

SEMÁNTICA DE TRAZADO PARA ASUNTOS TRANSVERSALES

Marta S. Tabares¹, Ana Moreira², Fernando Arango¹, Raquel Anaya³, João Araújo²

1: Escuela de Sistemas e Informática, Universidad Nacional de Colombia
Calle 59A No 63 - 20, Colombia, (574) 4255350
e-mail: {mstabare, farango}@unal.edu.co, web: <http://pisis.unalmed.edu.co>

2: CITI/Departamento de Informática, FCT da Universidade Nova de Lisboa
2829-516 Caparica, Portugal, +351 212948536
e-mail: {amm, ja}@di.fct.unl.pt, web: <http://ctp.di.fct.unl.pt/~amm/>

3: Escuela de Ingeniería - Departamento de Sistemas, Universidad EAFIT
Calle 59A No 63 - 20, Colombia, (574) 4255350
e-mail: ranaya@eafit.edu.co, web: <http://eafit.edu.co>

Palabras clave: Trazabilidad, asuntos transversales, semántica, join point, composición, Early Aspects, Aspect-Oriented Modeling, software engineering, requisitos, requirements.

Abstract. *Este artículo presenta una semántica de trazado para soportar y controlar la evolución de los asuntos transversales durante el proceso de desarrollo de software. Esta semántica define un conjunto de elementos de traza y establece caminos de trazado para preparar un ambiente donde sea posible verificar la consistencia semántica y la completitud en cualquier aproximación de modelado de asuntos transversales. La aproximación se ilustra a través de un caso de estudio.*

1. INTRODUCCIÓN

La trazabilidad es un atributo de calidad que ha sido tratado ampliamente en la ingeniería de software. Diferentes modelos y técnicas han sido propuestos para soportar el trazado en el proceso de desarrollo de software y lograr alta calidad en los productos que de este se generan, pero generalmente están orientados a trabajar con requisitos funcionales [1, 2, 3]. En el desarrollo de software orientado a aspectos (DSOAⁱ), se han realizado estudios con

ⁱ Trata con la identificación, separación, composición y trazado de los asuntos transversales. www.aosd.net

respecto a la práctica de la trazabilidad [4, 5]. Estos permiten ver la escasez de modelos o técnicas que permitan controlar y soportar el trazado así como la correspondiente validación de consistencia semántica y completitud para los modelos de asuntos transversales. El objetivo de nuestra aproximación es, proveer una semántica de trazado con elementos y caminos de trazado que permitan efectuar trazabilidad sobre cualquier aproximación de asuntos transversales. Esto permitirá crear un ambiente que identifique puntos de corte donde el trazado pueda ser verificado formalmente, además de permitir controlar algunas características típicas de evolución y algunas otras relacionadas con acciones generales de trazado.

La estructura de este artículo es como sigue. La sección 2 presenta una visión corta del caso de estudio. La sección 3 presenta la semántica de trazado para asuntos transversales. La sección 4 presenta trabajos relacionados. Finalmente, la sección 5 se concluye el artículo.

2. CASE DE ESTUDIO

Para ilustrar la propuesta desarrollada en esta aproximación, utilizamos el caso de estudio “Sistema de Subasta” [6] y el acercamiento de aspectos temprano AOSD/UC [7]. Específicamente, el requisito a tratar es el siguiente: “*El sistema de subasta es concurrente - los clientes pueden cotizar unos contra otros en paralelo, y un cliente coloca licitaciones en diferentes subastas además de aumentar su crédito paralelamente*”. Además, hemos creado un escenario de cambio para mostrar la disposición de la semántica de los elemento para el trazado: “*El cliente debe estar autorizado cuando un límite es excedido para continuar colocando licitaciones en subastas diferentes*”. Estos requisitos son representados en diagramas de casos de uso y diagramas de clases (ver Figuras 2-4 y 6 respectivamente) para la fase de análisis y diseño.

3. SEMÁNTICA PARA EL TRAZADO DE ASUNTOS TRANSVERSALES

El objetivo a lograr con esta aproximación es construir una semántica de trazado que permita verificar formalmente la consistencia semántica y la completitud en cualquier aproximación de modelado de asuntos transversales en fases tempranas del ciclo de vida. Para lograr esto hemos analizado de cada aproximación su meta-modelo el cual está conformado por constructores base definidos, las relaciones entre dichos elementos, las reglas utilizadas para transformar los diferentes artefactos en diferentes fases y posibles actividades de trazado requeridas para realizar la transformación que garanticen la consistencia de los elementos.

Nosotros hemos creado un modelo para soporta la semántica de trazado (ver Figura 1). Éste esta conformado por dos paquetes: el *EABaseModels* y el *TraceabilityDimensions*. El *EABaseModels* permite identificar, de las diferentes aproximaciones, los elementos (e.g. clases abstractas, atributos, relaciones, tablas de especificación, descripciones en XML, etc) y sus semánticas derivadas de los meta-modelos definidos en los *Join Point Models* – JPM y los *Composition Models* - CM. El *TraceabilityDimensions* toma los elementos reconocidos en el *EABaseModel* y los clasifica en dimensiones para establecer una semántica unificada que permita mantener el control del trazado y dar soporte a las transformaciones que los

elementos de los modelos experimenten a lo largo del ciclo de vida de desarrollo. Estas dimensiones son:

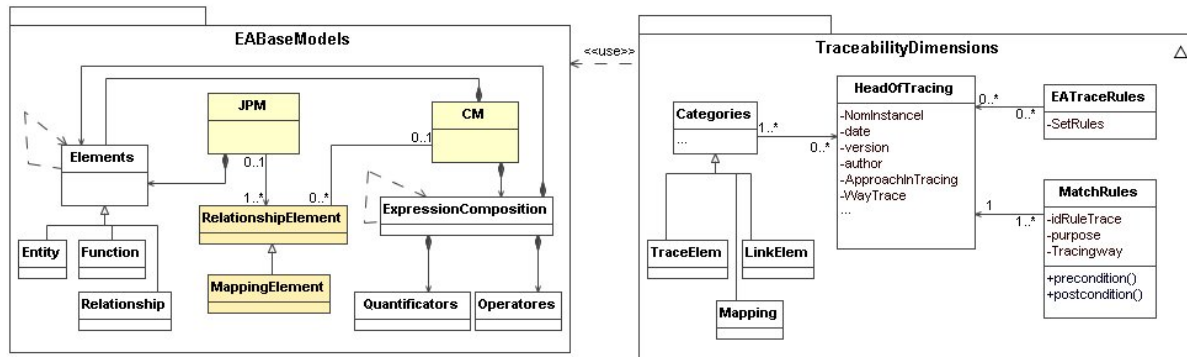


Figura 1. Modelo de la semántica de trazado

Dimensión <metaDim:TracingProvide> En esta dimensión se registran las características generales para controlar el estado general del trazado (ver Tabla 2, que presenta su definición y una instancia para el caso de estudio). Cada evento de trazado es registrado en *EncabezadoDeTrazado* a través de las siguientes características: <nombreInstancia>, <versión>, <autor> y <fecha> del trazado.

| EncabezadoDeTrazado | | | | | |
|------------------------------------|----------------------------|--------------------|----------------------|------------------------|--------------|
| <nombreInstancia>: SistemaSubastas | | <versión>: 1.0.x.x | | <autor>: Marta Tabares | |
| <fecha>: 12/09/2006 | | | | | |
| ApproachInTracing | WayTrace | DirTrace | FaseInTrack | MapElem | ActOfTracing |
| AOSD/UC | <Vertical> <Horizontal> | <NA> <Forward> | <Analysis>, <Design> | <NA> ⁱⁱ | <False> |

Tabla 1. Registro del Encabezado de Trazado para el caso de estudio.

Además, por cada aproximación involucrada en la actividad de trazado se lleva un registro de las características generales del trazado. <ApproachInTracing>: indica la aproximación de los modelos a ser trazados. <WayTrace>: indica el espacio donde se realiza el trazado; esta puede ser: *vertical*, que indica el trazado de productos de trabajo dentro de un nivel de abstracción (e.g. requisitos); *horizontal*, indica el trazado de la transformación de los requisitos en diferentes niveles de abstracción (e.g. entre análisis y diseño)ⁱⁱⁱ. <DirTrace>: indica la orientación en que se hace el trazado horizontal: forward (e.g. desde análisis al diseño) y backward en sentido inverso. <FaseInTrack>: indica cada uno de los niveles de abstracción en los cuales se realiza el trazado (e.g. análisis, diseño). <MapElement>: valor booleano (<true>, <false>) que indica si la aproximación del modelo trazado provee correlación entre los elementos de diferentes niveles de abstracción. <MethodOfTracing>: valor booleano (<true>, <false>) que indica si la aproximación poseen su propio método de

ⁱⁱ “<NA>” significa *No Aplica* y se registra cuando esa característica no es aplicable para esa práctica de trazado.

ⁱⁱⁱ Ver más sobre producto de trabajo, trazado vertical y horizontal en [1].

control del trazado. En caso de que el valor sea verdadero, se hará un manejo de estas para ser tenidas en cuenta como directrices de trazado.

Dimensión <metaDim:Categories> El paquete *EABaseModels* provee un modelo que permite identificar los elementos definidos en los metamodelos de las diferentes aproximaciones. Dichos elementos son clasificados en categorías con el objetivo de definir su función, significado y dependencias en la semántica de trazado. Se han creado tres categorías: <Fase>.<TraceElem>, <Fase>.<LinkElem> y <Mapping>.

(1) <Fase>.<TraceElem> En esta categoría los elementos de los modelos adquieren un significado dentro de la semántica de trazado. Estos son representados en *EABaseModels.Elements* como entidades (unidades funcionales o estructurales) o funciones (regla, restricción o función computacional) y se identifican por el nombre del elemento dado en cada aproximación (ElementName). A partir de esto se definen dos grupos de <TraceElem>: (a) los elementos de corte JE (*Join Elements*) que representan asuntos del problema base que pueden ser cortados o compuestos por otros elementos (asociados a *EABaseModels.JPM*); (b) los elementos de composición CE (*Composition Elements*) que representan la composición de los asuntos transversales con los asuntos base que son cruzados (asociados a *EABaseModels.CM*). La identificación de cada elemento y su función dentro de la semántica de trazado, son la base para la creación de caminos de trazado, reglas de transformación y reglas de verificación de consistencia y completitud. En la Figura 2 se representa el requisito descrito en el caso de estudio (Sección 2) y se muestra la definición de cada elemento en la semántica de trazado.

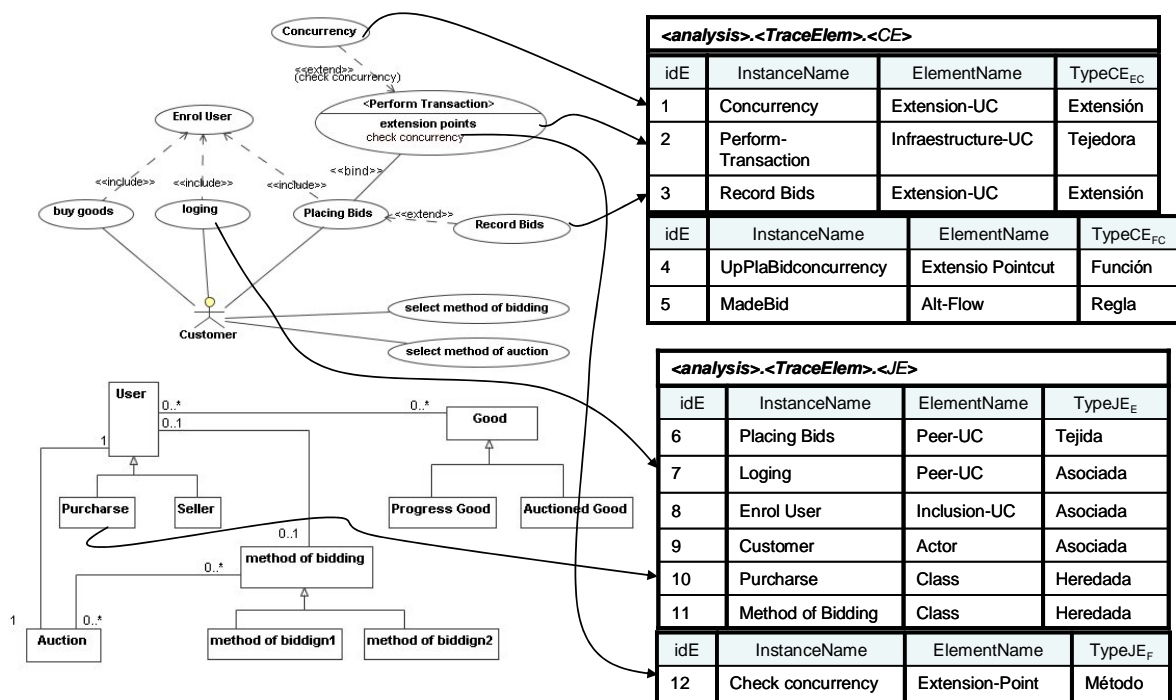


Figura 2. Representación AOSD/UC y definición de los *TraceElem* para la instancia del Sistema de Subasta.

Cada $\langle TraceElem \rangle$ es identificado de forma única (idE) y es definido en uno de los siguientes cuatro tipos:

- JE_E : elemento de corte tipo entidad. Se definen cuatro tipos de entidades: *agrupada*, *asociada*, *heredada* o *tejida* (entidad asociada con un CE). Esta diferenciación se hace con el fin de enlazar reglas de transformación y consistencia asociadas a cada tipo. Los elementos de este grupo pueden ser: casos de uso tipo peer-use-cases, clases tipo entidad o interfases, features, etc.
- JE_F : elemento de corte tipo función que la semántica distingue como métodos u operaciones asociadas a elementos de corte tipo entidad.
- CE_{EC} : elemento de composición tipo entidad. Se definen tres tipos de entidades: *de composición* (e.g paquetes, plantillas, componentes, $\langle\langle use\text{-}case\text{-}slice \rangle\rangle$), *tejedora* que agrupan asuntos transversales con asuntos base (e.g $\langle\langle perform\text{-}transaction\text{-}use\text{-}case \rangle\rangle$, colaboraciones, $\langle\langle aspect \rangle\rangle$) y *de extensión* que contiene flujos alternativos que pueden cruzar cualquier JE (e.g. case de uso de extensión, class extensions, etc).
- CE_{FC} : elemento de composición tipo función. Se define para identificar funciones, reglas o expresiones de composición^{iv} asociados a los CE_{EC} de extensión.

(2) $\langle Fase \rangle.\langle LinkElem \rangle$ En esta categoría se definen las dependencias o relaciones de trazado vertical entre los $\langle TraceElem \rangle$ definidos en la categoría anterior. Se construyen a partir de las relaciones que los modelos describen en *EABaseModels.RelationshipElement*.

| $\langle analysis \rangle.\langle LinkElem \rangle.\langle Rel \rangle$ | | | |
|---|-------------|----------------------------|--------------------------|
| idLE | ElementName | From | To |
| 1 | Include | Placing Bids (JE_E (6)) | Enrol User (JE_E (8)) |
| 2 | Include | Logging JE_E (7) | JE_E (8) |
| 3 | Association | Customer JE_E (9) | JE_E (6) |

(a)

| $\langle analysis \rangle.\langle LinkElem \rangle.\langle DepComp \rangle$ | | | |
|---|-------------|---------------------------|--------------------------------------|
| idLE | ElementName | From | To |
| 4 | Extend | Currency (CE_{EC} (1)) | Perform-Transaction (CE_{EC} (2)) |
| 5 | Bind | JE_E (6) | CE_{EC} (2) |
| 6 | Extension | JE_E (6) | CE_{EC} (3) |

(b)

| $\langle analysis \rangle.\langle LinkElem \rangle.\langle Trade \rangle$ | | | |
|---|-------------|---------------|-----------|
| idLE | ElementName | From | To |
| 7 | Trade | CE_{EC} (1) | $JE_E=10$ |
| 8 | Trade | CE_{EC} (1) | $JE_E=11$ |

(c)

Figura 3. Definición de los *LinkElem* para la instancia del Sistema de Subastas.

Cada $\langle LinkElem \rangle$ es identificado de forma única (idLE), y es definido en uno de los siguientes tres tipos:

- $\langle Rel \rangle$: identifica todos los tipos de relaciones de traza entre JE_E s (ver Figura 3(a)). A cada $\langle Rel \rangle$ pueden ser asociada restricciones definidas en el modelo. Este $\langle LinkElem \rangle$ también se declara explícitamente con expresiones de trazado tales como e.g

^{iv} *ExpressionComposition* = { $\langle all/any/none \rangle$, { JE_E , JE_F , $Rel_{(JEE)}$ }, $\langle where, less\ than \rangle$, $\langle and/or/none \rangle$, { JE_E , JE_F , $Rel_{(JEE)}$ }}. Sus elementos pueden ser usados de forma recursiva y son declarados en *EABaseModels.ExpressionComposition*.

ASOCIADO_CON, *AGRUPADO_EN* o *AGRUPA_A*, *ES_HEREDADO_DE*, *ES_TEJIDO_EN*, entre otras.

- $\langle \text{DepComp} \rangle$: identifica la *dependencia composicional de trazado* entre CEs y JEs (ver Figura 3(b)). Ésta es considerada una dependencia especial ya que nuestro propósito es liderar el trazado desde los elementos de composición (CE) y por esa razón es definida de la siguiente forma: un JE es composicionalmente dependiente de un CE, denotado por $\text{JE} \langle \text{DepComp} \rangle \text{CE}$, si y solo si para cada valor específico *JE*, existe exactamente otro valor *CE* que lo traza. Esta definición también aplica a la dependencia composicional entre CEs. Este $\langle \text{LinkElem} \rangle$ también es declarado explícitamente con dos expresiones de trazado: *COMPUESTO_EN* y *ES_TEJEDOR_DE*. Por ejemplo, *Perform-Transaction-UC ES_TEJEDOR_DE Concurrency-UC*, lo que significa que $\text{CE}_{\text{EC}}(1) \langle \text{DepComp} \rangle \text{CE}_{\text{EC}}(2)$.
- $\langle \text{Trace} \rangle$: son definidas como las *relaciones de trazado* entre elementos de diferentes representaciones de un requisito dentro de un mismo nivel de abstracción (ver Figura 3(b)). La expresión de trazado *TRAZA_A* es declarada para este tipo.

Una serie de vínculos de trazado pueden establecerse como posibles caminos de trazado verticales como se mostrarán en la sección 4.

(3) $\langle \text{Mapping} \rangle$ En esta categoría se definen las dependencias o relaciones de trazado horizontal entre los $\langle \text{TraceElem} \rangle$ de diferentes niveles de abstracción^v. Si las aproximaciones proveen correlaciones ya predefinidas entre elementos de diferentes niveles de abstracción, estas son descritas en *EABaseModels.MappingElement*. Cada correlación (MEi: Mapping Element) es identificado de forma única (idMap), y es definido en uno los siguientes tipos de correlación^{vi}.

- $\text{ME}_1 = \langle \text{Fase}_i \rangle . \langle \text{JE}_E \rangle \text{ map } \langle \text{Fase}_j \rangle . \langle \text{JE}_E \rangle$: Define una traza (1-1) de un elemento de corte tipo entidad en dos niveles de abstracción diferentes y continuos.
- $\text{ME}_2 = \langle \text{Fase}_i \rangle . \langle \text{JE}_F \rangle \text{ map } \langle \text{Fase}_j \rangle . \langle \text{JE}_F \rangle$: Define una traza (1-1) de un elemento de corte tipo función en dos niveles de abstracción diferentes y continuos.
- $\text{ME}_3 = \langle \text{Fase}_i \rangle . \langle \text{CE}_{\text{EC}} \rangle \text{ map } \langle \text{Fase}_j \rangle . \langle \text{CE}_{\text{EC}} \rangle$: Define una traza (1-1) de un elemento de composición tipo entidad en dos niveles de abstracción diferentes y continuos.
- $\text{ME}_4 = \langle \text{Fase}_i \rangle . \langle \text{CE}_{\text{FC}} \rangle \text{ map } \langle \text{Fase}_j \rangle . \langle \text{CE}_{\text{FC}} \rangle$: Define una traza (1-1) de un elemento de composición tipo función en dos niveles de abstracción diferentes y continuos.
- $\text{ME}_5 = \langle \text{Fase}_i \rangle . \langle \text{JE}_E \rangle \text{ map } \langle \text{Fase}_j \rangle . \langle \text{JE}_F \rangle$: Define una traza entre un elemento de corte tipo entidad que se transforma en un elemento de corte tipo función (o viceversa) en dos niveles de abstracción diferentes y continuos.
- $\text{ME}_6 = \langle \text{Fase}_i \rangle . \langle \text{JE}_E \rangle \text{ map } \langle \text{Fase}_j \rangle . \langle \text{CE}_{\text{EC}} \rangle$: Define una traza entre un elemento de corte tipo entidad que se transforma en un elemento de composición tipo entidad (o viceversa) en dos niveles de abstracción diferentes y continuos.

En la Figura 4 se muestra la transformación del caso de uso extendido *concurrency* en la fase

^v En esta dimensión la dirección del trazado se realiza de acuerdo al valor registrado en *EncabezadoDeTrazado.DirTrace*. Para el caso de estudio el valor correspondiente al trazado horizontal es $\langle \text{Forward} \rangle$

^{vi} Por falta de espacio solo se enuncian algunos tipos de correlación.

de diseño con la definición de sus elementos y algunas de las correlaciones. En la aproximación AOSD/UC, un caso de uso de la fase de análisis es representado, en la fase de diseño, por un *use-case slice* que conserva la especificación de un *use-case realization*.

En esta dimensión toma gran relevancia la agrupación de los $\langle TraceElem \rangle$ y sus correspondientes tipos para la construcción de reglas de transformación, consistencia y completitud. Por ejemplo, un JE_E tipo agrupadora puede continuar siendo del mismo tipo o transformarse en otros JE_E tipo, asociada o de composición. La expresión de trazado cada *MAPPING_TO* indicará la correlación de dos elementos.

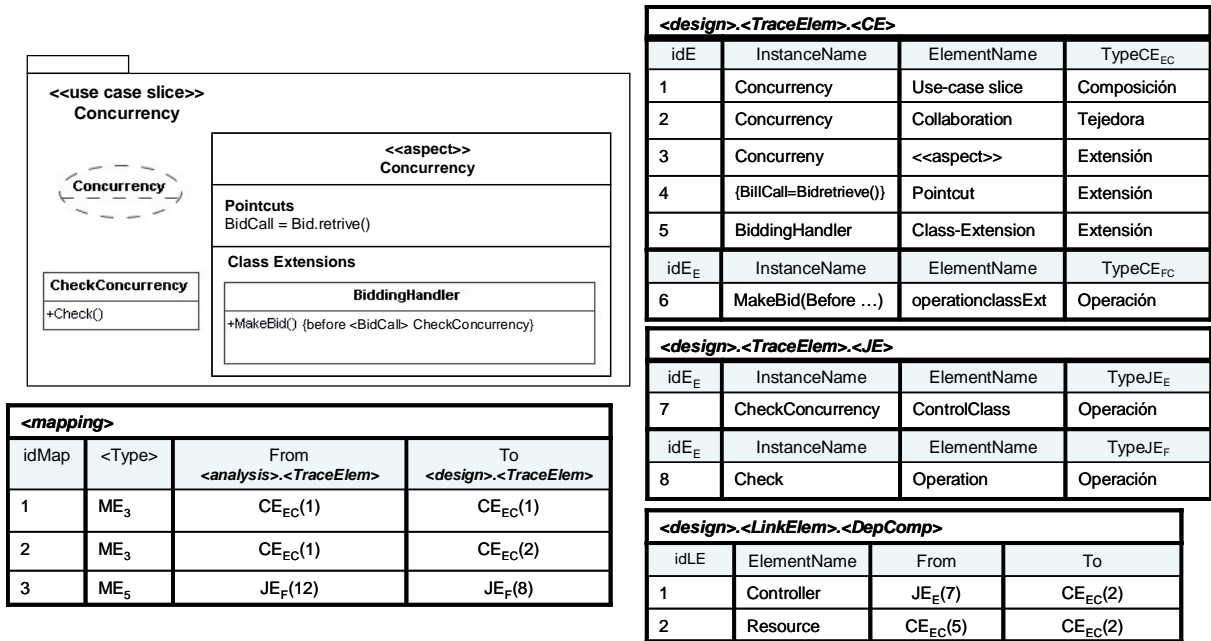


Figura 4. Representación AOSD/UC y definición de los *Mapping* para la instancia del Sistema de Subasta.

Dimensión $\langle metaDim:MatchRules \rangle$. Esta dimensión se definen caminos de trazado con el objetivo de poder evaluar la consistencia semántica y la completitud de un requisito a medida que se va transformando en el ciclo de vida. Estos caminos son creado con base en reglas o políticas generales en términos de *precondiciones* para verificar las condiciones de entrada o estado inicial del trazado definido en las dimensiones $\langle metaDim:TracingProvide \rangle$ y $\langle metaDim:Categories \rangle$, y las *postcondiciones* para soportar una correcta operación de verificación con respecto a la consistencia semántica y la completitud. Para lograr esto se han formulado las siguiente reglas:

- o **R1: *BaseTracingModel*** \rightarrow creación de una *línea base* del trazado para determinar el conjunto de caminos de trazado básicos donde se realizarán las correspondientes verificaciones.

pre: Cada elemento de modelo involucrado en el trazado debe estar compuesto por un *EncabezdoDeTrazado* y su correspondiente descripción de acuerdo a la semántica definida por $\langle metaDim:Categories \rangle$.

post: Cada camino de trazado es construido a partir de: un nodo cabeza de camino que representa un CE_{EC} o un JE_E y un conjunto de nodos secundarios de JEs o CEs que son vinculados a través de los $\langle LinkElem \rangle$ definidos entre ellos (ver ejemplo en la Figura 5).

- **R2: *IsReflectingIn*** → existe al menos una camino de trazado que puede ser recorrido de forma *top-down* sobre el cual pueda validarse consistencia semántica con base en sus dependencias de trazado, reglas de transformación y los elementos que la conforman.

pre: Todo recorrido de validación debe iniciar por un nodo cabeza de camino que corresponda a un asunto transversal válido e.g. $\langle Analisis \rangle . \langle CE_{EC}(1) \rangle$.

post: El estado del recorrido del camino de trazado es verdadero y puede reflejarse en los siguientes elementos: $\langle Analisis \rangle . \langle CE_{EC}(2) \rangle$, $\langle Analisis \rangle . \langle JE_E(6) \rangle$ (para el camino de trazado 1).

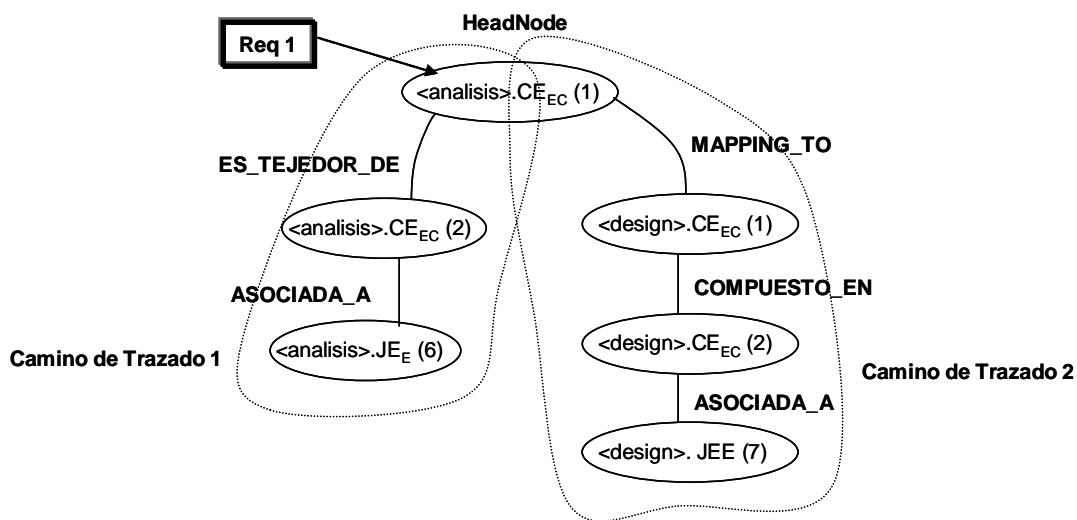


Figura 5. Representación de caminos de trazado para el caso de estudio.

R3: *IsEncapsulatedBy* → existe al menos una camino de trazado que puede ser recorrido de forma *bottom-up* sobre el cual pueda validarse completitud en el modelo con base en sus dependencias de trazado, reglas de transformación y los elementos que la conforman.

pre: Todo recorrido de validación debe iniciar por un nodo cabeza de camino que corresponda a un asunto transversal válido e.g. un nodo cabeza de traza sería $\langle Analisis \rangle . \langle JE_E(6) \rangle$.

post: El estado del recorrido del camino de trazado es verdadero y los elementos $\langle Analisis \rangle . \langle JE_E(1) \rangle$ y $\langle Analisis \rangle . \langle CE_{EC}(2) \rangle$ pueden encapsularse en $\langle Analisis \rangle . \langle CE_{EC}(1) \rangle$ (para el camino de trazado 1).

- **R4: *ChangeScenarioDependency*** → los elementos del modelo que representan el escenario de cambio son descritos de acuerdo a una aproximación reconocida y con su correspondiente descripción de acuerdo a la semántica definida por

<metaDim:Categories> (ver figura 6) .

pre: un nuevo evento de trazado con la operación de cambio debe ser válido (Create/Update/Delete) & los nuevos elementos de modelo deben ser descritos en la semántica de trazado.

post: los valores de *<metaDim:TracingProvide>*, los modelos y los caminos de trazado actualizados deben reportar estados válidos.

- **R5: ExceptionsTrace** → existen elementos de los modelos que no tienen correspondencia con otros elementos de la misma fase o entre fases.

pre: identificar todos los elementos que no tengan camino de trazado vertical u horizontal.

post: los elementos de modelo validados son marcados como *excepciones de traza*.

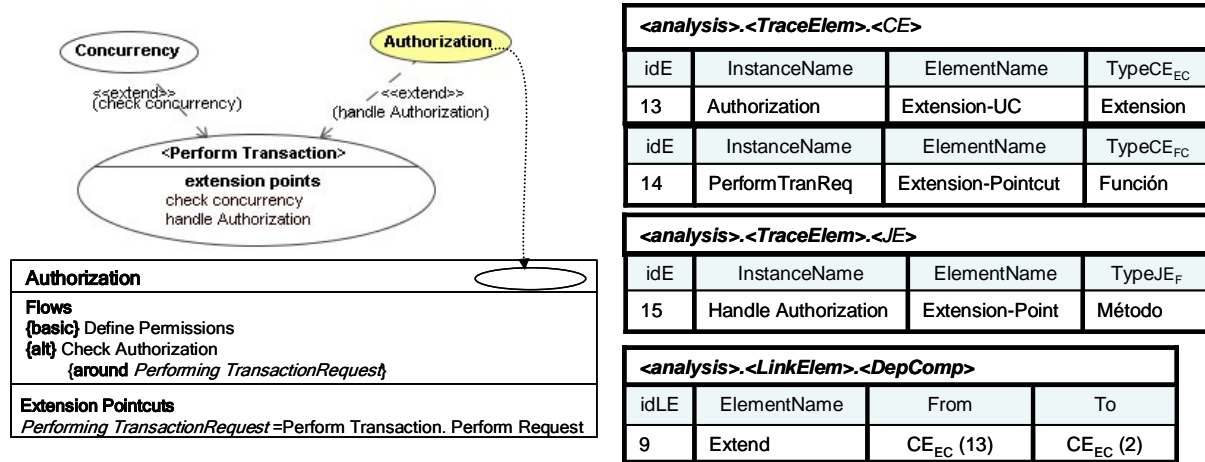


Figura 6. Representación del escenario de cambio en la aproximación AOSD/UC y su correspondiente descripción en la semántica de trazado.

4. TRABAJOS RELACIONADOS

A medida que las aproximaciones de aspectos tempranos se fortalecen, el soporte a la trazabilidad de los asuntos transversales cobra mayor interés. Bakker presenta un marco de trabajo para el trazado de asuntos usando XML [8]. Esta permite determinar el impacto de los escenarios de cambio al nivel de requisitos y arquitectura. Esta aproximación provee trazabilidad entre fases del ciclo de vida. La semántica usada es completa con respecto al trazado de asuntos, sin embargo la trazabilidad de los asuntos transversales aún no ha sido explorada. Para nuestra aproximación es importante el aporte que Bakker hace con respecto a la semántica XQuery garantizando el trazado de asuntos entre fases.

Chitchyan et al., presenta una aproximación bajo XML. Este marco de trabajo incluye los esquemas de todos los artefactos del ciclo de vida y permite realizar la correlación entre requisitos y arquitectura identificando elementos correspondientes entre RDL y DAOP- ADL; además de controlar el trazado a través del registro histórico de cada actividad realizada sobre de asuntos transversales [9].

5. CONCLUSIONES Y TRABAJO FUTURO

En este artículo hemos presentado la especificación de una semántica de trazado desde los asuntos transversales. Las aproximaciones de aspectos tempranos están orientadas a identificar asuntos de corte transversal, lo que permite que nuestra aproximación de trazado lleve el control de la evolución de los requisitos a partir de rutas de trazado constituidas desde asuntos transversales como seguridad, *performance*, persistencia, entre otros. Generar rutas de trazado a partir de los asuntos transversales permite controlar y soportar dinámicamente los cambios que afecten piezas de software en cualquier nivel de abstracción o iteración realizada por un analista. Esta semántica se convierte en el ambiente de trazado que nos permitirá verificar los modelos en consistencia semántica y completitud a través de formulas bien formadas. Poder controlar y soportar formalmente la trazabilidad desde los modelos de composición abre posibilidades prometedoras para garantizar la consistencia del proceso de modelado, analizar el impacto de los cambios y facilitar las pruebas del software en fases tempranas de desarrollo. En el futuro, esta propuesta estará soportada bajo un lenguaje como XML, de tal forma que pueda ser adaptada a CASEs orientados a aspectos.

REFERENCIAS

- [1] Lindvall, M., A Study of Traceability in Object-Oriented Systems Development, Licentiate Thesis 462, Dep. of Computer and Information Science, Linkping University, Sweden 1994.
- [2] B. Ramesh, M. Jarke, "Toward Reference Models for Requirements Traceability. IEEE Transactions On Software Engineering", Vol. 27, No. 1, January 2001.
- [3] P. Letelier, V. Anaya, "Entegrando Especificaciones Textuales y Elementos de modelado UML en un Marco de Trabajo para Trazabilidad de Requisitos". JISBD pag. 151-162. 2002.
- [4] R. Chitchyan, A. Rashid, P. Sawyer, J. Bakker, M. Pinto Alarcon, A. Garcia, B. Tekinerdogan, S. Clarke and A. Jackson, "Survey of Aspect-Oriented Analysis and Design Approaches", AOSD-Europe-ULANC-9, AOSD-EUROPE network of excellence. May 2005.
- [5] M. Tabares, A. Moreira, "Towards a Meta Aspect for Traceability". AOSD'06. Early Aspects: Traceability of Aspects in the Early Life Cycle. Workshop March 2006, Germany.
- [6] F.T. Sheldon, K. Jerath, Y-J. Kwon and Y-W. Ba, "Case Study: Implementing a Web Based Auction System using UML and Component-Based Programming," 26th Annual International Computer Software and Applications Conference, COMPSAC 2002: 211-216.
- [7] I. Jacobson and P.-W. Ng, Aspect-Oriented Software Development with Use Cases: Addison Wesley Professional, 2005.
- [8] P.J. Bakker, "Traceability of concerns". Master Thesis. University of Twente. 2005.
- [9] R. Chitchyan, M. Pinto, L. Fuentes, A. Rashid: Relating AO Requirements to AO Architecture, Early Aspects 2005 at OOPSLA.