

Evaluando la Calidad de Métodos para el Diseño de Aplicaciones Web¹

Magalí González, Silvia Abrahão, Joan Fons, Oscar Pastor

Departamento de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia
Camino de Vara s/n, P.O. Box: 22012, E-46020 - Valencia, España.
[magali, sabrahao, jjfons, opastor]@dsic.upv.es

Resumen

El incremento en la complejidad de los Sistemas de Información Basados en Web (WIS), hace que cada vez sea más evidente la necesidad de aplicar técnicas ingenieriles al desarrollo de los mismos. A raíz de esto, en estos últimos años han surgido una serie de métodos que permiten modelar y generar aplicaciones Web complejas. Este artículo presenta un marco de evaluación de métodos para el diseño y desarrollo de aplicaciones Web, y su aplicación a un caso de estudio.

Palabras claves: Sistemas de Información Basados en Web, Marco Conceptual, Métodos de Modelado Conceptual, Calidad de Métodos

Abstract

Nowadays, with the rapid expansion of the Internet, there are a number of initiatives which are intended to provide a solution for the creation of Web-based Information System (WIS) within a well-defined software production process. Considering this, may be very interesting for WIS designers to carry out an analysis or evaluation of the existing design methods that would be most appropriate to use in each case. This study presents a proposal of a framework for the evaluation process of WIS design methods. In order to clarify the proposal, this framework is also applied to a case study.

Key Words: Web Based Information Systems, Conceptual Framework, Conceptual Modelling Methods, Methods quality

1. Introducción

La Ingeniería Web intenta definir un conjunto de métodos, técnicas y herramientas que deben ser utilizadas para abordar el desarrollo correcto de aplicaciones Web. Hoy en día, no es sólo importante obtener un producto correcto en el sentido de libre de errores, más importante aún es que el producto obtenido sea de calidad. En este sentido, en los últimos años han surgido varias propuestas que presentan métodos, técnicas y herramientas para medir la calidad de las aplicaciones que se desarrollan. Por tal motivo, la evaluación de calidad corresponde a un aspecto fundamental en la Ingeniería Web.

Entonces, la primera pregunta que debería surgir a raíz de todo esto es: ¿Qué evaluar para asegurar un producto de calidad?

¹ Esta investigación está soportada por el Programa CYTED, en el proyecto VII.18, WEST y el Proyecto CICYT, con ref. TIC 2001-3530-C02-01

El estándar ISO/IEC 9126-1 presenta un marco conceptual para el modelo de calidad.

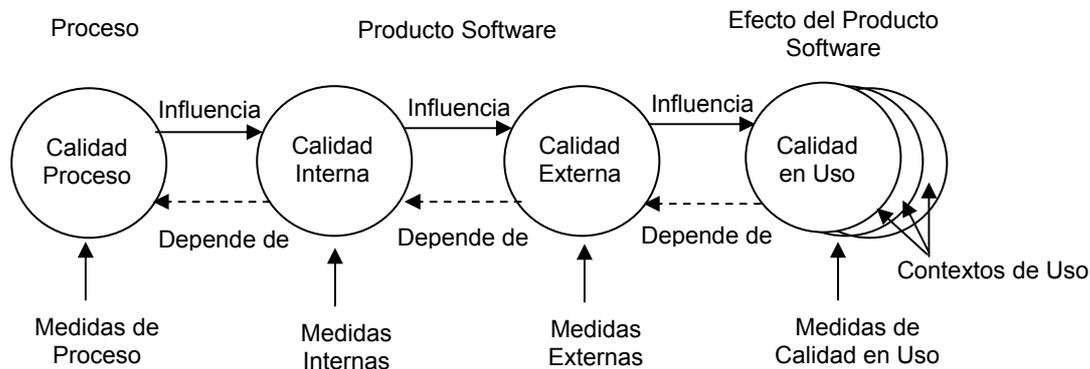


Figura 1. Marco conceptual para el modelo de calidad

En esta figura podemos ver claramente que la calidad del proceso (calidad de cualquiera de los procesos del ciclo de vida definido en ISO/IEC 12207) contribuye a mejorar la calidad del producto y que la calidad del producto contribuye a mejorar la calidad en uso. Entonces, mejorar el proceso de desarrollo ayuda a obtener un producto de mejor calidad, evaluar y mejorar la calidad del producto mejora la calidad en uso. Presentamos a continuación algunos trabajos realizados para cada caso:

1. Una propuesta muy aceptada para medir la *calidad del proceso* corresponde al estándar CMM (en inglés, Capability Maturity Model), siendo un modelo que ayuda a juzgar la madurez de un proceso software de una organización e identifica las prácticas claves requeridas para incrementar la madurez los mismos. Otro estándar bien conocido es ISO/IEC 15504 (SPICE).
2. Existen diversas propuestas que intentan medir la *calidad de los productos*, siendo alguna de ellas a nivel cualitativo y comparativo [19, 20], y otras que comprenden métodos robustos y formales aplicando conceptos cuantitativos como la metodología Web Site QEM (en inglés, Web Site Quality Evaluation Method) [17]. También existen diferentes estándares presentados por la ISO como el ISO/IEC 9126 que propone un modelo de calidad compuesto de una serie de características de calidad y una serie de métricas externas e internas [21]. El estándar ISO/IEC 14598 presenta además pautas que ayudan al proceso de evaluación considerando diferentes actores (desarrolladores, evaluadores, adquirentes).
3. En cuanto a la calidad en uso podemos mencionar al estándar ISO/IEC 9126-4 que contiene ejemplos de métricas para medir la productividad, efectividad, seguridad y satisfacción.

La evaluación de métodos puede considerarse un trabajo muy interesante, ya que, considerando a los métodos como recursos del proceso de desarrollo, el hecho de utilizar un método de calidad para diseñar (es decir, un método que posee las primitivas necesarias para modelar aplicaciones web), puede ayudar a obtener una aplicación Web de calidad, aunque evidentemente la experiencia del diseñador juega también un rol importante para utilizar apropiadamente tales primitivas. Existen algunos trabajos de evaluación de métodos, entre ellos, podemos mencionar a Christodolou que propone un marco para la evaluación de métodos hipermediales presentando una serie de atributos y aplicando métodos cuantitativos para la evaluación de los mismos [22]. En su propuesta define un conjunto de atributos a evaluar agrupados según aquellos que tienen relación

con el método propiamente dicho, el ambiente de desarrollo y las propiedades del sistema. Existen también otros trabajos de evaluación de métodos pero que aplican métodos cualitativos y que se realizan a nivel comparativo [6].

Una vez respondida a la pregunta de qué es conveniente evaluar, deberíamos preguntarnos lo siguiente ¿cómo evaluar?

Franca Garzotto presenta una clasificación de las diferentes formas de evaluación [25]:

1. **Automáticos:** aplicando software que realizan evaluación.
2. **Métodos empíricos:** observando usuarios que usan el sistema. En el caso de los métodos podría ser aplicando los métodos a casos reales y así verificando las propiedades de los mismos.
3. **Métodos Informales:** expertos observan el sistema (o el método).
4. **Métodos formales:** técnicas que formalizan el proceso de evaluación.

El objetivo principal de este estudio es proponer un marco robusto y flexible para evaluar de forma cuantitativa distintos métodos de desarrollo de aplicaciones Web, presentando una estructura compuesta de características, sub-características y atributos que intenta capturar toda la expresividad² necesaria que los métodos deberían poseer, para medir así la calidad de los mismos. El marco propuesto aplicará métodos empíricos, formales e informales para realizar el proceso de evaluación de diferentes métodos de diseño de aplicaciones Web.

El resto del artículo se organiza de la siguiente manera: en el apartado 2 se presenta el marco de evaluación, en el apartado 3 se aplica el marco a la evaluación de dos métodos bien conocidos, y en el apartado 4 se presentan las conclusiones y trabajos futuros.

2. Marco de evaluación

El marco de evaluación de métodos se divide básicamente en tres fases. La primera fase denominada captura de requisitos de calidad pretende identificar cuáles son los atributos que interesan ser medidos, y también definir una estructura de características, sub-características, y atributos. La segunda fase, comprende el proceso de medición de atributos, y el cálculo de características y sub-características a través de un proceso de agregación y puntaje. En la última fase se realiza un estudio de los resultados obtenidos. El marco general se resume en la figura 2.

2.1. Fase de captura de requisitos

Paso 1. Paradigma GQM (Goal-Question-Metrics)

El GQM [16] provee un marco que comprende los siguientes tres niveles:

- a. Nivel conceptual (objetivo): se define un objetivo compuesto de un objeto, un propósito, un proceso, un punto de vista y una característica de calidad.
- b. Nivel operacional (pregunta): se usa una serie de preguntas para caracterizar la forma en que los logros de un objetivo específico serán alcanzados. Las preguntas intentan caracterizar el objeto a ser medido con respecto a una característica de calidad desde el punto de vista especificado.

² Un método es más expresivo que otro cuando sus primitivas y constructores son capaces de captar de mejor forma la realidad que se desea modelar (aplicaciones web complejas, sistemas de información, etc). En este sentido, cuando un método posee mayor expresividad, puede decirse también que es de mejor calidad.

- c. Nivel cuantitativo (métrica): se asocia un conjunto de métricas a cada pregunta de forma a responderlas de manera cuantitativa.

En este sentido, el paso 1 se compone de los siguientes sub-pasos:

- 1) Aplicar el nivel conceptual al paradigma GQM: para identificar el objetivo principal de la evaluación, junto con el propósito, proceso, punto de vista y hecho de calidad. Ya que intentamos evaluar métodos de desarrollo de aplicaciones Web, el punto de vista a evaluar será el de los diseñadores.
- 2) Aplicar el nivel operacional para especificar una serie de preguntas que intentan contestar el objetivo principal y que en el paso 2 pasarán a convertirse en características y sub-características.
- 3) Se define una serie de atributos cuantificables que intentan responder a cada pregunta.

El paradigma GQM no es utilizado para realizar las mediciones, solamente para obtener la estructura que se presenta en el paso 2 (árbol de características, sub-características y atributos).

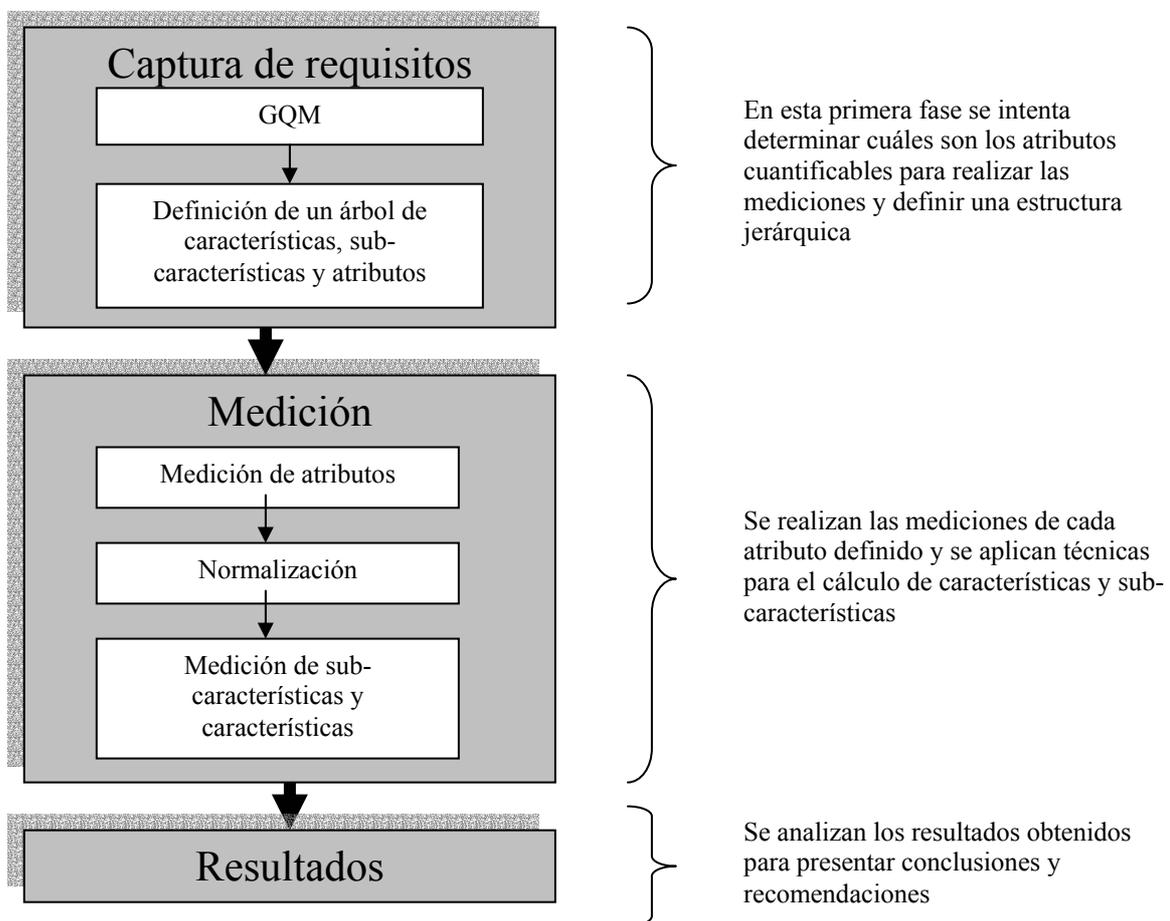


Figura 2. Fases del marco de evaluación de métodos

Paso 2. Especificación de un árbol de atributos

Se crea un árbol aplicando los siguientes sub-pasos:

1. La raíz del árbol corresponde a la *característica de calidad* definida en el objetivo del paradigma GQM (nivel conceptual).

2. Las características y sub-características se obtienen a partir de las preguntas surgidas del nivel operacional.
3. Las hojas del árbol serán los atributos cuantificables definidos en el nivel operacional del paradigma GQM. Estos atributos pueden ser cuantificados aplicando métricas directas o indirectas dependiendo del criterio que será utilizado para medir.

El estándar ISO 9126-1 [21] presenta un modelo de calidad compuesto de una serie de características que ayudan a la evaluación de la calidad de un producto. Se podría pensar que una manera de obtener el árbol descrito anteriormente debería ser a partir del modelo presentado por este estándar; pero creemos que para medir la calidad de un método, los atributos a considerar deben ser distintos, ya que el objeto no es el mismo (el modelo presentado en el estándar ayuda a evaluar un producto, nosotros intentamos evaluar un método).

2. 2. Fase de Medición

Paso 3. Medición de los atributos

En este paso se realiza la cuantificación de cada atributo, es decir, se asigna un valor numérico a cada atributo conforme a los criterios establecidos para su medición. Comprende los siguientes sub-pasos:

1. *Descripción*: se da una definición de cada atributo especificado, y la razón por la cual se encuentra especificado.
2. *Observaciones*: se realizan observaciones del atributo para identificar la forma en que los métodos lo expresan, y así justificar cualitativamente el valor que se asignará cuando adquiera un valor numérico.
3. *Implantación de la métrica*
 - a. *Parámetros*: se identifican los parámetros disponibles (en caso de que la métrica sea indirecta).
 - b. *Definición de la métrica*: se especifican los criterios utilizados para la medición.
 - c. *Rango de aceptabilidad*: se especifica un menor y mayor grado de aceptabilidad, esto significa que el menor grado de aceptabilidad corresponde al grado menos deseado y cualquier valor por debajo de este grado será considerado como no aceptado; en caso contrario, cualquier valor por encima del grado mayor será considerado como aceptado con el mayor grado de satisfacción.

Paso 4. Normalización de los atributos

El objetivo de este paso es obtener los criterios de calidad elementales para determinar la preferencia elemental de cada atributo. Cada atributo medido en el paso anterior tendrá asociado un valor numérico, que por medio de un criterio elemental producirá una preferencia elemental, que corresponde al grado de requerimiento del usuario satisfecho para el atributo. Para más detalles sobre preferencia elemental ver [17].

La preferencia elemental corresponde a un valor comprendido entre 0 y 1, en donde 0 significa que el valor del atributo no satisface el requerimiento de calidad (en este caso el valor del atributo es menor o igual que el menor grado de aceptabilidad), y 1

significa que el valor del atributo satisface completamente el requerimiento de calidad (en este caso el valor del atributo es igual o mayor al mayor grado de aceptabilidad). Para ello se utiliza una fórmula llamada función elemental que se encarga de realizar el mapeo del valor cuantificado del atributo al valor de preferencia elemental.

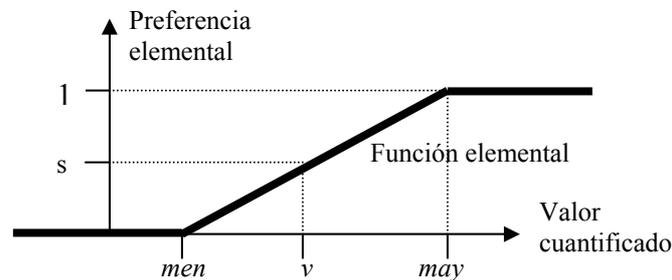


Figura 3. Criterio elemental para la obtención de la preferencia elemental

En la figura 3 podemos ver que los valores *men* y *may* corresponden al menor y mayor grado de satisfacción para cada atributo medido; el valor *v* corresponde al valor obtenido en la medición del atributo; *s* corresponde al grado de preferencia elemental que coincide con el grado de satisfacción. Esto se obtiene a partir de la siguiente fórmula matemática que corresponde a la función paramétrica de la figura 3.

$f(v) = s$, donde

$$\begin{cases} 0, & \text{si } v \leq \text{men} \\ 1, & \text{si } v \geq \text{may} \\ \frac{v - \text{men}}{\text{may} - \text{men}}, & \text{si } \text{men} < v < \text{may} \end{cases} \quad (1)$$

Esta fórmula nos dice que si el valor de *v* es menor que *men*, entonces se asigna el valor 0 a *s* (ya que indica que está por debajo del grado de satisfacción mínimo), y si el valor de *v* es mayor a *may*, se asigna el valor 1 (ya que indica que está por encima del grado de satisfacción máximo). Si *men*=0 y *may*=1, entonces tenemos que la preferencia elemental *s* coincide con el valor *v* (siempre y cuando *v* esté en el rango [0,1]), esto puede verse fácilmente despejando la fórmula matemática (1).

Paso 5. Cálculo de las sub-características y características

Para realizar el proceso de cálculo de las características y sub-características, se ha decidido aplicar el modelo de agregación y cálculo propuesto por Dujmovic [23, 24] denominado modelo de Agregación Lógica de Preferencias (en inglés, Logic Scoring of Preference, LSP).

LSP es un modelo de agregación y puntaje para evaluar sistemas complejos en donde sus resultados representan el grado de satisfacción de los usuarios conforme a los requerimientos de calidad establecidos. Es una generalización de los modelos y técnicas de puntajes aditivos y lineales, y tiene sus fundamentos en principios y modelos matemáticos y de lógica [17].

Las actividades básicas del proceso de agregación se presentan a continuación [17]:

1. Selección del tipo de relación lógica entre atributos, sub-características o características.

1. Polarización lógica conjuntiva (simultaneidad): cuando los participantes en el proceso de evaluación perciben que dos o más entradas deben estar presentes simultáneamente.
2. Polarización lógica disyuntiva: cuando los participantes en el proceso de evaluación perciben que dos o más entradas pueden estar presentes alternativamente (ej. la presencia de un atributo puede reemplazar la ausencia de otro).
3. Polarización neutra: cuando se percibe que dos o más preferencias de entrada pueden agruparse de un modo independiente.
2. Selección del tipo de función conforme a la relación de entradas entre atributos, sub-características o características.
 1. Relación simétrica: cuando se percibe que dos o más preferencias de entrada pueden agruparse de un modo independiente.
 2. Relación asimétrica: Cuando se requiere modelar requerimientos mandatorios con requerimientos no-mandatorios (atributos obligatorios se combinan con otros deseables y/u opcionales), o cuando condiciones necesarias se combinan con condiciones suficientes.
3. Selección del operador conforme al nivel de intensidad de la polarización lógica. Para más detalles de los operadores principales ver [17, pp. 107].
4. Selección de la importancia relativa de cada entrada conforme a los requerimientos: el evaluador puede valorar la relativa importancia de los elementos de entrada a las funciones de acuerdo a su intuición y experticia, o puede utilizar mecanismos como encuestas y establecer fórmulas de relativa importancia para computar los pesos.
5. Cómputo de las preferencias parciales y globales: una vez que fueron estructurados y acordados todos los criterios y funciones de agregación, los tomadores de decisión deben ejecutar el programa que calcula las preferencias de calidad parciales y globales para cada sistema participante aplicando la fórmula de la *media de potencia pesada* (del inglés, *weighted power mean*, *wpm*) [23].

2.2.3. Fase de Resultados

En esta fase se realizan actividades de análisis y comparaciones de las mediciones obtenidas de los atributos cuantificables, de las sub-características y las características. A partir del objetivo de evaluación definido en el paso 1 (en donde se considera el propósito, punto de vista, objeto y la característica de calidad), el proceso culmina con las conclusiones y recomendaciones del caso.

3. Evaluación de dos métodos

El marco de evaluación fue aplicado a dos métodos de diseño de aplicaciones Web. Se han seleccionado los siguientes métodos: OOHDM y OOWS [3, 4, 10, 11]; debido a que estamos interesados en evaluar métodos orientados a objetos en primera instancia y ambos aplican enfoques orientados a objetos, y además son conocidos, y mencionados en la literatura.

El trabajo de evaluación fue realizado por un equipo de investigadores de aplicaciones Web, pertenecientes a un grupo que trabaja fuertemente en el campo de modelos conceptuales y generación automática de código a partir de modelos conceptuales.

En las siguientes secciones presentaremos cada uno de los pasos del marco propuesto.

3.1.1. Paso 1: *paradigma GQM*

La siguiente tabla se presenta un ejemplo de la forma en que fue determinado tanto el objetivo, las preguntas y los atributos cuantificables. En esta sección no se presenta la tabla completa por cuestiones de espacio y en la siguiente sección se mostrará una primera versión del árbol (modelo de calidad) obtenido a partir del mapeo del paradigma GQM.

Objetivo:	Objeto Propósito Característica Punto de vista	Evaluar Métodos Orientados a Objetos para diseñar WIS Expresividad Diseñador
Pregunta		¿Qué acerca del modelado de la información?
Pregunta		¿Qué es necesario para modelar la estática?
Pregunta		¿Qué es necesario para modelar las clases?
Atributo cuantificable		Atributos
Atributo cuantificable		Operaciones
Pregunta		¿Qué es necesario para modelar las relaciones?
Atributo cuantificable		Agregación
Atributo cuantificable		Herencia
Pregunta		¿Qué es necesario para modelar la dinámica?
Pregunta ...		¿Qué acerca del modelado de navegación? ...

Tabla 1. Plantilla GQM

3.1.2. Paso 2: *Especificación de un árbol de atributos*

Como consecuencia de la identificación del objetivo, preguntas y métricas del paso 1 y en base a experiencias de desarrollo de aplicaciones Web se define el árbol de atributos de la figura 4.

El objetivo principal de crear una estructura en forma de árbol, es que una vez obtenido cada uno de los valores numéricos de los atributos cuantificables, pueden aplicarse métodos de agregación y puntaje para calcular las sub-características y características (modelo LSP presentado en el apartado 2).

El árbol tiene como raíz la característica expresividad y se descompone en aspectos que tienen relación con el dominio de información y la navegación. Consideramos que otro aspecto importante es la presentación, y en trabajos posteriores será considerado.

3.1.3. Paso 3: *Medición de los atributos*

En este apartado presentaremos las mediciones algunos de los atributos relacionados con la navegación. Los resultados de las mediciones de los demás atributos pueden verse en la tabla 2, ya que por cuestiones de espacio no serán presentados en detalle en esta sección. En particular, nos centraremos en los atributos especificados para la sub-característica vistas, y los agruparemos para simplificar la lectura.

Raiz: Expresividad

1. Dominio de información

- 1.1. Requisitos
 - 1.1.1. Funcionales
 - 1.1.2. No funcionales
- 1.2. Aspectos del modelado
 - 1.2.1. Estática
 - 1.2.1.1. Objetos
 - 1.2.1.1.1. Establecer estado
 - 1.2.1.1.1.1. Tipos de atributos
 - 1.2.1.1.1.2. Restricciones
 - 1.2.1.1.2. Modificar estado
 - 1.2.1.1.2.1. Operaciones
 - 1.2.1.2. Relaciones
 - 1.2.1.2.1. Agregación
 - 1.2.1.2.1.1. Asociación
 - 1.2.1.2.1.2. Composición
 - 1.2.1.2.2. Herencia
 - 1.2.1.2.2.1. Generalización
 - 1.2.1.2.2.2. Especialización

1.2.1.2.3. Accesibilidad de Información

- 1.2.2. Dinámica
 - 1.2.2.1. Cambios de estado
 - 1.2.2.2. Interacción de objetos
 - 1.2.2.3. Efectos de un evento

2. Navegación

- 2.1. Requisitos
- 2.2. Aspectos del modelado
 - 2.2.1. Vistas
 - 2.2.1.1. Globales
 - 2.2.1.2. Por usuarios
 - 2.2.1.3. Por subsistemas
 - 2.2.2. Unidades de navegación
 - 2.2.2.1. Información
 - 2.2.2.2. Estructura interna
 - 2.2.2.3. Tipos de Navegación
 - 2.2.3. Estructuras de acceso
 - 2.2.3.1. Tipos
 - 2.2.3.2. Estructuras modeladas
 - 2.2.3.3. Ordenación

Figura 4. Árbol de atributo que corresponde al modelo de calidad

Sub-característica 2.2.1. Vistas

1. Descripción: Esta sub-característica está compuesta de tres atributos (vistas globales, vistas parciales, por subsistemas) e intenta determinar si el método posee primitivas que modelan una estructura global de la navegación, una estructura por cada usuario y la posibilidad de agrupar estructuras a partir de un objetivo (identificando así módulos o subsistemas).

2. Observaciones:

OOHDM: Presenta los diagramas de clases navegacionales y el modelo de esquemas de contextos navegacionales que permiten identificar una estructura global de la aplicación. Los diagramas de clases navegacionales corresponden a vistas del esquema conceptual y los esquemas de contexto modelan el espacio de navegación incluyendo estructuras de acceso y contextos (que corresponde a un conjunto de instancias de una clase navegacional). Se podrían crear vistas parciales por usuario agrupando los contextos a partir de los tipos de usuarios que tienen acceso a los mismos. Las vistas por módulos o subsistemas no las modela de manera explícita, pero en los esquemas de contextos pueden modelarse fácilmente sub-módulos.

OOWS: Para representar la navegación este método usa el modelo de mapas navegacionales que es generado para cada agente identificado en una fase anterior. Este modelo contiene contextos que representan vistas del modelo de objetos y que en el espacio de la solución corresponde a una página web. Por tal motivo, presenta una estructura global para cada agente, pero no para toda la aplicación. Este método aún no permite la definición de subsistemas de navegación.

3. Definición de la métrica: La medición de este atributo se realiza de forma binaria, ya que lo único que nos interesa saber es si el método posee o no primitivas que

modelan lo especificado en la descripción (grado de disponibilidad). En otros casos podríamos tener otros criterios de medición.

Atributo	OOHDM	OOWS
Vistas globales	1	0
Vistas por usuarios	0	1
Vistas por Subsistemas	1	0

Tabla 2. Resultado de la medición de los atributos

4. Mayor y menor grado de satisfacción: en este caso el mayor grado de satisfacción corresponde al valor 1, lo cual significa que el método posee la expresividad necesaria para modelar lo deseado (variable *may*), y el menor grado de satisfacción corresponde al valor 0, queriendo decir que el método no posee la expresividad necesaria para modelar lo deseado (variable *men*).

3.1.4. Paso 4: Normalización de atributos

En este paso se realiza el cálculo de la preferencia elemental aplicando la fórmula (1) a cada atributo medido. Los resultados de la normalización pueden verse en la tabla 2. Estos resultados corresponden al valor de s de la fórmula (1) cuyas variables son v , may , y men , siendo v el valor numérico obtenido después de aplicar la implantación de la medición; may el mayor grado de satisfacción; y men el menor grado de satisfacción (ver sub-paso 3 de la medición de atributos). Estos valores de preferencias serán usados en el paso 5 para el cálculo de las sub-características y características.

Atributo	OOHM	OOWS
Requisitos funcionales (1.1.1)	1	0,5
Requisitos no funcionales (1.1.2)	0	0
Tipos de atributos (1.2.1.1.1.1.)	1	1
Restricciones (1.2.1.1.1.2.)	0	1
Operaciones (1.2.1.1.2.1.)	1	1
Asociación (1.2.1.2.1.1.)	1	1
Composición (1.2.1.2.1.2.)	1	1
Generalización (1.2.1.2.2.1.)	1	1
Especialización (1.2.1.2.2.2.)	1	1
Accesibilidad de Información (1.2.1.2.3.)	0	1
Cambios de estado (1.2.2.1.)	0	1
Interacción de objetos (1.2.2.2.)	0	1
Efectos de un evento (1.2.2.3.)	0	1
Requisitos navegacionales (2.1.)	1	0,5
Vistas globales (2.2.1.1.)	1	0
Vistas por usuarios (2.2.1.2.)	0	1
Vistas por subsistemas (2.2.1.3.)	1	0
Información (2.2.2.1.)	1	1
Estructura interna (2.2.2.2.)	1	1
Tipos de Navegación (2.2.2.3.)	1	1
Tipos (2.2.3.1.)	1	1
Estructuras modeladas (2.2.3.2.)	1	1
Ordenación (2.2.3.3.)	1	1

Tabla 3. Cálculo de las preferencias elementales

3.1.5. Paso 5: Cálculo de las sub-características y características

La medición de las sub-características y características se realiza aplicando el modelo de agregación LSP descrito anteriormente.

La estructura de agregación de preferencias parciales puede verse en la figura 5. Los valores que se encuentran en color rojo corresponden a los pesos asignados a cada uno de los atributos, las letras corresponden al tipo de polarización lógica definida entre atributos [24] y los números en negrita corresponden a las referencias de las sub-características y características del árbol. En particular los operadores lógicos aplicados en el árbol son: A (polarización neutra), C (conjunción pura), C- (conjunción débil), D- (disyunción débil), DA (disyunción media), D++ (disyunción fuerte). Para más detalles ver [17 pp. 105].

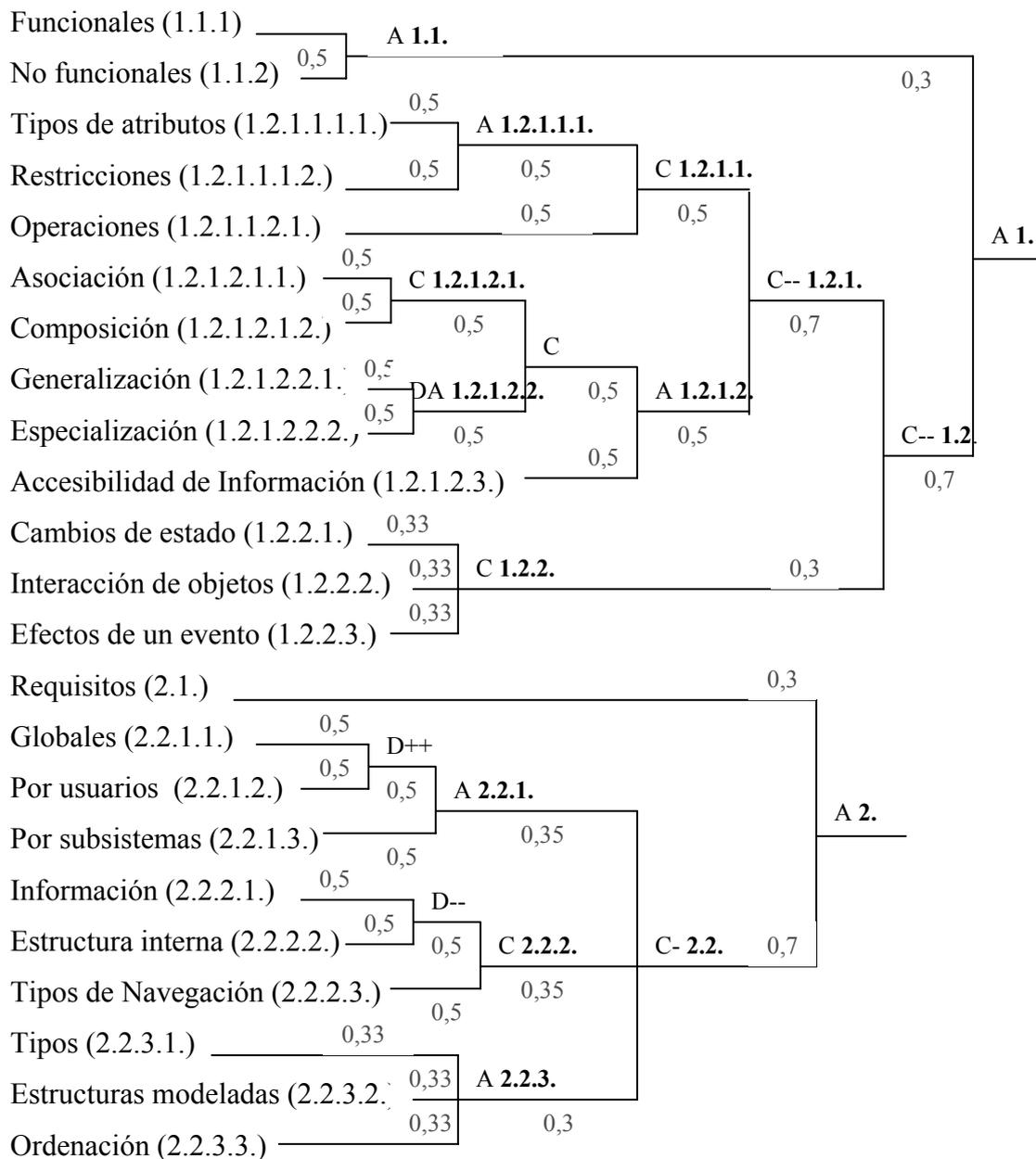


Figura 5. Agregación LSP

3.1.6. Paso 6: Resultados

En esta fase se realiza un análisis de los resultados obtenidos. Por cuestiones de espacios, no presentamos el análisis completo, pero sí algunas observaciones y resultados generales.

La preferencia elemental de cada atributo puede verse en la tabla 3, y los resultados al aplicar el modelo LSP a las sub-características y características se presentan en la tabla 4. Para una mayor comprensión de los resultados, podrían utilizarse otras tablas y gráficos, pero nosotros nos centraremos en presentar un análisis comparativo de los métodos a partir de los resultados obtenidos.

Características y sub-características / Métodos	OOHDM	OOWS
1. Dominio de información	0,346	0,775
1.1. Requisitos	0,5	0,25
1.2. Aspectos del modelado	0,281	1
1.2.1. Estática.....	0,5	1
1.2.1.1. Objetos.....	0,5	1
1.2.1.1.1. Establecer estado.....	0,5	1
1.2.1.1.2. Modificar estado.....	1	1
1.2.1.2. Relaciones	0,5	1
1.2.1.2.1. Agregación.....	1	1
1.2.1.2.2. Herencia	1	1
1.2.1.2.3. Accesibilidad.....	0	1
1.2.2. Dinámica.....	0	1
2. Navegación.....	0,996	0,701
2.1. Requisitos	1	0,5
2.2. Aspectos del modelado	0,994	0,787
2.2.1. Vistas	0,983	0,483
2.2.2. Unidades de navegación	1	1
2.2.3. Estructuras de acceso	1	1

Tabla 4. Resultados del cálculo de agregación y puntaje LSP

El modelo de requisitos para el dominio de información y la navegación se encuentra fuertemente definido en OOHDM, por tal motivo ha adquirido un puntaje mayor.

En cuanto a los aspectos del modelado del dominio de información podemos ver que OOWS ha adquirido un mayor puntaje. Esto se debe a que OOWS corresponde a la extensión de OO-Method [9] (un método tradicional orientado a objetos de generación automática de código a partir de modelos conceptuales), que modela de manera mucho más precisa aspectos de comportamiento (dinámica) de la aplicación, a través de su modelo dinámico para los cambios de estado e interacción de objetos, y el modelo funcional que permite modelar los efectos de un evento.

En los aspectos del modelado de la navegación OOHDM ha adquirido un mayor puntaje ya que por un lado, aspectos relacionados a la vista global del sistema, y separación por módulos o subsistemas puede fácilmente modelarse, cosa que no ocurre con OOWS. Un análisis posterior, y que deseamos realizar con estudios más detallados de evaluación de métricas, es refinar los atributos obtenidos, y proponer atributos relacionados con la presentación.

Otro aspecto importante a mencionar es que OOHDM surge inicialmente para el diseño de sistemas hipermediales y posteriormente es extendido para aplicaciones Web. OOWS sin embargo, corresponde a la extensión de un método para el diseño de sistemas de información tradicionales. Esto también se refleja en los resultados, ya que

en aspectos referentes a información OOWS ha adquirido mayor puntuación, no así en los aspectos de navegación ya que OOHDM lo ha superado.

Ambos métodos son muy potentes, y presentan primitivas que cubren requerimientos de las aplicaciones Web, con ciertas diferencias, pero que a partir de una evaluación comparativa pueden ser catalogados y analizados de una manera mucho más precisa.

4. Conclusiones

En este artículo se ha presentado un marco de evaluación de métodos para el diseño y desarrollo de aplicaciones Web, y su aplicación a un caso de estudio.

El aporte principal de este trabajo es un enfoque sistemático y cuantitativo para; por un lado determinar un modelo de calidad que permita evaluar y comparar la expresividad de métodos de desarrollo para aplicaciones web; realizar la evaluación de dos métodos; y a partir de los resultados y recomendaciones identificar las semejanzas y diferencias, fortalezas y debilidades de los mismos.

El propósito de esta investigación es llegar a obtener un árbol de atributos que capture toda la expresividad necesaria para el diseño de aplicaciones Web. Por tal motivo, como trabajo futuro tenemos previsto aplicar el árbol a otros casos de estudio, para así refinarlo. También se tiene previsto realizar validaciones tanto a nivel de atributos y de las mediciones aplicando técnicas de validaciones empíricas y teóricas [18].

AGRADECIMIENTOS: Nos gustaría agradecer a Luis Olsina por sus útiles aportes y recomendaciones en este trabajo.

5. Referencias

[1] Cachero C., Gomez J., and Pastor O. "Conceptual Design of Electronic Product Catalogs Using OOH-Method Techniques". In *19th International Conference on Conceptual Modeling (ER'00)*, Salt Lake City, USA, 2000. Springer-Verlag.

[2] Lowe D. and Hall W. *Hypermedia and the Web: An Engineering Approach*. Hohn Wiley & Sons Ltd, 1999.

[3] Schwabe D. and Rossi D. "Developing Hypermedia Applications using OOHDM". In *Workshop on Hypermedia Development Processes, Methods and Models (Hyper-Text '98)*, Pittsburgh, USA, 1998.

[4] Rossi G. "Uma Metodología Orientada a Objetos para o Projeto de Aplicativos Hipermedia". *PhD thesis*, Departamento de Informática da PUC-Rio, <http://www-lifia.info.unlp.edu.ar/~fer/oohdm/>, 1996.

[5] *The Object - Oriented Paradigm*. <http://home.earthlink.net/salhir/theobjectorientedparadigm.html>, 1998.

[6] Cernuzzi L., Rossi G., and Ortiz S. "Propuesta de un Marco de Evaluación para Modelos y Metodologías de Desarrollo de Aplicaciones Hipermedia". In *XXV Conferencia Latinoamericana de Informática (CLEI99)*, pages 1-12, Asunción, Paraguay, Septiembre 1999.

[7] Cernuzzi L. and González M. "Evaluación de Metodologías para el Diseño de WIS: la Propuesta de un Framework". In *4th Iberoamerican Workshop on Requirements Engineering and Software Environments (IDEAS'2001)*, pages 91-103, San Juan, Costa Rica, Abril 2001.

[8] Fenton N. and Pfleeger S. *Software Metrics: A Rigorous and Practical Approach*. PWS Publishing Company, 1997.

- [9] Pastor O., Insfrán E., Pelechano V., Romero J., and Merseguer J. “OO-Method: An OO Software Production Environment Combining Conventional and Formal Methods”. In *9th Conference on Advanced Information Systems Engineering (CAiSE'97)*, pages 145-159, Barcelona, Spain, Junio 1997. Springer-Verlag. LNCS (1250).
- [10] Pastor O., Abrahão S., and Fons J. J. “Building E-Commerce Applications from Object-Oriented Conceptual Models”. In *Newsletter of the ACM SIGecom Exchanges*, volume 2(2), pages 24-32, Junio 2001.
- [11] Pastor O., Abrahão S., and Fons J. J. “Object-Oriented Approach to Automate Web Applications Development”. In *2nd International Conference on Electronic Commerce and Web Technologies (EC-Web'01)*, Munich, Germany, Septiembre 2001. Springer-Verlag.
- [12] Diaz P., Aedo I., and Panetsos F. “Modeling the Dynamic Behavior of Hypermedia Applications”. In *IEEE Transactions on Software Engineering*, volume 27(6), pages 550-572, Junio 2001.
- [13] Ceri S., Fraternali P., and Bongio A. “Web Modeling Language (WebML): a Modeling Language for Designing Web Sites”. In *WWW9*, volume 33 (1-6), pages 137-157. Computer Networks, 2000.
- [14] Murugesan S., Deshpande Y., Hansen S., and Ginige A. “Web Engineering: A New Discipline for Development of Web Based Systems”. In *21th International Conference on Software Engineering*, Los Angeles, USA, Mayo 1999.
- [15] Isakowitz T., Stohr E., and Balasubramanian P. “A Methodology for the Design of Structured Hypermedia Applications”. In *Communications of the ACM*, volume 8(38), pages 34-44, 1995.
- [16] Basili V. and Rombach H. “The TAME Project: Towards Improvement Oriented Software Environments”. In *IEEE Transactions on Software Engineering*, volume 14(6), pages 758-773, 1988.
- [17] Olsina L. “Metodología Cuantitativa para la Evaluación y Comparación de la Calidad de Sitios Web”. *Tesis PhD*, Facultad de Ciencias Exactas, Universidad Nacional de la Plata, Argentina, 1999.
- [18] Genero M. “Defining and Validating Metrics for Conceptual Models”. *Tesis PhD*, Departamento de Informática, Universidad de Castilla-La Mancha, España, 2001.
- [19] McCall J.A, Richards P.K. and Walters, G.F. “Factores in Software Quality”, RADC TR-77-369, 1977.
- [20] White D.R.J Scott, D.L. and Schultz. “POED: a method of Evaluating System Performance”, *IEEE TEM*, Diciembre, 1963.
- [21] ISO/IEC 9126-2001 International Standard, “Information technology – Software product evaluation – Quality characteristics and guidelines for their use”, 2001.
- [22] Christodoulou S.P., Styliaras G. D., Papatheodorou T. S. “Evaluation of Hypermedia Application Development and Management Systems”. *ACM Hypertext '98*, Pittsburgh, PA, USA, June 20-24, 1998.
- [23] Dujmovic J.J. “Weighted Conjunctive and Disjunctive Meands and their Application in System Evaluation”, *Journal of the University of Belgrade*, EE Dept, Series Mathematics and Physics, Nro. 483, pp. 147-158, 1974.
- [24] Dujmovic J.J and Elnicki, R. “A DMS Cost/Benefit Decision Model: Mathematical Models for Data Management System Evaluation, Comparison and Selection”, *National Bureau of Standards*, Washington D.C. Nro. GCR 82-374. NTIS Nro. PB 82-170150, 1982.
- [25] Garzotto F., Mainetti L. and Paolini P. “Hypermedia Design, Analysis and Evaluation Issues”, *Communication of the ACM*, Vol. 38, Nro. 8, August 1995.