



Computación y Sistemas

ISSN: 1405-5546

computacion-y-sistemas@cic.ipn.mx

Instituto Politécnico Nacional

México

Silva, Josep; Carsí, José A.; Ramos, Isidro
Análisis teórico-experimental de criterios de comparación de esquemas conceptuales orientados a
objeto
Computación y Sistemas, vol. 9, núm. 1, julio-septiembre, 2005, pp. 1-16
Instituto Politécnico Nacional
Distrito Federal, México

Disponible en: <http://www.redalyc.org/articulo.oa?id=61590101>

- Cómo citar el artículo
- Número completo
- Más información del artículo
- Página de la revista en redalyc.org

redalyc.org

Sistema de Información Científica
Red de Revistas Científicas de América Latina, el Caribe, España y Portugal
Proyecto académico sin fines de lucro, desarrollado bajo la iniciativa de acceso abierto

Análisis Teórico-Experimental de Criterios de Comparación de Esquemas Conceptuales Orientados a Objeto *Theoretic-Experimental Analysis of Comparison Criteria for Object-Oriented Conceptual Schemas*

Josep Silva, José A. Carsí e Isidro Ramos

Universidad Politécnica de Valencia
Departamento de Sistemas Informáticos y Computación
Camino de Vera s/n, E-46071 Valencia, España
{jsilva, pcarsi, iramos}@dsic.upv.es

Artículo recibido en enero 10, 2002; aceptado en abril 05, 2005

Resumen

En algunos sistemas se produce la necesidad de comparar dos esquemas conceptuales orientados a objeto; el criterio de comparación utilizado, determinará en gran medida el resultado de la comparación. En este trabajo se realiza una clasificación de criterios de comparación de esquemas conceptuales orientados a objeto, a partir de su división en grupos dependientes de la información utilizada para la comparación (metainformación, información sobre su población, información exacta o estadística, etc.). Se enumeran las características de cada criterio, y se hace hincapié en sus puntos fuertes y débiles citando ejemplos. Finalmente, se presentarán estadísticas de comparación de esquemas conceptuales industriales. Las conclusiones finales, contrastarán dichas estadísticas con el análisis teórico previo. El presente trabajo forma parte del desarrollo de una herramienta para la migración automática de bases de datos, que está siendo desarrollada en el Departamento de Sistemas Informáticos y Computación de la Universidad Politécnica de Valencia.

Palabras Clave: Comparación de esquemas, Orientación a Objetos, Criterios de comparación, Definición de métricas, Evolución, Migración de datos, Estadísticas de comparación.

Abstract

Some systems have the need to compare object-oriented conceptual schemas; the comparison criteria used have a big influence on the result produced. In this work, a classification of object-oriented conceptual schemas comparison criteria is presented, based on the kind of the information used to compare (meta-information, population information, exact or statistical information, etc.). For each criterion, its main characteristics and strong and weak points are detailed and validated with some results from industrial conceptual schema comparisons. Final conclusions contrast the statistical results with the theoretical analysis previously done. The present work is part of a data migration tool under development by the Department of Computer Science and Information Systems of the Technical University of Valencia – Spain.

Keywords: Schema Comparison, Object-Orientation, Comparison Criteria, Metrics, Evolution, Data Migration.

1 Introducción

La comparación de esquemas conceptuales tiene un especial interés en la migración automática de bases de datos. Idealmente, la evolución del software se debe realizar a nivel de esquema conceptual, y a partir del mismo evolucionan automáticamente las aplicaciones y las bases de datos que forman el sistema de información. En ocasiones ocurre que las bases de datos del sistema original contienen información, y dicha información debe ser trasladada al nuevo sistema, este hecho es conocido como migración de datos. El principal inconveniente en la migración de datos es que las bases de datos entre las que se quiere realizar la migración suelen ser muy distintas, y en ocasiones incompatibles.

En una migración de datos, es fundamental conocer exactamente el grado de evolución sufrido por el esquema y la base de datos asociada que va a ser migrada. Se hace indispensable conocer qué modificaciones han sufrido los elementos del esquema origen, así como qué inserciones y borrados se han producido durante el proceso de evolución. También es una tarea importante determinar qué información se debe usar para identificar el origen y el destino de la migración. Por ejemplo, si en la BD del sistema origen se dispone de la tabla α , y en el destino también se dispone de la tabla α , ¿qué nos asegura que la información de α en el origen debe ser trasladada a la tabla α del destino? Puede haber ocurrido que la tabla α del origen se llame β en el destino, y esa información es determinante para la migración. En este contexto, adquiere especial relevancia un algoritmo capaz de automatizar el proceso de comparación de esquemas conceptuales (Silva et al., 2001).

El objetivo principal de los algoritmos de comparación de esquemas conceptuales es establecer una correspondencia¹ para cada elemento de los esquemas. Intuitivamente una correspondencia es una relación entre dos elementos, uno perteneciente a cada esquema, que indica que un elemento es una evolución del otro y las instancias del primero deben ser migradas al segundo. Todos estos algoritmos tienen en común que gran parte de su complejidad y de su tasa de acierto depende del criterio de comparación empleado. Se llama criterio de comparación a la información que se utiliza para comparar un conjunto de elementos dados. En un esquema conceptual la información disponible para comparar es muy rica (identificadores, nombres, relaciones con otros elementos, número de atributos en el caso de las clases, etc.) y dicha información puede determinar en gran medida el éxito de la comparación.

En este trabajo se presenta una relación de criterios de comparación de esquemas conceptuales, junto a su clasificación. Además, se presentan los resultados de un experimento de comparación de esquemas conceptuales industriales, que tiene por objetivo evaluar la calidad de los criterios de comparación presentados.

En el apartado 2, se presenta la estrecha relación existente entre la definición de criterios de comparación, y la definición de métricas. Posteriormente, en el apartado 3, se presentarán tres clasificaciones no disjuntas para criterios de comparación según distintas propiedades de los mismos. En el apartado 4, veremos una caracterización completa de criterios de comparación en función de la información utilizada para realizar la comparación. El apartado 5 presenta los resultados obtenidos en un experimento que consistía en realizar diversas comparaciones reales con algunos de los tipos de criterios presentados en los anteriores apartados. Finalmente, en el apartado 6 se extraerán las conclusiones fruto del estudio realizado.

2 Estado del Arte

Comparar siempre implica medir. Existen trabajos relativos al diseño de métricas para esquemas conceptuales. En (Dolado, 2000) puede encontrarse una recopilación de métricas destinadas a medir cada uno de los componentes de un esquema conceptual orientado a objetos. La aplicación de una métrica tiene como principal objetivo extraer una medida de alguna propiedad; pero si al aplicar una misma métrica a dos elementos distintos obtenemos medidas diferentes, nos daremos cuenta de que esos elementos son distintos. Realmente, cuando se comparan dos elementos, lo que se está haciendo es comparar dos medidas realizadas sobre alguna o algunas propiedades de esos elementos. De todo lo anterior podemos concluir que el diseño de métricas está íntimamente relacionado con el diseño de criterios de comparación.

Podemos aprovechar los estudios realizados sobre métricas para extraer, a partir de las mismas, criterios de comparación. Por ejemplo, en (Calero, 2001) se propone medir el tamaño de los servicios de una clase a partir de la complejidad ciclomática de McCabe (1976). Dicha métrica puede extrapolarse y ser redefinida como criterio de comparación de clases. A pesar de que en la mayoría de los casos se podrá obtener un criterio de comparación a partir de una métrica, no debemos olvidar que el propósito de las métricas y los criterios es esencialmente distinto; y no siempre existe un criterio adecuado que pueda derivarse a partir de una métrica. Si bien es cierto que puede ocurrir que alguna métrica no sea adecuada para derivar un criterio a partir de la misma; también lo es que todos los criterios tienen una métrica asociada; es decir, no todas las métricas pueden utilizarse para derivar un criterio de comparación de esquemas conceptuales, pero todos los criterios de comparación utilizan alguna métrica para comparar.

Se ha considerado conveniente no realizar en este punto una descripción de las métricas propuestas en la literatura; y realizarlo en un punto posterior (apartado 4) para, de este modo, poder contrastar cada criterio de comparación con su métrica asociada.

En la actualidad, el problema de la obtención de diferencias entre esquemas, es tratado por diferentes herramientas comerciales utilizando la traza de las operaciones de evolución. Este es el caso de VDIFF de Rational Rose. El principal problema que plantean estas herramientas es que los EC deben ser dependientes. Además, el hecho de que deba existir una traza de evolución entre los esquemas obliga a que los esquemas sean del mismo formato o pertenezcan a la misma aplicación. Estos problemas desaparecen con un proceso de comparación.

En (Silva et al., 2001) se presenta un algoritmo específico de comparación de esquemas conceptuales; en ese artículo se pone de manifiesto el hecho de que los algoritmos de comparación no pueden garantizar una tasa de aciertos del 100 %. Además, dicha tasa de aciertos es función directa del criterio de comparación utilizado. En el artículo se apela a la necesidad de un estudio que clasifique criterios de comparación con el fin de conocer sus propiedades y su adecuación a los distintos contextos que puede presentar una migración de datos. Si bien es cierto que existe una gran cantidad de artículos que dedican gran parte de sus esfuerzos a la definición de métricas para esquemas conceptuales; no se han encontrado en la bibliografía trabajos relativos a la clasificación de criterios de comparación. Los trabajos actuales más cercanos a la defini-

¹ El término correspondencia es la traducción al castellano de 'Mapping', su definición formal puede encontrarse en (Silva et al., 2001).

ción de criterios para comparar esquemas conceptuales, tienen por objetivo comparar los esquemas con el fin de estimar la calidad de la especificación.

En los siguientes apartados se aborda el problema de la comparación desde una perspectiva de evolución, tratando de definir criterios que identifiquen la evolución de los elementos de esquemas conceptuales orientados a objeto.

La comparación de esquemas conceptuales es una necesidad independiente del lenguaje de modelado subyacente que se utiliza. Por ello, la caracterización de criterios que se va a presentar es genérica y, por lo tanto, válida para cualquier modelo orientado a objetos que se utilice. No obstante, la mayoría de los lenguajes de modelado presentan características propias que permiten enriquecer con más información algunos criterios de comparación. A continuación, se va a presentar el lenguaje de modelado que se ha utilizado para elaborar el experimento, y al cual están referidos la mayoría de los ejemplos.

‘Open and Active Specification for Information Systems’, OASIS (Letelier et al., 1998), es un lenguaje de especificación formal para el modelado conceptual de sistemas de información usando el paradigma orientado a objetos. En OASIS todos los objetos son especificados a partir de su plantilla de objeto. Una plantilla de objeto está totalmente definida mediante la tripleta (I, τ, ρ) , donde I es una función de identificación utilizada para identificar unívocamente al objeto, τ indica el tipo del objeto, y ρ especifica la parte pública de τ . El tipo de un objeto se define a partir de la tupla (A, X, ϕ, π) donde A es el conjunto de atributos, X el conjunto de eventos, ϕ es el conjunto de formulas de la lógica dinámica que especifican el comportamiento de los objetos, y π es un conjunto de términos del álgebra de procesos. ρ es el par $(A_p, X_p) : A_p \subseteq A \wedge X_p \subseteq X$. Toda la información de un objeto OASIS susceptible de ser utilizada para comparar está incluida en los elementos de la plantilla de objeto.

En OASIS está contemplada la herencia simple y la múltiple. El hecho que dos objetos están relacionados se modela mediante la agregación que pueden ser inclusiva o relacional, disjunta o no-disjunta, flexible o estricta, estática o dinámica, univaluada o multivaluada. Desde el punto de vista de OASIS, las asociaciones son un caso especial de agregación dinámica y multivaluada.

Desde la versión 3.0, OASIS contempla en su especificación el metanivel de un esquema conceptual formado por metaclases. Una metaclase es una clase cuyas instancias, también llamadas metaobjetos, son objetos especiales que a su vez son clases de objetos. Una petición de servicio en un metaobjeto que modifique su estado, está modificando la plantilla de la clase que define, modificando, de esta manera el esquema conceptual. El hecho de considerar las clases del sistema como objetos, permite aprovechar información adicional de las mismas durante un proceso de comparación.

3 Clasificación de Criterios de Comparación

La clasificación de criterios de comparación puede enfocarse desde distintos puntos de vista ortogonales entre sí: dependencia o independencia de los esquemas; exactos o estadísticos; basados en información del esquema o en su población.

3.1 Criterios Basados en la Dependencia o Independencia de los Esquemas

Esta clasificación divide los distintos criterios en dos grupos atendiendo a si tienen en cuenta o no la dependencia o independencia de ambos esquemas. Dos esquemas son independientes si ninguno de los dos es una evolución directa del otro, o si lo es, no conserva sus propiedades de edición (como fecha de creación, identificadores, etc.). Tal es el caso de la migración de datos en los ‘Legacy Systems’. En contraposición a los esquemas independientes están los esquemas dependientes o evolutivos, donde uno de los dos es una evolución directa del anterior fruto de una secuencia de modificaciones; en algunos sistemas dicha secuencia de modificaciones se realiza a través de llamadas a servicios de evolución. En (Carsí, 1999) se propone un marco en el que hacer evolucionar el software basado en la introducción de metaclases en el modelo de objetos OASIS (Letelier et al., 1998). Dichas metaclases definen como servicios propios más de 50 servicios que permiten hacer evolucionar los esquemas conceptuales de sistemas de información.

En adelante, indicaremos que un criterio está basado en la independencia de los esquemas con una ‘I’, y en la dependencia con una ‘D’.

3.2 Criterios Exactos y Estadísticos

Algunos criterios pueden utilizar más de una propiedad para comparar; en este tipo de criterios se puede obtener un rango de valores como resultado de aplicar la función criterio. Llamaremos criterios exactos a aquellos que aceptan una pareja de elementos como iguales sí y solo sí, el valor es igual a uno dado; llamaremos criterios estadísticos a aquellos que aceptan un rango de valores como buenos. Veámoslo con un ejemplo:

Supongamos que tenemos que comparar dos clases usando un conjunto de propiedades $\delta_1, \delta_2, \dots, \delta_n$. Y que esas propiedades están ponderadas mediante $\rho_1, \rho_2, \dots, \rho_n$, respectivamente (ρ_i indica el peso asignado a cada propiedad). El valor de la función de comparación será:

$$V = (\mu_1 * \rho_1) + (\mu_2 * \rho_2) + \dots + (\mu_n * \rho_n)$$

donde μ_i es un valor lógico (0 ó 1) que indica si la comparación se cumple (1), o no (0); y ρ_i puede adoptar cualquier valor.

Un criterio exacto sólo daría por exitosa la comparación, si la función tiene exactamente un cierto valor (dicho valor suele ser el máximo posible), mientras que un criterio estadístico aceptaría un margen de error (una aproximación). En resumen:

Exactos: comparan por la igualdad de una o más propiedades.

Estadísticos: comprueban si estadísticamente puede tratarse del mismo elemento por su similitud. En adelante, haremos referencia a los criterios exactos con una 'X' y a los estadísticos con una 'E'.

3.3 Criterios Poblacionales o No Poblacionales

Cuando se comparan esquemas conceptuales en un contexto de migración de datos, los esquemas suelen estar vinculados a una población de instancias que van a ser migradas. Dicha población contiene una valiosa información que puede ser explotada durante la comparación. En adelante, haremos referencia con una 'P' a los criterios que utilizan información relativa a la población de los esquemas, y con una 'N' a los que no la utilizan.

4 Caracterización de Criterios Comparativos

En este apartado se va a enumerar una serie de criterios de comparación junto a su descripción y su clasificación. Se ha tratado de que el estudio sea bastante exhaustivo en cuanto a lo que a migración de datos se refiere. La muestra de criterios presentada contiene una representación de todos los tipos de clasificaciones.

4.1 Criterios

Inicialmente, se van a presentar los diferentes criterios que se pueden aplicar de manera individual a la hora de comparar dos esquemas conceptuales.

4.1.1 Comparación por Nombre (I, X, N)

La comparación de elementos por el nombre es la forma más intuitiva y fácil de comparar: Dos elementos (por ejemplo clases) son el mismo elemento si tienen el mismo nombre. Este es el criterio que utilizan los algoritmos de comparación de árboles de Selkow (Selkow, 1977) y Kuo-Chung Tai (Kuo-Chung, 1979) porque es idóneo para comparar árboles etiquetados. En un árbol etiquetado, la etiqueta es la única información que tienen asociada los nodos, y por lo tanto si cambia la etiqueta cambia todo el nodo; sin embargo, en un esquema conceptual el nombre no es más que un subconjunto de información, y puede ocurrir que dicho nombre cambie sin que eso signifique el cambio total del elemento. Este criterio fallará siempre que haya cambios de nombre en los elementos de los esquemas.

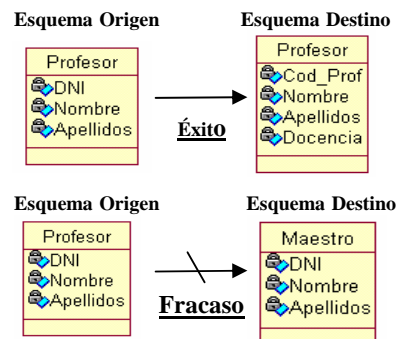


Fig. 1. Éxito y fracaso en la comparación de clases por nombre

4.1.2 Comparación por Identificador (D, X, N)

Cuando hablamos de identificador no se trata del identificador de las instancias de los elementos de los esquemas (Oid), sino que se hace referencia al identificador interno que tiene asociado cada elemento de los esquemas conceptuales. No obstante, en este artículo se hará referencia a este tipo de criterios como comparación por Oid, debido a que dicho identificador sí se corresponde con el Oid de los metaobjetos en el esquema conceptual. En una herramienta CASE, generalmente, este identificador será un valor interno de la herramienta, pero en un contexto no soportado por tecnología software puede ocurrir que este valor no exista.

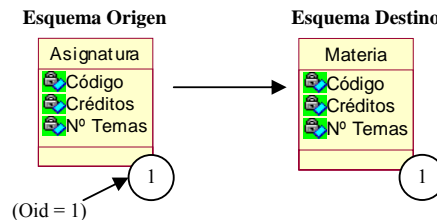


Fig. 2. Los identificadores se mantienen invariantes a la evolución

Comparar utilizando identificadores puede ser peligroso porque está (o puede estar) vinculado al editor de los esquemas. Por ejemplo, si la clase 'Asignatura' evoluciona a 'Materia' mediante un cambio de nombre, este criterio funciona perfectamente, puesto que el identificador es invariante a la evolución (ver figura 2). Sin embargo, debido al vínculo existente con el editor, cuando una misma clase que evoluciona sin ningún cambio aparente, es borrada y vuelta a crear, el editor le asignará un nuevo identificador, y hará que el criterio de comparación por identificador falle (ver figura 3).

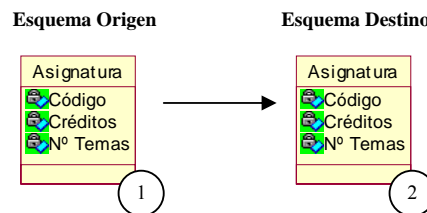


Fig. 3. Fracaso en la comparación por Identificador

4.1.3 Comparación por Metainformación (D, X, N)

Cuando los esquemas conceptuales han sido desarrollados por una herramienta CASE, a menudo ocurre que el editor almacena metainformación sobre los esquemas. Dicha metainformación puede variar en riqueza dependiendo de la herramienta, incluso puede no existir, con lo cual, no se podrían aplicar este tipo de criterios. Como ejemplos de metainformación, el editor puede almacenar la fecha de creación de cada elemento, comentarios asociados a los elementos, o incluso información dejada por el analista con el fin de guiar la comparación.

Toda esta información puede ser explotada a la hora de comparar. A partir de esta información, se podría averiguar el orden de creación de los elementos y servir de esta manera como criterio de comparación; o, como ya se ha mencionado, la metainformación podría estar destinada específicamente a guiar la comparación, asociando marcas a los elementos que el comparador podrá posteriormente utilizar.

La metainformación tiene la ventaja de que el diseñador de los esquemas puede aprovecharla para modelar de una determinada manera, guiando de esta forma la futura comparación (ver figura 4); por ejemplo, puede crear en el mismo orden los elementos del esquema, y posteriormente utilizar un criterio que compare el orden de creación. En cualquier caso, no olvidemos que en muchas ocasiones, no se dispone de dos esquemas evolucionados uno del anterior, o ni siquiera han sido hechos por el mismo editor.

Las posibilidades de la metainformación son muy grandes, pero están fuertemente ligadas a las posibilidades del editor de esquemas subyacente. Si el editor no ofrece ninguna metainformación, este criterio no podrá ser aplicado.

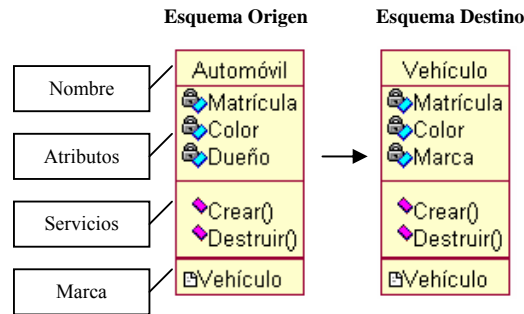


Fig. 4. La metainformación puede ser usada para dirigir la comparación

En la figura 4 se observa un ejemplo en el que el analista dejó una marca en la clase Vehículo en el momento de su modelado para que el algoritmo pueda identificar fácilmente los elementos correspondientes. En el ejemplo se observa que la marca 'Vehículo' se mantiene tras la evolución.

4.1.3.1 Comparación Utilizando Servicios de Evolución ($D, _2, N$)

Realmente no se trata de un criterio de comparación, puesto que no se sigue un proceso de comparación. Algunas herramientas CASE, son capaces de guardar la traza de llamadas a servicios que hacen evolucionar un esquema.

Estos servicios son llamados servicios de evolución, y dicha traza determina totalmente la evolución (Carsí, 1999). A partir de la traza de servicios de evolución, se puede averiguar mediante procesos heurísticos de deducción qué elementos proceden de cuales. Por ello, en caso de disponer de la traza de servicios de evolución la tasa de aciertos del comparador sería del 100%. Es obvio que para disponer de dicha traza, el esquema conceptual final ha de ser una evolución directa del inicial.

4.1.3.2 Comparación por Orden Relativo de los Elementos (D, X, N)

Si representamos un esquema conceptual como un árbol etiquetado, este criterio trata de comparar el orden relativo de los elementos dentro del árbol. Para el caso de atributos consiste en comparar el orden relativo de los mismos en el subárbol que representa una clase.

Se trata de comparar cierta metainformación que muchos (prácticamente todos) editores guardan, y que consiste en el orden en que fueron creados los atributos de las clases. Se parte de la idea de que en una evolución ese orden es conservado en su mayoría.

En este ejemplo vemos que el atributo 'C.lectivos' de la clase 'Materia' está exactamente en la misma posición que el atributo 'Créditos', lo cual le otorga una probabilidad alta de ser el mismo atributo. De hecho, si fuera un atributo nuevo estaría a la derecha del atributo 'Nº Temas' (obviamente, es dependiente del algoritmo de inserción que siga el editor).

En la mayoría de los casos el orden de creación de los elementos puede inferirse a partir del fichero donde la herramienta CASE guarda el modelo, y el citado algoritmo de inserción.

² Por su propia naturaleza, no es aplicable la clasificación de criterios exactos y estadísticos.

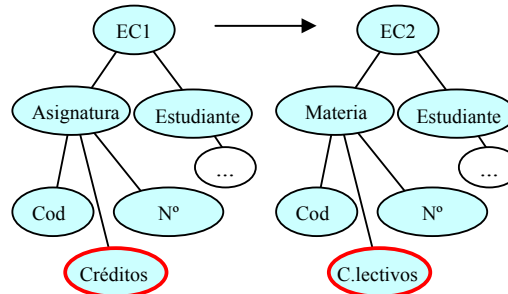


Fig. 5. El orden relativo de los elementos de conserva en la evolución

4.1.4 Comparación por Propiedades de Clase (I, E, N)

Existe mucha información asociada a las clases que puede ser explotada para comparar. Es el caso de los servicios. El número de servicios de una clase, así como su tamaño (Li y Henry (1993) proponen una métrica basada en la complejidad ciclomática de McCabe) pueden utilizarse para comparar. Asociado a este criterio, surge la idea de que todos los servicios de las clases no deberían ser tratados por igual, sino que deberían ponderarse asociando un peso mayor a los servicios compartidos. Surgen de esta manera diversos criterios asociados a los servicios de las clases. Finalmente, podemos utilizar el tamaño de las clases para comparar, considerando el tamaño de una clase como la suma del tamaño de sus atributos y sus métodos (este criterio está asociado a la métrica Sc de Li y Henry (1993), y a la versión ponderada de Thomas y Jacobson (1989)).

4.1.4.1 Comparación por Similitud de Atributos (I, E, N)

Este criterio está basado en la idea de que si dos clases evolucionan, es posible que cambien algunos atributos, pero es poco probable que cambien todos o la mayor parte de ellos. Se fundamenta en la idea de que en contextos de migración de datos, los cambios aplicados en un esquema conceptual suelen ser cambios evolucionarios y no revolucionarios (Staudt, 2000). Por su propia naturaleza, sólo puede utilizarse para comparar clases, y suele ser bastante acertado utilizarlo como segundo criterio de comparación³. Normalmente, se acepta un porcentaje de similitud como suficiente para dar la comparación por exitosa; por ejemplo, si dos clases tienen más del 75% de los atributos iguales se asume que dichas clases son la misma clase. Como puede suponerse, puede ocurrir que haya empates, lo que es un problema si no se utilizan criterios de desempate.

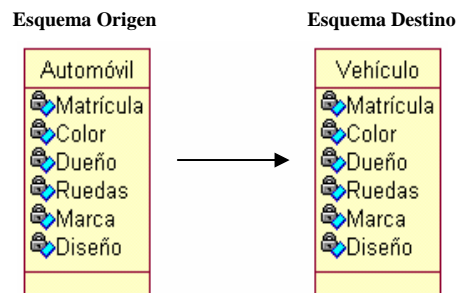


Fig. 6. Gran parte de los atributos suelen conservarse en una evolución

Existe otra variante que no mide la similitud de atributos, sino el número de atributos de la clase; este criterio puede ser útil cuando sabemos que la evolución se ha producido por renombrado de las clases; en estos casos el número de atributos permanece constante y puede ser un criterio a tener en cuenta. Debido a que también presenta el problema de los empates, este criterio puede ser muy útil combinado con otros criterios. Asociadas a este criterio está la métrica NOA (Lorenz y Kidd 1994) y la métrica WAC (Sharble y Cohen 1993).

³ Como veremos en el apartado dedicado a la combinación de criterios, lo usual es combinar varios criterios.

4.1.4.2 Comparación por Similitud de Tipos (I, X, N)

En el caso de los atributos, existe una propiedad muy propensa a ser utilizada para comparar; dicha propiedad es el tipo del atributo. Aunque es cierto que el tipo de un atributo puede cambiar en una evolución, el rango de tipos al que puede cambiar está limitado. Es posible que un atributo ‘fecha’ que era de tipo ‘Date’ evolucione a tipo ‘Long’ porque así lo indiquen los requerimientos; Sin embargo es poco probable que evolucione a tipo ‘Char’. Aprovechando esta idea, un algoritmo de comparación puede utilizar tablas de evolución de tipos en las cuales se especifiquen las probabilidades de posibles cambios de tipo que se pueden dar en una evolución. De esta manera el algoritmo podría calcular, por ejemplo, la probabilidad de que un atributo de tipo ‘Integer’ evolucionara a ‘long’ y así descartar posibles correspondencias.

En el caso de que se comparen estrictamente los tipos de los atributos este criterio está clasificado como exacto. En el caso de que se admita una probabilidad de cambio de tipo este criterio está clasificado como estadístico.

4.1.4.3 Comparación por Función de Identificación (I, X, N)

Las funciones de identificación se forman a partir de uno o más atributos y permiten identificar a los objetos por parte de los usuarios en el espacio del problema. En algunos sistemas (Letelier et al., 1998) (Grau, 1998) las funciones de identificación son conocidas como alias. Los criterios de comparación que utilizan el alias de la clase para comparar son un caso particular de la comparación por similitud de atributos (donde los atributos a comparar son los que forman el alias de la clase). Pero es importante reseñar una diferencia importante: Este criterio es de tipo exacto, ya que todos los atributos que forman el alias deben coincidir en la comparación, mientras que la comparación por similitud de atributos es un criterio clasificado como estadístico.

4.1.5 Comparación por Propiedades de Relación (I, E, N)

Existen tres tipos de relaciones en un esquema conceptual, cada una de ellas dispone de ciertas propiedades que se pueden utilizar en el proceso de comparación. Las relaciones de agregación y asociación pueden ser comparadas utilizando las cardinalidades máximas y mínimas de la relación. La especialización dispone del tipo de especialización (temporal / permanente, por-condición / por-evento). En los tres casos, existe la posibilidad de utilizar los roles de las clases participantes, además del nombre y el identificador de la relación.

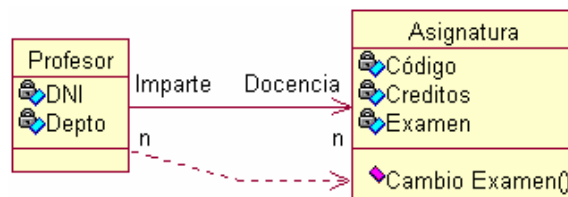


Fig. 7. Relaciones de asociación y agente entre clases

Dependiendo del modelo de objetos concreto que sirve de soporte a los esquemas conceptuales existirán variaciones, o características especiales para comparar. En OASIS existe la posibilidad de definir una agregación como inclusiva o relacional, disjunta o no-disjunta, flexible o estricta, estática o dinámica, inclusiva o multivaluada. En OmTroll (Grau, 1998) no está permitida la herencia múltiple ni la especialización dinámica, pero se puede especificar la interacción existente entre los elementos del esquema a partir del ‘Communication Diagram’, donde aparecen propiedades inherentes al lenguaje Troll. En la figura 7 se observan dos relaciones entre las clases ‘Profesor’ y ‘Asignatura’. La primera relación es una relación de asociación caracterizada por sus roles y su cardinalidad; la segunda relación es inherente al lenguaje Troll, y representa una relación de agente entre ambas clases que indica que los objetos de la clase ‘Profesor’ pueden invocar el servicio ‘Cambio Examen’ de la clase ‘Asignatura’.

4.1.6 Comparación por Similitud de Relaciones (I, E, N)

Del mismo modo que la comparación de clases puede estar guiada por la similitud de sus atributos, también lo puede estar por la similitud de sus relaciones con otras clases. Este criterio es totalmente resistente a los cambios sufridos en la plantilla

de una clase, siempre y cuando se mantengan sus relaciones de agregación, asociación y especialización con otras clases. Igual que ocurría en el caso de la similitud de atributos, se acepta un porcentaje de similitud como suficiente para dar la comparación por exitosa; y del mismo modo, este criterio también plantea el problema de los empates.

Debido a la independencia que mantiene este criterio respecto de la plantilla de las clases que se comparan, puede tener éxito en situaciones en las que los demás criterios suelen fallar; en la figura 8, puede observarse que a pesar de que 'Asignatura' ya no conserva ni su nombre, ni su Oid, ni ninguno de sus atributos, ni (podemos suponer) ninguna otra propiedad; si que conserva su relación con la clase 'Matemáticas', y esto hace que la comparación tenga éxito por este criterio. Este criterio también admite otra variante que no mide la similitud de relaciones, sino el número de relaciones que mantiene la clase. Asociada a este último criterio está la métrica CBO (Chidamber y Kemerer 1994).

Es evidente que para que puedan establecerse correspondencias entre clases utilizando este criterio, previamente habrán de establecerse todas las correspondencias entre las relaciones del esquema. Este es un claro ejemplo en el que se ve la interdependencia existente entre el algoritmo de comparación y el criterio utilizado para comparar.

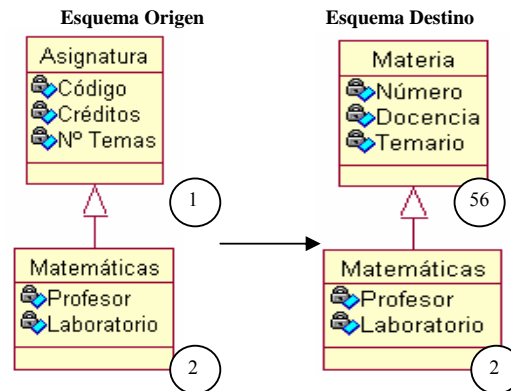


Fig. 8. Éxito en la comparación por similitud de relaciones

4.1.7 Comparación por Similitud de Clases (I, E, N)

La filosofía de este criterio es similar a la del anterior; no se fundamenta en las propiedades de los elementos que compara, sino en las de los que están relacionados con él. Se utiliza para comparar relaciones comparando las clases implicadas en la relación. En el caso de las relaciones de agregación y asociación se mirarán las propiedades de las clases componente y compuesta, y en el caso de las relaciones de especialización y generalización las de las clases padre e hija. En ambos casos se podrán aplicar las ideas comentadas en los criterios anteriores.

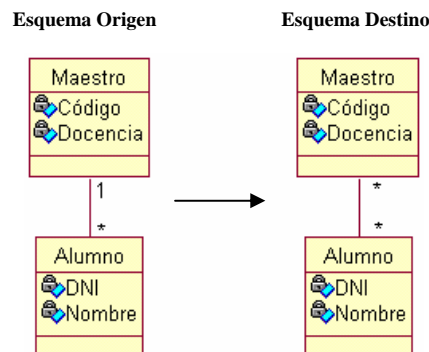


Fig. 9. Éxito en la comparación por similitud de clases

Este tipo de criterios es muy propenso a ser utilizado con la técnica de los anclajes que será explicada en la última sección de este apartado. Si el algoritmo de comparación ya ha identificado como correspondientes a dos pares de clases (existen dos anclajes), es muy probable que una relación entre ellas sea identificada como correspondiente.

4.1.8 Comparación por Población (I, E, P)

En algunas ocasiones, se da la circunstancia de que la migración entre las bases de datos asociadas a los esquemas conceptuales se realiza paulatinamente. En estas situaciones, puede ocurrir que ambos esquemas estén poblados⁴ cuando se produzca la comparación. En los casos en los que la población ya migrada no es borrada de las bases de datos iniciales, existe información duplicada en las bases de datos. Los criterios de comparación por población explotan dicha información para comparar.

Este criterio puede ser muy interesante cuando la población migrada no ha sufrido cambios durante la migración; en estos casos, por mucho que pueda cambiar el esquema, los datos asociados permanecen invariantes; este hecho supone una garantía en la comparación. Los algoritmos de comparación, solo pueden comparar aquella información almacenada que sea del mismo tipo. En la figura 10 se observa que la clase vehículo ha cambiado de nombre en la evolución; sin embargo, sus instancias han permanecido inmutables y permiten a los algoritmos identificar correctamente la evolución.

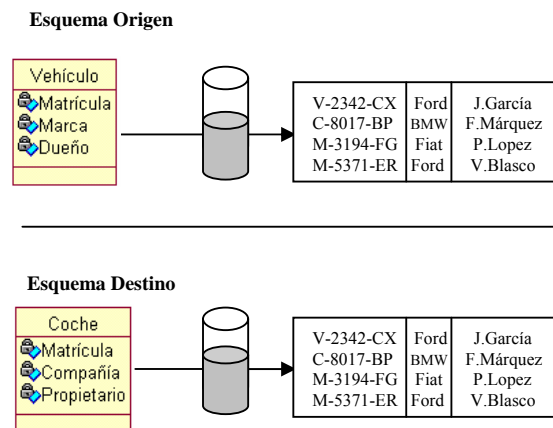


Fig. 10. La información almacenada en la base de datos puede ser utilizada para comparar

4.2 Combinación de Criterios

En los anteriores apartados se ha puesto de manifiesto que cada tipo de criterio presenta una mejor adaptación en determinados contextos. En la práctica, lo deseable es combinar varios criterios con el objetivo de mejorar la comparación cubriendo de esta manera las deficiencias de cada criterio con el otro al cual se combina.

La combinación de dos criterios puede realizarse de dos formas:

- * *Combinación secuencial*: Se aplica el primer criterio (dominante⁵) a cada elemento, y el segundo criterio a cada elemento que no haya sido emparejado por el primer criterio.
- * *Combinación paralela*: Se aplican los dos criterios a la vez a cada elemento, y se da una de las tres situaciones siguientes:
 - o Los dos aciertan: Se acepta la pareja.
 - o Solo uno acierta: Se acepta o rechaza la pareja.
 - o Los dos fallan: Se rechaza la pareja.

Si en la situación en la que sólo un criterio acierta, se rechaza la pareja, la combinación será llamada estricta; si la pareja es aceptada, la combinación será llamada no-estricta.

⁴ Que un esquema esté poblado quiere decir que existen instancias almacenadas en la base de datos asociada al esquema.

⁵ El otro criterio será llamado criterio débil.

Sea λ el conjunto de correspondencias encontradas por un criterio δ , sea α y β los subconjuntos de λ tales que α contiene las correspondencias correctas de λ y β las incorrectas ($\lambda = \alpha + \beta$). Cualquier criterio combinación secuencial δ' con δ como criterio dominante cumple lo siguiente:

- a) $|\lambda'| \geq |\lambda|$
- b) $|\alpha'| \geq |\alpha|$
- c) $|\beta'| \geq |\beta|$
- d) $\forall a \in \lambda \rightarrow a \in \lambda'$
- e) $\forall a \in \alpha \rightarrow a \in \alpha'$
- f) $\forall a \in \beta \rightarrow a \in \beta'$

Cualquier criterio combinación paralela estricta δ'' con δ como criterio participante cumple las siguientes propiedades:

- a) $|\lambda''| \leq |\lambda|$
- b) $|\alpha''| \leq |\alpha|$
- c) $|\beta''| \leq |\beta|$
- d) $\forall a \in \lambda'' \rightarrow a \in \lambda$
- e) $\forall a \in \alpha'' \rightarrow a \in \alpha$
- f) $\forall a \in \beta'' \rightarrow a \in \beta$

Para la combinación paralela no-estricta:

- a) $|\lambda''| \geq |\lambda|$
- b) $|\alpha''| \leq |\alpha|$ No tiene porqué cumplirse.
- c) $|\beta''| \leq |\beta|$ No tiene porqué cumplirse.
- d) $\forall a \in \lambda'' \rightarrow a \in \lambda$ No tiene porqué cumplirse.
- e) $\forall a \in \alpha'' \rightarrow a \in \alpha$ No tiene porqué cumplirse.
- f) $\forall a \in \beta'' \rightarrow a \in \beta$ No tiene porqué cumplirse.

Aunque a priori podría parecer que la combinación secuencial es similar a la combinación paralela en la que se acepta un acierto de cualquier criterio, esto no es así. En la combinación secuencial se prima mucho más el criterio dominante, de tal forma que todos los elementos que hayan sido emparejados por el criterio débil no podrán ser emparejados por el dominante.

En cambio, en la combinación paralela puede ocurrir que un elemento que haya sido emparejado por el criterio débil sí pueda ser emparejado por el dominante. Es importante saber combinar los criterios, y tenerlos en cuenta a la hora de modelar, puesto que pueden mejorar sustancialmente la comparación.

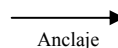
4.3 Utilización de Anclajes

La comparación de esquemas conceptuales es un proceso iterativo en el que se va obteniendo información a medida que se avanza en el proceso. En este contexto, el término anclaje, hace referencia a una correspondencia previamente obtenida durante el proceso de comparación. De manera formal, un anclaje en el instante t_i es una correspondencia obtenida por el algoritmo de comparación en un instante t_j donde $t_j < t_i$. Los anclajes representan información que puede ser tenida en cuenta por el criterio (y el algoritmo) de comparación.

Los anclajes están sujetos a la implementación particular del algoritmo de comparación; y su validez es dependiente del esfuerzo computacional y de la cantidad de información que el algoritmo emplea para calcularlos. Por ejemplo, un algoritmo de comparación podría realizar una primera comparación entre los elementos de los esquemas con el fin de establecer un cierto número de anclajes. Utilizando dichos anclajes realizar otra comparación (con algún criterio basado en anclajes) de la que se obtendrían a su vez nuevos anclajes, y así sucesivamente hasta establecer todas las correspondencias.

Esquema Origen

Esquema Destino



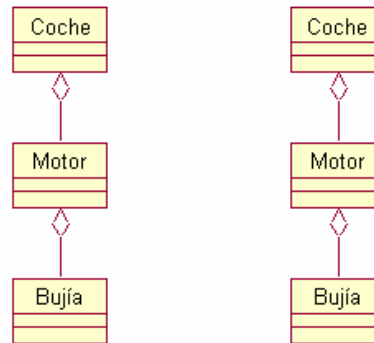


Fig. 11. Los anclajes establecen puntos de fijos para futuras comparaciones

Como puede apreciarse, el uso de anclajes supone un proceso incremental, en el que cada iteración obtiene información que podrá ser utilizada en la siguiente. De esta manera, los criterios utilizados en las primeras iteraciones deben ser criterios muy exigentes para garantizar que los anclajes sobre los que se soportarán el resto de comparaciones sean correctos.

Es importante hacer notar que un criterio de comparación puede tener una complejidad elevada si utiliza anclajes. Aunque este tipo de criterios presenta mayores costes y su complejidad ciclomática es de un orden superior (debido a su interdependencia con el propio algoritmo de comparación), este tipo de criterios será en la mayoría de los casos el criterio de comparación apropiado, puesto que por termino medio, presenta una mejoría sustancial en sus tasas de evaluación (Silva et al., 2001).

5 Análisis Estadístico de Comparación de Criterios

Una vez concluido el análisis teórico, disponemos de toda una estructura conceptual que es necesario validar de manera empírica. La experimentación que se ha realizado en este sentido ha seguido la línea que se conoce como diseño intra-sujetos. Este tipo de experimentos están caracterizados por el hecho de que una vez se ha seleccionado la muestra que va a participar en el experimento, a todos los sujetos se les aplican todas las condiciones experimentales.

A pesar de que sí se han tenido en cuenta posibles efectos distorsionantes como pudiera ser el aprendizaje, la fatiga, o la práctica; no ha existido una aleatorización rigurosa a la hora de seleccionar la muestra con la que se pretendía realizar la experimentación. Este hecho se debe a que los esquemas conceptuales no han sido seleccionados aleatoriamente, sino que han sido escogidos por sus distintas propiedades y características. Se ha seleccionado una muestra de quince esquemas conceptuales orientados a objetos, y todos ellos tienen entre sí características distintas (algunos han sufrido muchos renombres, modificaciones masivas, simplificaciones... ; se han utilizado esquemas dependientes e independientes; se ha forzado el caso de reconstrucción de clases para la no coincidencia de Oids; etc.). Se dispone así de estadísticas de comparación, fruto de la experimentación, que ponen de manifiesto cómo se comportan los distintos criterios ante una gran diversidad de circunstancias. Este fue el objetivo principal de la experimentación y ha sido ampliamente satisfecho como se puede ver en la tabla 1.

La cual está dividida en dos bloques independientes; el primer bloque muestra las tasas de acierto obtenidas por siete criterios distintos habiendo sido utilizados frente a una muestra de quince esquemas. La fila ‘Total’ muestra la tasa media de acierto para cada criterio. El segundo bloque representa la muestra utilizada compuesta por quince esquemas conceptuales orientados a objeto. En esta tabla, la columna ‘Evolución’ indica la dependencia o independencia entre el esquema origen y el destino; el resto de campos son autoexplicativos.

La principal conclusión que se extrae de las estadísticas es que los criterios que se especializan en comparar por información identificativa obtienen mejores tasas de éxito; es el caso, por ejemplo, del alias. En todas las pruebas, la comparación por alias ha superado a la comparación por similitud de atributos. Teniendo en cuenta que el alias es un subconjunto de los atributos, queda de manifiesto que existen atributos que deberían pesar más que los demás en una ponderación. Del estudio se deduce que deben otorgarse los pesos más elevados a aquellos atributos que participen en alguna función de identificación.

Tabla 1. Estadísticas de comparación obtenidas en el experimento

Análisis Teórico-Experimental de Criterios de Comparación de Esquemas Conceptuales Orientados a Objeto

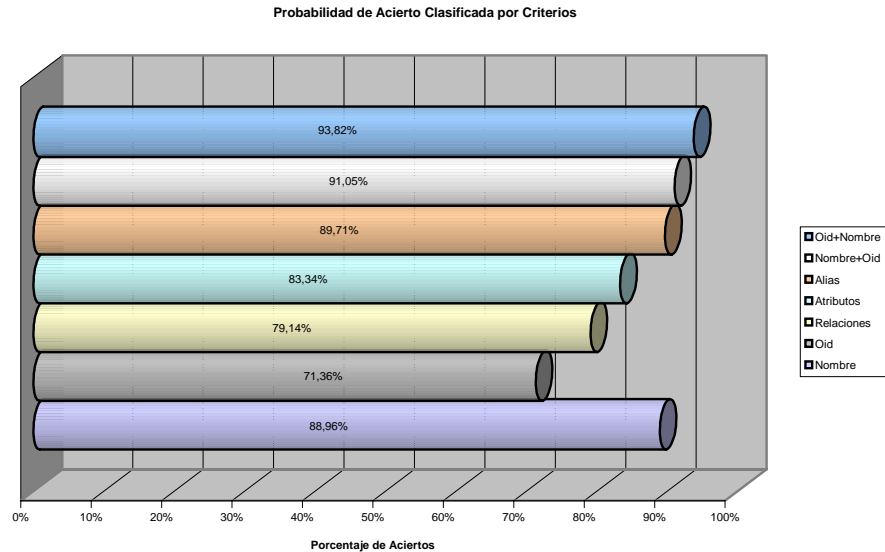
Esquema	Nombre	Oid	Relaciones	Atributos	Alias	Nombre+Oid	Oid+Nombre
1	66,67%	100%	37,50%	66,67%	66,67%	66,67%	100%
2	100%	100%	82,45%	100%	100%	100%	100%
3	91,67%	0%	100%	83,33%	100%	91,67%	91,67%
4	92,86%	0%	85,71%	78,57%	85,71%	92,86%	92,86%
5	88,89%	83,33%	77,78%	83,33%	83,33%	94,44%	100%
6	100%	100%	75%	100%	100%	100%	100%
7	100%	100%	100%	100%	100%	100%	100%
8	77,78%	22,22%	66,67%	77,78%	88,89%	77,78%	77,78%
9	88,23%	94,12%	82,35%	76,47%	88,23%	88,23%	100%
10	100%	100%	100%	100%	100%	100%	100%
11	63,15%	0%	52,63%	47,37%	68,42%	63,15%	63,15%
12	83,33%	100%	83,33%	91,67%	91,67%	100%	100%
13	100%	88,89%	77,78%	88,89%	100%	100%	100%
14	100%	100%	75%	100%	100%	100%	100%
15	81,81%	81,81%	90,90%	72,72%	72,72%	90,90%	81,81%
Total	88,96%	71,36%	79,14%	83,34%	89,71%	91,05%	93,82%
Esquema	Nombre	Evolución	Inserciones	Borrados	Modificaciones	Nº Clases	
1	Gestión de facturas	Si	2	0	2	8	
2	EvEs20a	Si	3	0	0	11	
3	Discografía	No	4	1	5	12	
4	Zoológico	No	5	2	3	14	
5	Club Tenis	Si	7	3	6	18	
6	Gestoría Pardo	Si	4	0	0	11	
7	Comparador	Si	7	0	12	18	
8	Hipermercado	No	4	2	4	9	
9	Viajes Márquez S.A.	Si	6	8	12	17	
10	Fotógrafos	Si	1	5	3	10	
11	Venta Software	No	8	6	10	19	
12	Consultoría	Si	5	3	6	12	
13	Seguridad Cajeros	Si	3	2	4	9	
14	Videoclub	Si	2	2	3	8	
15	Competición Judo	Si	5	4	6	11	

En los casos en los que los esquemas sean dependientes, los criterios que utilizan el Oid son muy aconsejables. En los casos en que sean independientes no tendrá sentido aplicar ni combinar ningún criterio clasificado como dependiente. En las estadísticas queda confirmada la idea de que la combinación de criterios mejora en la mayoría de los casos el éxito en la comparación (ver gráfica 1). Por otra parte queda de manifiesto que la combinación de criterios no cumple la propiedad conmutativa puesto que la combinación de Oid y Nombre no obtiene las mismas tasas de acierto que la combinación de Nombre y Oid. En esta combinación particular es preferible utilizar como dominante el criterio Oid frente al de Nombre.

En la tabla 2 se muestran las desviaciones típicas del conjunto de pruebas de cada criterio. Se observa que a pesar de que los esquemas conceptuales presentaban condiciones de comparación muy diversas, la desviación típica es pequeña; lo cual indica que en la mayoría de los casos los criterios tienen una tasa de aciertos similar y un comportamiento estable.

Tabla 2. Desviación típica obtenida en las pruebas de cada criterio

Nombre	OID	Relaciones	Atributos	Alias	Nombre+Oid	Oid+Nombre
0,123	0,418	0,172	0,152	0,122	0,123	0,111



Gráfica 1. Porcentaje medio de acierto obtenido en 7 criterios de comparación

Los valores de la desviación típica de la tabla 2 dan una idea de la variabilidad de comportamiento de cada criterio ante diversas situaciones experimentales. En el caso de la comparación por Oid la desviación típica es elevada; esto se debe a que se ha utilizado este criterio de tipo independiente para comparar esquemas dependientes lo cual ha hecho que se obtengan tasas de acierto muy bajas y aumente la desviación típica. Muy lejos de este valor, los criterios de comparación por Oid son extraordinariamente estables, y presentan varianzas muy pequeñas. En el caso de nuestro experimento, el valor de la desviación típica para el criterio de Oid calculado solamente a partir de las comparaciones de esquemas dependientes es de 0,072.

La experimentación que ha sido presentada, forma parte del desarrollo de una herramienta para la migración automática de datos que está siendo desarrollada en el Departamento de Sistemas Informáticos y Computación de la Universidad Politécnica de Valencia. Dicha herramienta es capaz de generar de manera totalmente automatizada un plan de migración de datos entre bases de datos de sistemas orientados a objetos.

La primera fase de la herramienta es la comparación automática de los esquemas a partir de una combinación de criterios de comparación, llegándose incluso a realizar combinaciones secuenciales de hasta cuatro criterios. Dicha comparación identifica los orígenes y los destinos de los datos en la migración, y a partir del mismo es generado un plan de migración por defecto. Cada comparación automática puede ser supervisada por un analista que determinará cuando ha existido un error en la comparación y podrá solucionarlo realizando nuevas comparaciones solamente sobre aquellos elementos que él determine. La experiencia obtenida hasta el momento concluye que los errores son muy escasos, obteniéndose tasas de acierto que oscilan sobre el 98%.

6 Conclusiones

El éxito en un proceso de comparación de esquemas conceptuales está totalmente ligado al criterio de comparación empleado. Podemos clasificar los criterios en función de la información que utilizan para llevar a cabo la comparación; existen tres clasificaciones primarias no disjuntas entre sí:

- * Criterios que utilizan la dependencia o independencia de los esquemas.
- * Criterios que requieren exactitud o aproximación en la comparación.
- * Criterios que utilizan o no, la población de los esquemas a comparar.

No todos los criterios obtienen las mismas tasas de acierto, ni existe un criterio que sea siempre mejor que los demás. Cada criterio de comparación tiene unas características propias que le hacen ser más apropiado en unas situaciones, y me-

nos apropiado en otras. Deberá ser el analista quien decida qué criterio es apropiado, en función del tipo de evolución que haya entre los esquemas.

En general, lo más acertado es utilizar una combinación de criterios que se complementen entre sí. Dicha combinación puede realizarse de una manera paralela o secuencial. También es aconsejable el uso de anclajes, puesto que suponen una mejora en las tasas de acierto que justifica su incremento en el coste computacional.

Se ha establecido una tipología de criterios de comparación clasificada en función de la información utilizada para comparar. Para cada tipo de criterio, se han determinado sus características principales detalladas con ejemplos.

Finalmente, las estadísticas extraídas de comparaciones de esquemas conceptuales industriales han permitido extraer conclusiones fruto de pruebas reales. Se ha podido estudiar qué criterios mantienen un comportamiento medio más adecuado ante diversos tipos de esquemas conceptuales. Además, a partir de la varianza se ha podido estudiar cómo y cuanto varía dicho comportamiento. El análisis experimental ha contrastado el estudio teórico previo, permitiendo cuantificar las conclusiones establecidas.

Referencias

1. **Aho A.V., Hirschberg D.S. y Ullman J.D.**, "Bounds on the complexity of the longest common subsequence problem", *Journal of the ACM*, 23(1):1-12, January 1976.
2. **Anaya R.**, "Desarrollo y gestión de componentes reutilizables en el marco de Oasis", Tesis Doctoral, Universidad Politécnica de Valencia, Noviembre 1999.
3. **Assenova P. y Johannesson P.**, "Improving Quality in Conceptual Modelling by the Use of Schema Transformations", in: B. Thalheim (Ed.): *Conceptual Modeling: proceedings ER '96*, Springer, Berlin et al., S. 277-291, 1996.
4. **Calero C.**, "Definición de un conjunto de métricas para la mantenibilidad de bases de datos relacionales, activas y objeto-relacionales", Tesis Doctoral, Universidad de Castilla-La Mancha, 2001.
5. **Carsí J.A.**, "OASIS como marco conceptual para la evolución del software", Tesis Doctoral, Universidad Politécnica de Valencia, Octubre 1999.
6. **Dolado J.J. y Fernández L.**, "Medición para la gestión en la ingeniería del software", Libro Editorial RA-MA, ISBN 84-7897-403-2, 2000.
7. **Grau A.**, "Computer-Aided Validation of formal conceptual models", Tesis Doctoral, Technical University of Braunschweig, 1998.
8. **Herden O.**, "Measuring Quality of Database Schemas by Reviewing - Concept, Criteria and Tool", Oldenburg Research and Development Institute for Computer Science Tools and Systems, Escherweg 2, 26121 Oldenburg, Germany
9. **Kuo-Chung Tai**, "The tree-to-tree correction problem", *Department of Computer Science, North Carolina State University, Raleigh, Journal of the ACM*, vol. 26 no 3, pp. 422-433, 1979.
10. **Letelier P., Ramos I., Sánchez P. y Pastor O.**, "Oasis 3.0: Un enfoque formal para el modelado conceptual orientado a objeto", Editorial SPUPV- 98.4011, Departamento Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, 1998.
11. **Moreno M., García F. y Polo M.**, "Medición de la calidad del software en el ámbito de la especificación de requisitos", Universidad de Salamanca, Departamento de Informática y Automática, Taller JISBD, 2000.
12. **Pérez J., Carsí J.A., Ramos I., Anaya V. y Silva J.**, "Generación Automática de un plan de migración entre poblaciones de esquemas conceptuales orientados a objetos", Jornadas de Trabajo de DOLMEN, Sevilla 2001.
13. **Ramos I. y Pastor O.**, "OO-Method: Una metodología OO para la producción automática de software.", Technical Reports II/8/95, Departamento Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, Valencia, España. Junio 1995
14. **Sankoff D.**, "Matching sequences under deletion/insertion constraints", *Proc. Nat. Acad. Sci., USA* 69, pp. 4—6, 1972.
15. **Selkow S.**, "The tree-to-tree editing problem", *Department of Computer Science, University of Tennessee, Knoxville, Information Processing Letters*, 6(6):184-186, December 1977.
16. **Silva J.**, "Comparación de esquemas conceptuales orientados a objeto y generación automática de planes de migración", Proyecto Final de Carrera, Facultad de Informática, Universidad Politécnica de Valencia, 2001.
17. **Staudt B.**, "A model for compound type changes encountered in schema evolution", *University of Massachusetts, Amherst, ACM Transactions on Database Systems*, Vol. 25, No. 1, March 2000, Pages 83–127.
18. **Varas M. y Pradenas J.**, "Hacia la definición de métricas de calidad para esquemas conceptuales de bases de datos", *Revista Electrónica del DIIC*, edición nº 6, año 3, 2000.
19. **Wagner R.A. y Fisher M.J.**, "The string to string correction problem", *Journal of the ACM*, 21(1):168-173, January 1974.
20. **Wong C.K. y Chandra A.K.**, "Bounds for the string editing problem", *Journal of the ACM*, 23(1):13-16, January 1976.

Josep Silva, José A. Carsí e Isidro Ramos



Josep Silva. *Se graduó como Ingeniero Superior en Informática en la Universidad Politécnica de Valencia, España. Terminó sus estudios de Especialista en Programación Declarativa e Ingeniería de la Programación en la misma universidad obteniendo finalmente el Diploma de Estudios Avanzados. Actualmente trabaja como investigador para el grupo de Investigación ELP de la UPV en España. Sus campos de interés incluyen metodologías formales de análisis y diseño, incluyendo transformaciones de programas como la fragmentación estática y dinámica, depuración y optimización de código en lenguajes funcionales y lógico-funcionales, y la construcción de plataformas de desarrollo para lenguajes multiparadigma.*



José A. Carsí Cubel. *Es Licenciado en Informática en 1993 y Doctor en Informática en 1999 por la Universidad Politécnica de Valencia. Actualmente es profesor Titular de Universidad en la Universidad Politécnica de Valencia siendo miembro del grupo de investigación “Programación Lógica e Ingeniería del Software”. Áreas de interés: evolución del software, gestión de modelos, lenguajes de descripción de arquitecturas y modelado orientado a objetos.*



Isidro Ramos Salavert. *Es Licenciado en Ciencia Físicas por la Universidad de Valencia (España) desde 1966, Licenciado en Informática por la Universidad Politécnica de Madrid (España) desde 1972, Doctor en Ciencias Físicas por la Universidad Complutense de Madrid (España) desde 1971. Áreas de interés: modelado conceptual de sistemas de información, lenguajes de descripción de arquitecturas y gestión de modelos. Actualmente es Catedrático de Universidad y director del grupo de Investigación “Programación Lógica e Ingeniería del Software” de la Universidad Politécnica de Valencia.*