

# **IV JORNADAS sobre calidad del software**


**8 y 9 de julio de 1999. Madrid**

## **LIBRO DE PONENCIAS**

*Patrocinado por:*



C/ Gregorio del Amo, 6  
Teléfs: 91 349 56 00/35  
Fax: 91 349 56 74  
28040 Madrid  
Dep. Legal: M-27297-1999  
Impresión: Artes Gráficas Mañas S. L.

 Esta publicación se ha realizado en papel reciclado

## ÍNDICE

Introducción. La calidad del software. ¿Una misión imposible? Gloria Nistal. ATI Madrid	5
Las ventajas de la certificación ISO-9001 Pablo Borondo. Sadiel	11
La Gestión de la Configuración como base de la calidad Patricia de la Fuente. UNI2	19
Implementación de una Intranet para trabajo en grupo en desarrollo de software Manuel Galindo. INSS	33
Implantación de un Sistema de Aseguramiento de la Calidad: Un método práctico Idoia Alarcón. UAM	45
Evaluación y mejora de un currículum en Ingeniería del Software utilizando SPICE Javier Tuya, José García. Universidad de Oviedo	59
Utilización de la simulación de proyectos para estimar el coste de la calidad del software Mercedes Ruiz, Isabel Ramos. Universidad de Cádiz, Universidad de Sevilla	75
Métricas para la calidad de bases de datos relacionales Coral Calero, Mario Piattini, Francisco Ruiz, Macario Polo. UCLM	89
El Papel de la trazabilidad de requisitos en la calidad y la mejora de procesos de software María N. Moreno. Universidad de Salamanca	103
Iniciación de la calidad del producto final a partir de la evaluación de la calidad de la arquitectura del software Williams L. de Oliveira, Juan C. Dueñas, Juan A. de la Puente. Universidad Politécnica de Madrid.	115
Las métricas como arma competitiva en el entorno de S.I. Marta D'Amore . CENTRISA	127
SW-CMM. <i>Light assessment for managers</i> . Informe de una experiencia. Juan José López Espinosa. ERICSSON ESPAÑA S.A.	137
De un proyecto de mejora a los proyectos de software. Asier Ares. IBERMÁTICA.	149

Aplicación del modelo europeo de calidad a un centro de Producción de Sistemas de Información. Carlos Herrán. INDRA.	159
Evolución de las normas de desarrollo software en el DoD. José Manuel Pastor. ISDEFE.	171
Composición del Grupo de Calidad del Software de la Asociación de Técnicos de Informática (ATI)	189

# Métricas para la calidad de bases de datos relacionales<sup>1</sup>

Coral Calero Muñoz, Mario Piattini Velthuis, Francisco Ruiz González, Macario Polo Usaola

*Grupo ALARCOS*

*E.S.Informática. Dpto. Informática*

*Universidad de Castilla-La Mancha*

*Tfno: 926295300 Ext:3715*

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

## Resumen

Debido al auge que han cobrado las bases de datos dentro de los sistemas de información en los últimos años, resulta fundamental disponer de algún método que nos permita asegurar la calidad de las mismas. Una forma de controlar la calidad es mediante la utilización de métricas específicas. Estas métricas, no deben ser solamente sometidas a verificaciones formales sino también a validaciones empíricas que nos permitan asegurar su utilidad. En este trabajo presentamos una serie de métricas para medir la calidad de las bases de datos relacionales junto con un experimento realizado con dos de las métricas relacionadas con la integridad referencial.

## 1. Introducción

Una de las formas de controlar y conocer la calidad de una base de datos es mediante la utilización de métricas que midan su complejidad. Está reconocido que las métricas del software son un buen medio para entender, monitorizar, controlar, predecir y probar el desarrollo software y los proyectos de mantenimiento [Briand et al., 1996]. Pero las medidas no solamente se utilizan para entender, controlar y probar, sino que también pueden ser utilizadas para que los profesionales e investigadores puedan tomar las mejores decisiones [Pfleeger, 1997]. Por otra parte, en general, la medición permite ganar control, comprensión de las situaciones y posibilita la mejora [Fernández, 1999].

Los ingenieros de software han propuesto grandes cantidades de métricas para productos, procesos y recursos software [Melton, 1996; Fenton y Pfleeger, 1997]. Sin embargo, desde que McCabe propusiera su complejidad ciclométrica [McCabe, 1976] hasta ahora, la gran mayoría

---

<sup>1</sup> Este trabajo forma parte del proyecto MANTICA, parcialmente financiado por la CICYT y la Unión Europea (IFD97-0168)

de las métricas estaban enfocadas a las características del programa dejando a un lado las bases de datos.

Sin embargo, diseñar métricas para medir los datos es importante debido a la relevancia que tanto el tamaño de los datos como su naturaleza tienen en aspectos como el esfuerzo del desarrollo. Por tanto, medir datos puede ayudar a controlar y predecir aspectos del modelo de datos durante el proceso de desarrollo software [MacDonell et al., 1997].

Sneed y Foshag [1998] justificaron la escasez de métricas específicas para bases de datos basándose en el hecho de que, hasta hace poco, las bases de datos se consideraban como algo secundario sin apenas repercusión en la complejidad total del sistema. Sin embargo, las bases de datos se han constituido en una parte esencial, pasando a ser el núcleo de los sistemas de información, debido a lo cual resulta fundamental su medición.

Dentro de los factores que afectan la calidad está la mantenibilidad, en la que influyen tres factores: entendibilidad, modificabilidad y testeabilidad, los cuales a su vez están influenciados por la complejidad [Li y Cheng, 1987]. Sin embargo, una métrica general para medir la complejidad es como el “santo grial” [Fenton, 1994]. Henderson-Sellers [1996] divide la complejidad en tres: computacional, psicológica y representacional, y para la psicológica define tres componentes: complejidad del problema, factores cognitivos humanos y complejidad del producto. En esta última hemos centrado nuestro trabajo.

En este artículo se presentan diferentes métricas para medir la complejidad del esquema de bases de datos relacionales: NA, DRT, RD, NR y COS.

## **2. Métricas propuestas para bases de datos relacionales**

Desde que a finales de los sesenta el Dr. Codd propusiera su modelo relacional [Codd, 1970], se ha intensificado la investigación en el campo de las bases de datos y los productos de bases de datos relacionales han generado una importante industria.

Date [1995] define un sistema de gestión de base de datos relacional como “un sistema, en el que como mínimo:

- Los datos son vistos por el usuario como tablas (y sólo así) y
- Los operadores disponibles para el usuario generan nuevas tablas a partir de otras antiguas. Entre los operadores se encuentran, como mínimo la selección, (SELECT), la proyección (PROJECT) y la combinación (JOIN)”.

El único indicador utilizado para medir la calidad de una base de datos relacional fue la teoría de la normalización, a partir de la cual Gray et al. [1991] proponen un ratio de normalidad. Por nuestra parte, en este trabajo, proponemos cinco métricas:

- **Número de atributos (NA)**

NA es el número de atributos en todas las tablas del esquema y NA(A) es el número de atributos de la tabla A.

- **Grado de Referenciabilidad (Referential Degree RD)**

RD se define como el número de claves ajenas del esquema relacional y RD(A) es el número de claves ajenas de la tabla A.

- **Profundidad del árbol referencial (Depth Referential Tree DRT)**

DRT se define como la longitud del máximo camino referencial del esquema relacional. Los ciclos sólo se consideran una vez.

- **Ratio de normalidad (Normality ratio NR)**

NR se define como el número de tablas en tercera forma normal o superior dividido entre el número de tablas en el esquema:

$$NR = \frac{NT_{3NF}}{NTS} \quad \dots \text{donde:} \quad \begin{array}{l} NT_{3NF} \text{ es el número de tablas en 3NF (o superior)} \\ NTS \text{ es el número de tablas en el esquema} \end{array}$$

- **Cohesión del esquema (Cohesion of the schema COS)**

COS se define como la suma del cuadrado del número de tablas en cada subgrafo no relacionado dentro de los subgrafos que representan la base de datos:

$$COS = \sum_{i=1}^{[US]} NTUS_i^2 \quad \begin{array}{l} [US] \text{ es el número de subgrafos no relacionados.} \\ NTUS_i \text{ es el número de tablas en el subgrafo no relacionado "i".} \end{array}$$

Aplicamos las métricas definidas al ejemplo de la tabla 1, tomado de Elmasri y Navathe [1997]:

<pre> CREATE TABLE EMPLEADO (   NOMBREP      VARCHAR(15)  NOT NULL,   INIC        CHAR,   APELLIDO    VARCHAR(15)  NOT NULL,   NSS         CHAR(9)      NOT NULL,   FECHAEN     DATE,   DIRECCION   VARCHAR(30),   SEXO        CHAR,   SALARIO     DECIMAL(10,2),   NSSUPER     CHAR(9),   ND          INT          NOT NULL,   CONSTRAINT CLPEMP     PRIMARY KEY (NSS),   CONSTRAINT CLESUPEREMP     FOREIGN KEY (NSSUPER) REFERENCES EMPLEADO(NSS)       ON DELETE SET NULL ON UPDATE CASCADE,   CONSTRAINT CLEDEPTOEMP     FOREIGN KEY (ND) REFERENCES DEPARTAMENTO (NUMEROD)       ON DELETE SET DEFAULT ON UPDATE CASCADE);  CREATE TABLE DEPARTAMENTO (   NOMBRED     VARCHAR(15)  NOT NULL,   NUMEROD     INT          NOT NULL,   NSSGTE      CHAR(9)      NOT NULL,   FECHAINICGTE DATE,   CONSTRAINT CLPDEPTO     PRIMARY KEY (NUMEROD),   CONSTRAINT CLSDEPTO     UNIQUE(NOMBRED),   CONSTRAINT CLEGTDEPTO     FOREIGN KEY (NSSGTE) REFERENCES EMPLEADO(NSS)       ON DELETE SET DEFAULT ON UPDATE CASCADE); </pre>	<pre> CREATE TABLE LUGAR_DEPTS (   NUMEROD     INT          NOT NULL,   LUGARD      VARCHAR(15)  NOT NULL,   PRIMARY KEY (NUMEROD, LUGARD),   FOREIGN KEY (NUMEROD) REFERENCES DEPARTAMENTO (NUMEROD)       ON DELETE CASCADE ON UPDATE CASCADE);  CREATE TABLE PROYECTO (   NOMBREPR    VARCHAR(15)  NOT NULL,   NUMEROPR    INT          NOT NULL,   LUGARPR     VARCHAR(15),   NUMD        INT          NOT NULL,   PRIMARY KEY (NUMEROP),   UNIQUE (NOMBREP),   FOREIGN KEY (NUMD) REFERENCES DEPARTAMENTO (NUMEROD));  CREATE TABLE TRABAJA_EN (   NSSE        CHAR(9)      NOT NULL,   NUMP        INT          NOT NULL,   HORAS       DECIMAL(3,1) NOT NULL,   PRIMARY KEY (NSSE, NUMP),   FOREIGN KEY (NSSE) REFERENCES EMPLEADO (NSS),   FOREIGN KEY (NUMP) REFERENCES PROYECTO (NUMEROP));  CREATE TABLE DEPENDIENTE (   NSSE        CHAR(9)      NOT NULL   NOMBRE_DEPEND VARCHAR(15) NOT NULL,   SEXO        CHAR,   FECHAAN     DATE,   RELACION    VARCHAR(8),   PRIMARY KEY (NSSE, NOMBRE_DEPEND),   FOREIGN KEY (NSSE) REFERENCES EMPLEADO (NSS)); </pre>
--	--

Tabla 1. Ejemplo



En la figura 1 se presenta el grafo relacional para el ejemplo anterior, donde las flechas indican las relaciones de integridad referencial:

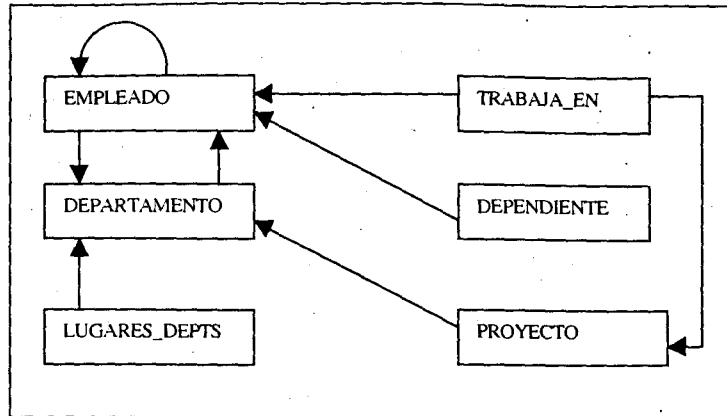


Figura 1. Esquema del ejemplo

Los valores de las métricas correspondientes al ejemplo se muestran en la tabla 2.

	NA	RD	DRT	NR	COS
EMPLEADO	10	2			
DEPARTAMENTO	4	1			
LUGARES_DEPTS	2	1			
PROYECTO	4	1			
TRABAJA_EN	3	2			
DEPENDIENTE	5	1			
ESQUEMA	28	8	5	1	36

Tabla 2. Valores de las métricas para el ejemplo

En la tabla, podemos observar todos los valores obtenidos para las métricas a partir de la información que nos da tanto el ejemplo como su esquema. Por ejemplo, el valor para la métrica DRT es cinco, que se obtiene del camino referencial siguiente:

TRABAJA\_EN → PROYECTO → DEPARTAMENTO → EMPLEADO → EMPLEADO → DEPARTAMENTO.

En este punto conviene recordar que, por definición, nuestra métrica DRT sólo considera los ciclos una vez.

### 3. Validación empírica de las métricas

Puede que una métrica sea correcta desde un punto de vista formal, pero no tener relevancia práctica para un problema determinado. Es por esto que resulta necesaria la

validación empírica. Según Dolado y Fernández [1998], los experimentos sirven para contrastar predicciones teóricas con la realidad y nos permiten llegar dónde el análisis deductivo no llega. Además, los autores defienden que la experimentación permite eliminar explicaciones alternativas y generar nuevas teorías.

Así pues, el estudio empírico resulta necesario para comprobar y entender las implicaciones de las medidas de nuestros productos y procesos. Esto se consigue a través de hipótesis en el mundo real, mas allá de la pura teoría, que habrá que comprobar con datos empíricos.

Es importante, cuando se va a realizar un estudio empírico, tener especial cuidado con los datos que se eligen ya que aunque un experimento pueda estar correctamente diseñado, los datos pueden resultar incorrectos o insuficientes por lo que el resultado del experimento no será indicativo de la realidad. Esto se evita utilizando las técnicas de muestreo adecuadas a la hora de recoger los datos.

Una vez que los datos están disponibles, hay que aplicar uno de entre todos los posibles métodos estadísticos que existen para interpretar de forma correcta los datos. Las distintas alternativas fueron recogidas por Pfleeger [1995] y se muestran en la tabla 3:

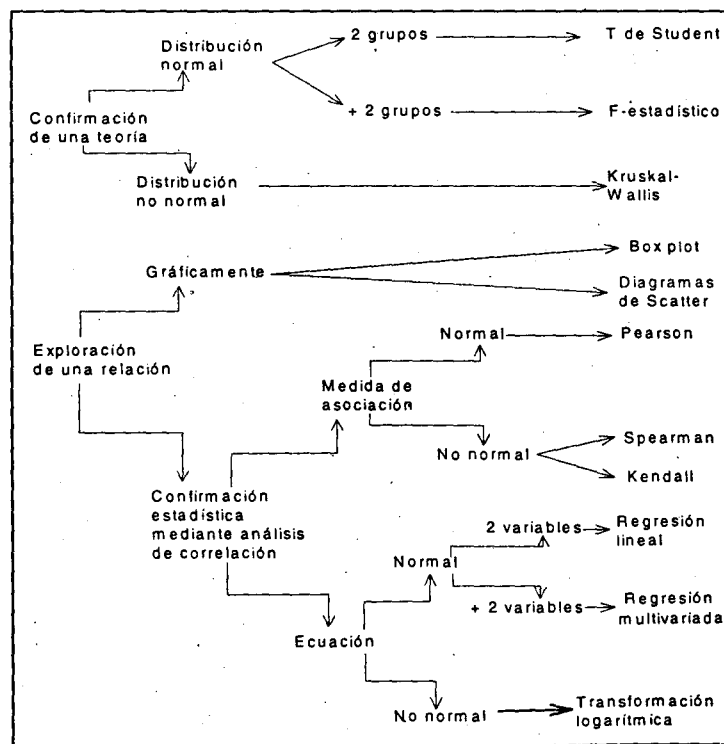


Tabla 3. Relación de métodos para validación según Pfleeger

## 4. Validación empírica de las métricas relacionadas con la integridad referencial

Una de las características fundamentales del modelo relacional son las claves ajenas. Relacionada con las claves ajenas, el modelo presenta la integridad referencial. La validación empírica que presentamos, se utiliza para demostrar la influencia de las dos métricas propuestas relacionadas con la integridad referencial (TDRT y RD) con la entendibilidad del esquema de una base de datos relacional.

El objetivo de esta validación es demostrar que las métricas pueden ser utilizadas para medir la complejidad del esquema de bases de datos relacional, la cual influye en su entendibilidad.

### 4.1. Hipótesis

Las hipótesis formales son:

- Hipótesis nula: La comprensión del esquema de la base de datos no está influida por los diferentes valores de las métricas.
- Hipótesis alternativa 1: La comprensión del esquema de la base de datos depende del valor de la métrica TDRT
- Hipótesis alternativa 2: La comprensión del esquema de la base de datos depende del valor de la métrica RD
- Hipótesis alternativa 3: La comprensión del esquema de la base de datos depende de la combinación de los valores de las métricas TDRT y RD.

### 4.2. Sujetos

Los participantes del experimento fueron alumnos del tercer curso de la Escuela de Informática de Ciudad Real, los cuales cursan una asignatura de bases de datos durante todo el curso escolar.

Hasta el día en el que se realizó el experimento, los alumnos no tenían conocimiento del mismo. Este fue realizado por un total de 60 alumnos aunque sólo 59 fueron finalmente aceptados para obtener los resultados.

Se intentó reducir al mínimo la variabilidad entre los participantes haciendo que todos pertenecieran al mismo curso y que además fuera el último de la carrera. Además, todos los participantes realizaron las mismas pruebas y tuvieron todos el mismo tiempo para llevarlas a cabo (diez minutos por test).

#### *4.3. Materiales del experimento*

Se diseñaron cuatro tests diferentes en cada uno de los cuales el valor de las dos métricas era diferente. Así, la métrica RD podía tomar dos valores (8 ó 5) y TDRT otros dos (2 ó 5).

La documentación presentada para realizar el experimento fue de aproximadamente siete páginas incluyendo los esquemas de las bases de datos, las tablas con sus columnas y la hoja de preguntas y respuestas. Para cada diseño, se mantuvo fijo en seis el número de tablas para no afectar la complejidad de los diseños, siendo los distintos valores de las métricas los que diferenciaban los esquemas ya que entre las seis tablas había que conseguir tener más o menos claves ajenas con mayor o menor longitud en los caminos referenciales. Por ejemplo en el caso de  $RD=8$  y  $TDRT=2$ , entre las seis tablas debía de haber ocho claves ajenas que no formaran caminos de longitud mayor que dos.

Sobre cada uno de estos cuatro diseños los sujetos del experimento debieron ver a cuantas columnas y de que tablas se afectaba al realizar una inserción, un borrado y una modificación en alguna de las tablas del esquema. La figura 2 muestra el aspecto de la hoja de preguntas y respuestas:

1. ¿Qué tablas y cuantas columnas por tabla se verán afectadas si borramos de la tabla 5 la columna con cod1=210?

Tabla 1	Tabla 2	Tabla 3	Tabla 4	Tabla 5	Tabla 6

2. ¿Qué tablas y cuantas columnas por tabla se verán afectadas si modificásemos la columna X de la fila con X=11 en la tabla 3?

Tabla 1	Tabla 2	Tabla 3	Tabla 4	Tabla 5	Tabla 6

3. ¿Cuántas filas, de que tablas y cuantas columnas por tabla habría que modificar si quisiéramos añadir una nueva fila (con todos los datos nuevos en la base de datos) a la tabla 6?

Tabla 1	Tabla 2	Tabla 3	Tabla 4	Tabla 5	Tabla 6

Figura 2. Ejemplo de hoja de preguntas y respuestas para el experimento

Previamente a la realización del experimento, las pruebas fueron realizadas por un pequeño grupo de personas para poder asegurar su bondad.

#### 4.4. Diseño experimental

Debido a que cada nivel de un factor aparece con cada nivel del otro factor se eligió un diseño de tipo cruzado. Esta relación cruzada se denota como  $A \times B$  donde A es la métrica RD y B es la TDRT. La Tabla 3 representa de forma gráfica esta relación.

		Factor B (RD)	
		BAJO	ALTO
Factor (TDRT) A	BAJO	2,5	2,8
	ALTO	5,5	5,8

Tabla 3. Diseño cruzado del experimento

Para incrementar el poder del test se eligió el valor de  $\alpha$  como 0,1 en vez de 0.05, valor más comúnmente elegido.

#### 4.5. Resultados del experimento

Cuando hay que elegir la técnica de análisis de los datos hay que tener en cuenta tres factores fundamentales: la naturaleza de los datos, con que fin se ha desarrollado el experimento y el tipo de diseño experimental utilizado. Teniendo en cuenta las características del experimento realizado, la técnica más conveniente para obtener los resultados es el F-estadístico. La tabla 4 muestra los resultados obtenidos a partir de los datos recogidos en el experimento.

Factor de variación	$Q_i$	Grados de libertad	$S_i^2$	F-Ratio
TDRT	18.457	1	18.5	1.67
RD	531.000	1	531	48,1
Interaction	31.339	1	31.3	2.84
Error	2560.304	1	11.0	
Total	3141.102	232		

Tabla 4. Resultados del experimento

Comparando estos valores con el de  $F_{1,232}=2.73$ , extraído de la tabla de valores, podemos afirmar que:

- ⇒ Hipótesis 1. Como  $1.67 < 2.73$ , el valor de TDRT no afecta a los resultados del experimento. Por tanto esta hipótesis es NO VÁLIDA.
- ⇒ Hipótesis 2. Como  $48.1 > 2.73$ , el valor de RD afecta a los resultados del experimento. Por tanto esta hipótesis es VÁLIDA.
- ⇒ Hipótesis 3. Como  $2.84 > 2.73$ , la interacción de ambas métricas afecta a los resultados del experimento. Por tanto esta hipótesis es VÁLIDA.

Finalmente, en vista de estos resultados, concluimos que el número de claves ajenas en el esquema de base de datos relacional es un indicador sólido de su entendibilidad y que la longitud del camino referencial por si mismo no resulta indicativo, aunque modula los efectos en combinación con RD, es decir a igualdad en el valor de RD, TDRT nos podrá indicar cual de los dos esquemas será más entendible.

## 5. Conclusiones y trabajo futuro

Es necesaria una mayor investigación en los aspectos de la medida del software, tanto desde el punto de vista teórico como práctico [Neil, 1994; Glass, 1996; Basili, 1999; Fernández, 1999].

Pensamos que es muy interesante disponer de métricas para bases de datos relacionales. Estas métricas pueden utilizarse para llamar la atención de los jefes de proyecto sobre determinados esquemas relacionales que tengan unos requisitos muy estrictos, de mantenibilidad, de forma parecida a la propuesta por Chidamber et al. [1998] para modelos orientados a objetos.

En este trabajo hemos presentado una serie de propuestas para medir la complejidad de bases de datos relacionales, que vaya más allá de la sola teoría de la normalización. Creemos que es también fundamental llevar a cabo una validación empírica para demostrar la utilidad de la medida en la práctica. Una medida puede ser correcta desde el punto de vista de la teoría de la medida (consistente con el sistema relacional empírico) pero irrelevante de cara al problema a resolver. Con este fin se ha llevado a cabo el experimento que demuestra la influencia del valor de la métrica RD en la comprensibilidad de esquemas relacionales. De todas maneras, somos conscientes de las limitaciones de los experimentos controlados [Basili, 1999] por lo que se han iniciado diversos análisis de bases de datos "reales" en colaboración con organismos y empresas. También hay que tener en cuenta que la utilidad de las métricas mejoraría mucho si se diera una guía clara de aplicación a lenguajes específicos [Churcher y Shepperd, 1995]. Por ello, estamos construyendo herramientas para obtener de forma automática estas medidas en entornos que soporten el lenguaje SQL.

## Bibliografía

[Basili, 1999] Basili, V. R., "Using experiments to build a body of knowledge". 20/1/99. Madrid. España. 1999

[Briand et al., 1996] Briand, L.C., Differding, C.M. y Rombach, D., "Practical Guidelines for Measurement-Based Process Improvement", *Software Process-Improvement and Practice*, Vol. 2, 253-280, 1996.

[Chidamber et al., 1998] Chidamber, S.R., Darcy, D.P. y Kemerer, C., "Managerial Use of Metrics for Object-Oriented Software: An Exploratory Analysis", *IEEE Trans. On Software Engineering*, Vol. 24, Nº 8, 629-639, 1998.

[Churcher y Shepperd, 1995] Churcher, N.J. y Shepperd, M.J., Comments en "A Metrics Suite for Object-Oriented Design", *IEEE Trans. on Software Engineering*, Vol. 21(3), 263-265, 1995.

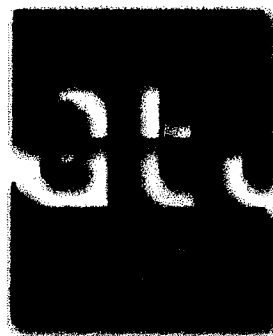
[Codd, 1970] Codd, E.F., "A Relational Model of Data for Larged Shared Data Banks", *Communications of the ACM*, Vol. 13 (6), 377-387, 1970.

[Date, 1995] Date, C.J., *An Introduction to Database Systems*. 6<sup>th</sup>. Addison-Wesley, Reading, Massachusetts, 1995.

- [Dolado y Fernández, 1998] Dolado, J. y Fernández, L., "¿Hay que experimentar en Ingeniería del Software?", *III Jornadas de Calidad del Software, SIMO-TCI'98*, Madrid, 6 de Noviembre 1998.
- [Elmasri y Navathe, 1997] Elmasri, R. y Navathe, S., *Database Systems*. Second edition. Addison-Wesley. Massachussets, 1997.
- [Fenton, 1994] Fenton, N., "Software Measurement: A Necessary Scientific Basis", *IEEE Transactions on Software Engineering*, Vol. 20(3), 199-206, 1994.
- [Fenton y Pfleeger, 1997] Fenton, N. y Pfleeger, S. L., *Software Metrics: A Rigorous Approach* 2<sup>nd</sup>. edition. London, Chapman & Hall, 1997.
- [Fernández, 1999] Fernández, L., "Una revisión breve de la medición de software", *Novática*, Enero-Febrero 1999, n°137, 20-24, 1999.
- [Glass, 1996] Glass, R., "The Relationship Between Theory and Practice in Software Engineering", *IEEE Software*, Vol. 39 (11), 11-13, 1996.
- [Gray et al., 1991] Gray R.H.M., Carey B.N., McGlynn N.A. y Pengelly A.D., "Design metrics for database systems", *BT Technology Journal*, Vol 9, 4 Oct, 69-79, 1991.
- [Henderson-Sellers, 1996] Henderson-Sellers, B., *Object-oriented Metrics - Measures of complexity*. Prentice-Hall, 1996.
- [Li y Cheng, 1987] Li, H.F. y Cheng, W.K., "An empirical study of software metrics", *IEEE Trans. on Software Engineering*, Vol. 13 (6): 679-708, 1987.
- [MacDonnell et al., 1997] MacDonell, S.G., Shepperd, M.J. y Sallis, P.J., "Metrics for Database Systems: An Empirical Study", *Proc. Fourth International Software Metrics Symposium - Metrics'97*, Albuquerque. IEEE Computer Society, pp. 99-107, 1997.
- [McCabe, 1976] McCabe, T.J., "A complexity measure", *IEEE Trans. Software Engineering*, Vol. 2(5), 308-320, 1976.
- [Melton, 1996] Melton, A. (ed.), *Software Measurement*. London, International Thomson Computer Press, 1996.
- [Neil, 1994] Neil, M., "Measurement as an Alternative to Bureaucracy for the Achievement of Software Quality", *Software Quality Journal*, Vol. 3 (2), 65-78, 1994.
- [Pfleeger, 1995] Pfleeger, S.L., "Experimental Design and Analysis in Software Engineering", en *Annals of Software Engineering*, pg. 219-253, Science Publishers, 1995.
- [Pfleeger, 1997] Pfleeger, S. L., "Assessing Software Measurement", *IEEE Software*, marzo/abril, pp. 25-26, 1997.



[Sneed y Foshag, 1998] Sneed, H.M. y Foshag, O., "Measuring Legacy Database Structures", *Proc of The European Software Measurement Conference FESMA 98, Antwerp, May 6-8*, Coombes, Van Huysduynen and Peeters (eds.), 199-211, 1998.



***LA ASOCIACIÓN DE TÉCNICOS DE INFORMÁTICA***

**Capítulo Territorial de Madrid**

**CERTIFICA QUE**

**D. Mario Piattini Velthuis**

**Ha participado como ponente en las IV Jornadas de Calidad del Software celebradas los días 8 y 9 de julio de 1999, en la Escuela de Organización Industrial de Madrid.**

**Madrid, 9 de julio de 1999**

A handwritten signature in black ink, appearing to read 'Gloria Nistal Rosique', written over a horizontal line.

**Gloria Nistal Rosique**  
Coordinadora del Grupo de Calidad  
del Software de ATI-Madrid

