

Evaluación de propuestas para la gestión de trazabilidad en el contexto de la Ingeniería Dirigida por Modelos

Iván Santiago Viñambres, Valeria de Castro, Juan Manuel Vara, Esperanza Marcos,

Grupo de Investigación Kybele, Departamento de Lenguajes y Sistemas Informáticos II,
Universidad Rey Juan Carlos, Avda. Tulipan S/N,
28933 Móstoles (Madrid), España
{ivan.santiago, valeria.decastro, juanmanuel.vara, esperanza.marcos}@urjc.es

Resumen. El mantenimiento y gestión de la trazabilidad ha sido siempre un reto en el contexto de la Ingeniería del Software. La llegada de la Ingeniería Dirigida por Modelos y el papel clave que los modelos pasan a jugar en cualquier actividad relacionada con el desarrollo software proporciona un nuevo escenario para introducir mejoras en la gestión de la trazabilidad y sus aplicaciones. En este contexto, el trabajo que se presenta resume los principales resultados de una revisión sistemática sobre la gestión de la trazabilidad, prestando especial interés a las propuestas relacionadas con la Ingeniería Dirigida por Modelos. Dicha revisión pretende aunar criterios en torno a las operaciones básicas para gestión de la trazabilidad y sus aplicaciones con dos objetivos; por un lado, obtener una visión global de las propuestas para la gestión de trazabilidad, y por otro lado, detectar posibles carencias.

Palabras clave: Mantenimiento, Gestión de Trazabilidad, Ingeniería Dirigida por Modelos.

1 Introducción

La trazabilidad ha sido siempre un tema relevante en Ingeniería del Software. El mantenimiento de los enlaces desde los requisitos hasta los artefactos de diseño, código y casos de prueba, ha resultado de interés para los investigadores como una forma de llevar a cabo el análisis de impacto, las pruebas de regresión, la validación de requisitos, etc. [6].

Con la llegada de la Ingeniería Dirigida por Modelos (MDE, *Model Driven Engineering*) [18] y la Arquitectura Dirigida por Modelos (MDA, *Model Driven Architecture*) [22] cuyo principio fundamental es trasladar el foco de atención de las actividades de codificación a las actividades de modelado; la gestión de la trazabilidad puede adquirir mayor relevancia. El papel clave de los modelos en el proceso de desarrollo facilita el mantenimiento de las trazas ya que estas pueden ser enlaces entre los elementos de los modelos. Dichos enlaces, a su vez, podrían ser recogidos en otros modelos y por tanto, procesados mediante transformaciones,

generadores de código, etc. De hecho, estas trazas pueden ser generadas automáticamente si los modelos se conectan mediante una transformación de modelos y el lenguaje utilizado para desarrollar la transformación ofrece soporte para mantener la información de traza [23]. De esta forma, si algún elemento del modelo origen es modificado, podría llegar a replicarse la modificación sobre los elementos correspondientes del modelo destino.

Ante las facilidades que proporciona MDE para la gestión de la trazabilidad, se plantean las siguientes cuestiones de investigación: ¿para qué se usa la información de trazabilidad? ¿Qué operaciones de gestión se llevan a cabo cuando trabajamos con información de trazabilidad? ¿Qué características presentan las aproximaciones basadas en MDE para soportar la trazabilidad? Para dar respuesta a estas cuestiones se ha llevado a cabo una revisión sistemática. Una revisión sistemática consiste en una revisión metodológica y rigurosa de resultados de investigaciones. Su realización no sólo aporta evidencias sobre una cuestión de investigación, sino que también pretende apoyar el desarrollo de directrices basadas en evidencias para los profesionales [12]. Este trabajo se centrará en mostrar los principales resultados y conclusiones obtenidas mediante la revisión sistemática, aunque por motivos de espacio, no se incluye aquí toda la información obtenida ni se detalla el proceso desarrollado.

En relación con las preguntas anteriormente planteadas, el trabajo se estructura en tres grandes bloques: (1) aplicación de la información de trazabilidad en la ingeniería del software. Esta evaluación permitirá dar respuesta al interrogante sobre qué uso tiene la información de trazabilidad. (2) Operaciones fundamentales para la gestión de información de trazabilidad, que darán respuesta a la segunda cuestión planteada. Por último (3) se evalúan propuestas basadas en MDE para identificar características comunes o diferenciadores entre ellas. Conviene mencionar que debido a la amplitud de algunos trabajos, estos han sido incluidos en más de un bloque de la revisión, ya que por ejemplo, tratan el uso de la información de trazabilidad y también soportan operaciones para la gestión de la trazabilidad.

El resto del artículo se estructura de la siguiente forma. En la sección 2 se presenta el primer bloque de la revisión, centrado en identificar posibles aplicaciones para la información de trazabilidad. En esta sección, en primer lugar se introducen los criterios seguidos, a continuación se resumen las principales aportaciones de los trabajos seleccionados y finalmente se resumen las principales conclusiones. Esta estructura es seguida por las siguientes secciones 3 y 4 que presentan los otros dos grandes bloques de la revisión: las operaciones para la gestión de la trazabilidad y las propuestas basadas en MDE para trazabilidad. La sección 5 presenta los trabajos relacionados y, por último, la sección 6 presenta una discusión sobre los resultados obtenidos y las conclusiones obtenidas a tenor de las evaluaciones realizadas.

2 Uso de la información de trazabilidad

Esta sección intenta dar respuesta al interrogante sobre el uso de la información de trazabilidad. Consideramos de gran importancia conocer el para qué es interesante poseer información de trazabilidad y qué aplicaciones se puede llevar a cabo a partir de ella.

2.1 Criterios para la evaluación

En primer lugar, conviene mencionar que todas las aplicaciones de la información de trazabilidad analizadas, están fuertemente ligadas entre sí. Por ejemplo, la posibilidad de realizar ingeniería inversa no sólo favorece al mantenimiento del software, sino también a la sincronización de modelos, a la propia gestión del proyecto, ya que permite elevar el nivel de abstracción del software y facilitar su entendimiento. Por tanto, para facilitar la interpretación de los datos, se han estructurado las distintas aplicaciones en cinco grupos, que serán los criterios objeto de evaluación.

- El grupo de aplicaciones de análisis de *impacto del cambio* permiten determinar el efecto que tendrá el cambio sobre los artefactos en todo el sistema a través del ciclo de vida de desarrollo del software [10]. Además, se han incluido otras aplicaciones relativas al análisis de cobertura definido como “el grado en el que algunos objetos del sistema son seguidos por otros artefactos en el sistema” y al análisis de huérfanos, utilizado para buscar artefactos que no han sido originados por ningún enlace o traza de un tipo específico [14].
- El grupo de mejora de la *entendibilidad* incide en que una de las principales ventajas que proporciona la información de trazabilidad es que hace más comprensible el software. El poder determinar qué elementos dan lugar a qué otros, permite elevar el nivel de abstracción al que se diseña y concibe el software, reduciendo la complejidad del propio software y facilitando el entendimiento. Dos aplicaciones concretas son disponer de un soporte para decisiones de diseño y disponer de un mecanismo para la comprensión y depuración de transformaciones [27].
- El grupo de *gestión de proyectos* incluye aplicaciones de la trazabilidad relativas a la gestión de proyectos, incluyendo actividades como el mantenimiento del software, la reutilización, utilización en procesos de ingeniería inversa y en consecuencia la calidad del software.
- El grupo de *gestión de la configuración* incluye aplicaciones relativas a la gestión de la configuración y a la planificación de versiones.
- Finalmente, el grupo de sincronización de *modelos* incluye aplicaciones relativas al uso de la trazabilidad como entrada para modelos de transformación y sincronización de modelos.

2.2 Trabajos evaluados

A continuación se presentan seis propuestas estudiadas que consideran posibles aplicaciones para la información de trazabilidad.

En [24] Valderas y Pelechano presentan una aproximación que introduce capacidades de trazabilidad de requisitos en el contexto de MDE para aplicaciones Web, mediante transformaciones entre modelos, y desde el modelo de requisitos hasta el modelo de navegación. Proporcionan un método para la obtención de conformidad de la transformación. Hacen uso de herramientas para la generación de informes de trazabilidad a partir del modelo de navegación y para la aplicación automática de reglas de transformación gráficas de modelos de requisitos basados en tareas. Aportan también

un metamodelo para enlaces de trazabilidad. Respecto al uso de la información de trazabilidad se identifican aplicaciones relativas al análisis del impacto del cambio.

Dee [7] propone un *framework* que permite analizar el impacto del cambio de los requisitos. Concretamente permite realizar un seguimiento de los requisitos desde su concepción hasta los elementos que los implementan. Se basa en transformaciones de modelos y propone un metamodelo de traza personalizado. Al igual que Valderas y Pelechano, solo hacen uso de la información de trazabilidad para tareas relacionadas con el análisis del impacto del cambio.

Melby [14] presenta una herramienta genérica que permite definir tipos de trazabilidad y capturar información de trazabilidad de estos tipos. El objetivo final es definir un esquema de clasificación para la trazabilidad en MDE. Considera tres aspectos relativos a la información de trazabilidad; en relación a productos, en relación a procesos y en relación a enlaces de trazabilidad con semántica similar. En este trabajo se identifican aplicaciones relativas al análisis del impacto del cambio, actividades relativas a la gestión de la configuración y a la utilización de la información de trazabilidad como entrada para modelos de transformación.

Winkler y Von Pilgrim [27] estudian dos enfoques. Por un lado evalúan algunos criterios relativos a principios básicos de trazabilidad (que se estudiarán en el siguiente apartado) y por otra parte se evalúan tres criterios referentes al trabajo con trazas tales como actividades, uso y visualización. Respecto al uso de las trazas, en concreto a la identificación de escenarios específicos, se distinguen ocho casos posibles; soporte a decisiones de diseño, pruebas de adecuación/validación, entendimiento y mantenimiento de artefactos, entendimiento y depuración de transformaciones, derivación de visualizaciones utilizables, análisis del impacto del cambio, sincronización de modelos e impulso al desarrollo de líneas de producto.

Anquetil et. al. [5] presentan un *framework* para trazabilidad a través de diversas actividades relacionadas con el desarrollo de líneas de productos software bajo el proyecto europeo AMPLE. Se propone un metamodelo de trazabilidad y destaca la aplicación de información de trazabilidad en actividades relacionadas con la gestión de la configuración y la planificación de versiones.

Schwarz [19] presenta una aproximación genérica para la definición de información de trazabilidad. Considera como elementos trazables las entidades de negocio y el código; tanto su estructura como su semántica. En este trabajo se identifican actividades relativas a la de gestión de proyectos, mantenimiento, reutilización, ingeniería inversa, calidad del software, gestión de la configuración, entendibilidad y gestión del cambio.

2.3 Conclusiones de la evaluación

La Tabla 1 resume las conclusiones obtenidas sobre el uso de la información de trazabilidad. El eje horizontal representa los trabajos evaluados y el eje vertical cada uno de los grupos de actividades. Se han marcado aquellos trabajos que incluyen o tratan las actividades enmarcadas en cada grupo.

Tabla 1. Aplicación de la información de trazabilidad en distintas propuestas.

	[24]	[7]	[14]	[27]	[5]	[19]
Impacto del Cambio	✓	✓	✓	✓		✓
Entendibilidad				✓		✓
Gestión de Proyectos				✓		✓
Gestión de la Configuración			✓		✓	✓
Modelos			✓	✓		

A la vista de los resultados que muestra la Tabla 1, se puede concluir que el grupo de aplicaciones de análisis del impacto del cambio, es el más abordado por las distintas propuestas. El uso de la información de trazabilidad para llevar a cabo el análisis del impacto del cambio, el análisis de cobertura y el análisis de huérfanos es donde la comunidad investigadora ha intentado proponer más soluciones, ya que la mayor parte de los trabajos cubren estas aplicaciones. Por el contrario, el uso de información de trazabilidad en actividades relacionadas con la entendibilidad del software y la sincronización de modelos, son las menos afrontadas. Esto puede ser derivado de la inexistencia de herramientas que proporcionen un análisis adecuado a partir de la información de trazabilidad identificada y almacenada.

Así mismo, el uso de la información de trazabilidad en actividades relacionadas con la gestión de la configuración y la planificación de versiones son afrontadas por la mitad de los trabajos seleccionados en este bloque. Una de las posibles causas de esta situación sea la carencia de herramientas para el control de versiones de modelos.

Uno de los objetivos perseguidos mediante el mantenimiento de la información de trazabilidad es la entendibilidad del software, proporcionando un método eficaz para la depuración de transformaciones. Parece evidente que transformaciones bien depuradas mejorarían la calidad de las transformaciones, y facilitarían la gestión de proyectos. La sincronización de los modelos y el uso de la trazabilidad como entrada para modelos de transformación, también reforzarían el resto de aplicaciones incluidas en la gestión de proyectos, como la ingeniería inversa o la reutilización.

3 Gestión de la trazabilidad

Esta sección intenta dar respuesta al interrogante sobre qué operaciones para la gestión de la trazabilidad son usadas. Para ello, se estudia qué operaciones básicas soporta cada propuesta.

3.1 Criterios para la evaluación

En primer lugar conviene mencionar que las operaciones incluidas como criterios para la evaluación, han sido elegidas por ser las más utilizadas en la literatura. A continuación son definidas.

- *Definición*: clasificación de los tipos de entidad que serán trazados y el tipo de relaciones de trazabilidad existente entre ellos. En la literatura se diferencian dos tendencias; diseño de metamodelos para trazabilidad y relaciones de trazabilidad como restricciones de integridad (por ejemplo, formalización matemática, lógica, evento-condición, acción...).
- *Identificación*: es la operación que permite describir relaciones de trazabilidad previamente desconocidas entre entidades, ya sea de forma manual, automática o semi-automática.
- *Almacenamiento*: es considerada como una operación fundamental en la gestión de la trazabilidad. En algunas propuestas se define como la posesión material de relaciones de trazabilidad en forma de estructuras de datos. A través de las distintas propuestas se distinguen dos posibilidades; trazabilidad interna, donde la información de trazabilidad se registra dentro de las entidades trazadas mediante la introducción de referencias textuales o hipervínculos y trazabilidad externa, que requiere de esquemas de base de datos o metamodelos específicos.
- *Importación/Exportación*: determina las propuestas capaces de importar y/o exportar la información de trazabilidad o enlaces de traza en un formato adecuado para su intercambio entre diferentes aplicaciones. Es decir, analiza la interoperabilidad entre distintas propuestas.
- *Consulta*: suele estar condicionada a que la información de trazabilidad haya sido previamente identificada y almacenada. Son varias las herramientas que permiten consultar la información de trazabilidad en; bases de datos relacionales mediante el lenguaje SQL, repositorios basados en gráficos, lenguajes de consulta de ontologías, expresiones regulares y/o herramientas textuales.
- *Visualización*: representación de las relaciones de trazabilidad existentes entre los diferentes artefactos. Existen diversas estructuras que son usadas para tal fin como matrices de trazabilidad, matrices de referencias cruzadas, gráficos de dependencias.
- *Modificación*: de las relaciones existentes de trazabilidad. Al igual que la operación de identificación, puede llevarse a cabo de forma manual, automática o semi-automática.

3.2 Trabajos evaluados

A continuación se presentan las propuestas incluidas en este bloque. Se han incluido los trabajos que, al menos, han considerado una de las operaciones para la gestión de la información de trazabilidad.

En [4] Almeida et al. proponen un *framework* que trata de simplificar el mantenimiento de las relación entre requisitos y modelos. Incorporan una notación de conformidad entre los modelos propuestos. Para la visualización de la información de trazabilidad se utilizan tablas cruzadas de requisitos por modelos.

Paige et al. [16] hacen énfasis en el proceso de construcción de clasificaciones de trazabilidad para operaciones de ingeniería dirigida por modelos. Considera operaciones tanto manuales como automáticas. Las actividades identificadas para llevar a cabo el proceso de construcción de clasificaciones se divide en tres fases; fase de elicitación

ción, donde se realiza una identificación básica de enlaces de traza y relaciones; fase de análisis de restricciones en relaciones y generalización de relaciones y enlaces de trazas y, por último fase de construcción de la clasificación. Las operaciones de gestión de trazabilidad identificadas son la definición, la identificación y la consulta.

La propuesta de Melby [14] ya fue presentada en el apartado anterior. Para la evaluación en base a operaciones de gestión de información de trazabilidad, únicamente añadiremos que considera las operaciones de identificación y consulta.

Al igual que en el trabajo anterior, la aportación de Winkler y von Pilgrim [27] ya ha sido presentada. Bajo esta sección cabe destacar la caracterización de operaciones de definición, almacenamiento, visualización y consulta.

Seibel et al. [21] presentan un enfoque de trazabilidad completo que combina enfoques de trazabilidad clásicos para MDE y modelos globales en forma de megamodelos dinámicos-jerárquicos integrando el mantenimiento eficiente de la información de trazabilidad. Propone un metamodelo de trazabilidad y un enfoque de modelado basado en Eclipse. Llevan a cabo operaciones para gestión de información de trazabilidad como definición, identificación, almacenamiento, consulta, visualización y modificación.

La propuesta de Anquetil et al. [5] ya se presentó en la sección anterior. Destacaremos en este apartado que considera operaciones de definición, identificación, almacenamiento, consulta, visualización y modificación.

Aleksy et al. [3] presentan un diseño conceptual y la implementación de un prototipo con capacidades de trazabilidad en el proyecto OMEGA. La propuesta da soporte al trazado de requisitos a través de diferentes modelos y niveles de abstracción en transformaciones entre modelos y de modelos a texto. Utilizan el lenguaje XML para el almacenamiento de modelos y Velocity para las transformaciones de modelo a texto. Definen dos precondiciones básicas para un mantenimiento efectivo y eficiente de la trazabilidad; por un lado recuperación automática, validación y modificación de enlaces de trazabilidad, y por otra parte herramientas de soporte. Para la identificación utilizan la herramienta TRACES y para la visualización de información de trazabilidad utilizan TraVis.

En [26], Walderhaug et al. proponen una solución genérica para trazabilidad en MDE que; (1) proporcione una definición flexible de modelos de traza que permita la personalización de artefactos y relaciones, (2) ofrezca soporte para todo el ciclo de vida, (3) soporte el trazado de artefactos a través de diferentes herramientas. Se basa en el metamodelo propuesto por Ramesh y Jarke [17] que consiste en un conjunto de metamodelos para ser personalizados y una base de datos de trazas actuales. Se identifican operaciones para la gestión de la trazabilidad como la definición, almacenamiento y modificación de relaciones.

El trabajo de Schwarz [19] ya ha sido presentado en el apartado anterior. En referencia al soporte de operaciones de gestión de la información de la trazabilidad considera la definición, identificación, almacenamiento y consulta.

En [2] Aizenbud-Reshef et al. definen una semántica operacional para la trazabilidad en UML, que permita capturar diferentes tipos de trazabilidad. Se utiliza una notación común para todos los casos. La representación se lleva a cabo mediante reglas, condiciones y acciones. Definen dos tipos de semántica, la semántica preventiva donde se realiza una descripción de las cosas que no deberían suceder y la semántica reactiva, donde se describe lo que debería dispararse cuando sucede algo en

uno o varios elementos relacionados. Respecto a la información de los enlaces de trazabilidad, también se distinguen dos categorías; qué información tiene que ser gestionada entre elementos relacionados y qué acciones deben ser disparadas cuando hay cambios que afectan a elementos relacionados.

En [8] Drivalos et al. proponen un enfoque riguroso para definir enlaces de trazabilidad enriquecidos semánticamente entre modelos expresados en diferentes lenguajes de modelado usando metamodelos de trazabilidad específicos.

Finalmente, Schwarz et al. [20] presentan un visión global sobre la trazabilidad en el proceso de desarrollo software. Se estructura de acuerdo a seis operaciones específicas; definición, registro, identificación, mantenimiento, recuperación y utilización de la información de trazabilidad anteriormente obtenida en escenarios de aplicación concretos.

3.3 Conclusiones de la evaluación

La Tabla 2 resume las conclusiones obtenidas sobre las operaciones para la gestión de la trazabilidad. Las propuestas están representadas en el eje horizontal, respecto a las operaciones soportadas, representadas en el eje vertical.

Tabla 2. Operaciones de gestión de información de trazabilidad.

	[4]	[16]	[14]	[27]	[21]	[5]	[3]	[26]	[19]	[1]	[8]	[20]
Definición		✓		✓	✓	✓		✓	✓	✓	✓	✓
Identificación		✓	✓		✓		✓		✓	✓		✓
Almacenamiento				✓	✓			✓	✓			✓
Import/Export						✓						
Consulta		✓			✓				✓			✓
Visualización	✓		✓	✓	✓	✓	✓				✓	
Modificación				✓	✓			✓				✓

A la vista de los resultados, podemos concluir que ninguna propuesta aborda todas las operaciones para la gestión de información de trazabilidad. La operación menos abordada por las distintas propuestas es la importación y/o exportación de información de trazabilidad. En cambio, las operaciones de definición, identificación y visualización son operaciones más maduras, abordadas por prácticamente todas las propuestas.

Las operaciones de almacenamiento, consulta y modificación solo son soportadas por algunas propuestas. Estos trabajos parecen estar más orientados a la adquisición de información de trazabilidad, para posteriormente poder modificar enlaces de traza y realizar consultas sobre un soporte de almacenamiento. Destaca el hecho de que todas las propuestas que ofrecen soporte al almacenamiento de la información, también ofrecen la posibilidad de modificación. De igual forma, la consulta de información de trazabilidad, está ligada al almacenamiento previo.

No existe, a la vista de los resultados, una dependencia clara entre la operación de identificación y la de almacenamiento. Esto puede estar originado porque algunos trabajos se centran en identificar información de trazabilidad implícita y las propuestas que apuestan por el soporte del almacenamiento, se centran en el establecimiento manual de relaciones entre los distintos artefactos.

Por último, la carencia de un metamodelo estándar para facilitar el intercambio de información de trazabilidad, puede ser el motivo principal por el que la operación de importación y/o exportación sea la menos abordada por las distintas propuestas.

4 Propuestas basadas en MDE para la gestión de trazabilidad

Esta sección intenta responder a la cuestión de qué características presentan las aproximaciones basadas en MDE para soportar la trazabilidad. El objetivo es identificar características comunes o diferenciadores entre ellas.

4.1 Criterios para la evaluación

En primer lugar, aclararemos que se han seleccionado aquellas características que aparecen en todos los trabajos incluidos en este grupo. Las características que se han estudiado en las propuestas son:

- El criterio de *Aproximación* nos permite distinguir la naturaleza del trabajo evaluado. Para este criterio se establecen dos posibles valores; aproximaciones basadas en requisitos y aproximaciones basadas en transformaciones.
- El criterio de *Transformación* nos permite identificar el tipo de transformación que se lleva a cabo en la propuesta. Existen dos posibles valores para este criterio; transformaciones entre modelos (M2M) y/o transformaciones de modelo a texto (M2T).
- La *Dimensión* de trazabilidad evalúa si la propuesta soporta trazabilidad de modelos y/o elementos del modelo a diferentes niveles de abstracción. Consideramos la dimensión horizontal para propuestas que soportan trazabilidad entre modelos (y/o elementos del modelo) del mismo nivel de abstracción y dimensión vertical a las propuestas que soportan trazabilidad en distintos niveles de abstracción.
- El criterio de *Nivel* indica si la trazabilidad se lleva a cabo entre modelos (alto nivel) o entre elementos del modelo (bajo nivel).
- El criterio de *Lenguajes* enumera los distintos lenguajes de transformación y/o generación de código que utilizan las propuestas.
- La *Representación* define las principales estructuras que son usadas para representar la información de trazabilidad.
- El criterio de *Metamodelo* indica el propósito del metamodelo presentado en cada trabajo analizado. Hemos establecido dos posibles valores para el criterio de metamodelo; genérico o específico para el dominio del problema.

4.2 Trabajos evaluados

A continuación se presentan las propuestas incluidas en este bloque. Han sido seleccionadas las ocho propuestas para MDE que se han mostrado más maduras en cuanto a la inclusión de algún tipo de soporte para gestión de la trazabilidad.

El trabajo de Valderas y Pelechano [24] ya se presentó en el primer bloque. Esta aproximación se centra en la trazabilidad de requisitos para aplicaciones Web a través de transformaciones entre modelos. Se identifica una dimensión de trazabilidad vertical y para la representación de la información de trazabilidad utiliza dos herramientas; AGG Tool (*Attributed Graph Grammars Tool*) para la realización de gráficos y TaskTracer para la generación de informes de trazabilidad.

En [7] Dee centra su propuesta en la trazabilidad de requisitos y en transformaciones entre modelos. Utiliza los lenguajes OCL (*Object Constraint Language*) y QVT (*Query/View/Transformation*) para la dimensión de trazabilidad en vertical. Propone un metamodelo específico y para representación de la información utiliza gráficos de dependencias y tablas cruzadas.

Aleksy et al. [3] realizan una contribución basada en requisitos, a través de transformaciones de modelos y transformaciones de modelo a texto, cubriendo las dimensiones horizontal y vertical. Utiliza XML para el almacenamiento de modelos y Velocity para llevar a cabo las transformaciones de modelo a texto.

El resto de propuestas son aproximaciones basadas en transformaciones. Jouault [11] presenta una propuesta para añadir trazabilidad a programas escritos en ATL, mediante el uso de transformaciones de modelos y apoyándose en modelos de *weaving* AMW (*Atlas Model Weaver*). Presenta un metamodelo de trazas básico cubriendo la dimensión de trazabilidad horizontal y vertical.

Falleri et al. [9] proponen un *framework* para trazabilidad basado en el lenguaje *kermeta*. Parten de un metamodelo básico de trazas fundamentado en [11]. Consideran la dimensión vertical y horizontal de trazabilidad. Para la representación utiliza modelos *kermeta* y la especificación XMI.

Von Pilgrim et al. [25] discuten cómo una cadena de transformaciones puede ser modelada y ejecutada en un lenguaje de transformación de modo independiente. Estas transformaciones de modelos se desarrollan bajo un *plugin* de Eclipse llamado UniTL (*Unified Transformatin Infrastructure*) utilizando lenguajes como Java, ATL o MTF (*Model Transformation Framework*) de IBM. Proponen dos extensiones a un metamodelo básico de traza; *UniTI Traceability Metamodel* y *MTM (Merged Traceability Metamodel)*. Para la representación proponen GEF3D (*Eclipse Graphical Editing Framework*).

Olsen y Oldevik [15] discuten sobre el uso de escenarios para trazabilidad, presentando su propuesta de implementación. Describen cómo la trazabilidad y los enlaces de traza pueden ser usados para soportar el desarrollo de sistemas. Utilizan transformaciones de modelo a texto mediante el lenguaje MofScript. Consideran la dimensión de trazabilidad horizontal y vertical.

Por último, en [13] Kolovos et al. proponen la unificación de dos enfoques de almacenamiento de información de trazabilidad a través de un modelo de fusión. El primer enfoque considera el almacenamiento de forma interna al modelo, es decir, dentro del propio modelo; mientras que el segundo enfoque propone almacenar la información en modelos externos. Se sugiere que la información de trazabilidad debe

mantenerse en modelos por separado y que pueda ser combinada con el primer modelo bajo demanda. El objetivo es producir modelos anotados para fines de inspección. Por otra parte, el trabajo se centra en transformaciones de modelos mediante el lenguaje de transformación EML (*Epsilon Merging Language*) incluyendo un metamodelo específico de trazas EML. Para la representación utilizan metamodelos EML y modelos de traza UML.

4.3 Conclusiones de la evaluación

La Tabla 3 resume las conclusiones sobre las distintas características estudiadas a lo largo de las ocho propuestas analizadas. El eje horizontal representa los criterios estudiados y en el eje vertical las propuestas.

Tabla 3. Características de propuestas para trazabilidad en Ingeniería Dirigida por modelos.

	Aproximación	Transformación	Dimensión	Nivel	Lenguaje	Representación	Metamodelo
[24]	Requisitos	M2M	Vertical	Alto	XML	Gráficos AGG Tool Informes TaskTracer	Específico
[7]	Requisitos	M2M	Vertical	Alto	QVT OCL	Gráficos de dependencias Tablas cruzadas	Específico
[3]	Requisitos	M2M M2T	Horizontal Vertical	Alto	XML Velocity	Visualizaciones con TraVis	Específico
[11]	Transformación	M2M	Horizontal Vertical	Alto	ATL	Modelo de traza ATL Enlaces de traza AMW	Genérico
[9]	Transformación	M2M	Horizontal Vertical	Bajo	Kermeta	Modelos Kermeta XMI	Genérico
[25]	Transformación	M2M	Horizontal	Alto	Java ATL MTF	Diagramas GEF3D	Específico
[15]	Transformación	M2T	Horizontal Vertical	Alto	MofScript	TAP UML (modelo de traza)	Específico
[13]	Transformación	M2M	Horizontal	Bajo	EML	Metamodelos EML UML (modelo de traza)	Específico

Las primeras conclusiones que se obtienen, a la vista de los resultados obtenidos, es que, al parecer, en las aproximaciones basadas en requisitos predomina la dimensión vertical, mientras que en las aproximaciones que basadas en transformaciones, consideran ambas dimensiones: horizontal y vertical. También llama la atención el predominio de trazabilidad de alto nivel (trazabilidad entre modelos) con respecto a la trazabilidad de bajo nivel (entre elementos del modelo) que únicamente es considerada por las propuestas [9] y [13]. El tipo de transformación más predominante en los trabajos son las transformaciones de modelos. Únicamente [3] y [15] consideran también las transformaciones de modelo a texto. Respecto al metamodelo utilizado, la tendencia general es partir de un metamodelo genérico de traza, para construir un metamodelo específico centrado en las necesidades perseguidas en cada trabajo.

Por último, tanto para la representación de la información de traza, como para los lenguajes utilizados, es difícil establecer una línea de actuación común. Cada propuesta utiliza un lenguaje específico para realizar las correspondientes transformaciones o para almacenar los modelos. Lo mismo sucede con la representación de la información.

5 Trabajos Relacionados

Este trabajo se centra en mostrar los resultados obtenidos en una revisión sistemática sobre trazabilidad en Ingeniería Dirigida por Modelos, persiguiendo dos objetivos; por un lado aunar criterios e identificar actividades en torno al uso de la información de trazabilidad en la Ingeniería del Software y, por otro lado, operaciones básicas para la gestión de información de trazabilidad así como características comunes o diferenciadoras de propuestas basadas en Ingeniería Dirigida por Modelos para soporte de trazabilidad.

En relación con otras revisiones como [10], donde Galvao y Goknil presentan un estudio donde son evaluados cinco criterios a lo largo de diferentes propuestas en el contexto de MDE, como la representación de la información de trazabilidad, la correspondencia, escalabilidad, análisis del impacto del cambio y/o herramientas de soporte. Se valora la actividad de análisis del impacto del cambio, pero no son evaluadas otras actividades relativas a la gestión de proyectos y/o a la gestión de la configuración, tales como reutilización, ingeniería inversa, planificación de versiones, etc. Tampoco se abordan ninguna operación para la gestión de la trazabilidad en MDE.

Schwarz et al. en [20] presentan una visión global sobre trazabilidad, en lo referido al proceso de desarrollo software. El trabajo se estructura de acuerdo a seis operaciones específicas como; definición, registro (almacenamiento), identificación, mantenimiento, recuperación (consulta) y/o uso de la información en escenarios concretos. No considera las operaciones de importación/exportación, ni tampoco se evalúa el uso de información de trazabilidad en actividades relacionadas con el análisis de impacto del cambio, entendibilidad, gestión de proyectos, gestión de la configuración, etc.

6 Discusión y Conclusiones

El mantenimiento y gestión de la trazabilidad ha sido siempre un reto en el contexto de la Ingeniería del Software. La llegada de la Ingeniería Dirigida por Modelos y el papel clave de los modelos en el proceso de desarrollo, puede introducir mejoras en la gestión de la trazabilidad y sus aplicaciones.

Uno de los objetivos planteados al principio de este trabajo era conocer para qué es usada la información de trazabilidad. A partir de los resultados obtenidos, se puede concluir que el uso de la información de trazabilidad en actividades relativas al análisis del impacto del cambio, análisis de cobertura y análisis de huérfanos, son las más frecuentes. El uso de enlaces de traza en entre los diferentes elementos permite determinar el efecto del cambio sobre los artefactos en todo el sistema a través del ciclo de vida de desarrollo del software. Por el contrario, se han encontrado pocas propuestas donde se utilice la información de trazabilidad para mejorar la entendibilidad del software. En nuestra opinión, la aplicación de la información de trazabilidad para actividades de sincronización de modelos y como entrada para transformaciones de modelos resultaría muy beneficiosa para potenciar actividades de ingeniería inversa y reutilización, mejorando la calidad y ayudando al mantenimiento del software. Este hecho motiva nuestros esfuerzos a trabajar en el uso de la información de trazabilidad para hacer el software más entendible.

En relación a la segunda cuestión planteada, respecto a qué operaciones son soportadas para la gestión de trazabilidad, se concluye que ninguna propuesta soporta todas las operaciones evaluadas. Las operaciones de definición, identificación y visualización son las más maduras, soportadas casi en su totalidad por las distintas propuestas. En cambio, la operación de importación y/o exportación es identificada como la tarea menos abordada. Una de las posibles causas sea la carencia de un metamodelo estándar para facilitar el intercambio de este tipo de información. Consideramos, por tanto, interesante trabajar en la construcción de un metamodelo estándar que dotará de mayor interoperabilidad a las distintas propuestas. Además este estándar puede traer consigo nuevas operaciones de gestión de la trazabilidad, como la comparación de información de trazabilidad.

Respecto a qué características presentan las propuestas basadas en MDE para soportar trazabilidad, parece ser que las aproximaciones basadas en requisitos tienden a ofrecer soporte a la trazabilidad vertical, mientras que las propuestas basadas en transformaciones se centran tanto en la trazabilidad horizontal como en vertical. Así mismo, llama la atención el predominio de la trazabilidad de alto nivel, frente a la trazabilidad de bajo nivel. Finalmente, se identifica que la tendencia general de las distintas propuestas es partir de un metamodelo genérico de traza para, posteriormente, poder personalizarlo en metamodelos específicos centrados en satisfacer las necesidades perseguidas en cada trabajo.

Actualmente, se está trabajando en la definición de una propuesta para la gestión de trazabilidad en el contexto de la Ingeniería Dirigida por Modelos que permita el uso de la información de trazabilidad en los distintos campos evaluados en este trabajo. Asimismo también se intentará dar soporte a todas las operaciones para la gestión de la información de trazabilidad.

Agradecimientos. Este trabajo se ha llevado a cabo en el marco de los proyectos MODEL CAOS (Ref. TIN2008-03582) y Agreement Technologies (CONSOLIDER CSD2007-0022) financiado por el Ministerio de Ciencia y Tecnología de España.

Referencias

1. Aizenbud-Reshef, N., et al., *Model Traceability*. IBM Systems Journal - Model-driven software development, 2006. **45**(3): p. 515 - 526.
2. Aizenbud-Reshef, N., et al. *Operational Semantics for Traceability*. in *European Conference on Model-Driven Architecture - Traceability Workshop*. 2005.
3. Aleksy, M., et al. *A Pragmatic Approach to Traceability in Model-Driven Development*. in *Process Innovation for Enterprise Software*. 2009. Mannheim, Germany: PRIMIMUM'2009.
4. Almeida, J.P.A., M.-E. Iacob, and P. Van Eck, *Requirements Traceability in Model-Driven Development: Applying Model and Transformation Conformance*. Information Systems Frontiers, 2007. **IX**(4): p. 327-342.
5. Anquetil, N., et al., *A model-driven traceability framework for software product lines*. Software and Systems Modeling, 2010. **9**(4): p. 427-451.
6. Balasubramaniam, R., et al., *Requirements traceability: Theory and practice*. Annals of Software Engineering, 1997. **3**: p. 397 - 415.

7. Dee, M.R., *Traceability-based Change Management in the Model Driven Architecture*, in *Electrical Engineering, Mathematics and Computer Science2007*, Twente: Twente, Holanda. p. 157.
8. Drivalos, N., et al. *Towards Rigorously Defined Model-to-Model Traceability*. in *European Conference on Model Driven Architecture - Traceability Workshop*. 2008. Berlin, Germany: ECMDA-TW'2008.
9. Falleri, J.R., M. Huchard, and C. Nebut. *Towards a traceability framework for model transformations in kermeta*. in *European Conference on Model Driven Architecture - Traceability Workshop*. 2006. Bilbao, Spain: ECMDA-TW'2006.
10. Galvao, I. and A. Goknil. *Survey of Traceability Approaches in Model-Driven Engineering*. in *11th IEEE International Enterprise Distributed Object Computing Conference*. 2007. Annapolis, Maryland, EEUU: EDOC'2007.
11. Jouault, F. *Loosely Coupled Traceability for ATL*. in *European Conference on Model Driven Architecture - Traceability Workshop*. 2005. Nuremberg, Germany: ECMDA'05.
12. Kitchenham, B., et al., *Systematic literature reviews in software engineering - A systematic literature review*. Information and Software Technology, 2009. **51**(1): p. 7-15.
13. Kolovos, D., R. Paige, and F. Polack, *On-Demand Merging of Traceability Links with Models*, in *European Conference on Model-Driven Architecture - Traceability Workshop2006*, ECMDA-TW'06: Bilbao, Spain.
14. Melby, S.J., *Traceability in Model Driven Engineering*, in *Department of Informatics2007*, Osloensis: Oslo, Noruega. p. 130.
15. Olsen, G.K. and J. Oldevik, *Scenarios of Traceability in Model to Text Transformations*, in *Model Driven Architecture- Foundations and Applications2007*, Lecture Notes in Computer Science. Springer Berlin / Heidelberg. p. 144 - 156.
16. Paige, R.F., et al. *Building Model-Driven Engineering Traceability Classifications*. in *European Conference on Model Driven Architecture - Traceability Workshop*. 2008. Berlin, Germany: ECMDA-TW'08.
17. Ramesh, B. and M. Jarke, *Toward Reference Models for Requirements Traceability*. IEEE Trans. Softw. Eng., 2001. **27**(1): p. 58-93.
18. Schmidt, D.C., *Model-Driven Engineering*. IEEE Computer, 2006(39).
19. Schwarz, H., *Taxonomy and Definition of the Explicit Traceability Information Suppliable for Guiding Model-Driven, Ontology-Supported Development*, 2009.
20. Schwarz, H., J. Ebert, and A. Winter, *Graph-Based Traceability: A Comprehensive Approach*. Software and Systems Modeling, 2010. **9**(4): p. 473 - 492.
21. Seibel, A., S. Neumann, and H. Giese, *Dynamic hierarchical mega models: comprehensive traceability and its efficient maintenance*. Software & Systems Modeling, 2009. **9**(4): p. 493-528.
22. Selic, B., *MDA Manifestations*. Upgrade. The European Journal for the Informatics Profesional, 2001. **IX**(2): p. 12-16.
23. Tratt, L., *Model transformations and tool integration*. Journal of Software and Systems Modelling, 2005. **4**(2): p. 112 - 122.
24. Valderas, P. and V. Pelechano, *Introducing requirements traceability support in model-driven development of web applications*. Information And Software Technology, 2009. **51**(4): p. 749-768.
25. von Pilgrim, J., et al., *Constructing and Visualizing Transformation Chains*, in *Model Driven Architecture – Foundations and Applications*, I. Schieferdecker and A. Hartman, Editors. 2008, Springer Berlin / Heidelberg. p. 17-32.
26. Walderhaug, S., et al. *Towards a Generic Solution for Traceability in MDD*. in *European Conference on Model Driven Architecture - Traceability Workshop*. 2006. Bilbao, Spain: ECMDA-TW'2006.
27. Winkler, S. and J. von Pilgrim, *A survey of traceability in requirements engineering and model-driven development*. Software and Systems Modeling, 2010. **9**(4): p. 529-565.