



Deusto - Bilbao, 13 - 16 Octubre 1998

IV Congreso TECNOLOGIA OBJETOS

Facultad de
Ingeniería
ESIDE

Universidad de
Deusto



Prólogo	i
Índice	v
Comités	xi
Comité de Honor	xiii
Comité Organizador	xv
Comité de Programa Académico	xvii
Comité de Programa Industrial	xix
Conferenciantes	xxi
Brian Henderson-Sellers	xxiii
Bertrand Meyer	xxv
Trygve Reenskaug	xxvii
Kim Walden.....	xxix
Saba Zamir	xxxi
Ponencias de Partners	xxxiii
IBM: Simon Phipps	xxxv
ORACLE: Javier Castro	xxxvii
TRANSTOOLS: César Pérez Chirinos	xxxix
Ponencias de Colaboradores	xli
ESI: Ray Fernández Rupérez, Stephan Schuster	xliii
HEWLETT PACKARD: José Míghel Mahave	xliv
IDEAL OBJECTS: Luis M. Doreste.....	xlvi
INFORMÁTICA EL CORTE INGLÉS: Rodrigo Becerra.....	xlix
PROYECTOS ESPRIT: José Luis Fidalgo	li
SAP: José Manuel García	liii
STERLING SOFTWARE: Paul Geldart	lv
SYSECA: Enrique Martín	lvii
WORLDNET CONSULTING – OBJECTSPACE: Lois Goldwaite.....	lix
Ponencias Académicas	1
<i>La Clarividencia Razonada</i>	3
Ricardo Devis Botella, José Luis del Val Román	
<i>Sincronizador por Comportamientos</i>	5
Jesús Álvarez Gómez, Julio García Martín	
Universidad Politécnica de Madrid	
<i>Procesamiento de Periódicos Electrónicos para su Almacenamiento con Oracle 8</i>	19
M. J. Aramburu, I. Sanz, S. García, R. Berlanga	
Universitat Jaume I de Castellón	
<i>Propuesta de Definición de Métricas para Sistemas de Bases de Datos Objeto-Relacionales</i>	29
Coral Calero, Mario Piattini, Macario Polo, Francisco Ruiz	
Grupo ALARCOS, Universidad de Castilla – La Mancha	

<i>Una Visión de las Posibilidades del Estándar OpenGIS para la Interoperabilidad de Aplicaciones GIS sobre CORBA</i>	35
S. Comella, J. Zarazaga, J. Ezpeleta, D. Infante, R. López, P. R. Muro-Medrano Centro Politécnico Superior, Universidad de Zaragoza	
<i>Change Case Analysis</i>	41
Oscar Díaz, Juan José Rodríguez Universidad del País Vasco – Euskal Herriko Unibertsitatea	
<i>Esfuerzo en la Comprensión de Jerarquías de Clases Complejas</i>	53
Antonio Domínguez, Javier Rodeiro Universidad de Vigo	
<i>Invocación Implícita: Una Visión de Conjunto</i>	61
Luis Felipe Fernández, Juan Ramón López, Cristina Gómez, Antoni Olivé Universitat Politècnica de Catalunya, Universidade da Coruña	
<i>Entorno Visual Integrado para Gestión de Sistemas y Redes de Datos</i>	73
Luis L. Fernández, Nuria B. García Universidad Pontificia de Comillas	
<i>Integración de Técnicas de Inteligencia Artificial para la Reutilización de Clases en Java</i>	81
C. J. Fernández Conde, P. A. González Calero, M. B. Díaz Agudo Universidad Complutense de Madrid	
<i>YODA: Construcción de Aplicaciones Basada en Componentes</i>	89
D. Fernández Lanvín, R. Izquierdo Castanedo, A. A. Juan Fuente, J. M. Cueva Universidad de Oviedo	
<i>An Approach to Algorithm Design by Patterns</i>	97
Javier Galve-Francés, Julio García-Martín, José M. Burgos-Ortiz Universidad Politécnica de Madrid	
<i>Una Experiencia Práctica con la Arquitectura de Tres Modelos de OORAM</i>	107
Julio García Molina, Manuel E. Acacio Sánchez, Ginés García Mateos Universidad de Murcia	
<i>Problemática de la Entrada / Salida en C++, Comparación con Java</i>	121
José Baltasar García Pérez-Schofield, Manuel Pérez Cota Universidad de Vigo	
<i>Especificación Formal de Componentes en Java (Programación por Contrato)</i>	131
A. A. Juan Fuente, R. Izquierdo Castanedo, J. M. Cueva Lovelle, C. Viejo Vigil Proyecto Jedi – Universidad de Oviedo	
<i>Herencia Múltiple Débilmente Acoplada</i>	139
David Martínez Oliveira, Xulio Fernández Hermida Universidad de Vigo	

<i>Técnicas de Indexación para BDOviedo3</i>	149
A. B. Martínez, J. M. Cueva, D. Álvarez, M. García Universidad de Oviedo	
<i>Un Mecanismo de Control de Concurrencia para Objetos Replicados</i>	159
Francesc D. Muñoz Escoí, Pablo Galdámez Saiz, José M. Bernabéu Aubán Universidad Politécnica de Valencia	
<i>Gestión de Tareas de Usuario en ATOMS</i>	171
F. García, P. Rodríguez, J. Contreras, R. Moriyón Universidad Autónoma de Madrid, Instituto de Ingeniería del Conocimiento	
<i>Diseño de Ambientes Software Orientados al Objeto de Prototipación Automática</i>	179
José Romero, Oscar Pastor Universitat Politècnica de Valencia	
<i>CORBA y Java como Soporte de Aplicaciones y Servicios de Ingeniería Distribuida</i> .	187
Juan Carlos Yelmo, José Fernán Martínez Universidad Politécnica de Madrid, Universidad del Cauca (Colombia)	
<i>Una Aproximación para la Utilización de Metainformación en Aplicaciones de C++</i>	197
F. J. Zarazaga, S. Comella, J. Noguerras, J. Valiño Universidad de Zaragoza	

Propuesta de Definición de Métricas para Sistemas de Bases de Datos Objeto-Relacionales¹

Coral Calero, Mario Piattini, Macario Polo, Francisco Ruiz

Grupo ALARCOS

Departamento de Informática

Universidad de Castilla-La Mancha

Ronda de Calatrava, 5

13071 Ciudad Real

e-mail: {ccalero, mpiattin, mpolo, fruiz }@inf-cr.uclm.es

Resumen

Las métricas para los sistemas de bases de datos han sido descuidadas por la ingeniería del software. Sin embargo, hoy en día, estos sistemas tienen una alta productividad y un gran impacto económico. La aparición de nuevos SGBD avanzados pueden conllevar la aparición de dificultades para sus usuarios debido a la alta complejidad del esquema de estas bases de datos, que se caracterizaba en el modelo relacional clásico por su simplicidad.

Palabras clave

Bases de datos objeto-relacionales; métricas; complejidad; mantenibilidad.

1. Introducción

Prácticamente todas las métricas propuestas hasta la fecha se han centrado en medir programas [3]. Estas métricas fueron definidas para controlar algunos atributos internos de elementos software (como la complejidad) que influyen en atributos externos (por ejemplo la mantenibilidad) que son los que normalmente interesan a los responsables de desarrollo software [5].

Sin embargo, en los sistemas de información actuales, las bases de datos han cobrado un papel fundamental. Por ello creemos interesante trabajar en métricas para bases de datos, que no han sido suficientemente consideradas hasta el momento. Esta carencia puede ser debida al hecho de que las bases de datos han sido hasta hace poco "simples" tablas que no aportaban apenas complejidad al sistema en su conjunto.

¹ Esta investigación se encuadra dentro del proyecto MANTEMA realizado en colaboración con la empresa ATOS ODS, S.A. y subvencionado por la iniciativa ATYCA de la Dirección General de Tecnología y Seguridad Industrial del Ministerio de Industria y Energía.

Actualmente y debido a la aparición de bases de datos avanzadas, especialmente las orientadas a objetos, el componente de base de datos de un sistema de información ha ido incrementando su complejidad incorporando nuevos tipos de datos, reglas, generalizaciones, objetos complejos, etc. Recientemente, han aparecido nuevos productos "objeto-relacionales" en el mercado [6]. Este es el caso de Informix/Illustrá, Oracle 8 y DB2/2. Además, aunque con unos años de retraso, el nuevo estándar para SQL3 debería ser publicado en pocos meses.

En nuestra opinión, los sistemas de información desarrollados con estos nuevos SGBD se encontrarán con grandes problemas de mantenimiento debido principalmente al esquema de la base de datos. En este artículo proponemos un conjunto de métricas que permiten determinar la complejidad de un esquema de base de datos objeto-relacional.

2. Bases de datos objeto-relacionales

Las bases de datos objeto-relacionales combinan las características de las bases de datos tradicionales (modelo de datos, seguridad, concurrencia, lenguaje de alto nivel...) con los principios de la orientación a objetos (encapsulamiento, generalización, agregación, polimorfismo...). Estos productos ofrecen la posibilidad de definir clases o tipos abstractos de datos además de las tablas de las bases de datos relacionales.

También, se pueden definir jerarquías de generalización entre clases (super y subclases) y entre tablas (CREATE TABLE subtabla UNDER supertable).

Por otro lado, las bases de datos objeto-relacionales soportan generalmente herencia múltiple (una subtabla puede tener más de una supertable). Por tanto se pueden establecer dos tipos de asociaciones entre tablas como puede verse en la figura 1.

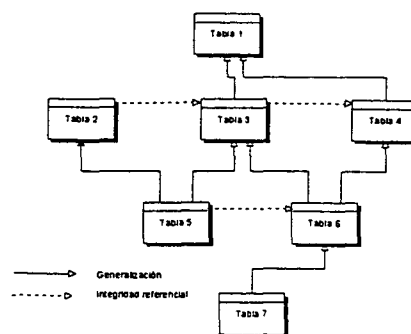


Figura 1. Ejemplo de esquema de base de datos objeto-relacional

Los atributos de una tabla se pueden definir en un dominio simple (por ejemplo char) o en una clase (por ejemplo un número complejo, una imagen...). Estas clases pueden, además, formar parte de una generalización.

3. Métricas para bases de datos objeto-relacionales

El objetivo de nuestra investigación es proponer un conjunto de métricas que puedan ser obtenidas automáticamente del catálogo de la base de datos. Este conjunto puede también ser suficiente para propósitos prácticos.

Siguiendo las definiciones dadas por [1] consideramos tres tipos de métricas.

3.1. Métricas de tamaño

3.1.1. Tamaño del sistema.

El tamaño de sistema (SS) se define como la suma de los tamaños de cada tabla del esquema.

$$SS = \sum_{i=1}^{NT} TS_i$$

Donde NT es el número de tablas del sistema

3.1.2. Tamaño de la tabla.

El tamaño de una tabla es la suma del tamaño de sus columnas simples (SCS), las cuales se consideran con un tamaño igual a 1 por lo que SCS corresponderá justamente con el número de columnas simples, junto con el tamaño de sus columnas complejas (CCS). Este tamaño se caracteriza mediante la siguiente expresión:

$$TS = SCS + \sum_{i=1}^{NCC} CCS$$

Siendo NCC el número de columnas complejas en la tabla

3.1.3. Tamaño de la columna compleja.

Las columnas complejas son aquellas que se definen sobre una clase. De esta forma, una columna de una tabla puede ser definida sobre una clase (la cual a su vez puede pertenecer a una jerarquía de clases) y también varias columnas pueden ser definidas sobre la misma clase. Con todo esto, el tamaño de una columna simple (SCS) se calcula en función del tamaño de la clase sobre la que se define el cual, a su vez, se define recursivamente como:

$$CCS = CS_p + \sum_{i=1}^{NCF} CS_i$$

Donde NCF es el número de clases padre de la clase

3.1.4. Tamaño propio de la clase.

Para obtener una expresión para el tamaño propio de una clase hemos de considerar tanto los atributos como los métodos de la clase. Debido al hecho de que los atributos y los métodos tienen diferentes tamaños, será conveniente ponderar su suma.

Así, el tamaño propio de una clase (CS_p) puede ser definido como la suma de sus atributos ponderados (ACS) más sus métodos ponderados (MCS):

$$CS_p = (ACS + MCS)$$

3.1.5. Tamaño de los atributos de una clase.

Definimos ACS considerando que los atributos tienen un tamaño igual a 1 por lo que esta métrica corresponderá exactamente al número de atributos de la clase (NOAC):

$$ACS = NOAC$$

3.1.6. Tamaño de los métodos de una clase.

$$MCS = \sum_{i=1}^{nm} V_{ij} (G)$$

Definimos MCS usando la versión de la complejidad ciclomática de McCabe dada por [5]:

Siendo nm el número de métodos por clase

3.2. Métricas de longitud

La figura 1 muestra como las tablas pueden relacionarse de dos formas distintas. Por un lado mediante generalización (simple o múltiple) y por el otro mediante integridad referencial (claves ajenas). Teniendo esto en cuenta definimos dos métricas de longitud.

La primera se basa en la métrica DIT de [2] y la definimos como la profundidad máxima en el árbol de herencia (TDIT).

La segunda también se basa en la métrica DIT pero usando el otro tipo de relación posible entre las tablas. La definimos como el máximo número de niveles de integridad referencial entre tablas en el esquema (TDRT).

Por ejemplo en la figura 1 los valores de estas métricas serían: TDIT=3 and TDRT=2.

3.3. Métricas de complejidad

Podemos definir el grado de referenciabilidad (RD) de un esquema como el número de claves ajenas (NFK) del esquema:

S. Añadir relaciones entre elementos de m_1 y de m_2 no decreta la longitud de S.

- Propiedad 5. La longitud de un sistema S formado por la unión de dos módulos disjuntos m_1 y m_2 es el máximo de las longitudes de m_1 y m_2 .

La demostración de estas propiedades para las métricas TDIT y TDRT es la siguiente:

1. La profundidad de un árbol...

- Propiedad 2. La complejidad de un sistema es nulo si el conjunto de las relaciones es el conjunto vacío.
- Propiedad 3. La complejidad de un sistema no debe depender de la convención escogida para representar las relaciones entre sus elementos.
- Propiedad 4. La complejidad de un sistema no es menor que la suma de las complejidades de cualesquiera dos de sus

IV Congreso sobre Tecnología de Objetos

$$RD = \sum_{i=1}^{NT} NFK$$

Por ejemplo en la figura 1, $RD=3$

4. Descripción formal de las métricas propuestas

En esta sección usamos los axiomas propuesto por [1] para poder así caracterizar las métricas propuestas con anterioridad.

4.1. Métrica SS

Podemos considerar que un esquema relacional está compuesto por un conjunto de columnas de las tablas del esquema. Las tablas se pueden considerar entonces como los módulos del sistema.

Una métrica de tamaño se caracteriza por las siguientes propiedades (Briand et al., 1996):

- Propiedad 1. El tamaño de un sistema es siempre mayor o igual que cero
- Propiedad 2. Si E es el conjunto vacío, el tamaño es cero
- Propiedad 3. El tamaño de un sistema es igual a la suma del tamaño de dos de sus módulos siempre y cuando cualquier elemento de S pertenezca a alguno de esos dos módulos.

Probar estas propiedades para la métrica SS es fácil:

1. Viendo las distintas expresiones que componen la métrica, se comprueba que es imposible obtener un valor negativo.
2. Si no hay definidos atributos ($E=\emptyset$), entonces $SCS=0$ y $CCS=0$ por lo que $TS=0$ y $SS=0$.
3. Aquí habrá que estudiar tres posibilidades:

Caso 1. En un módulo hay un atributo simple y en el otro también

Caso 2. En un módulo hay un atributo complejo y en el otro también (de cualquier tipo)

Caso 3. En un módulo hay un atributo simple y en el otro uno complejo (de cualquier tipo)

Es indiferente estudiar el caso de que haya un solo atributo o varios por lo que se estudia el caso más simple.

En el caso 1, tendremos que $TS = SCS$ para los dos casos y por tanto, SS será la suma de los dos TS .

En el caso 2, tendremos que $TS = CCS$ para los dos casos y por tanto, SS será la suma de los dos TS .

En el caso 3, tendremos que el primer $TS = SCS$ y el segundo $TS = CCS$ y por tanto, SS será la suma de ambos.

Por tanto podemos concluir que SS es una métrica de tamaño.

De forma similar es fácil comprobar que TS , SCS , CCS y CS_p también son métricas de tamaño.

4.2. Métricas TDIT y TDRT

En este caso, consideramos que un esquema relacional se caracteriza por el conjunto de sus elementos que corresponde al conjunto de tablas de un esquema junto con sus relaciones. Estas relaciones serán de herencia cuando estudiemos la métrica TDIT y de integridad referencial cuando sea la métrica TDRT.

Según [1] la longitud de un sistema es una función caracterizada por las siguientes propiedades:

- Propiedad 1. La longitud de un sistema es siempre un valor positivo.
- Propiedad 2. La longitud de un sistema es cero si el conjunto de elementos es el conjunto vacío
- Propiedad 3. Sean, S un sistema y m un módulo de S tal que m se representa como un componente conexo dentro del grafo que representa a S. Añadir relaciones entre elementos de m no incrementa la longitud de S.
- Propiedad 4. Sean, S un sistema y m_1 y m_2 dos módulos de S tales que están representados por componentes conexos diferentes dentro del grafo que representa a

tiene algunas deficiencias [4] y no es el único propuesto ([7], [3]); por tanto, estamos intentando verificar estas métricas utilizando otros axiomas.

Actualmente estamos llevando a cabo una validación empírica de las métricas propuestas, no sólo para probar su validez, sino para obtener algunos valores límite que puedan ser útiles para los diseñadores, principalmente a la hora de evaluar diseños alternativos. De momento se ha realizado una validación experimental sobre las métricas de integridad referencial y se están preparando, para realizar en breve, nuevos experimentos para la validación de las métricas específicas de bases de datos orientadas a objetos.

[7] Weyuker, E.J. "Evaluating software complexity measures". *IEEE Transactions on Software Engineering* Vol. 14, Nº 9, pp. 1357-1365, Sept.1988.

Bibliografía

- [1] Briand, L., Morasca, S. and Basili, V. "Property-Based Software Engineering Measurement". *IEEE Transactions on software Engineering*, Vol. 22, No. 1, Pg 68-86. 1996
- [2] Chidamber, S. and Kemerer, C. "A metrics suite for object-oriented design", *IEEE Trans. Software Eng.*, vol. 20,no. 6, pp. 476-493, June, 1994.
- [3] Fenton, N. "Software Metrics: A Rigorous Approach". London, Chapman & Hall.. (1991)
- [4] Kitchenham, B.A., Stell, J.G. "The danger of using axioms in software metrics". *IEE Proc.-Soft. Eng.*, Vol. 144, No. 5-6, Pg 279-285. (1997).
- [5] Li, W. and Henry, S. "Object-Oriented metrics that predicts maintainability". *J.Sys.Software*, 23, pp. 111-122. (1993).
- [6] Stonebraker, M. "Object-Relational Databases. The next-Wave". Morgan-Kauffman. (1996).

>From jjoo@orion.deusto.es Wed Jul 22 13:14 MET 1998
Return-Path: <jjoo@orion.deusto.es>
Comments: Authenticated sender is <jjoo@orion.deusto.es>
From: "Jornadas de Tecnología de Objetos" <jjoo@inf.deusto.es>
Organization: Universidad de Deusto
To: ccalero@inf-cr.uclm.es
Date: Wed, 22 Jul 1998 13:14:48 +0000
Subject: JJ00'98: Aceptación
Cc: mpiattin@inf-cr.uclm.es, mpolo@inf-cr.uclm.es, fruiz@inf-cr.uclm.es
X-Confirm-Reading-To: "Jornadas de Tecnología de Objetos" <jjoo@inf.deusto.es>
X-Pmrc: 1
Return-Receipt-To: "Jornadas de Tecnología de Objetos" <jjoo@inf.deusto.es>
Priority: normal
Content-Length: 210499

Buenos días,

En primer lugar, nuestras disculpas por el retraso en el envío de las respuestas de aceptación de trabajos. El artículo 'Propuesta de Definición de Métricas para Sistemas de Bases de Datos Objeto-Relacionales' ha sido aceptado.

Comentarios de los miembros del Comité de Programa:

1º Sería deseable que los autores presentaran un poco más detallado un caso práctico, si lo tienen. Una validación experimental.

Os recuerdo que debemos recibir la versión definitiva antes del 11 de Septiembre. Para facilitar el trabajo de edición, adjuntamos un fichero con la plantilla Word97 y sus estilos. En cuanto a vuestra franja de tiempo, dispondreis de un total de 20 minutos: 15 minutos de presentación y 5 dedicados a las preguntas. Cuando tengamos la versión definitiva de la agenda, os comunicaremos la fecha y hora exactas. Para cualquier duda o consulta, estaremos a vuestra disposición hasta el 24 de Julio y a partir del 31 de Agosto. Saludos.

The following section of this message contains a file attachment prepared for transmission using the Internet MIME message format. If you are using Pegasus Mail, or any another MIME-compliant system, you should be able to save it or view it from within your mailer. If you cannot, please ask your system administrator for assistance.

---- File information -----

File: jjoo98.dot
Date: 22 Jul 1998, 11:48
Size: 154112 bytes.
Type: Unknown

Attachment Converted: "c:\archivos de programa\eudora\attach\jjoo98.dot"