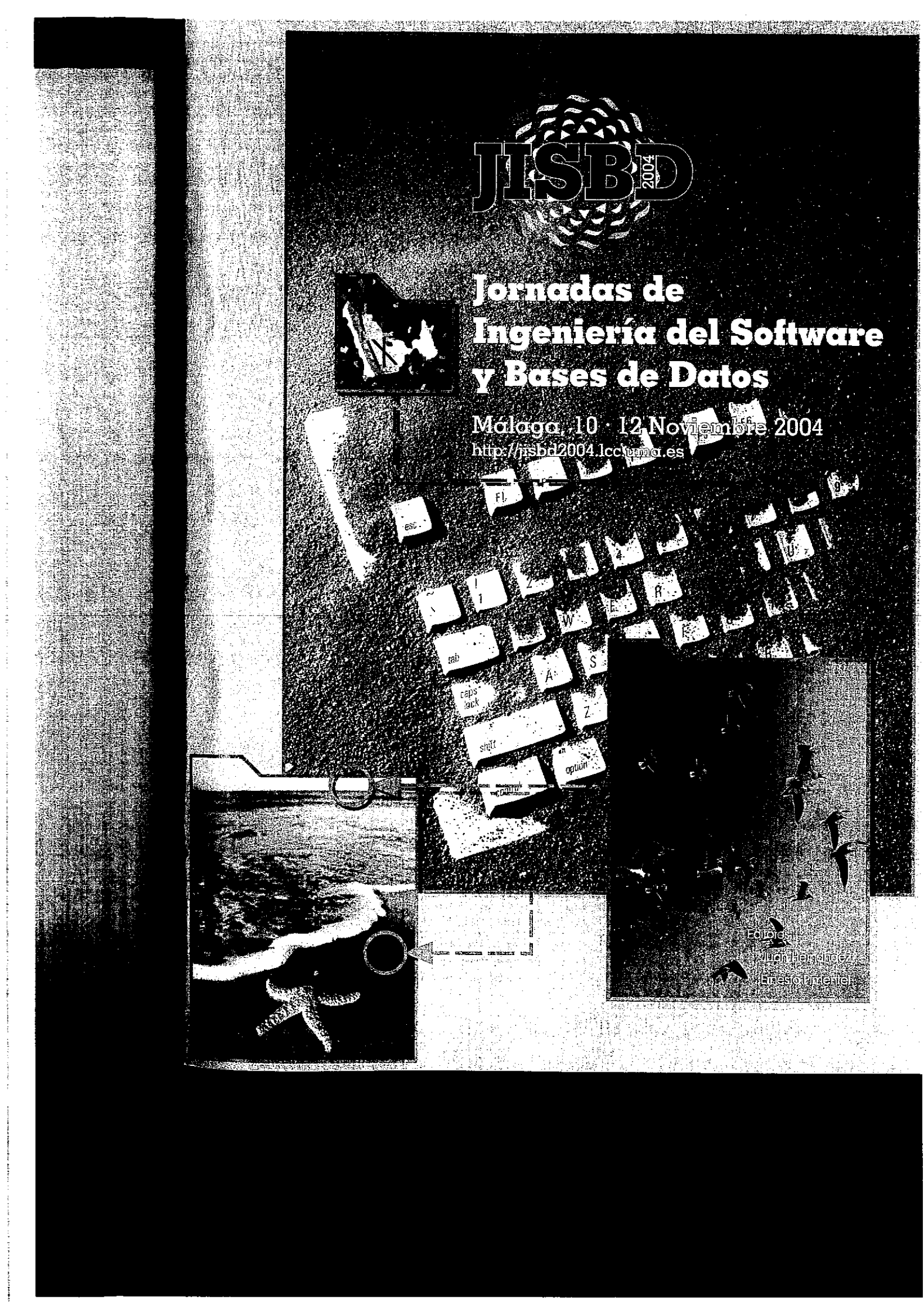
JISBD 2004
Jornadas de Ingeniería del Software y Bases de Datos
Málaga, 10 - 12 Noviembre 2004
http://jisbd2004.lcc.uma.es
Editors
Juan Hernández
Ernesto Pimentel

ACTAS

IX Jornadas de Ingeniería del Software y Bases de Datos

Málaga, 9 al 12 de Noviembre de 2004

I.S.B.N. 84-688-8983-0
D.L.: MA-1468-2004

Imprime: AltaGrafics

Prólogo

Las IX Jornadas de Ingeniería del Software y Bases de Datos han constituido, de nuevo, un punto de encuentro en el que profesionales y académicos de España y Portugal de ambos campos han podido compartir experiencias y resultados en un ambiente de colaboración entre distintos grupos de investigación, desarrollo e innovación tecnológica. Estas jornadas vieron su nacimiento en Sevilla en 1996, cubriendo inicialmente el área de la Ingeniería del Software, e incorporando en la edición que se celebró en Cáceres en 1999 el campo de las Bases de Datos. Desde entonces, se han consolidado como uno de los eventos celebrados en España de mayor prestigio en estas áreas.

Las IX Jornadas de Ingeniería del Software y Bases de Datos representan una edición especial, con la existencia de un Comité de Programa único formado por investigadores expertos en las materias de interés de las Jornadas.

El presente volumen contiene los trabajos seleccionados por el Comité de Programa para su presentación en la novena edición de estas Jornadas, ***JISBD'2004***, celebradas en Málaga durante los días 10, 11 y 12 de Noviembre de 2004. Se recibieron un total de 121 contribuciones, y cada una de ellas fue revisada por al menos tres miembros del Comité de Programa, quienes tras una ardua labor, dado el número y calidad de los artículos recibidos, seleccionaron 38 trabajos. Adicionalmente, se consideraron 14 trabajos para su presentación en la modalidad de artículos cortos. El programa científico se ha completado con dos excelentes conferencias desarrolladas por investigadores de gran prestigio a nivel internacional. La conferencia de apertura, *Towards Active Software*, fue dictada por el Dr. Ivar Jacobson, quien actualmente trabaja para Rational y Jaczone AB. Las contribuciones del Dr. Ivar Jacobson en múltiples campos de la Ingeniería del Software son ampliamente conocidas, entre las que merece la pena destacar el diseño del Lenguaje Unificado de Modelado (UML) y el proceso para el desarrollo de software basado en componentes (RUP). La conferencia en el campo de las Bases de Datos, *Enhancing the Web with DB Technology*, fue impartida por el Prof. Timos Sellis, Profesor de la *National Technical University* de Atenas, Grecia. El Profesor Timos Sellis, uno de los investigadores europeos más importantes en el campo de las Bases de Datos, recibió el premio Presidencial Jóvenes Investigadores del Presidente de los Estados Unidos, así como el premio de VLDB 1997, por sus trabajos en el área de las Bases de Datos.

Otra actividad que, con cada edición, va adquiriendo mayor relevancia, y que complementan el atractivo de las jornadas es la organización de talleres y tutoriales sobre temas muy específicos y/o de especial interés. Tuvieron lugar, en su mayor parte el día 9 de Noviembre, precediendo a la conferencia principal, un total de 9 talleres y 4 tutoriales con una nutrida y diversa participación.

La celebración de un evento de las características de JISBD, con una participación cada vez más numerosa y consolidada, y con unas exigencias de calidad que se van incrementando en cada edición, no podría realizarse sin la dedicación de los miembros del Comité Organi-

zador del Grupo de Ingeniería del Software de la Universidad de Málaga, a los que queremos agradecer el trabajo desarrollado en beneficio del éxito de las jornadas. Asimismo, queremos agradecer la labor del Grupo Quercus de Ingeniería del Software de la Universidad de Extremadura, quienes han estado a cargo de todo el sistema de recepción y revisión de artículos. Del mismo modo, no podemos olvidar que el objetivo último de este congreso es hacer posible que los investigadores y desarrolladores compartan y debatan sus ideas, y para ello hemos de agradecer y reconocer el trabajo desarrollado por los miembros del Comité de Programa y los revisores adicionales, que han contribuido a conformar un programa científico atractivo y de gran calidad. También queremos agradecer el soporte recibido por las entidades patrocinadoras y colaboradoras.

Deseamos, finalmente, que la próxima edición de estas jornadas, que se celebrarán en Granada en el 2005 como parte del I Congreso Español De Informática (CEDI), mantenga la tendencia de estas últimas ediciones en el nivel de participación .

Málaga, Noviembre 2004

Juan Hernández
Presidente del Comité de Programa
JISBD'2004

Ernesto Pimentel
Presidente del Comité de Organización
JISBD'2004

Índice

Artículos cortos

PRESIDENCIA DEL COMITÉ PROGRAMA

Juan Hernández	Universidad de Extremadura

MIEMBROS COMITÉ DE PROGRAMA

José Aldana	Universidad de Málaga
Maria José Aramburu	Universidad de Castellón
Luís Arriaga da Cunha	Universidad do Évora
Orlando Belo	Universidad do Minho
Pere Botella	Universidad de Politécnica de Cataluña
Nieves Brisaboa	Universidad de La Coruña
Matilde Celma	Universidad Politécnica de Valencia
Rafael Corchuelo	Universidad de Sevilla
Carmen Costilla	Universidad Politécnica de Madrid
Yania Crespo	Universidad de Valladolid
Carlos Delgado Kloos	Universidad Carlos III
Óscar Díaz	Universidad del País Vasco
João Falcão e Cunha	Universidad do Porto
Xavier Franch	Universidad Politécnica de Cataluña
Mario J. Gaspar da Silva	Universidad de Lisboa
Jaime Gómez	Universidad de Alicante
Alfredo Goñi	Universidad del País Vasco
Natalia Juristo	Universidad Politécnica de Madrid
Antonia Lopes	Universidad de Lisboa
Henrique Madeira	Universidad de Coimbra
Esperanza Marcos	Universidad Rey Juan Carlos
Ana Moreira	Universidad. Nova de Lisboa
Juan José Moreno	Universidad Politécnica de Madrid
Juan Manuel Murillo	Universidad Extremadura
Óscar Pastor	Universidad Politécnica de Valencia
Mario Piattini	Universidad de Castilla la Mancha
Antonio Polo	Universidad de Extremadura
Celia Ramos	Universidad do Algarve
Isidro Ramos	Universidad Politécnica de Valencia
José C. Riquelme	Universidad de Sevilla
Antonio Rito Silva	Universidad Técnica de Lisboa
Francisco Ruiz	Universidad de Castilla la Mancha
José Samos	Universidad de Granada
Isabel Sofia Sousa Brito	Instituto Politécnico Beja
Miguel Toro	Universidad de Sevilla
Ambrosio Toval	Universidad de Murcia
José M. Troya	Universidad. de Málaga
Toni Urpi	Universidad Politécnica de Cataluña
Antonio Vallecillo	Universidad de Málaga

patrones y los índices invertidos. Asimismo, se ha presentado un caso práctico de cómo pueden ser fácilmente integrados en un sistema completo.

Como trabajos futuros se plantea la posibilidad de aplicar búsquedas por contenido sobre textos comprimidos. Existen en la actualidad algoritmos de compresión que además de reducir sensiblemente el tamaño de los textos almacenados en la biblioteca, permiten realizar búsquedas directamente sobre el texto comprimido más rápido que si se hiciera sobre la versión normal.

9 Referencias

[1] [en línea] : *Biblioteca Virtual Lluís Vives.* [consulta: 9 de junio de 2004]. Universitat d'Alacant, Institut Joan Lluís Vives. <http://www.lluisvives.com/>

[2] [en línea] : *CELT: corpus of electronic texts.* [consulta: 9 de junio de 2004]. University College Cork. <http://www.ucc.ie/celt/>

[3] [en línea] : *Biblioteca Virtual Galega* [consulta: 9 de junio de 2004]. Universidade da Coruña. <http://bvg.udc.es/>

[4] Baeza-Yates, R.; Ribeiro Neto, Berthier: *Modern Information Retrieval.* Addison Wesley Longman, 1999

[5] Peña, R.; Baeza-Yates, R. et al; *Gestión digital de la información: de Bits a Bibliotecas Digitales y la Web.* RA-MA Editorial, 2002

[6] [en línea]: *ACM digital library* [consulta: 9 de junio de 2004]. Association for Computing Machinery <http://portal.acm.org/dl.cfm?coll=portal&dl=ACM&CFID=22465613&CFTOKEN=99693354>

[7] [en línea] : *The Universal Library* [consulta: 9 de junio de 2004]. Carnegie Mellon Univesity. <http://www.ulib.org/html/index.html>

[8] [en línea]: *Proyecto Gutemberg* [consulta: 9 de junio de 2004]. <http://www.gutenberg.net/>

[9] [en línea]: *Biblioteca Virtual Miguel de Cervantes* [consulta: 9 de junio de 2004]. <http://www.cervantesvirtual.com/>

[10] [en línea]: *Cornell University Library Gateway* [consulta: 9 de junio de 2004]. Cornell University <http://campusgw.library.cornell.edu/>

[11] [en línea]: *Gallica, Bibliothèque numérique de la Bibliothèque nationale de France* [consulta: 9 de junio de 2004]. Bibliothèque nationale de France. <http://gallica.bnf.fr/>

[12] [en línea]: Jakarta Lucene [consulta: 22 de junio de 2004]. The Apache Software Foundation. <http://jakarta.apache.org/lucene/docs/index.html>

[13] Navarro, G.; Raffinot, M: *Flexible pattern matching in strings.* Cambridge University Press, 2002.

[14] Knuth, D. E.; Morris, J. H.; Pratt, V. R.: Fast pattern matching in strings, *SIAM Journal on Computing* 6 (1): 323 – 350.

[15] Navarro, G.; Raffinot, M: A Bit-Parallel Approach to Suffix Automata: Fast Extended String Matching, in *Proceeding of the 9th Annual Symposium on Combinational Pattern Matching*, Lecture Notes in Computer Science 1448, Springer-Verlag, Berlin, 14 –31

[16] Levenshtein, V. I., Binary codes capable of correcting deletions, insertions, and reversals, Doklady Akademii Nauk SSSR, 163(4):845-848, 1965.

Incorporando Control de Acceso y Auditoría en el Modelado Multidimensional de Almacenes de Datos

Rodolfo Villarroel[1], Eduardo Fernández-Medina[2], Juan Trujillo[3], Mario Piattini[2]

(1) Dep. Computación e Informática. Universidad Católica del Maule (Chile)
rvillarr@spock.ucm.cl
(2) Dep. Informática. Universidad de Castilla-La Mancha (España)
{Eduardo.FdezMedina, Mario.Piattini}@uclm.es
(3) Dep. Lenguajes y Sistemas Informáticos. Universidad de Alicante (España)
jtrujillo@dlsi.ua.es

Resumen. Los Almacenes de Datos (*Data Warehouses, DW*) se usan como un mecanismo muy poderoso para descubrir información de negocio crucial. Considerando la extrema importancia de la información manejada por este tipo de aplicaciones, es esencial definir medidas de seguridad desde las primeras etapas del diseño del DW en el modelado MD y hacerlas cumplir. En los últimos años, se han propuesto aproximaciones para representar las principales propiedades del modelado MD a nivel conceptual. Sin embargo, ninguna de ellas considera la seguridad como un elemento importante en sus modelos, y por tanto, no permiten especificar restricciones de confidencialidad para ser cumplidas por las aplicaciones que usarán estos modelos. En este artículo, presentamos un modelo de Control de Acceso y Auditoría (ACA) para modelos MD. A continuación, extendemos el Lenguaje de Modelado Unificado (*Unified Modeling Language, UML*) con el modelo ACA que nos permite incluir la información de seguridad (reunidas en el modelo ACA) en modelos conceptuales MD, permitiéndonos diseñar modelos MD seguros. Además, usamos OCL (*Object Constraint Language*) para especificar las restricciones del modelo ACA, evitando un uso arbitrario de las mismas. De esta forma, integramos la seguridad en el modelado MD, cumpliendo uno de los requisitos más solicitados en ingeniería de software.

1. Introducción

El Modelado Multidimensional (MD) es la base de los Almacenes de Datos (*Data Warehouses, DW*), Bases de Datos Multidimensionales, y aplicaciones de Procesamiento Analítico En-Línea (*On-Line Analytical Processing, OLAP*). Estos sistemas se usan como un mecanismo muy poderoso para descubrir información de negocio crucial en los procesos de toma de decisiones estratégicas. Muchos gobiernos están muy preocupados respecto a la privacidad (también importante para la empresa), y promulgan leyes para proteger la privacidad individual, tales como el HIPAA (*Health Insurance Portability and Accountability Act*), que regula la privacidad de la información de salud de las personas. Por lo tanto, considerando que

la supervivencia de las organizaciones depende de la correcta gestión, seguridad y confidencialidad de la información [8], y la extrema importancia de la información que se puede descubrir usando este tipo de aplicaciones, es crucial especificar medidas de confidencialidad en el modelado MD, y hacerlas cumplir. De hecho, como algunos autores defienden [7, 9], la información de seguridad es un requisito serio que se debe considerar, no como un aspecto aislado, sino como algo presente en todas las etapas del ciclo de vida de desarrollo. Además, la confidencialidad es un requisito especialmente importante para aplicaciones basadas en modelos MD, ya que la información de negocio, que es muy sensible, puede ser descubierta ejecutando simples consultas. En el mundo real de implementaciones de DW, esto se realiza normalmente limitando las operaciones OLAP a los usuarios finales. Sin embargo, creemos profundamente que estos aspectos de seguridad deberían también ser considerados junto a los datos en el correspondiente modelo conceptual MD. De esta forma, consideraremos los aspectos de seguridad de DW desde las etapas tempranas de un DW y hacerlas cumplir en la futura implementación.

Podemos considerar tres tecnologías que han sido ampliamente usadas para proteger la información contra el acceso o modificaciones inapropiadas: Autenticación, Control de Acceso, y Auditoría, que en conjunto proveen la base para la seguridad de la información [18]. Autenticación permite comprobar que el usuario final es quien dice ser. Control de acceso determina los permisos con respecto a recursos y objetos. La auditoría recopila datos respecto a la actividad del sistema y los analiza para descubrir violaciones de seguridad o para diagnosticar su causa. Autenticación es un mecanismo que es independiente del diseño y depende más de las políticas de la compañía, y por lo tanto, está fuera del alcance de este artículo. Sin embargo, el control de acceso y la auditoría tienen un componente importante de diseño. De hecho, las consideraciones sobre control de acceso y auditoría se deberían tomar en cuenta a través de todo el proceso de diseño, y no sólo cuando el sistema está completamente desarrollado. Por lo tanto, consideramos que estos aspectos deberían ser incluidos en el modelado conceptual de un sistema, además de considerar todos los aspectos de seguridad en la implementación del sistema final.

En este artículo, definimos un modelo preliminar de Control de Acceso y Auditoría (ACA) que nos permite especificar consideraciones de control de acceso y auditoría en el modelado conceptual MD de DW's. Debido a la variedad de propuestas de modelos MD y procesos de diseño de DW's, este modelo debería ser independiente, pero fácilmente adaptable a cualquiera de estas propuestas. Nos hemos basado en la propuesta [14, 19], que permite fácilmente modelar las propiedades MD a nivel conceptual y, hemos extendido UML para que nos permita especificar todos los conceptos definidos previamente en el modelo ACA. Según nuestro conocimiento, éste es el primer enfoque formal para el diseño conceptual de DW's que considera aspectos de seguridad y auditoría como parte del modelado conceptual.

El resto de este artículo se estructura como sigue: La sección 2 introduce el trabajo relacionado. La sección 3 propone nuestro modelo ACA. La sección 4 presenta un caso de estudio y aplica nuestro modelo ACA y extensión de UML para el modelado MD seguro. Finalmente, la sección 5 presenta las principales conclusiones y nuestro trabajo futuro inmediato.

2. Trabajo Relacionado

Muchas propuestas han sido desarrolladas para proteger la información del acceso inadecuado. Todas ellas explotan las particularidades de los sistemas con los cuales tratan, tales como los tipos de objetos, sujetos, privilegios, signos, resolución de conflictos, etc. Por ejemplo, hay modelos para archivos de datos [2], sistemas de bases de datos [1, 15], documentos XML [6], e incluso para documentos multimedia [5]. Por otro lado, existen algunas propuestas interesantes que tratan de definir un modelo de autorización para DW's [11, 12, 21, 22], pero tratan principalmente con operaciones OLAP, y no han sido concebidas para ser integradas en el modelado MD. Además, estas propuestas sólo consideran el control de acceso y no la auditoría.

Sugerimos leer [20] para una referencia de trabajos relacionados con modelado multidimensional e integración de la seguridad dentro del proceso de diseño.

3. Modelo de Control de Acceso y Auditoría (ACA)

El Control de Acceso no es una solución completa para la seguridad de un sistema [18] debido a que debe ser acoplado con la auditoría, para el registro y análisis posterior de todos los requisitos y actividades de los usuarios. Por lo tanto, en nuestro enfoque, consideramos ambos conceptos para que sean integrados en el diseño del modelado conceptual MD.

Aunque existen muchos modelos de autorización que permiten una especificación fácil y flexible de autorizaciones, éstos dependen de las propiedades particulares del modelo de datos considerado [10]. Como resultado, estos modelos de autorización no se pueden extender con facilidad a otros modelos de datos, tales como el modelo MD.

Los modelos de control de acceso están típicamente compuestos de un conjunto de reglas de autorización que regulan el acceso a los objetos. Cada regla de autorización generalmente especifica el sujeto al cual se aplica la regla, el objeto al cual se refiere la autorización, la acción a la cual se refiere la regla, y el signo que describe si la regla otorga un permiso o deniega un acceso.

De manera de regular el acceso a los objetos en un modelo MD, hemos considerado el modelo de Control de Acceso Obligatorio (*Mandatory Access Control, MAC*), y un conjunto de reglas de autorización, que representan excepciones a las reglas multinivel de manera general. Así, el modelo ACA estará compuesto de un conjunto de reglas de asignación de información de seguridad, donde el diseñador define la información de seguridad para todos los elementos del modelo MD, un conjunto de reglas de autorización donde el diseñador puede especificar diferentes situaciones en la cual las reglas multinivel deberían ser cumplidas, y finalmente, un conjunto de reglas de auditoría, que representan los requisitos de auditoría que considera el diseñador.

En las siguientes subsecciones, introducimos todos los detalles del modelo ACA: El modelo de control de acceso que ha sido considerado, sujetos de autorización, objetos de autorización, acciones, reglas de asignación de información sensible, reglas de autorización, reglas de auditoría, y resolución de conflictos.

3.1. El Modelo de Control de Acceso

El modelo de control de acceso que hemos considerado como base para los modelos MD es MAC, el cual ha sido ampliamente estudiado [16, 17] y muchas vulnerabilidades han sido detectadas, tales como su falta de flexibilidad, la poli-instanciación, etc. Sin embargo, la mayoría de estos problemas son provocados por las operaciones de lectura y escritura en el sistema. Afortunadamente, podemos considerar inicialmente que la única operación que será usada por los usuarios en este tipo de sistemas es *read*, de esta manera, MAC sería apropiada. Además, MAC está siendo integrada en algunos de los más importantes SGBD, tales como Oracle*9i* Label Security [13] y DB2 Universal Database (UDB) [4]. Esto es importante, debido a que los modelos MD podrían ser implementados por alguno de estos SGBDs.

Hemos considerado en nuestro modelo tres diferentes pero compatibles formas de clasificación de usuarios, por su nivel de seguridad, el rol de usuario que ellos juegan y las categorías de usuario a la cual pertenecen:

- Niveles de seguridad: Indican el nivel de acreditación del usuario.
- Roles de usuario. Cada organización usa para organizar a sus empleados una estructura de roles jerárquica, de acuerdo a las responsabilidades de cada tipo de trabajo. Cada usuario puede jugar uno o más roles.
- Categorías de usuario. Cada organización también usa para su organización un conjunto de categorías o grupos organizacionales, tales como separación territorial, área de trabajo, etc. Cada usuario puede pertenecer a una o más categorías.

Así, por cada objeto del modelo, los requisitos de acceso del usuario pueden ser definidos, y especificados con gran precisión lo que puede acceder de cada objeto. De esta manera, la regla de control de acceso para operaciones *read* es la siguiente: Un usuario puede acceder a una información sólo si, a) el nivel de seguridad del usuario es mayor o igual que el nivel de seguridad de la información, b) todas las categorías de usuario que han sido definidas para la organización están asignadas al usuario, y c) el usuario juega al menos uno de los roles que han sido definidos para la información.

3.2. Sujetos de Autorización

La especificación de sujetos en las reglas de control de acceso tiene a menudos dos requisitos aparentemente opuestos [16]. Por un lado, una referencia a un sujeto debe ser simple, de manera de permitir un control de acceso eficiente y para explotar las posibles relaciones entre sujetos en la resolución de conflictos entre las autorizaciones (por ejemplo, las relaciones más específicas entre roles y subroles). Por otro lado, uno quisiera ver más expresividad que la simple referencia a identidades o roles de usuario, categorías o niveles de seguridad, proveyendo soporte de autorizaciones dependientes de un perfil cuya validez depende de las propiedades asociadas con los usuarios (por ejemplo, edad, nacionalidad, etc.). Nuestra solución satisface ambos requisitos soportando a la vez conceptos de clasificación de usuarios (niveles de seguridad, roles y categorías) y perfiles de usuario. El perfil puede ser modelado como un objeto estructurado que representa todas las propiedades relevantes de cada sujeto. Por lo tanto, el componente *sujeto* de nuestro modelo ACA incluye dos partes:

- Una *identidad*, que puede estar compuesta de uno o más de los siguientes atributos:
  - *userID*: El identificador del usuario.
  - *roleId*: Los identificadores de uno o más roles de usuario.
  - *compartmentID*: Los identificadores de una o más categorías de usuario.
  - *securityLevel*: El nombre de un nivel de seguridad o el intervalo de niveles de seguridad.
- Una *subjectExpression*: Es una expresión OCL sobre perfiles de usuario.

Consideramos, para la definición de los elementos en el modelo ACA, una gramática que utiliza notación EBNF (*Extended Backus Naur Form*), en la cual | significa una elección, ? significa opcionalmente, * significa cero o más veces, y + significa una o más veces. Los sujetos de autorización son definidos como se ilustra en la Tabla 1 (a).

Tabla 1. Gramática ACA

a)	Subjects := subjectIdentification subjectExpression subjectIdentification := subjectIdentifier (logicalOperator subjectIdentifier)* subjectIdentifier := ("ALLSUBJECTS" | "ID" userID) | ("RID" roleID) | ("CID" compartmentID) | ("SL" securityLevel) logicalOperator := "AND" | "OR" subjectExpression := ("COND" OCLExpression[1])*
b)	Objects := objectIdentifier objectExpression objectIdentifier := ("CL" className) | ("ATT" className"."attributeName) objectExpression := ("COND" OCLExpression)*
c)	Actions := action (logicalOperator action)* Action := "READ" | …
d)	SIAR := "OBJECTS" Objects ("INVCLASSES" involvedClasses)? ("SECINF" securityInformation | "COND" conditionAssignment) involvedClasses := Objects (logicalOperator Objects)* securityInformation := ("SL" securityLevel)? ("SR" userRole+)? ("SC" userCompartment+)? conditionAssignment := "IF" booleanExpression "THEN" (securityInformation | conditionAssignment) ("ELSE" (securityInformation | conditionAssignment))? "ENDIF"
e)	AUR := "SUBJECTS" Subjects "OBJECTS" Objects "ACTIONS" Actions "SIGN" Sign ("INVCLASSES" involvedClasses)? Sign := "+" | "-"
f)	AR := "OBJECTS" Objetos "LOGTYPE" logType "LOGINFO" logInformation logType := "none" | "all" | "frustratedAttempts" | "successfulAttempts" logInformation := subjectID? objectID? action? Time? response?

Por ejemplo, el elemento *sujeto "**RID** administrativo **AND CID** USA **AND CID** servicios financieros **COND** profile.edad>18"* denota todos los sujetos que juegan un rol *administrativo*, pertenecen a las categorías *USA* y *servicios financieros*, y también satisfacen la condición.

3.3. Objetos de Autorización

De acuerdo a la descripción de modelos MD presentada en [14, 19], identificamos estos tipos de objetos de protección: clases de hecho, clases de dimensión, clases base, atributos e instancias. El componente *objeto* de nuestro modelo ACA incluye dos partes:

- Una *identidad*, que puede ser uno de los siguientes subatributos:
  - *className*: El identificador de la clase. Éste puede estar relacionado a un hecho, dimensión o clase base.
  - *attributeName*: El identificador del atributo.

[1] Representa una condición basada en los elementos del diagrama de clases (por ejemplo, clases de hecho, clases de dimensión, etc.)

– Una *objectExpression* la cual es una expresión OCL en el modelo de clases que representa el modelo MD. Esta condición selecciona algunas instancias de una clase.

Los objetos de autorización son definidos en la Tabla 1 (b). Por ejemplo, el elemento objeto "***CL** diagnóstico **COND** diagnóstico.tipo=SIDA*" denota todas las instancias de la clase *Diagnóstico* cuyo tipo de atributo tiene el valor *SIDA*.

3.4. Acciones de Autorización

Para simplicidad, en este modelo ACA preliminar, sólo hemos considerado la acción *read*. Otras acciones relevantes orientadas a bases de datos para procesos ETL (*Extraction-Transformation-Loading*) tales como *insert, delete, update*, y tal vez otras acciones orientadas a OLAP tales como *drilling-through, drilling-down, rolling-up, slice, dice*, etc., están fuera del alcance de este trabajo, y serán tratados en trabajos futuros. Por lo tanto, la componente de acción de nuestro modelo tiene sólo un elemento que identifica el tipo de acción (ver Tabla 1 (c))

3.5. Reglas de Asignación de Información de Sensibilidad

Debido a que el control de acceso que hemos considerado es MAC, tenemos que especificar la información de sensibilidad de cada elemento en el modelo multinivel. Así, para cada regla de asignación de información de seguridad (SIAR), necesitamos especificar los siguientes conceptos:
– *Objetos* a los cuales la regla es aplicable.
– *Información de seguridad* que es asignada. Como mencionamos previamente, podemos especificar información de sensibilidad por medio de niveles de seguridad, roles de usuario y categorías de usuario.
– *Clases involucradas* en la consulta. En modelos MD, la sensibilidad de la información puede depender de las clases involucradas en una consulta.
– *Una condición* puede ser especificada para asignar diferente información de sensibilidad de acuerdo al valor de algunos atributos. Si se define una condición, diferentes valores de sensibilidad deben ser especificados, dependiendo de la evaluación de la condición.

Las reglas de asignación de información de sensibilidad son definidas como se ilustran en la Tabla 1 (d). Podemos considerar el siguiente ejemplo de SIAR: "***OBJECTS CL** diagnóstico **COND IF** diagnóstico.areaSalud=oncología **THEN SL** Secret **ELSE SL** Confidential **ENDIF***". Este SIAR define el nivel de seguridad para cada instancia de la clase *diagnóstico* dependiendo de la evaluación de esa condición.

3.6. Reglas de Autorización

Hay diferentes tipos de reglas de autorización (AUR). La más importante son implícita, explícita, positiva, negativa, débil y fuerte [9]. Además, los modelos de control de acceso pueden ser abiertos o cerrados [16]. En un sistema multinivel, podemos considerar que todos los objetos tienen información de sensibilidad (la

menos restrictiva por defecto), así podemos considerar que nuestro sistema es cerrado. Sin embargo, en nuestro sistema tanto AURs positivos y negativos pueden coexistir. AURs implícitos y explícitos pueden también ser definidos, pero no consideraremos explícitamente autorizaciones fuertes y débiles. Por cada AUR, especificaremos los siguientes conceptos:
– *Sujetos* a los cuales la regla es aplicable.
– *Objetos* a los cuales la regla es aplicable.
– *Acciones* consideradas. Tal y como hemos comentado, sólo consideramos read
– *Signo*, que define si la autorización es positiva o negativa.
– *Clases involucradas* en la consulta. Especifica las clases que están involucradas en la consulta para que el AUR sea aplicable.

Es importante mencionar que si el AUR es positivo, la regla será evaluada en todos los usuarios que no pueden acceder a la información, de acuerdo a los criterios multinivel (ver Sección 3.8). Por otro lado, si el AUR es negativo, la regla será evaluada en todos los usuarios que tienen acceso a la información. Las reglas de autorización son definidas como se ilustra en la Tabla 1 (e).

Por ejemplo, podemos considerar el siguiente AUR: "***SUBJECTS RID** administrativo **AND CID** USA **AND CID** servicios financieros **COND** profile.edad>18 **OBJECTS CL** diagnóstico **COND** diagnóstico.tipo=SIDA **ACTIONS READ SIGN** +*"

AUR1 es una regla de autorización positiva que permite leer todas las instancias de la clase diagnóstico si el tipo de diagnóstico es SIDA, a todos los sujetos con el rol, categorías y edad especificada en la condición.

3.7. Reglas de Auditoría

Los controles de auditoría son útiles como un medio para analizar el comportamiento del usuario en el uso del sistema, de manera de descubrir posibles intentos o violaciones actuales. Adicionalmente, la auditoría es esencial para asegurar que los usuarios no autorizados no abusen de sus privilegios [18]. Las reglas de auditoría (AR) pueden ser especificadas considerando los siguientes conceptos:
– *Objetos* a los cuales la regla es aplicable.
– *Tipo de log*. Especifica si el acceso debe ser registrado. Los valores pueden ser ninguno, acceso total, sólo accesos frustrados, o sólo accesos satisfactorios.
– *Información que será registrada*. Según la situación, podemos registrar información como el sujeto que solicita el acceso, el objeto que será accedido, la operación requerida, el tiempo, y la respuesta del modelo de control de acceso.

Las reglas de auditoría son definidas como se ilustra en la Tabla 1 (f).

3.8. Resolución de conflictos

Algunos conflictos pueden aparecer entre diversos AURs, varios SIARs y aún entre AURs y SIARs. Podemos resolver estos conflictos considerando las siguientes reglas:
– Si hay conflicto entre un AUR positivo y un SIAR, los usuarios que serán capaz de acceder a la información estará compuesta por los usuarios que cumplen el SIAR y los usuarios que no cumplen el SIAR pero cumplen la condición del AUR.

- Si hay conflicto entre un AUR negativo y un SIAR, el conjunto de usuarios que no podrá tener acceso a la información serán los usuarios que no cumplen el SIAR y los usuarios que cumplen el SIAR pero cumplen la condición del AUR.
- Si hay un conflicto entre dos SIARs, la información de seguridad para cada instancia, será el resultado de aplicar el SIAR más restrictivo.
- Un AUR que se refiere a un usuario individual tiene más preferencia que cualquier otro AUR que se refiere a un conjunto de usuarios.
- Un AUR que se refiere a un rol de usuario particular *r* tiene más preferencia que cualquier AUR que se refiere a un ascendente de r en el árbol de roles de usuario.
- Dos AURs que se refieren a diferentes conjuntos de usuarios (por ejemplo, a una categoría y un rol) tienen la misma preferencia.
- Si dos AURs tienen la misma preferencia, seleccionaremos la negativa. Si tienen el mismo signo no hay conflicto.

4. Un caso de estudio de aplicación de nuestra extensión para el Modelado MD Seguro

En esta sección, aplicamos nuestro modelo ACA y extensión de UML (*profile*[2]) para el diseño conceptual de un modelo MD seguro en el contexto de un sistema típico de salud. De acuerdo a [3], una extensión de UML comienza con una breve descripción y a continuación se describen todos los estereotipos, valores etiquetados, y restricciones de la extensión. Sugerimos al lector referirse a [20] para una descripción completa de la extensión de UML para el diseño de modelos MD seguros. Hemos considerado un ejemplo reducido para poder enfocar nuestra atención en las especificaciones de seguridad. La Figura 1 (a) muestra la jerarquía simplificada de roles de usuario del sistema, y la Figura 1 (b) muestra los niveles de seguridad que han sido definidos[3]. En este ejemplo no hemos considerado categorías de usuario.

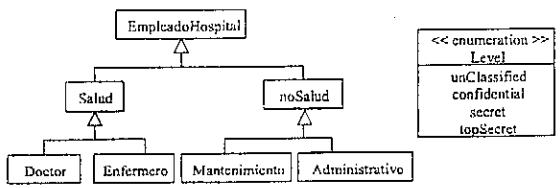

Fig. 1. (a) Jerarquía de roles de usuario (b) Niveles de seguridad

Considerando el modelo MD que es mostrado en la Figura. 2, el cual está compuesto de la clase de hecho *Admisión*, clases de dimensión *Diagnóstico* y

[2] Un *profile* extiende un tipo de diagrama UML existente para un uso diferente. Es especificado por medio de los mecanismos de extensibilidad provistos por UML (estereotipos, propiedades y restricciones) de manera de adaptarse a un nuevo método o modelo.

[3] *Level* (Fig. 1) es un nuevo tipo de dato heredado del tipo de dato *enumeration* de UML.

Paciente, y las clases base *Grupo_Diagnos y* Ciudad, podemos definir su modelo ACA definiendo los SIARs, AURs and ARs que son especificados en la Tabla 2.

Tabla 2. Modelo ACA

SIAR 1	Por cada instancia de la clase de hecho *Admisión*, el nivel de seguridad será al menos *Secreto*, y los roles de seguridad serán *Salud* y *Admin*.
	`OBJECTS CL Admisión SECINF SL Secret SR (Salud, Admin)`
SIAR 2	Los roles de seguridad para el atributo *coste* de la clase de hecho *Admisión* será sólo *Admin*
	`OBJECTS ATT Admisión.coste SECINF SR Admin`
SIAR 3	Por cada instancia de la clase de dimensión *Diagnóstico*, el nivel de seguridad será al menos *Secreto* y el rol de seguridad será *Salud*
	`OBJECTS CL Diagnóstico SECINF SL Secret SR Salud`
SIAR 4	Por cada instancia de la clase de dimensión *Paciente*, el nivel de seguridad será al menos *Secreto* y los rol es de seguridad serán *Salud* y *Admin*
	`OBJECTS CL Paciente SECINF SL Secret SR (Salud, Admin)`
SIAR 5	El rol de seguridad para el atributo *dirección* de la clase de dimensión *Paciente* será sólo *Admin*
	`OBJECTS ATT Paciente.dirección SECINF SR Admin`
SIAR 6	Por cada instancia de la clase base *Grupo_Diagnos*, el nivel de seguridad será al menos *Confidencial*
	`OBJECTS CL Grupo_diagnos SECINF SL Confidential`
SIAR 7	Por cada instancia de la clase base *Ciudad* , el nivel de seguridad será al menos *Confidencial*
	`OBJECTS CL Ciudad SECINF SL Confidential`
SIAR 8	Por cada instancia de la clase de hecho *Admisión*, si la descripción del grupo de diagnóstico o su diagnóstico es especialmente sensible (cáncer o SIDA), entonces su nivel de seguridad será *altoSecreto*, en caso contrario será *Secreto*.
	`OBJECTS CL Admisión INVCLASSES CL Diagnóstico AND CL Grupo_diagnos AND CL Paciente COND IF self.Diagnóstico.Grupo_Diagnos.descripción='Cáncer' or self.Diagnóstico.Grupo_Diagnos.descripción='SIDA' THEN SL topSecret ELSE SL Secret ENDIF`
SIAR 9	Por cada instancia de la clase de hecho *Admission*, si su costo es mayor que $10000, entonces su nivel de seguridad será *altoSecreto*, en caso contrario será *Secreto*
	`OBJECTS CL Admisión INVCLASSES CL Paciente COND IF self.coste>10000 THEN SL topSecret ELSE SL Secret ENDIF`
AUR 1	Si una consulta involucra las clases *Diagnóstico*, *Grupo_Diagnos* y *Paciente*, la información que puede ser obtenida sobre los pacientes es muy sensible. Así que permitiremos el acceso de la información sólo a los miembros del rol de seguridad *Salud*, y si su área de trabajo es la misma que el área de salud de los pacientes.
	`SUBJECTS RID Salud COND userProfile.areaDeTrabajo<>self.diagnóstico.areaSalud OBJECTS CL Admisión ACTION READ SIGN - INVCLASSES CL Diagnóstico AND CL Grupo_diagnos AND CL Paciente`
AUR 2	Los pacientes serán usuarios especiales del sistema, por lo tanto, desearemos que ellos puedan tener acceso a sus propios datos.
	`SUBJECTS ALLSUBJECTS COND userProfile.nombre=self.nombre OBJECTS CL Paciente ACTION READ SIGN +`
AR 1	Deseamos registrar el sujeto, objeto y tiempo para todos los intentos de acceso frustrados, de manera de analizar quién trata de acceder ilegalmente a la información de nuestro DW.
	`OBJECTS CL Admisión LOGTYPE frustratedAttempts LOGINFO subjectID ObjectID time`

La Figura 2 muestra un modelo MD que incluye todas las clases descritas previamente, y una clase adicional (*UserProfile*). La clase *UserProfile* (estereotipo *UserProfile*) contiene la información de todos los usuarios que tendrán acceso a este modelo MD. Se puede observar que usamos varios valores etiquetados para modelar todas nuestras reglas del modelo ACA. SIAR 1 es representada definiendo los valores etiquetados asociados al nivel de seguridad *SL* (*security level*) y a los roles de

seguridad *SR* (*security roles*) en la clase de hecho *Admisión*. La misma estrategia es usada para modelar los SIAR 2 al 7. SIAR 8 y 9 son modelados con notas UML (rotuladas con números 1 y 2 respectivamente) asociadas con la clase *Admisión*. En estas notas, incluimos valores etiquetados que nos permiten representar todos los conceptos importantes que han sido previamente identificados en el modelo ACA. AUR 1 y 2 son modeladas con las notas 4 y 5 respectivamente, y finalmente, AR 1 es modelada con la nota 3.

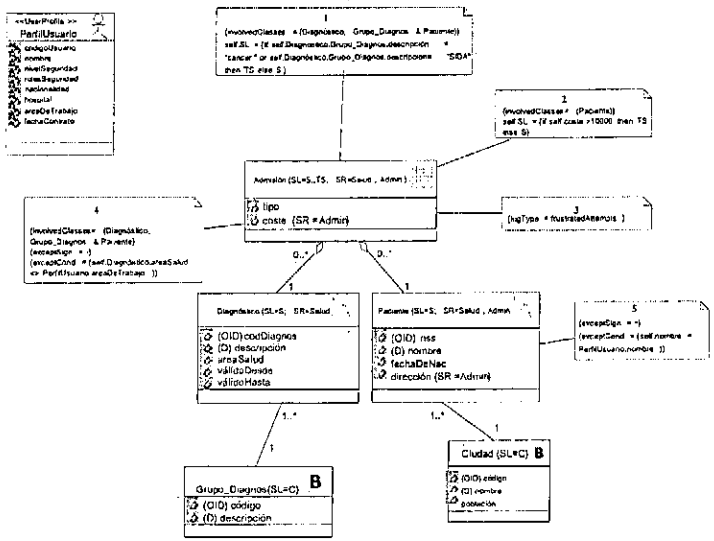

Fig 2. Ejemplo de modelo multidimensional con información y restricciones de seguridad

5. Conclusiones y Trabajo Futuro

En este artículo hemos presentado un modelo de Control de Acceso y Auditoría (ACA) que nos permite especificar y representar los principales aspectos de confidencialidad en el modelado conceptual de DW. El modelo ACA nos permite definir reglas para especificar la información de seguridad (SIAR), reglas para la representación de reglas de autorización (AUR) y reglas que nos permiten especificar requisitos de auditoría (AR). La extensión de UML contiene los estereotipos necesarios, valores etiquetados y restricciones para realizar un modelado MD seguro. Hemos usado OCL para especificar las reglas ACA y las restricciones asociadas a estos nuevos elementos definidos, evitando así su uso arbitrario.

Nuestro trabajo futuro inmediato es mejorar el modelo ACA, extendiendo el conjunto de privilegios considerado en este artículo (read) que nos permita especificar aspectos de seguridad en los procesos ETL (*Extraction-Transformation-Loading*) considerando otras operaciones como *delete, insert* y *update*. Además, abordaremos temas de implementación para utilizar los aspectos de seguridad considerados cuando consultemos un modelo MD a partir de herramientas OLAP.

Agradecimientos

Esta investigación es parte de los proyectos CALIPO (TIC2003-07804-C05-03) y RETISTIC (TIC2002-12487-E) de la Dirección General de Investigación del Ministerio de Ciencia y Tecnología, y la red VII-J.RITOS2 financiada por CYTED.

Referencias

1. Bertino, E., Jajodia, S., y Samarati, P., *A Flexible Authorization Mechanism for Relational Data Management Systems*. ACM Transactions on Information Systems. 17 (1999) 101-140
2. Bonatti, P., Damiani, E., De Capitani di Vimercati, S., y Samarati, L. *An Access Control Model for Data Archives*. in *IFIP-TC11 International Conference on Information Security*. Paris, France (2001)
3. Conallen, J., *Building Web Applications with UML*. Object Technology Series.Addison-Wesley (2000)
4. Cota, S., *For Certain Eyes Only*. DB2 Magazine. 9(1) (2004) 40-45
5. Damiani, E., De Capitani di Vimercati, S., Fernandez-Medina, E., y Samarati, P. *An Access Control System for SVG Documents*. in *Sixteen Annual IFIP WG11.3 Working Conference on Data and Application Security*. Cambridge (U.K.) (2002)
6. Damiani, E., De Capitani di Vimercati, S., Paraboschi, S., y Samarati, P., *A Fined-Grained Access Control System for XML Documents*. ACM Transactions on Information and Systems Security. 5 (2002) 169-202
7. Devanbu, P. y Stubblebine, S., *Software engineering for security: a roadmap*, in *The Future of Software Engineering*, Finkelstein, A., Editor, ACM Press (2000) 227-239
8. Dhillon, G. y Backhouse, J., *Information system security management in the new millennium*. Communications of the ACM. 43(7) (2000) 125-128
9. Ferrari, E. y Thuraisingham, B., *Secure Database Systems*, in *Advanced Databases: Technology Design*, Piattini, M. y Díaz, O., Editors, Artech House: London (2000)
10. Jajodia, S., Samarati, P., Sapino, M.L., y Subrahmanian, V.S., *Flexible Support for Multiple Access Contro Policies*. ACM Transactions on Database Systems. 26 (2001) 214-260
11. Katic, N., Quirchmayr, G., Schiefer, J., Stolba, M., y Min Tjoa, A. *A Prototype Model for Data Warehouse Security Based on Metadata*. in *9th International Workshop on Database and Expert Systems Applications (DEXA'98)*. Vienna, Austria.: IEEE Computer Society (1998)
12. Kirkgöze, R., Katic, N., Stolda, M., y Min Tjoa, A. *A Security Concept for OLAP*. in *8th International Workshop on Database and Expert System Applications (DEXA'97)*. Toulouse, France: IEEE Computer Society (1997)

13. Levinger, J., *Oracle label security. Administrator's guide. Release 2 (9.2)*http://www.csis.gvsu.edu/GeneralInfo/Oracle/network.920/a96578.pdf (2002)
14. Luján-Mora, S., Trujillo, J., y Song, I.Y. *Extending the UML for Multidimensional Modeling*. in *5th International Conference on the Unified Modeling Language (UML 2002)*. Desden, Germani: Springer-Verlag (2002)
15. Rabitti, F., Bertino, E., Kim, W., y Woelk, D., *A Model of Authorization for Next-Generation Database Systems*. ACM Transactions on Database Systems. 16(1) (1991) 88-131
16. Samarati, P. y De Capitani di Vimercati, S., *Access control: Policies, models, and mechanisms*, in *Foundations of Security Analysis and Design*, Focardi, R. y Gorrieri, R., Editors, Springer: Bertinoro, Italy (2000) 137-196
17. Sandhu, R. y Chen, F., *The Multilevel Relational Data Model*. ACM Transactions on Information and Systems Security (TISSEC). 1(1) (1998) 93-132
18. Sandhu, R. y Samarati, L., *Authentication, Access Control, and Intrusion Detection*, in *CRC Handbook of Computer Science and Engineering*, Tucker, A., Editor, CRC Press Inc. (1997)
19. Trujillo, J., Palomar, M., Gómez, J., y Song, I.Y., *Designing Data Warehouses with OO Conceptual Models*. IEEE Computer, special issue on Data Warehouses, (34) (2001) 66-75
20. Villarroel, R., Fernandez-Medina, E., Trujillo, J., y Piattini, M. *Un profile de UML para diseñar almacenes de datos seguros*. in *IX Jornadas de Ingeniería del Software y Bases de Datos (JISBD 04)*. Málaga, España (2004)
21. Wang, L., Jajodia, S., y Wijesekera, D. *Securing OLAP Data Cubes Against Privacy Breaches*. in *IEEE Symposium on Security and Privacy*. Berkeley, California (2004)
22. Weippl, E., Mangisengi, O., Essmayr, W., Lichtenberger, F., y Winiwarter, W. *An Authorization Model for Data Warehouses and OLAP*. in *Workshop on Security in Distributed Data Warehousing*. New Orleans, Louisiana, USA (2001)

Modelado Navegacional desde una Perspectiva rientada a Servicios de Usuario

Paloma Cáceres, Esperanza Marcos, Valeria de Castro

Grupo de Investigación Kybele
Universidad Rey Juan Carlos
Madrid (España)
{pcaceres,emarcos,vcastro}@escet.urjc.es

Resumen. Hoy en día el desarrollo de Sistemas de Información Web (SIW) hace posible que a través de Internet se pueda tanto comprar un billete de avión como realizar búsquedas de información altamente específica. Pero debido a la falta de métodos adecuados para contruir el modelo navegacional de los SIW, los usuarios se pierden mientras navegan en el sistema puesto que desconocen qué pasos han de realizar para completar una operación determinada. Las metodologías de desarrollo de SIW existentes no han evolucionado tan rápidamente como los sistemas, y generalmente sólo contemplan la dimensión estructural de los sistemas y no la dimensión de comportamiento presente ya en los SIW actuales. Por este motivo, nuestra propuesta incorpora la dimensión de comportamiento al modelado navegacional de SIW. Pero además plantea el modelado desde una nueva perspectiva orientada a servicios de usuario, es decir, a los servicios específicos que el usuario requiere del sistema. El resultado es que el modelo navegacional obtenido representa, para cada uno de dichos servicios, una ruta específica que guía al usuario en la navegación a través del SIW. En este artículo, se presenta el método para la construcción del modelo navegacional desde esta nueva perspectiva a través de un caso de estudio.

1 Introducción

Gracias al desarrollo de sistemas de información Web (SIW), hoy en día es posible tanto un billete de avión como buscar información altamente específica a través de Internet. Ahora bien, ¿el modelado navegacional de los SIW actuales facilita la navegación del usuario a través del sistema teniendo en cuenta que no dispone de manual de usuario? Como usuarios de la Web, pensamos que la navegación de los SIW actuales, no ayuda ni guía al usuario a moverse a través del sistema. De hecho, el usuario generalmente se pierde mientras navega y desperdicia su tiempo intentando encontrar el modo de realizar una operación concreta.

Nosotros creemos que esto es debido a que la navegación de los SIW sigue heredando las costumbres de los inicios de la Web, donde lo único que se presentaba era información y de una manera estática. La Web ha seguido prosperando y ya existen complejos SIW. Pero los métodos de desarrollo no han evolucionado tanto y en general están enfocados, con alguna excepción, a la visualización de información.

Por este motivo empezamos a trabajar en esta línea. En trabajos previos [10], presentamos una aproximación que incorporaba la dimensión de comportamiento al modelado navegacional de SIW, partiendo del modelo de casos de uso. En la actualidad, nuestro trabajo ha realizado un cambio de enfoque respecto a cómo tener