

LA INGENIERÍA DE REQUISITOS ORIENTADA A ASPECTOS: UNA EXPERIENCIA DE APLICACIÓN EN UN SISTEMA DE AYUDA EN LÍNEA

ASPECT-ORIENTED SOFTWARE ENGINEERING: AN EXPERIENCE OF APPLICATION IN A HELP DESK SYSTEM

MARTA S. TABARES

Departamento de Informática, Escuela de Ingeniería de Antioquia, Envigado, pfmstabare@eia.edu.co

RAQUEL ANAYA

Escuela de Ingeniería, Departamento de Sistemas, Universidad EAFIT, Medellín

FERNANDO ARANGO

Escuela de Sistemas e Informática, Universidad Nacional de Colombia, Medellín

Recibido para revisar octubre 20 de 2006, aceptado abril 27 de 2007, versión final julio 20 de 2007

RESUMEN: La Ingeniería de Requisitos Orientada a Aspectos provee enfoques para la elicitación y especificación de los asuntos y asuntos transversales en etapas tempranas de desarrollo de software. Este artículo presenta un caso de estudio para evaluar un modelo de la separación multidimensional de asuntos orientados a aspectos. Con el caso de estudio, se revisan sus características para la elicitación, análisis y trazabilidad de requisitos, así como algunas limitaciones del modelo y cómo este puede ser adaptado al proceso de desarrollo. Además, planteamos un conjunto de argumentos que podrían promover el desarrollo y evolución de este modelo.

PALABRAS CLAVE: asuntos, separación de asuntos, asuntos transversales, requisitos funcionales, requisitos no funcionales, RF, RNF, ingeniería de requisitos, modelo de requisitos, aspectos tempranos, ingeniería de requisitos orientada a aspectos, DSOA.

ABSTRACT: Aspect-oriented Requirement Engineering provides approaches for eliciting and specifying the concerns and crosscutting concerns in the early stages of software development. In this paper, we present a case study in order to assess a model of the aspect-oriented multidimensional separation of concerns. With the case study, we review their features for elicitation, analysis and traceability of requirements, as well as some limitations of the model and how it can be adapted to the development process. Also, we discuss some issues that might endorse the development and evolution of this model.

KEY WORDS: concerns, separation of concerns, crosscutting concerns, functional requirement, non-functional requirements, FR, NFR, requirement model, requirement engineering, early aspects, aspect-oriented requirement engineering, AORE.

1. INTRODUCCIÓN

La separación de asuntos (*separation of concerns*) ha sido tratada ampliamente en la ingeniería de software y específicamente por la ingeniería de requisitos. Está orientada hacia la descomposición de los dominios del problema y de la solución, con el fin de reducir la complejidad, eliminar fallas de interpretación, mejorar la relación entre usuarios y

desarrolladores, y estructurar sistemas complejos a través de subsistemas, módulos o elementos simples de una forma más natural [1, 2]. Para soportar dicha separación, diferentes enfoques de ingeniería de requisitos han creado modelos que son considerados buenas prácticas para la elicitación y análisis de los requisitos [3, 4, 5, 6, 7]. Los requisitos pueden ser separados de acuerdo a diferentes criterios; generalmente,

los analistas parten de los modelos de procesos del negocio y hacen la discriminación de acuerdo a las funciones del dominio del problema para así identificar los requisitos funcionales, no funcionales y restricciones, básicamente. Muchas veces se logra una separación mayor identificando reglas de negocio, requisitos de información y servicios, para determinar con mayor exactitud la arquitectura del sistema desde fases tempranas de desarrollo.

Es posible también lograr otro tipo de separación identificando los requisitos que cruzan a otros; estos son llamados *asuntos transversales* (*crosscutting concerns*). Comúnmente son difíciles de factorizar y soportar en módulos separados ya que se encuentran dispersos (*scattering*) o enmarañados (*tangling*) en diferentes asuntos o requisitos del sistema. Los requisitos más comunes que tienen esta naturaleza transversal son los que representan atributos de calidad del sistema. Por ejemplo, los requisitos asociados al asunto (*concern*) de “seguridad”, la mayoría de veces no son elicitados en etapas tempranas o son descritos de forma solapada en los requisitos funcionales y consecuentemente este asunto no logra ser separado fácilmente en una clase o método en posteriores fases del ciclo de vida.

La separación de este tipo de asuntos es tratada por el Desarrollo de Software Orientado a Aspectos (AOSD – *Aspect Oriented Software Development*, www.aosd.net). Esta disciplina nace inicialmente como modelo de programación y rápidamente se percibe su potencialidad como enfoque útil en las fases tempranas del desarrollo [8, 9, 10, 11]. Es así como surgen diferentes aproximaciones orientadas a aspectos que apoyan la elicitación y análisis de requisitos de software [14] y se constituye la Ingeniería de Requisitos Orientada a Aspectos (AORE - *Aspect oriented Requirement Engineering*, www.early-aspects.net).

El objetivo a lograr en este artículo es evaluar, por medio de un caso de estudio, el enfoque de la separación multidimensional de asuntos orientados a aspectos [12]. Esta experiencia permitirá divulgar sus características más

relevantes de elicitación, análisis y trazabilidad de requisitos, sus limitaciones y las posibilidades de adaptación y uso en el proceso de desarrollo de software.

El artículo está estructurado de la siguiente forma: Sección 2 describe las bases teóricas del modelo. Sección 3 presenta el análisis del caso de estudio desde la perspectiva del modelo. Finalmente en la sección 4 se presentan conclusiones y trabajos futuros.

2. MODELO DE INGENIERÍA DE REQUISITOS ORIENTADO A ASPECTOS AORE

AORE es un modelo para el tratamiento de requisitos que enfatiza el tratamiento de intereses transversales[11, 12]. Este modelo usa la técnica de PREview basada en viewpoints con un mecanismo de composición basado en XML (bajo la herramienta ARCADE). Un asunto es definido como un conjunto coherente de requisitos; los asuntos transversales son identificables en matrices donde es posible establecer cuándo algunos de sus requisitos influyen de manera positiva (*colabora*) o afectan de manera negativa (*restringe*) otros asuntos. La aproximación está orientada principalmente a:

- ♦ Identificar, separar y componer asuntos transversales.
- ♦ Soportar el establecimiento de compensaciones (*trade-off*) tempranas entre asuntos transversales y requisitos que se superponen .
- ♦ Negociar y tomar decisiones entre grupos de participantes (*stakeholders*).

La aproximación es refinada por Moreira et al. en [13] y adaptada desde una perspectiva multidimensional para tratar los asuntos en un “espacio de asuntos” de forma uniforme sin importar la naturaleza de sus requisitos, funcionales o no funcionales. Posteriormente, el modelo es soportado por PROBE, un framework para verificar con lógica temporal las reglas de composición [18] (ver Figura 1).

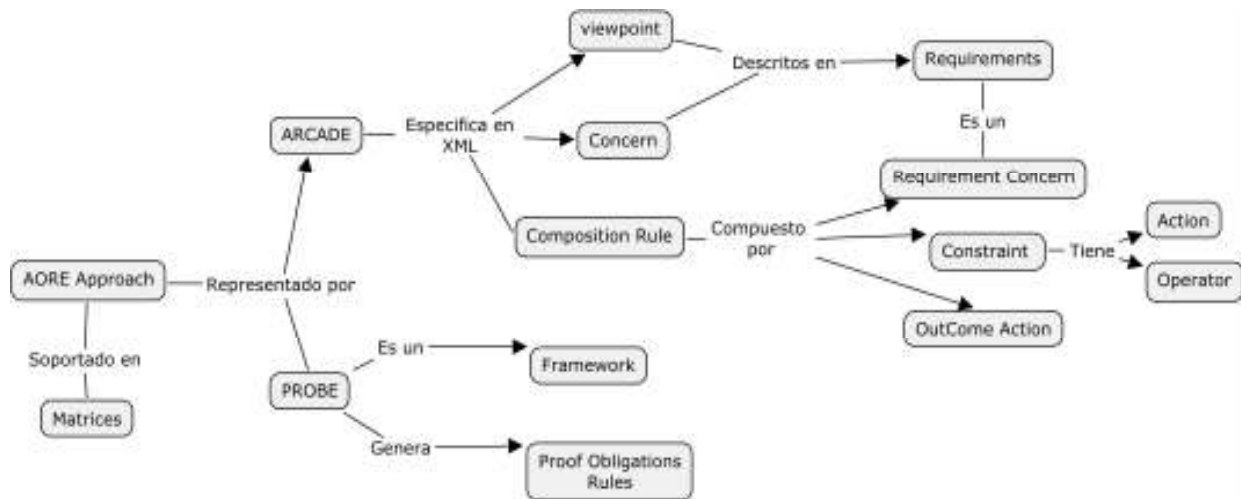


Figura 1: Un esquema conceptual de la aproximación AORE

Figure 1: A conceptual schema of the AORE approach

2.1 El Proceso de AORE

La aproximación desarrolla el modelo AORE en un proceso determinado por cinco actividades (ver Figura 2):

1. Identificar y especificar asuntos. Para esta actividad se utilizan técnicas de separación de requisitos como: *viewpoints* [3], casos de usos [4], y orientación a objetivos [5], entre otras. Su especificación es realizada en XML.

2. Identificar relaciones de grano grueso entre asuntos. Se buscan las posibles influencias entre los asuntos. Estos asuntos son relacionados entre sí en una matriz.

3. Especificar las proyecciones de los asuntos usando reglas de composición. Se especifican las relaciones entre los asuntos con base en las influencias encontradas entre los requisitos. Esta especificación (XML) se realiza a través de reglas de composición.

4. Manejo de conflictos entre asuntos. Actividad conformada por un conjunto de acciones que llevan a la solución de conflictos ocasionados por las contribuciones entre asuntos.

5. Especificación de las dimensiones de los asuntos. Se decide la manera como los asuntos serán soportados en la arquitectura con respecto a las dimensiones de influencia y mapeo:

- ♦ *Influencia:* indica en cuales etapas del ciclo de vida afectarán los asuntos; son particulares a cada proyecto o son determinadas por los desarrolladores.
- ♦ *Mapeo:* indica los artefactos en los cuales serán correlacionados los asuntos especificados:

* **Función:** cuando el asunto es posible relacionarlo a nivel de diseño como una clase estándar (o un conjunto de clases) que debe mantener el control de la relación con otras clases.

* **Decisión:** cuando el asunto es tenido en cuenta sólo para la toma de decisiones en cualquier etapa del ciclo de vida.

* **Aspecto:** cuando las propiedades de un asunto no pueden ser encapsuladas en una clase sencilla y estas propiedades podrían estar dispersas por diferentes clases del núcleo del sistema.

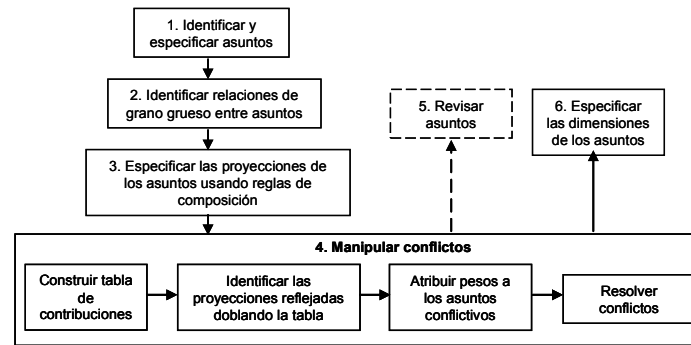


Figura 2: Modelo de ingeniería de requisitos orientado al tratamiento uniforme de asuntos [11-13]

Figure 2: Concern-oriented requirement engineering model [11-13]

2.2 Especificación De Aore En Arcade

La aproximación especifica el detalle de los asuntos, sus requisitos y reglas de composición en la herramienta ARCADE. Una instancia del modelo se desarrolla en la sección 3.

Las reglas de composición son especificadas para cada asunto (en XML) como intersecciones composicionales que permiten identificar puntos potenciales de compensación (trade-off) [12]. Las reglas están compuestas por los requisitos de cada asunto (*requirement concern*), las restricciones con la acción y el operador de tejido correspondiente (*constraint action - operator*) y la acción de resultado (*Outcome action*). Esta última acción toma gran importancia en la composición, ya que determinará la acción de salida al realizar el tejido, ya sea por realización satisfactoria de la regla o porque dicha regla satisface otro asunto.

Las restricciones de acción definidas en [12 - (Tabla 1)] especifican el modo en que un requisito puede afectar a otro con el que se relaciona. Así, es posible establecer las condiciones asociadas al comportamiento que será adicionado en el momento del tejido. Algunas de las restricciones de acción definidas son: *reforce* (reforzar), usada para imponer condición adicional sobre un conjunto de asuntos; *ensure* (asegurar), usada para determinar que una condición para un conjunto

de asuntos existentes; *provide* (proveer), usada para especificar características adicionales a ser

incorporadas en un conjunto de asuntos; *affect* (afecta), usado para especificar que un conjunto de requisitos de un asunto alterará el estado de otro.

Los operadores definidos en [12 - (Tabla 2)], permiten establecer la condición de evento, es decir “*el cuándo*” se hará el tejido. La diversidad de operadores lleva a reconocer todos los posibles estados de temporalidad a los cuales un asunto estará asociado en el momento del tejido. Algunos de los operadores definidos son: *during* (durante), describe el intervalo temporal en el cual un conjunto de requisitos están siendo satisfechos; *on* (sobre), describe el punto temporal después que un conjunto de requisitos ha sido satisfecho; *and*, *or*, *xor* (y, o, o exclusivo), ayudan a la conjunción y disyunción de requisitos de asuntos.

Los operadores de resultados [12 - (Tabla 3)], permiten representar los resultados del tejido hecho entre asuntos.

3. DESARROLLO DEL CASO DE ESTUDIO

A través del caso de estudio analizamos la aplicabilidad de la aproximación AORE, sus características de elicitación, análisis y trazabilidad de requisitos. De igual forma, en cada actividad hacemos un análisis de sus limitaciones y posibilidades de adaptación y uso en el proceso de desarrollo.

3.1 Descripción General Del Problema

El problema corresponde a un desarrollo orientado a objetos que se localiza en el Departamento de Mantenimiento de una empresa [15]. Su función es dar ayuda en línea a solicitudes de servicios de mantenimiento de edificios. El usuario afectado por el daño debe diligenciar una solicitud en la cual hace una descripción del problema; esta podrá ser ingresada a cualquier hora del día y cualquier día de la semana. Un empleado del área de mantenimiento la recibe y valida; si está correcta, programa los técnicos para la atención del servicio en una agenda de servicios.

Cuando se atiende el servicio, el técnico diligencia una planilla con la información correspondiente. Una solicitud puede involucrar la ejecución de más de un servicio para ser terminada. Un servicio de mantenimiento puede ser contratado con una empresa externa con la debida autorización del jefe de área y del departamento financiero. Todo el flujo de trabajo de una solicitud de servicio deberá ser reportado en diferentes estadísticas de servicio. El sistema deberá validar y autorizar el ingreso al sistema de acuerdo al rol de cada usuario, que a su vez deberán ser empleados. El sistema deberá soportar muchos usuarios concurrentes. El jefe del departamento de mantenimiento espera que el sistema tenga un tiempo de respuesta rápido y que el departamento de sistemas provea constante mantenimiento al sistema.

Para la elicitación de los asuntos se tuvieron en cuenta artefactos como el discurso general del problema, diagramas de procesos y casos de uso. A continuación se muestra el desarrollo del caso de estudio en cada uno de los pasos indicados por el proceso del modelo.

3.2 Identificación Y Especificación De Asuntos

Para la identificación de asuntos es necesario contar con modelos de negocio muy bien definidos y especificados. El uso de aproximaciones orientadas a objetivos (*Goal-Oriented Approaches*) y orientadas a

modelamiento de negocios (*business-oriented modeling*), son convenientes. Además, consideramos importante que los participantes identifiquen asuntos del sistema tales como seguridad, concurrencia, disponibilidad, entre otros.

Los requisitos deben ser agrupados de acuerdo al contexto funcional determinado por cada uno de los asuntos. Para lograr esto, realizamos varios refinamientos hasta conseguir las agrupaciones correspondientes.

Los siguientes asuntos fueron identificados:

- ♦ **Solicitudes:** Diligenciar y verificar información relacionada con solicitudes de servicio. Inicia el proceso de atención.
- ♦ **Servicios:** Manejar los servicios pendientes; asignación de técnicos, actualización de la agenda y generación de reporte de servicios.
- ♦ **Gestión de usuarios:** Administrar usuarios, roles y asignar privilegios a éstos.
- ♦ **Seguridad:** Validar el acceso de los usuarios, mostrar información de acuerdo a roles establecidos y garantizar la integridad de los datos.
- ♦ **Disponibilidad:** Habilitar disponibilidad del sistema 24x7.
- ♦ **Compatibilidad:** Interactuar con otros sistemas existentes.
- ♦ **Multiacceso:** Soportar múltiples usuarios concurrentes.
- ♦ **Soporte al sistema:** Mantener el sistema operativo, la base de datos, la aplicación y otros dispositivos lógicos y físicos que sean necesarios.
- ♦ **Sistemas externos:** Proveer desde sistemas externos información al sistema de ayuda, para la validación de usuarios, activos y presupuesto.
- ♦ **Estadísticas de Gestión:** Entregar estadísticas de las solicitudes, servicios y funcionalidades del mismo.

- ♦ **Contratos:** Permitir la gestión de la contratación de un servicio con una empresa externa.

La Figura 3(a) muestra la especificación del asunto “*Servicios*” y en la Figura 3(b) el asunto “*Estadísticas de Gestión*”, ambos con sus

correspondientes requisitos. Cada asunto y requisito es identificado; si un requisito necesita ser subdividido en varios requisitos, el requisito más detallado se identifica conservando el identificador del padre al cual se le adiciona un nuevo número para formar el identificador del requisito hijo.

```
<?xml version="1.0" ?>
<Concern name="Servicios">
  <Requirement id="1"> Crear servicios asociados a una solicitud.
</Requirement>
  <Requirement id="2"> Asignar técnicos a un servicio no regular.
  <Requirement id="2.1"> Los técnicos deben estar disponibles para la fecha a la cual se va a asignar el servicio.
</Requirement>
  <Requirement id="2.2"> Generar un reporte de cada uno de los servicios asociados a una solicitud para ser asignados a los técnicos disponibles.
</Requirement>
  <Requirement id="2.3"> Cuando todos los técnicos que pueden atender un servicio están ocupados, el servicio queda en estado de espera.
</Requirement>
</Requirement>
  <Requirement id="3"> Programar fechas de los servicios regulares del departamento de mantenimiento.
  <Requirement id="3.1"> Asignar técnicos a los servicios regulares programados.
</Requirement>
</Requirement>
</Concern>
```

(a)

```
<?xml version="1.0" ?>
<Concern name="Estadísticas de Gestión">
  <Requirement id="1"> El sistema debe permitir la generación de estadísticas de solicitudes.
  <Requirement id="1.1"> Calcular el total de las solicitudes atendidas entre fechas determinadas.
</Requirement>
  <Requirement id="1.2"> Calcular el total de solicitudes aceptadas/rechazadas.
</Requirement>
  <Requirement id="1.3"> Calcular el total de los servicios contratados/solicitados.
</Requirement>
</Requirement>
</Concern>
```

(b)

Figura 3: (a) Requisitos del asunto “*Servicios*”, (b) Requisitos asunto “*Estadísticas de Gestión*”

Figure 3: (a) Requirements of the “*Services*” concern, (b) Requirements of the “*Management statistics*” concern

Adicionalmente, los asuntos deben ser especificados en plantillas (*templates*) que permitan concretar su detalle. e.g. plantillas de *viewpoints* [3], casos de uso [4], entre otras. En este caso, nosotros seleccionamos una plantilla que permitiera preparar los asuntos para el

análisis de contribuciones o influencia de unos con otros, entonces hemos utilizado la plantilla presentada en [17]. Las Tablas 1 y 2 muestran la especificación para dos de los asuntos del caso de estudio.

Tabla 1. Especificación del asunto *Servicios*
Table 1. Specification of the “*Services*” concern

Nombre	Servicios
Fuente	Participantes, Conjunto inicial de requerimientos, Conocimiento del sistema.
Participantes	Secretaria, Coordinador, Técnico, Jefe
Descripción	En el sistema se ingresan los servicios que se deban atender, se asigna el personal encargado y se genera el reporte.
Clasificación	Funcional
Contribución	(+) Solicitudes, (+) Seguridad.
Prioridad	Muy importante
Asuntos requeridos	1. Solicitudes, y sus hijos

Tabla 2. Especificación del asunto “Soporte al sistema”
Table 2. Specification of “System Support” concern

Nombre	Soporte al sistema
Fuente	Conjunto de requerimientos iniciales, Usuarios del sistema
Participantes	Desarrolladores, Reparadores de Hardware, Usuarios del sistema.
Descripción	Los equipos y el sistema operativo sobre el que corre la aplicación del sistema de ayuda deben estar en óptimas condiciones para su operación. Cualquier cambio sobre el sistema debe ser conseguido a través del personal de soporte y no de los usuarios.
Clasificación	No-Funcional
Contribución	(+) Seguridad, (-) Disponibilidad
Prioridad	Media
Asuntos requeridos	<ninguno>

➤ Análisis de la actividad

La elicitación de requisitos es una actividad compleja que depende de las interacciones entre los participantes. Para el desarrollo de este caso de estudio, sólo contamos con los documentos de requisitos que habían sido obtenidos con anterioridad. En la tarea de agrupar los requisitos en asuntos se encontraron las siguientes dificultades: (a) algunos requisitos estaban descritos de forma inexacta o confusa, lo que ocasionó que fueran desechados o agrupados en asuntos diferentes de su función real; (b) algunos requisitos, si bien estaban descritos correctamente, estos eran agrupables en varios asuntos a la vez (superposición temprana); (c) otros requisitos eran no agrupables en ningún asunto, lo que significaba que faltaban asuntos por definir o se presentaban errores en la descripción del problema o en la elicitación del requisito. También es posible encontrar asuntos sin requisitos asociados; generalmente son requisitos del sistema que ellos mismos constituyen un asunto.

En general, esta actividad es de alto riesgo y requiere de mucho tiempo ya que es necesario llegar a un buen nivel de descripción de los requisitos. Es importante resaltar que los analistas deben procurar documentación complementaria que les permita disminuir factores de riesgo como la ambigüedad y la incompletitud de los requisitos. Algunas de estas podrían ser, catálogos de requisitos no

funcionales, catálogos de reglas de negocio y de asuntos transversales, entre otros [6, 16].

3.3 Identificación De Relaciones De Grano Grueso Entre Asuntos

Cuando un asunto tiene una descripción muy general o no tiene requisitos agrupados, no será fácil percibir la influencia de éste en otros asuntos. Para la identificación y análisis de relaciones de grano grueso, nosotros hemos tomado la fila *contribución* de la plantilla usada en las Tablas 1 y 2. De igual forma el analista puede hacer uso de las técnicas mencionadas en [12, 13].

Por ejemplo, en la Tabla 1 el asunto “*Servicios*” es influenciado positivamente por el asunto “*Solicitudes*”. Esto quiere decir que los requisitos del asunto “*Servicios*” se satisfacen si se cumplen los requisitos del asunto “*Solicitudes*”. De forma similar, podemos hacer la siguiente interpretación en la Tabla 2, el asunto “*Soporte al Sistema*” es influenciado positivamente por el asunto “*Seguridad*” porque si se satisfacen algunos de los requisitos del asunto “*Seguridad*” se garantiza que se cumplen algunos de los requisitos del asunto “*Soporte al Sistema*”. Por ejemplo, si se cumple el requisito 1 del asunto “*Seguridad*” (permitir el acceso al sistema de ayuda solamente a usuarios autorizados de acuerdo a roles establecidos), incluyendo su hijo (mostrar la información a los usuarios de acuerdo a los roles establecidos), se satisface el requisito 1 del asunto “*Soporte al Sistema*” (dar soporte al hardware y al sistema

operativo donde corre la aplicación del sistema de ayuda).

Por otro lado, el asunto “*Soporte al Sistema*” está influenciado negativamente por el asunto “*Disponibilidad*” porque si se satisfacen todos los requisitos de éste asunto (el sistema de ayuda debe estar disponible 24x7) se afecta el cumplimiento de todos los requisitos de “*Soporte al Sistema*”. Aunque ésta y la anterior actividad se asemejen a otras actividades de elicitación de requisitos bajo otros modelos, es importante ser consciente del aporte que este modelo hace para el tratamiento de asuntos y sus requisitos agrupados.

En la Tabla 3, se proyecta cada uno de los asuntos en una relación muchos a muchos. La Tabla se lee de fila a columna, por ejemplo: el asunto *Solicitudes* (So) influencia sobre los asuntos *Servicios* (Se), *Estadísticas de Gestión* (Est) y *Contratos* (Con).

Desde una vista de grano grueso, la influencia de un asunto en otro permite que los analistas tengan en cuenta situaciones importantes tales como: qué asuntos deberán ser tratados bajo reglas de composición, el nivel de complejidad del sistema, el nivel de trazado de los asuntos transversales candidatos y posibles conflictos entre los participantes.

Tabla 3. Matriz de asuntos relacionados

Table 3. Matrix relating concerns

	Se	So	Gu	Seg	Dis	Com	MA	Sop	Sext	Est	Con
Se										X	
So	X									X	X
Gu	X	X		X							
Seg							X	X			
Dis		X	X				X	X	X		
Com									X		
MA											
Sop				X	X						
Sext						X					X
Est											
Con										X	

(Se: Servicios, So: Solicitudes, Gu: Gestión de Usuarios, Seg: Seguridad, Dis: Disponibilidad, Com: Compatibilidad, MA: Multiacceso, Sop: Soporte, Sext: Sistemas Externos, Est: Estadísticas, Con: Contratos)

➤ Análisis de la actividad

Aunque identificar las relaciones entre asuntos de grano grueso requiere de un trabajo exhaustivo, en este proceso encontramos las siguientes utilidades para la elicitación de requisitos:

- *Tratar los requisitos funcionales y no funcionales bajo criterios similares de funcionalidad del sistema.* Es común dejar las tareas de elicitación de requisitos no funcionales a los arquitectos, pero realmente quien tiene las necesidades exactas sobre el sistema es el usuario. Por ejemplo, el requisito de seguridad para el caso de estudio, “*permitir el acceso al sistema de ayuda solamente a usuarios autorizadas de acuerdo a roles establecidos*”, no es una necesidad del desarrollador sino del usuario.

Posteriormente, el arquitecto identificará que atributo de calidad cruzará dicho requisito para transformarlo en un componente del sistema que podrán satisfacer dicha necesidad o en un lineamiento de arquitectura. Igualmente este tipo de requisitos proveerán a los desarrolladores criterios de pruebas muy importantes.

- *Verificar la calidad semántica de los requisitos cuando se está en la tarea de agruparlos.* Es una tarea extensa y dispendiosa, pero trae beneficios para la transformación de los requisitos en artefactos o componentes de software en etapas posteriores.
- *Establecer la influencia entre asuntos permite realizar control del trazado vertical (dependencia entre requisitos).* Esta tarea

proporciona características de fiabilidad, exactitud y permite disminuir la complejidad en los modelos de requisitos y casos de uso que determinen la arquitectura del sistema.

- *Identificar los asuntos transversales conduce al aseguramiento de una buena arquitectura del sistema.* El hallar solapamiento temprano tanto de requisitos funcionales como no funcionales, orienta la identificación de los asuntos transversales. Dos criterios para su identificación pueden ser: (a) asuntos con funcionalidad independiente que pueden contribuir con información adicional a otros asuntos; en este caso podríamos hablar de asuntos como “Seguridad” o “Estadísticas de Gestión”. (b) asuntos con funcionalidad solapada pero que no es posible identificarlos de forma independiente de los asuntos sobre los cuales contribuye (posible dependencia entre requisitos funcionales).

3.4 Especificación De Los Asuntos Usando Reglas De Composición

Una vez los asuntos han sido cruzados en la matriz (Tabla 3), procedemos a construir las reglas de composición (tejido). La composición es la actividad más compleja del modelo, pero hace el aporte más importante de la aproximación a la ingeniería de requisitos.

Para crear las reglas de composición nos basamos en el solapamiento temprano inicialmente detectado al nivel de requisitos y registrado al nivel de grano grueso en la Tabla 3. Esta parte del proceso corresponde a lo que se denomina intersección composicional. De acuerdo a la experiencia obtenida, la composición puede ser lograda de forma organizada y ágil a través de los siguientes pasos:

- Identificar de la matriz de asuntos relacionados, los asuntos fuentes (filas) y asuntos destino (columnas).
- Para cada uno de los requisitos del asunto fuente verificar la multiplicidad de la relación con los requisitos del asunto destino (ver Figura 4). Los requisitos del asunto

fuelle que tengan una multiplicidad de 1:n (uno a muchos) con requisitos del asunto destino se les asignará la restricción de acción y el operador correspondiente. Este es un caso de requisito diseminado (*scattering*) en otros requisitos, e.g. R_{A1} .

- Para cada uno de los requisitos del asunto destino verificar la multiplicidad de la relación con los requisitos del asunto fuente. Los requisitos del asunto destino que tengan una multiplicidad de 1:n con requisitos del asunto fuente se le asigna la restricción de acción y el operador correspondiente a cada relación o conjunto de relaciones. Este es un caso de requisito mezclado (*tangling*) en otros requisitos s , e.g. R_{B4} .
- Cuando algún requisito de un asunto destino participa de una multiplicidad n:m (muchos a muchos) se puede considerar que el asunto/requisito es transversal (candidato) al asunto/requisitos fuente, e.g. R_{B4} .

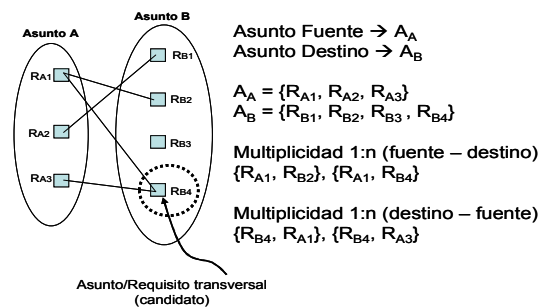


Figura 4: Relación de asuntos y requisitos
Figure 4: Relationship of the concerns and requirements

Por cada composición hallada, se debe determinar qué comportamiento preestablecido o adicional será involucrado. El uso de las restricciones de acción no está asociado a algunos asuntos específicos y son definidas genéricamente; estas ayudarán a que el analista entienda todas las posibles variables en el tejido de los asuntos en etapas posteriores. Para el uso de las restricciones de acción y los operadores hemos creado los siguientes criterios de uso:

- Si al realizar la composición es necesario que un asunto destino “adicione” comportamiento en otro conjunto de asuntos originados, se utilizan restricciones de acción

tales como *enforce*, *provide* y *affect*. El asunto que provee la información adicional en la composición bajo este tipo de restricciones de acción es un asunto transversal candidato.

- La acción *excluir* actúa como una excepción (*exception*) que puede ocurrir en un asunto fuente cuando un requisito agrupado en dicho asunto no será ejecutado en el momento de la composición. Ésta y las restricciones de *applied* y *ensure*, pueden orientar a la declaración de una acción más genérica llamada *exception* que permita manejar otro tipo de excepciones de tiempo, satisfacción de cumplimiento de la regla y de una composición, entre otras.
- Los operadores son usados según la temporalidad y las secuencias de ejecución de los asuntos en el momento del tejido. Estos pueden ser asociados a las restricciones de acción según sea la forma como pueden ser controladas las actividades de los procesos de negocio del problema tratado.

Para ilustrar el caso de estudio, en la Tabla 4 se muestran dos de las reglas de composición elaboradas para los asuntos “*Servicios*” y “*Disponibilidad*”. El buen entendimiento del uso

de operadores en estas reglas es fundamental para dar significado a la composición ya que estos son la restricción o condición de la regla.

➤ Análisis de la actividad

El análisis de la composición de requisitos no es un tema muy tratado por los modelos actuales de ingeniería de requisitos. No obstante, el uso de reglas de composición en etapas tempranas provee algunos de los siguientes beneficios:

- Diferenciar entre dependencia de requisitos y composición de requisitos.
- Identificar, con mayor certeza, en la especificación de los casos de uso, elementos tales como: precondiciones, postcondiciones, puntos de extensión y flujos alternos; además de relaciones tipo <<extend>> e <<include>>.
- Detectar y evitar futuros acoplamientos entre clases a partir de la correcta definición de las reglas de composición.
- Establecer relaciones de trazado que puedan conservar el sentido de la composición desde etapas tempranas.

Tabla 4. Reglas de composición para los asuntos **Servicios** y **Disponibilidad**
Table 4. Composition rule of the **Services** and **Availability** concerns

Especificación de la composición en XML (ArCaDe)	Interpretación en lenguaje natural
<pre><?xml version="1.0" ?> <Composition> <Requirement concern="Servicios" id="2" children="include"> <Constraint action="affect" operator=" on "> <Requirement concern="Estadísticas de Gestión" id="1.2" /> </Constraint> <Outcome action="fulfilled" /> </Requirement> </Composition></pre>	<p>El requisito 2 de Servicios (asignar técnicos a un servicio no regular) incluyendo (include) sus hijos (children) afecta (affect) el estado del requisito 1.2 del asunto Estadísticas de Gestión (total de los servicios internos solicitados, atendidos), con el resultado de que el requisito 2 de Servicios se da de manera exitosa (fulfilled).</p>
<pre><?xml version="1.0" ?> <Composition> <Requirement concern=" Disponibilidad " id="all" > <Constraint action="ensure" operator="during"> <Requirement concern="Sistemas Externos" id="all" /> <Requirement concern="Solicitudes" id="all" /> <Requirement concern="Gestion de Usuarios" id="all" /> <Requirement concern="Multiacceso" id="all" /> <Requirement concern="Soporte" id="all" /> </Constraint> <Outcome action="fulfilled" /> </Requirement> </Composition></pre>	<p>Todos (all) los requisitos del asunto Disponibilidad deben ser asegurados (ensure) durante (during) todos (all) los requisitos de los asuntos: Sistemas Externos, Solicitudes, Gestión de Usuarios, Multiacceso y Soporte; con el resultado de que todos los requisitos del asunto Disponibilidad se dan de manera exitosa (fulfilled).</p>

3.5 Manejo De Conflictos

El manejo de conflictos que propone la aproximación se basa en el análisis de compensaciones. Los pasos a seguir para la identificación y posterior administración de conflictos son:

1. Refinar la matriz de asuntos relacionados (Tabla 3), indicando en cada interrelación el tipo de contribución definitiva que un asunto ejerce sobre otro (ver Tabla 5). Por ejemplo, el asunto “*Servicios*” contribuye positivamente con el asunto “*Estadísticas de Gestión*” y el asunto “*Soporte al Sistema*” restringe (contribuye negativamente) el asunto “*Disponibilidad*”.
2. Doblar la matriz de contribución sobre su diagonal; ésta actividad permite observar el efecto acumulativo para las situaciones

donde dos asuntos influyen entre sí de manera recíproca (ver Tabla 6). Después de un análisis detallado, si al doblar la matriz se presenta más de un signo en la misma casilla (denotado en oscuro en la matriz) los signos de la contribución deben ser iguales. Si la correspondencia es de signos contrarios se debe revisar la especificación de los requisitos de los asuntos en conflicto, y se debe tomar alguna de las siguientes decisiones:

- Redactar de nuevo el (los) requisito(s) de uno o ambos asuntos.
- Subdividir el requisito existente, para restringir su generalidad.
- Verificar que el requisito sí pertenezca al asunto en cuestión.
- Eliminar o invalidar el requisito.

Tabla 5. Matriz de contribución
Table 5. Contribution matrix

	Se	So	Gu	Seg	Dis	Com	MA	Sop	Sext	Est	Con
Se										+	
So	+									+	+
Gu	+	+		+							
Seg							+	+			
Dis		+	+				+	-	+		
Com									+		
MA											
Sop				+	-						
Sext						+					+
Ind											
Con										+	

Tabla 6. Matriz de contribución de asuntos doblada sobre su diagonal
Table 6. The concern contribution matrix folded along its diagonal

Se	Se										
So	+	So									
Gu	+	+	Gu								
Seg			+	Seg							
Dis		+	+		Dis						
Com						Com					
MA				+	+		MA				
Sop				++	--			Sop			
Sext					+	+			Sext		
Ind	+	+								Est	
Con		+							+		Con

Como se observa en la matriz de contribución doblada, los asuntos “*Disponibilidad*” y “*Soporte al Sistema*” se influncian de forma

recíproca y de manera negativa. Esto puede significar que, cuando se satisfacen los requisitos de “*Soporte al Sistema*” se penaliza el

cumplimiento de algunos de los requisitos de “Disponibilidad”, y al aumentar la “Disponibilidad” del sistema el tiempo dedicado al soporte se ve afectado, disminuyendo la posibilidad del cumplimiento de los requisitos del asunto “Soporte al Sistema”. Lo contrario ocurre entre “Seguridad” y “Soporte al Sistema”, para éste caso se garantiza que los requisitos de “Soporte al Sistema” se cumplan; es decir, si se puede brindar “Seguridad” al sistema de la solución y al resto de sistemas que interactúan con éste, el soporte a dicho sistema es menos traumático y difícil.

El último paso para la negociación consiste en darle un peso a las contribuciones que tienen signo negativo en la matriz. La persistencia del conflicto debe ser solucionado según el nivel de

importancias que tiene para los participantes. El peso es un número real entre 0 y 1, y representa la prioridad del asunto fuente en relación con el asunto destino. Muy importante toma valores entre (0.8 - 1], importante (0.5 - 0.8], promedio (0.3 - 0.5], no tan importante (0.1 - 0.3], sin importancia [0, - 0.1] (ver Tabla 7).

En la matriz se observa que no se presenta ningún conflicto entre los asuntos, pues no hay más de una contribución negativa en diferentes filas en una misma columna. A la influencia bidireccional negativa entre los asuntos de “Disponibilidad” y “Soporte al Sistema” se le asignó un peso de 0.7 producto de la negociación entre los participantes del proyecto. Se consideró que ambos asuntos para el presente sistema se encuentran en la categoría de importante.

Tabla 7. Matriz doblada con ponderación en las contribuciones negativas
Table 7. Weighted (folded) negative contribution matrix

Se	Se											
So	+	Sol										
Gu	+	+	Gu									
Seg			+	Seg								
Dis		+	+		Dis							
Com						Com						
MA				+	+		MA					
Sop				++	0.7			Sop				
Sext					+	+			Sext			
Ind	+	+								Est		
Con		+							+		Con	

➤ Análisis de la actividad

- En esta parte, el nivel de refinamiento y entendimiento de los requisitos es mayor lo que permite detectar y corregir errores de la elicitación de requisitos de los pasos anteriores. Inclusive, es posible llegar a un conjunto de requisitos que puedan ser agrupados en un nuevo asunto de grano grueso.
- Esta buena práctica puede compensar conflictos proporcionados por situaciones tales como: carencia de definición o desconocimiento de los procesos de negocio que acompañan los requisitos elicitados, definición errada del alcance del proyecto, especificación de requisitos no controlada y ausencia del dimensionamiento de los riesgos, entre otras.

- Valorar la importancia del conflicto ayuda a los participantes a reevaluar la verdadera necesidad y prioridad de los requisitos solicitados.

3.5 Especificación De Las Dimensiones De Los Asuntos

Con el objetivo de estar cerca de una arquitectura lógica de alto nivel, el resultado final que arroja el modelo está orientado a determinar cuáles de los asuntos de grano grueso detectados se convertirán en, aspectos, decisiones o funciones (ver definiciones en la sección 2); además de determinar su influencia en posteriores fases del ciclo de vida. En la Tabla 8 se muestra la clasificación que se le a dado a algunos de los asuntos identificados.

Esta es una de las tareas más difíciles del modelo ya que la aproximación no provee criterios explícitos para determinar el tipo de mapeo y su influencia. Para lograr esto nosotros usamos los criterios creados en la actividad 3.4.

Tabla 8. Dimensiones de los asuntos
Table 8. Concern dimensions

Asunto	Influencia	Mapeo
Servicios	Especificación, Diseño, Evolución	Función
Solicitudes	Especificación, Diseño, Evolución	Función
Seguridad	Especificación, Diseño, Implementación	Aspecto
Disponibilidad	Cualquier etapa	Decisión
Multiacceso	Especificación, Diseño, Implementación	Aspecto
Soporte al Sistema	Cualquier etapa	Decisión
Sistemas Externos	Especificación, Diseño, Implementación	Aspecto
Indicadores de Gestión	Especificación, Diseño, Evolución	Función

➤ **Análisis de la actividad**

Proyectar la disposición de los artefactos y los elementos de modelo desde el análisis de los requisitos, traer beneficencia al nivel de la definición de la arquitectura. De esta forma, nosotros consideramos que es importante que en la fase influenciada sea posible tipificar los artefactos afectados; por ejemplo, en la fase de diseño el middleware, los componentes, las relaciones, entre otros, son artefactos que pueden ser afectados por uno o varios asuntos.

También, la influencia y el mapeo podrán ser usados como vínculos de trazado con restricción identificada que permitirá controlar los asuntos y sus requisitos tanto hacia delante como hacia atrás en el ciclo de vida de desarrollo.

4. CONCLUSIONES Y TRABAJOS FUTURO

Una vez hemos realizado el aporte crítico en cada una de las actividades del modelo AORE,

presentamos algunas conclusiones generales y trabajos futuros que pueden orientar el uso de esta aproximación en otras técnicas de ingeniería de requisitos.

La capacidad de AORE resalta la identificación y análisis de asuntos/requisitos transversales. Esta práctica no es usual o al menos no evidente cuando se trabaja bajo procesos de desarrollo de software conocidos, como el proceso unificado (RUP – Rational Unified Process). Aunque al principio parece difícil y costosa su manipulación, el modelo provee características para: (a) disminuir la complejidad de desarrollo; (b) disminuir el riesgo en el diseño e implementación a partir de la solución de conflictos; (c) controlar y soportar de la trazabilidad de requisitos al poder inferir vínculos de trazado identificando su influencia y mapeo en posteriores artefactos o fases del ciclo de vida. Estas características abren la posibilidad de orientar el manejo de requisitos en nuevos dominios de la ingeniería de software, como MDA (Model-Driven Architecture) [21], ya que sería posible controlar la multiplicidad de interrelación de los asuntos/requisitos fuente y destino en modelos CIM (*Computation independent model*) y su posterior transformación en modelos PIM (*platform independent model*) y PSM (*platform specific model*).

El modelo hace un aporte muy importante en relación a la separación y composición de asuntos. RUP dirige la descomposición del problema hacia la funcionalidad (casos de uso) que debe ser entregada como producto de software. Por la orientación de los casos de uso, los analistas limitan su especificación a la descripción de requisitos funcionales. Comúnmente, atributos de calidad, (conocidos también como requisitos no funcionales), y reglas de negocio que han sido elicitados, y que podrían ser transversales a muchas funcionalidades, son tratados por los arquitectos en fases avanzadas del ciclo de vida.

Realizando una separación orientada por AORE, todo asunto del sistema podría tratarse de manera homogénea en un espacio multidimensional de concerns. Por ejemplo, Jacobson y Ng en [19],

identifican asuntos transversales procedentes de los requisitos no funcionales o reglas de negocio representados en casos de uso extendidos. Sousa et al. [20], presentan casos de uso y relaciones estereotipadas para representarlos y pueda darse una separación plena de los asuntos que conciernen a muchos otros sin afectar las funcionalidades básicas.

La composición en RUP está dirigida a la agrupación de elementos de modelo (casos de uso y otros) en paquetes que determinen la arquitectura del sistema y guíen las iteraciones que permitirán la planificación y estimación del costo de sus productos. La elaboración de reglas de composición orientada por AORE puede ser obtenida desde la especificación de cada caso de uso. Es decir, elementos tales como precondiciones, postcondiciones y puntos de extensión podrían proveer las restricciones de acción, los operadores y las acciones que deberán ejecutarse una vez se efectue la composición con los flujos básicos, alternos y subflujos. Así, desde etapas tempranas, los arquitectos tendrán elementos para proyectar la construcción o uso de componentes y servicios en una arquitectura de diseño y su posterior implementación.

Como trabajo futuro se desarrollarán más casos de estudio bajo éste modelo que permitan: generar criterios de uso de restricciones de acción, operadores, influencias, correlaciones, entre otras características. Además consideramos importante formalizar los criterios de agrupación de requisitos, aplicación de reglas y manejo de conflictos.

REFERENCIAS

- [1] DIJKSTRA, E.W. *A Discipline of programming*, Prentice Hall, Englewood Cliffs, NJ, 1976.
- [2] PARNAS, D. *On the Criteria To Be Used in Decomposing Systems into Modules*. *Communications of the ACM*, 1053-1058, 1972.
- [3] SOMMERVILLE, I. *Software Engineering*, 7 ed: Addison-Wesley, 2004.
- [4] JACOBSON, I., CHIRSTERSON, M., JONSSON P. and Overgaard G., *Object-Oriented Software Engineering: A Use Case Driven Approach*, Addison-Wesley, 1992.
- [5] LAMSWEERDE, A. *Goal-Oriented Requirements Engineering: A Guided Tour*, presented at 5th IEEE International Symposium on Requirements Engineering, 2001.
- [6] CHUNG, L., NIXON, B., YU E. and MYLOPOULOS J. *Non-Functional Requirements in Software Engineering*: Kluwer Academic Publishers, 2000.
- [7] LAMSWEERDE, A.V. *Goal-Oriented Requirements Engineering: A Roundtrip from Research to Practice*, presented at Requirements Engineering (RE 2004), Kyoto, Japan, 2004.
- [8] KICZALES, G. HILSDALE, E., HUGUNIN, J., KERSTEN, M., PALM, J. and GRISWOLD W. G. *Getting Started with AspectJ*. *Comm. Of the ACM*, v. 44, n. 10, 59-65, 2001.
- [9] TARR, P., OSSHER, H., SUTTON, S.M. Jr., *N Degrees of Separation: Multidimensional Separation of Concerns 21st Int. Conf. On Software Eng.* ACM, 107-119, 1999.
- [10] CLARKE, S., BANIASSAD, E., *Aspect-Oriented Analysis and Design. The Theme Approach*. Addison-Wesley, Object Technology Series, 2005.
- [11] RASHID, A., MOREIRA, A. and ARAÚJO, J. *Modularisation and Composition of Aspectual Requirements*. *AOSD, ACM*, 11-20, 2003.
- [12] MOREIRA A., ARAUJO J., and RASHID A., "Multi-Dimensional Separation of Concerns in Requirements Engineering," presented at RE'05 Conference (RE 05), France, 2005.
- [13] MOREIRA, A., ARAÚJO, J. and RASHID A. *A Concern-Oriented Requirements Engineering Model*. *CAISE, LNCS Vol.3520*, 293-308, 2005.

- [14] CHITCHYAN, R., RASHID, A., SAWYER, P., BAKKER, J., PINTO, M., GARCIA, A., TEKINERDOGAN, B., CLARKE, S., JACKSON, A., "Survey of Aspect-Oriented Analysis and Design Approaches", AOSD-Europe-ULANC-9, AOSD-EUROPE network of excellence. May 2005.
- [15] PINEDA, O. Definition, Analysis, and Design for General Services Software at UNAL Medellín branch. Thesis Degree National University of Colombia. 2004.
- [16] Open Process Framework Repository Organization (OPFRO). www.opfro.org.
- [17] BRITO, I.S., MOREIRA, A. Integrating the NFR framework in a RE model". EA 2004: AORE and Architecture Design, workshop 3rd Inter. Conf. on Aspect-Oriented Software Development, Lancaster, UK, 22-26 2004.
- [18] SHMUEL, K., AWAIS, R., From Aspectual Requirements to Proof Obligations for Aspect-Oriented Systems. RE 2004: 48-57.
- [19] JACOBSON I., and NG, P.W., "*Aspect-Oriented Software Development with Use Cases*", Addison Wesley Professional, 2005.
- [20] SOUSA, G., SOARES, S., BORBA P., and Castro J., "Separation of Crosscutting Concerns from Requirements to Design: Adapting the Use Case Driven Approach", EA'04: Aspect-Oriented Requirements Engineering and Architecture Design. Workshop at AOSD 2004, pages 93-102. 2004, Lancaster UK.
- [21] MDA Guide Version 1.0.1. Copyright © 2003 OMG. www.omg.org.