia functionality" Hypermedia and multimedia Volume 2, 1996 Taylor.
- [Gotel'94] Gotel O, Finkelstein A. "An analysis of the Requirements Traceability Problem", International Conference on Requirements Engineering, 1994 . pp 94-101.
- [Jackson '83] Jackson, M. "Systems Development", Prentice-Hall, 1983.
- [Leite'95] Leite J.C.S.P, A. Pádua Albuquerque Oliveira. "A Client Oriented Requirements Baseline". Proceedings of the Second IEEE International Symposium on Requirements Engineering, IEEE Computer Society Press, 1995, pp.108-115.
- [Leite'97a] Leite J.C.S.P, G. Rossi, et al. "Enhancing a Requirements Baseline with Scenarios" Proceedings of RE 97: IEEE International Symposium on Requirements Engineering, IEEE Computer Society Press, 1997, pp 44-53.
- [Leite'97b] Leite J.C.S.P, "Notas de Aula", apuntes correspondientes a un curso dictado en la UNCPBA, Tandil, 10-14 de noviembre de 1997.

- [Leite '98] Leite J.C.S.P, Leonardi Ma. Carmen "Business rules as organizational Policies", Proceedings IEEE IWSSD9: Ninth International Workshop on Software Specification and Design, IEEE Computer Society pp 68-76.
- [Leonardi'97] Leonardi Ma. Carmen, Maiorana V, Balaguer F. "Una estrategia de Análisis Orientada a Objetos basada en escenarios" Actas de II Jornadas de Ingeniería del software, JIS97, Dpto. de Informática, Universidad del País Vasco, San Sebastián 1997, pp. 87-100.
- [Leonardi'98] Leonardi Carmen, Leite J.C.S.P, Petersen Laura "Integración de un Modelo de Reglas a la definición de Requisitos" Actas de la Jornadas Iberoamericanas de Ingeniería de Requisitos y Ambientes de Software, Universidad Federal do Rio Grande do Sul, Instituto de Informatica, pp. 273-285.
- [Petrounios'94] Petrounias, I. Loucopoulos, P. A Rule-Based Approach for the Design and Implementation of Information Systems. Systems, EDBT '94, M. Jarke (ed.), Springer-Verlag, Cambridge, U.K., 1994.
- [Nielsen'90] J. Nielsen "Hypertext and Hypermedia" Academic Press, 1990.
- [Rossi'96] Rossi G " An Object Oriented Method for Designing Hypermedia applications". PhD Thesis, Departamento de Informatica, PUC-Rio, Brasil, Julio de 1996.
- [Rosca'98] Daniela Rosca, Sol Greenspan, Mark "A Decision Making Methodology in support of the business rules Lifecycle" Proceedings of RE 97': International Symposium on Requirements Engineering, IEEE. (Boston, Enero 1997) pp 236-246.
- [Rumbaugh'91] Rumbaugh J., Blaha M., Premerlani W., Eddy F., Lorenzen W. 1991. Object-Oriented Modelling and Design. Prentice Hall.
- [Schwabe'98] Schwabe D, Rossi G, " An object Oriented approach to Web-Based

Application Design" a ser publicado en "Theory and Practice of Object Systems", Wiley and Sons.

[Wirfs'90] Wirfs-Brock R. et al. "Designing Object-Oriented Software", Prentice Hall.

1990.