

MECANOS: EXPOSICIÓN DE RESULTADOS Y LÍNEAS DE TRABAJO ABIERTAS EN LA REUTILIZACIÓN SISTEMÁTICA DEL SOFTWARE

Francisco J. García Peñalvo*
fgarcia@gugu.usal.es

José Manuel Marqués Corral[♥]
jmmc@infor.uva.es

María del Pilar Romay Rodríguez[♦]
mpromay@infor.uva.es

Yania Crespo González-Carvajal[▲]
yania@infor.uva.es

Resumen

En el presente trabajo se expone una visión global de los avances realizados en la definición de un modelo de reutilización soportado por unas estructuras complejas de reutilización denominadas Mecanos, de forma que facilite la inclusión de dichas estructuras en entornos avanzados de desarrollo software.

Los Mecanos constituyen unos elementos software reutilizables formados por assets clasificados en diferentes niveles de abstracción y por las relaciones existentes entre ellos, contribuyendo al aumento del nivel de abstracción en el proceso de reutilización.

El modelo de reutilización que se está construyendo sobre la base de los Mecanos reúne diferentes perspectivas en el área de la reutilización: un enfoque técnico centrado en la construcción y mantenimiento de Mecanos, un enfoque de proceso y un enfoque de cualificación.

Palabras Clave

Mecano, reutilización del software, estructura compleja de reutilización, relación estructural, relación semántica, modelo de reutilización, línea de producto.

1. Introducción

Son muchas las referencias en la literatura especializada que se encargan de ensalzar los beneficios potenciales de la reutilización del software, véase [2, 11, 12], y sin embargo son muchas menos las que citan ejemplos concretos y representativos de los beneficios reales de la reutilización del software.

Esto induce a pensar que la reutilización sistemática del software necesita nuevas aportaciones que la acerquen más a los entornos de ingeniería del software basados en herramientas automáticas y soportados por ambientes metodológicos acordes con las nuevas tendencias en el mundo del desarrollo del software.

Bajo la premisa de conseguir una *reutilización del software más automatizada* se plantea la necesidad de contar con una estructura de reutilización que permita construir elementos software reutilizables tanto de forma asistida por un experto en el desarrollo para la reutilización, como de forma automática en respuesta a una petición realizada por un desarrollador con reutilización.

* Departamento de Informática y Automática. Facultad de Ciencias. Universidad de Salamanca. Plaza de la Merced S/N. 37008, Salamanca (España).

♦ Becaria de Investigación del grupo GIRO. Ed. de Tecnologías de la Información y las Telecomunicaciones. Universidad de Valladolid. Campus Miguel Delibes. 47011, Valladolid (España).

♥ Departamento de Informática. Ed. de Tecnologías de la Información y las Telecomunicaciones. Universidad de Valladolid. Campus Miguel Delibes. 47011, Valladolid (España).

▲ Dpto. CC. de la Computación, Universidad de La Habana. Estudiante de Doctorado en la Univ. de Valladolid (Becaria del Instituto de Cooperación Iberoamericana - ICI). Ed. de Tecnologías de la Información y las Telecomunicaciones. Universidad de Valladolid. Campus Miguel Delibes. 47011, Valladolid (España).

A la hora de definir esta estructura compleja de reutilización se ha tenido en cuenta que los elementos reutilizables deben cumplir una serie de principios básicos:

- **Aumento del nivel de abstracción de la reutilización:** Se busca que el alcance de la reutilización no se quede estancado en el nivel de implementación (*nivel de abstracción al que tradicionalmente se ha asociado la reutilización*), dirigiéndose hacia niveles de mayor abstracción (*diseño y análisis*).
- **Soporte simultáneo de diferentes niveles de abstracción:** La estructura de reutilización debe dar soporte a un elemento software reutilizable de una granularidad tal que está compuesto por un conjunto de elementos reutilizables de grano fino (*los cuales a lo largo de este documento serán designados como assets¹*) relacionados entre sí y clasificados en diferentes niveles de abstracción.
- **Soporte para la trazabilidad:** Debe permitir la navegación entre los diferentes assets componentes a través de la red de enlaces existentes entre ellos, con especial interés en los enlaces entre los assets clasificados en diferentes niveles de abstracción.
- **Presencia tanto en el desarrollo para reutilización como en el desarrollo con reutilización:** Con esta estructura se busca una dicotomía que la haga válida tanto para la creación de elementos reutilizables por los desarrolladores para reutilización, como para la creación de elementos reutilizables en tiempo de reutilización como respuesta a solicitudes de desarrolladores con reutilización.

Los elementos reutilizables construidos bajo estas premisas reciben el nombre de Mecanos, constituyendo la base sobre la cual construir un modelo de reutilización que integre el proceso general de la reutilización software, en el que se encuadran las actividades propias del desarrollo para reutilización y las actividades propias del desarrollo con reutilización, tal y como se indica en el proyecto REBOOT [10].

En este documento se realiza una exposición del trabajo realizado en la definición del modelo de reutilización. Para ello, en la sección dos se realiza un repaso por la definición de los Mecanos como elementos software reutilizables, haciendo especial hincapié en el modelo de Mecano. En la sección tres se aborda la experiencia práctica de la adaptación de EUROWARE [14] para dar soporte a los Mecanos. La cuarta sección recorre las líneas de trabajo que actualmente se están explorando en relación con los Mecanos, centrando la atención en el modelo de reutilización que se está construyendo entorno a ellos y en la relación entre los Mecanos y las líneas de producto. En la sección cinco se hace un breve repaso por los principales trabajos relacionados con lo que aquí se presenta. Las conclusiones y el trabajo futuro cierran este artículo en la sección seis.

2. Mecanos como elementos reutilizables

La definición del modelo de reutilización está estrechamente relacionada con la definición del modelo de elemento reutilizable que le da soporte, el modelo de Mecano.

Los Mecanos son elementos reutilizables de granularidad gruesa que ofrecen soporte a diferentes niveles de abstracción de forma simultánea, y que se pueden definir de una forma más precisa como: “*Un sistema de elementos software reutilizables, clasificados en diferentes niveles y relacionados entre sí, ya sea dentro de un mismo nivel de abstracción (relaciones intranivel) o entre diferentes niveles de abstracción (relaciones internivel), cumpliéndose la restricción de que debe existir al menos una relación internivel*”. [5,7].

¹En el ambiente del presente trabajo se ha decidido utilizar el vocablo inglés **asset** para designar un elemento reutilizable de grano fino, independientemente de la fase del ciclo de vida en el que fue creado, en detrimento del término **componente**, cuya múltiple utilización en diferentes contextos puede dar lugar a una ambigüedad semántica que se prefiere evitar.

Estos elementos reutilizables, como se ha mencionado anteriormente, deben integrarse en el proceso general de la reutilización del software, esto es, deben estar presentes tanto en las actividades propias del desarrollo para reutilización como en las actividades propias del desarrollo con reutilización.

Dentro del desarrollo para la reutilización, la actividad más importante es la encargada de la creación de los Mecanos. Desde esta perspectiva, los Mecanos se crean con un enfoque compositivo. Se parte de un espacio de reutilización formado por assets clasificados en diferentes niveles de abstracción y por sus interrelaciones, obteniéndose como resultado los Mecanos que se almacenan en un repositorio.

Una vez que un Mecano queda almacenado en el repositorio se tiene un Mecano persistente [6], que no variará en el tiempo por acciones diferentes a las propias del mantenimiento del repositorio.

Sin embargo, limitar el proceso de generación de Mecanos únicamente al enfoque compositivo daría como resultado una estructura de reutilización estática, en el sentido de que sólo se podrían recuperar los Mecanos almacenados en el repositorio. Aumentar el potencial de reutilización ofertado por el repositorio pasa por introducir un enfoque generativo que permita generar estructuras complejas de reutilización, Mecanos, de acuerdo a las necesidades específicas del reutilizador dentro de las que proporciona el suministrador del repositorio. Esto es, generar nuevos Mecanos a partir de los assets del repositorio que se ajusten a unos requisitos expresados por el desarrollador con reutilización.

Los Mecanos generados en tiempo de reutilización están más en consonancia con las actividades propias del desarrollo con reutilización: selección de assets, un proceso de generación automática del Mecano navegando a través de las relaciones existentes entre los assets seleccionados, y por último el estudio, filtrado, adaptación e integración del Mecano resultante. El proceso finaliza, bien cerrando el ciclo y realimentando el repositorio con el Mecano generado (*el Mecano generado se convierte en un nuevo Mecano persistente*), o bien desechando el mecano generado tras su reutilización.

Esta dualidad en la creación de Mecanos da lugar a un modelo de reutilización mixto composición-generación.

2.1 Modelo de Mecano

La definición de la estructura compleja de reutilización se ve plasmada en el modelo de Mecano². La exposición detallada y justificación del modelo de Mecano se encuentran recogidas en [7].

En la Figura 1 se muestra el modelo de Mecano³, el cual se caracteriza por los siguientes aspectos:

- En el modelo aparece la clase **Mecano**, que representa el elemento reutilizable complejo formado por una agregación de assets.
- Que un Mecano sea una agregación y no una composición es fácil de argumentar. Cada uno de los componentes de un Mecano puede ser parte de otros Mecanos. Además, la eliminación de un Mecano como elemento reutilizable no conlleva obligatoriamente a la eliminación de sus componentes.
- De los dos puntos anteriores se puede concluir que la connotación semántica de la relación de agregación entre el **Mecano** y los **Assets** se corresponde con lo que **UML 1.1** denomina agregación compartida, y que no es más que un tipo de agregación débil que implica que si se destruye el “todo” de la agregación (*el Mecano*) no se tienen que eliminar las “partes” constituyentes (*los assets*), y lo que es más importante, las partes constituyentes pueden aparecer en varios agregados al mismo tiempo.

² Para la realización de este modelo se ha utilizado UML 1.1 como lenguaje de modelado.

³ El modelo de Mecano presentado en este documento es la culminación de un trabajo de refinamiento de la definición de la estructura de Mecano inicialmente presentada en [4] y posteriormente revisada en [5,7].

- El Mecano presenta una restricción de definición importante que debe quedar reflejada en el modelo: **'todo Mecano debe contar con al menos una relación internivel'**. Esta restricción aparece en el modelo como una sentencia **OCL** en una nota asociada a la clase semántica **Mecano**.
- Se da por sentado que el repositorio es el soporte necesario para el almacenamiento de los assets producidos como consecuencia de los desarrollos para reutilización siendo, además, el soporte al desarrollo de sistemas software con reutilización.
- El objetivo que se persigue con la definición de esta estructura no es la construcción de una herramienta **CASE**, ni de un motor de repositorios, sino definir, desde la perspectiva de los repositorios, un esquema que permita almacenar, manejar y recuperar elementos reutilizables, más concretamente Mecanos.
- Si un Mecano se ve como el modelo conceptual de una estructura de información formada por assets y sus relaciones, el Mecano está al mismo nivel que un esquema de bases de datos, siendo el sistema gestor de bases de datos el repositorio. Bajo esta perspectiva, el nivel de datos se corresponde con los assets introducidos en el repositorio, siendo el nivel de modelado el modelo de Mecano.
- El modelo de Mecano presenta una flexibilidad total para la inclusión de nuevos tipos de assets y de relaciones semánticas. Sin embargo, el modelo está cerrado en cuanto al tipo de relaciones estructurales soportadas.

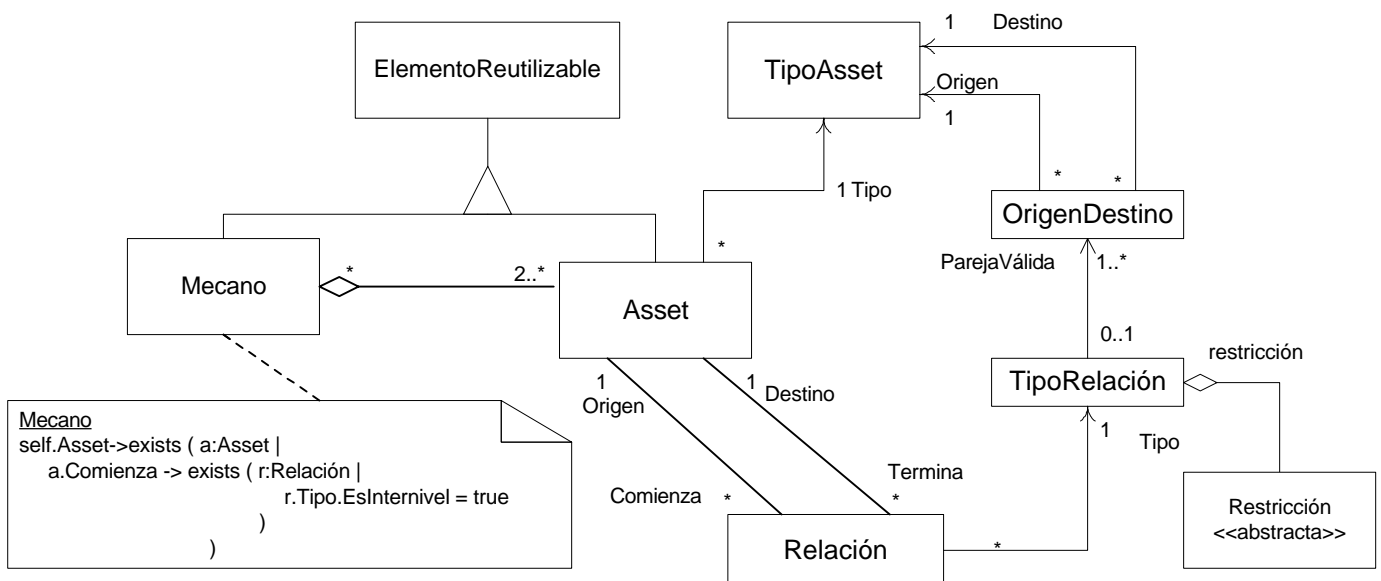


Figura 1. Modelo de Mecano.

2.2 Relaciones entre los assets componentes de un Mecano

Las relaciones entre assets tienen un papel protagonista dentro de un entorno de reutilización sistemática. Al hablar de relaciones entre assets se está haciendo referencia a las relaciones semánticas que se dan entre los objetos del universo de discurso en el que se plantea la definición de los Mecanos.

Cada uno de los enlaces que se mantiene entre dos assets debe pertenecer a un tipo de relación semántica que recoja las restricciones características de la familia de enlaces a la que da lugar.

Para capturar de una forma adecuada la semántica de los tipos de relaciones entre assets se ha decidido tomar como base de referencia un modelo objeto que aporte un conjunto reducido de relaciones estructurales, que sirvan para caracterizar los tipos de relaciones semánticas entre los assets.

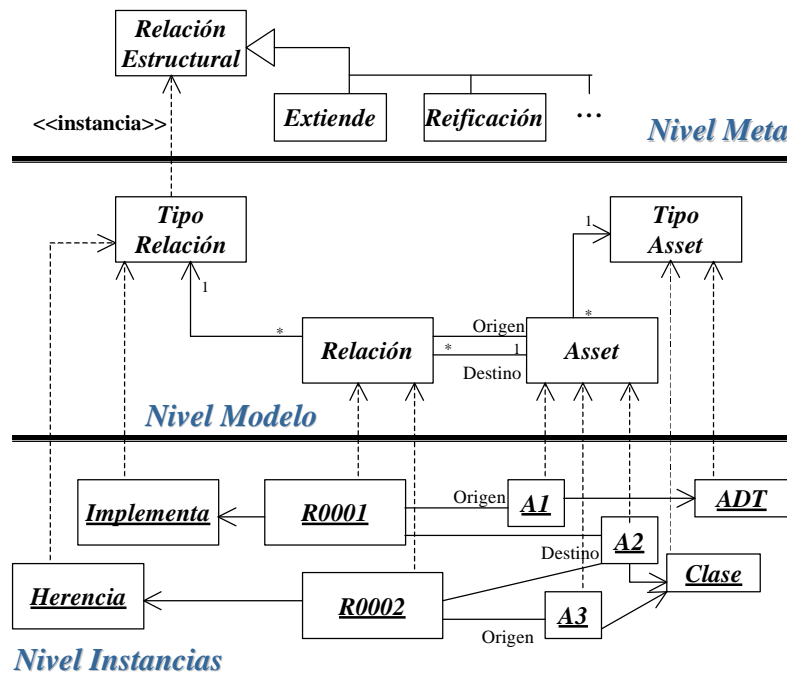


Figura 2. Niveles de abstracción en la definición de los Mecanos.

Las relaciones entre los assets componentes de un Mecano están definidas en tres niveles de abstracción:

- En el nivel de abstracción más bajo están los enlaces entre los assets. Cada uno de estos enlaces entre assets pertenece a un tipo de relación semántica.
- Tanto los enlaces como los tipos de relación semántica (*a los que pertenecen los enlaces*) son instancias de las clases que en el modelo de Mecano representan las relaciones (*clases **Relación** y **TipoRelación***), este nivel de modelo constituye el segundo nivel de abstracción en la definición de las relaciones entre assets.
- Pero a su vez cada uno de los tipos de relación semántica está soportado por una relación estructural definida en el modelo objeto, y que sirven para caracterizar los tipos de relaciones semánticas. Este es el tercer nivel de abstracción, correspondiéndose con un nivel meta.

La Figura 2 muestra la jerarquía de niveles de abstracción que se tiene en la definición de los Mecanos como estructuras complejas de reutilización. El nivel de modelo de esta figura se corresponde con una parte del modelo de Mecano ilustrado en la Figura 1.

Cada una de las relaciones estructurales a considerar debe concentrar en su definición unas connotaciones semánticas, que serán compartidas por todos los tipos de relaciones semánticas soportados por ellas, convirtiéndose así en la base constructiva y de soporte a la generación de los Mecanos.

Uno de los aspectos más relevantes de los Mecanos como estructura compleja de reutilización es su definición en diferentes niveles de abstracción simultáneamente, especialmente cuando se está haciendo referencia a los Mecanos que se almacenan en el repositorio como elementos software reutilizables complejos. Por tanto, al existir siempre en un Mecano assets clasificados en diferentes niveles de abstracción aparece la necesidad de diferenciar entre las relaciones que se dan entre assets clasificados en un mismo nivel de abstracción, relaciones intranivel, y las relaciones que enlazan assets clasificados en diferentes niveles de abstracción, relaciones internivel. Por consiguiente, es necesario que el modelo objeto subyacente aporte las relaciones estructurales adecuadas para dar soporte a ambos tipos de relaciones [6].

En la Tabla 1 se recogen las relaciones estructurales que deben estar presentes en el modelo subyacente a la estructura de Mecano.

Relaciones Estructurales	
Relaciones Internivel	Relaciones Intranivel
<ul style="list-style-type: none"> • Reificación 	<ul style="list-style-type: none"> • Agregación • Composición • Usa a • Extensión • Asociación

Tabla 1. Relaciones estructurales definidas en la estructura de un Mecano.

La relación de reificación es la relación estructural que soporta la semántica derivada del cambio de nivel de abstracción de un asset.

La semántica de inclusión se hace necesaria para representar la relación que existe entre aquellos assets de grano grueso que contienen un conjunto de assets de grano más fino, todos ellos clasificados en el mismo nivel de abstracción. Esta semántica se soporta con dos relaciones estructurales, la agregación y la composición. La primera de ellas presenta una contención débil, donde las partes componentes tienen un ciclo de vida independiente del asset agregado, pudiendo pertenecer a varios assets agregados a la vez. Por su parte la composición implica una pertenencia más fuerte de las partes al asset compuesto, no teniendo sentido la reutilización de las partes por separado, ni su pertenencia a otro asset compuesto.

La relación estructural “usa a” representa las relaciones cliente/servidor donde un asset especifica un comportamiento (*servicios*) que otro asset cliente utiliza sin que el primero tenga que tener un conocimiento expreso de ello.

La semántica de extensión caracteriza todas aquellas relaciones en las que un asset se define en función de la definición ofrecida por otro asset.

Las asociaciones se necesitan en el espacio de reutilización para expresar enlaces entre assets clasificados en el mismo nivel de abstracción, pero de manera que expresen los contenidos semánticos no soportados por los tipos de relaciones estructurales anteriores.

El estudio completo y la justificación de las relaciones dentro del modelo de Mecano se presenta en [6, 7].

3. Desarrollo de la biblioteca de reutilización: experiencias prácticas

Los trabajos teóricos presentados en la sección anterior han sido implementados y contrastados mediante el desarrollo de assets y Mecanos principalmente dentro del dominio del *Tratamiento Digital de Imágenes (TDI)*, aunque no de forma exclusiva, contando con EUROWARE⁴ [14] como soporte operativo a la reutilización. Sin embargo, para dar soporte a la estructura de Mecano por parte de EUROWARE, ha sido necesario la adaptación de dicho motor de reutilización. Esta versión adaptada del repositorio se puede consultar en la dirección <http://jupiter.dcs.fi.uva.es>.

3.1 Adaptación de EUROWARE

Para dar soporte al concepto de Mecano sobre EUROWARE se hace necesaria la definición en este entorno de un modelo de elemento software reutilizable que, salvando las limitaciones propias de

⁴ EUROWARE (Enabling Users to Reuse Over Wide AREAs) es el proyecto ESPRIT #8947, desarrollado dentro del proyecto SER ESPRIT #9809. EUROWARE es un conjunto de aplicaciones, construidas sobre la base de REBOOT [10], e integradas en un servidor WWW que permite que clientes remotos, conectados a una red TCP/IP, tengan acceso a los assets almacenados en el repositorio para su reutilización.

EUROWARE, permita introducir en este entorno la filosofía propia de la definición de los Mecanos, es decir, elementos reutilizables de grano grueso con soporte simultáneo de diferentes niveles de abstracción.

El modelo de elemento reutilizable que presenta EUROWARE no es válido para soportar directamente la estructura de Mecano, principalmente porque un Mecano se caracteriza por ser una agregación de assets clasificados en distintos niveles de abstracción, y esta idea no está soportada por el modelo de EUROWARE, como se puede deducir del conjunto de relaciones semánticas soportadas por el modelo (Figura 3).

El modelo de elemento reutilizable que se va a utilizar en EUROWARE para soportar el concepto de Mecano se muestra en la Figura 4. En él se utiliza una entidad denominada *Mecano* que se representa como una agregación de mecanos y/o de assets. Los assets van a poder estar relacionados por cinco relaciones intranivel (*agregación, composición, uso, extensión y asociación*) y una relación internivel (*reificación*), quedando de esta forma representadas las seis relaciones estructurales contempladas dentro del modelo de Mecano [6, 7].

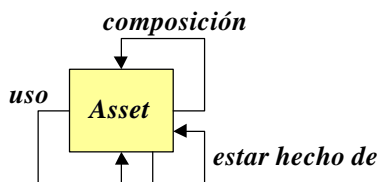


Figura 3. Modelo de elemento reutilizable de EUROWARE

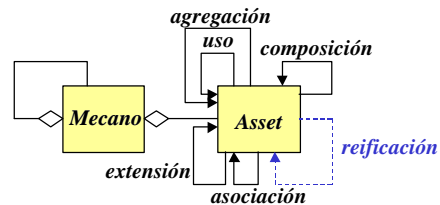


Figura 4. Modelo de elemento reutilizable adaptado sobre EUROWARE

La rigidez de la estructura interna de EUROWARE no lo convierten en el repositorio más adecuado para incorporar la estructura de Mecano. Para salvar los problemas derivados de su diseño interno, y poder así incorporar la estructura de Mecano en EUROWARE, se ha tenido que trabajar en los siguientes aspectos:

1. Ampliación del modelo del repositorio para incluir la estructura multinivel de Mecano, haciéndolo de forma independiente de la estructura de Asset que se mantiene.
2. Incorporación de las relaciones estructurales necesarias para la definición de Mecanos como elementos software reutilizables.
3. Ampliación de la funcionalidad necesaria para la manipulación de los Mecanos.

Para incorporar la estructura de Mecano, de la forma menos costosa en cuanto a esfuerzo, se ha optado por mantener el modelo interno de EUROWARE, añadiéndole una nueva faceta que permite distinguir entre assets y Mecanos. Estos elementos no se encuentran aislados dentro del repositorio sino enlazados mediante las dependencias entre los elementos reutilizables, las cuales se establecen mediante campos que representan a cada uno de los tipos de relaciones estructurales consideradas. Las funcionalidades propias del repositorio se mantienen de manera similar para los assets y los Mecanos en esta primera fase. La diferencia esencial entre assets compuestos y Mecanos dentro del repositorio modificado estriba en la agregación del tipo de elementos que entran a formar parte de los Mecanos, esto es, los Mecanos pueden estar compuestos por cualquier combinación de elementos de distinto nivel de abstracción (*con la única restricción de que estén representados al menos dos niveles de abstracción*), mientras que los assets compuestos se limitan a componentes clasificados en el mismo nivel de abstracción.

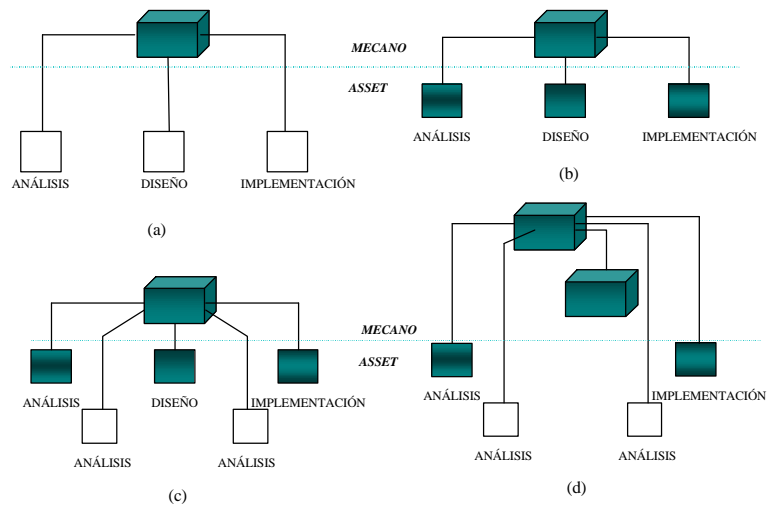


Figura 5. Alternativas de construcción de Mecanos en EUROWARE.

Sobre el nuevo esquema definido se tiene una estructura organizativa que permite la recuperación y el almacenamiento de cualquiera de los elementos que puede manejar el repositorio, es decir: a) *assets simples* interrelacionados entre sí por las relaciones de reificación, uso, asociación y extiende, b) *assets complejos* construidos a través de agregación de assets simples y c) *Mecanos* construidos por agregación de assets simples y/o complejos, con la restricción de que exista al menos una relación de reificación entre los elementos agregados.

Las modificaciones realizadas aportan un amplio grado de libertad para la construcción de Mecanos. Así, se tienen las posibilidades que se reflejan en la Figura 5.

La Figura 5(a) representa al Mecano formado por un conjunto de assets simples con la restricción de que al menos dos están clasificados en diferente nivel de abstracción y relacionados mediante una relación de reificación. La Figura 5(b) muestra un Mecano como agrupación de assets compuestos, agrupados en assets de análisis, diseño e implementación con la misma restricción anterior. La Figura 5(c) presenta una mezcla de las dos anteriores, y por último la Figura 5(d) refleja un Mecano como agrupación de assets simples, assets compuestos y otros Mecanos.

En las primeras experiencias llevadas a cabo en la población del repositorio con Mecanos se ha optado por la configuración exhibida por la Figura 5(b). El motivo principal de esta decisión es facilitar la introducción y extracción de assets compuestos y Mecanos, ya que los sistemas de consultas y de presentación de la información de EUROWARE no son demasiado adecuados.

3.3 Construcción de Mecanos en EUROWARE

El trabajo derivado de la población del repositorio, tanto con elementos de grano fino (*assets*) como con elementos de grano grueso (*Mecanos*) se está llevando a cabo mediante la realización de diversos proyectos de final de carrera, tomando como áreas experimentales los dominios de aplicación de *tratamiento digital de imágenes*, de *software para discapacitados* y de *gestión universitaria*.

Estas actividades han dado lugar a un *laboratorio de reutilización* adecuado para realizar experimentos de desarrollo de aplicaciones software con reutilización y de cualificación de assets⁵.

⁵ En el momento de escribir este trabajo el repositorio cuenta con 509 assets y 12 Mecanos accesibles públicamente. Estos números son sumamente cambiantes debido al carácter experimental del repositorio y a la cantidad de proyectos que actualmente se basan en el desarrollo para y con reutilización.

Para ilustrar el proceso de construcción de Mecanos dentro de EUROWARE se va a presentar un ejemplo en el que se establece un Mecano que representa un formato gráfico binario de una imagen, de forma que, considerando que se está trabajando bajo un paradigma objetual dirigido por responsabilidades, los componentes de este Mecano serán: *descripciones de requisitos funcionales* y *tarjetas CRC* clasificados en el nivel de abstracción de análisis, *diagramas de clases* en el nivel de diseño y *código* en el nivel de implementación.

Con el fin de facilitar las tareas de construcción y recuperación se ha adoptado una nomenclatura que permite distinguir los elementos reutilizables del repositorio de acuerdo al esquema: **Dominio-Nivel-Tipo-Nombre**.

El *dominio* hace referencia al campo de estudio donde se incluye el elemento reutilizable. Se representa mediante las iniciales de la frase que lo define. Interesa que el número de letras sea pequeño (*del orden de tres*). Si el elemento reutilizable no quiere asociarse a ningún dominio basta con dejar este campo en blanco. Dado que el ejemplo que se va a desarrollar cae dentro del dominio del *tratamiento digital de imágenes*, éste quedará representado por las letras **TDI**.

El *nivel* indica el nivel de abstracción en que se encuentra clasificado el elemento reutilizable, y puede ser **ANA** (*análisis*), **DIS** (*diseño*) o **IMP** (*implementación*).

El *tipo* expresa si el elemento reutilizable es un asset (**A**) o un Mecano (**M**).

Finalmente el *nombre* es un campo que se rellena libremente por el suministrador, indicando su contenido, de manera que sea descriptivo y no excesivamente grande.

Debido a las limitaciones intrínsecas de EUROWARE, el Mecano se va a ir construyendo por niveles de abstracción, generando un **asset complejo** por cada nivel de abstracción presente en el Mecano (*al menos deben existir dos por definición*) que aúne mediante agregación todos los assets del Mecano clasificados en dicho nivel de abstracción.

Así, en el caso que se desarrolla se crearán tres assets complejos, el primero de ellos viene referenciado por el identificador **TDI-ANA-A-formato** y representa el nivel de análisis o especificación del formato gráfico, el segundo es **TDI-DIS-A-formato** y representa el nivel de diseño, por último el asset complejo **TDI-IMP-A-formato** representa el nivel de implementación del formato gráfico. Tal y como se puede apreciar en la Figura 6, cada uno de estos assets complejos es una agregación⁶ de un conjunto de assets que están relacionados por las relaciones intranivel oportunas a cada caso.

Sin embargo, para que la estructura que se está formando pueda ser considerada como un Mecano, falta introducir las relaciones internivel. En este punto, y obligados por las limitaciones de EUROWARE, se establecen relaciones de reificación entre los assets complejos que representan cada nivel de abstracción en lugar de hacerlo entre sus componentes.

Una vez que se han establecido y relacionado los assets complejos, éstos se relacionan mediante agregación con un nuevo elemento reutilizable que representa el Mecano, y que viene representado por el identificador **TDI-M-formato**.

⁶ Con el fin de facilitar la asimilación del ejemplo al lector, en la Figura 6 se ha optado por representar la relación de agregación como contención de elementos, en lugar de utilizar una asociación con los adornos propios de algún lenguaje de modelado concreto.

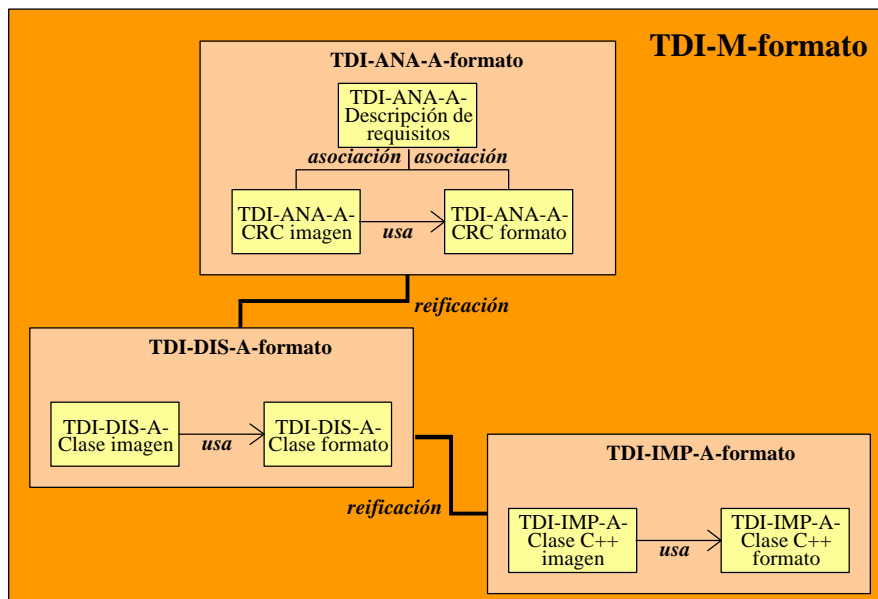


Figura 6. Mecano TDI-M-formato.

4. Líneas de trabajo abiertas

El modelo de reutilización que se está concibiendo presenta tres puntos de vista ortogonales: el modelo técnico de reutilización, el modelo de proceso de reutilización y el modelo de cualificación para la reutilización [8].

El modelo técnico de reutilización es el conjunto de conceptos y actividades que se utilizan para construir los elementos software reutilizables. Por tanto, va a hacer referencia a todas las actividades relacionadas con la creación de Mecanos, ya sea como función de un proceso de desarrollo para reutilización o como función de un proceso de desarrollo con reutilización. Además, como actividades relacionadas debe preocuparse por el almacenamiento, clasificación y recuperación de los Mecanos. El punto central de esta parte del modelo de reutilización es el propio modelo de Mecano.

El modelo de proceso de reutilización centra su atención en la organización lógica del personal, recursos, métodos y procedimientos, para definir el dominio de reutilización y las líneas de producto dentro del dominio.

Decir que un Mecano es un elemento reutilizable es quedarse en un nivel muy abstracto donde todo puede tener cabida. Esto es aceptable a la hora de definir la estructura del elemento reutilizable, pero para establecer el modelo de proceso es necesario darle un significado a los Mecanos. Así, un aspecto muy interesante es la utilización de Mecanos para el soporte de dominios y líneas de producto. Hacer que un número reducido de Mecanos modele un dominio completo daría lugar a elementos reutilizables demasiado gruesos, perdiéndose capacidad de reutilización. Por este motivo, una de las principales tareas dentro del modelo de proceso de reutilización será el estudio del dominio en el que una organización desea actuar, para encontrar sus subdominios o líneas de producto.

Una línea de producto sí puede ser representada por una serie de Mecanos. Inicialmente una línea de producto básica estará formada por su especificación y por su arquitectura [3] que desde un punto de vista de granularidad más fina no van a ser otra cosa que el conjunto de assets de la línea de producto con sus interrelaciones. Con un Mecano inicial que represente esta línea de producto básica, se puede empezar a generar los productos (*aplicaciones*) de la línea de producto, bien utilizando la arquitectura común

definida en la línea de producto y adaptándola con los requisitos específicos, o bien generando nuevos sistemas mediante la composición y posterior adaptación de los assets de la línea de productos. Con los nuevos productos se puede ir poblando la parte del repositorio donde se almacena la línea de producto, de forma que estos nuevos productos serán Mecanos que comparten componentes de los que inicialmente sirvieron para crear la línea de producto base, y todos ellos en conjunto forman la familia de productos que definen la línea.

Por último, el modelo de cualificación para la reutilización hace referencia a los criterios de evaluación y certificación de los Mecanos, así como a las métricas necesarias para medir tanto los productos como, en un futuro, los procesos.

5. Trabajos relacionados

En un principio puede pensarse que el trabajo realizado cae dentro del apartado del desarrollo basado en componentes (*CBD*). Sin embargo, el objetivo perseguido es la definición de una estructura de elemento reutilizable de grano grueso cuya aplicación práctica en el campo de la reutilización viene definida por el modelo de proceso de reutilización. Esto no significa que los componentes (*entendidos éstos como piezas reutilizables del nivel de implementación*) y los Mecanos sean antagónicos, ya que los componentes pueden formar parte de un Mecano, o bien un modelo de proceso de reutilización puede utilizar Mecanos para representar los componentes como un conjunto de assets que reflejan la evolución del componente desde su especificación a su implementación.

En la literatura especializada se pueden encontrar interesantes trabajos relacionados con la definición de elementos reutilizables, sin embargo la gran mayoría de ellos se centran en establecer elementos de granularidad fina clasificados en un solo nivel de abstracción. En este sentido cabe destacar el **BIDM** (*Basic Interoperability Data Model*) que es el estándar IEEE 1420.1 [9] para la catalogación de software en Internet, que salvando su pobreza en el tema de las relaciones semánticas, ha servido como referencia principal para la creación del modelo de componente de grano fino dentro del modelo de Mecano.

Otros trabajos más acordes con el tema que aquí se discute son los relacionados con la creación de estructuras complejas de reutilización, en especial el proyecto EEC-SPRIT II ITHACA [1] – el cual se centra en la definición de un elemento reutilizable complejo denominado *Application Frame* – y el modelo de biblioteca de software definido por CARDS (*Comprehensive Approach to Reusable Defense Software*) [13] – donde se defiende un paradigma de reutilización basado en la idea de que elementos software reutilizables de grano fino constituyen la entrada a la biblioteca, obteniéndose a la salida sistemas software.

6. Conclusiones y trabajo futuro

En este trabajo se han expuesto los avances realizados en el campo de la reutilización del software, y más concretamente en la construcción de elementos reutilizables complejos, de manera éstos elementos ofrezcan un soporte simultáneo de diferentes niveles de abstracción, a la vez que contribuyen a un incremento del nivel de abstracción en el proceso de reutilización. Todo ello con el fin último de conseguir integrar la reutilización en los entornos de ingeniería del software.

El trabajo realizado presenta dos vertientes. Por un lado está el desarrollo teórico de la estructura, que se refleja en el modelo de Mecano, el cual abre un número importante de líneas de trabajo que se están canalizando en la definición de un modelo de reutilización.

Por otra parte está el soporte práctico a la estructura creada. Como primera experiencia se ha optado por la adaptación de un motor de repositorios ya existente, EUROWARE, para el soporte de la estructura de Mecano. Este primer acercamiento ha aportado mucha experiencia al grupo, porque gracias a contar con un motor de repositorios ya desarrollado se ha podido trabajar con prototipos funcionales de un repositorio

donde realizar experimentos, primeramente con elementos de grano fino y posteriormente, tras la adaptación del mismo, con Mecanos. A este repositorio se puede acceder mediante la dirección <http://jupiter.dcs.fi.uva.es>.

La adaptación de EUROWARE aunque laboriosa ha sido fundamental para validar los trabajos teóricos con un menor esfuerzo que la construcción de un nuevo repositorio. No obstante, el trabajo realizado sobre EUROWARE ha sacado a la luz todas sus limitaciones intrínsecas, planteándose como una tarea próxima la creación de un nuevo repositorio que soporte completamente la estructura y filosofía de reutilización propias de los Mecanos, siendo fundamental para la obtención de los requisitos que éste debe cumplir el primer prototipo desechable realizado sobre EUROWARE.

7. Agradecimientos

Este documento ha sido generado en el seno del grupo **GIRO** (*Grupo de Investigación en Reutilización y Orientación al Objeto*) compuesto por miembros del Departamento de Informática de la Universidad de Valladolid y del Área de Lenguajes y Sistemas Informáticos de la Universidad de Burgos. Desde aquí queremos agradecer la inestimable colaboración y las correcciones sugeridas por el resto de los miembros de este grupo. Este trabajo ha sido parcialmente financiado por el proyecto **CICYT TIC97-0593-C05-05**.

8. Referencias

- [1] Ader, M., Nierstrasz, O., McMahon, S., Muller, G. and Pröfrock, A-K. "The ITHACA Technology: A Landscape for Object-Oriented Application Development". In the proceedings of ESPRIT'90 Conference. Kluwer Academic Publisher. November, 1990.
- [2] Biggerstaff, T. J. "An Assessment and Analysis of Software Reuse". Advances in Computers. Academic Press, Inc. Edited by Marshall C. Yovits. Vol. 34: 1-57. 1992.
- [3] Cohen, S., Friedman, S., Martin, L., Royer, M., Solderitsch, N. and Webster, R. "Concept of Operations for the ESC Product Line Approach". Technical Report CMU/SEI-96-TR-019 ESC-TR-96-018. Software Engineering Institute, Carnegie Mellon University. September, 1996.
- [4] García, F. J., Marqués, J. M. y Maudes, J. M. "Mecano: Una Propuesta de Componente Software Reutilizable". En las actas de las II Jornadas de Ingeniería del Software (Donostia-San Sebastián, Spain, 3-5 septiembre de 1997): 232-244. 1997.
- [5] García Peñalvo, F. J., Marqués Corral, J. M., Laguna, M. Á. y Maudes Raedo, J. M. "Estructuras Complejas de Reutilización: Definición de Mecano Estático". En las actas de las II Jornadas de Trabajo MENHIR. Editor José A. Carsí (Valencia, 19-20 de Febrero de 1998): 135-141. 1998.
- [6] García Peñalvo, F. J., Marqués Corral, J. M., Laguna, M. Á. y Maudes Raedo, J. M. "Influencia de las Relaciones entre Elementos Software Reutilizables en la Generación de Mecanos". En las actas de las III Jornadas de Ingeniería del software. Editores Ambrosio Toval y Joaquín Nicolás Ros (Murcia, 11-13 de Noviembre de 1998):155-166. 1998.
- [7] García Peñalvo, F. J., Marqués Corral, J. M., Laguna, M. Á. y Maudes Raedo, J. M. "Mecanos: Soporte de Diferentes Niveles de Abstracción en los Elementos Software Reutilizables". Technical Report (TR-GIRO-02-98V1.2.1), Universidad de Valladolid (España). Octubre 1998.
- [8] García, F. J., Marqués, J. M. and Maudes, J. M. "Mecanos as Basis of a Compositional/Generative Mixed Reuse Model". In the proceedings of the second edition of the European Reuse Workshop (Madrid, 4-6 November, 1998). Vol.2: 17-20. 1998.
- [9] IEEE. "IEEE Standard for Information Technology – Software Reuse – Data Model for Reuse Library Interoperability Data Model (BIDM)". IEEE Std 1420.1, 1995.
- [10] Karlsson, Even-André. "Software Reuse. A Holistic Approach". John Wiley & Sons Ltd. 1995.
- [11] Krueger, C. W. "Software Reuse". ACM Computing Surveys. Vol. 24. Nº 2: 131-183. 1992.
- [12] Mili, H., Mili, F. and Mili, A.. "Reusing Software: Issues and Research Directions". IEEE Transactions on Software Engineering. Vol. 21. Nº 6 (June): 528-562. 1995.
- [13] Petracca, A. and Bock, T. "Command Center Library Model Document Release 4.0 Comprehensive Approach to Reusable Defense Software (CARDS)". T.R. STARS-VC-B015/002/00. 31 May 1994.
- [14] SER Consortium. "Solutions for Software Evolution and Reuse". SER Esprit Project 9809. January, 1996.