

# Construcción de una Taxonomía de Componentes COTS Orientados a la Gestión de Requisitos<sup>‡</sup>

Claudia P. Ayala<sup>§</sup>, Pere Botella, Xavier Franch  
UPC-Campus Nord, c/Jordi Girona 1-3, 08034  
Barcelona (Catalunya, Spain)  
{cayala ,botella,franch}@lsi.upc.es  
<http://www.lsi.upc.es/~gessi>

**Resumen.** Este artículo presenta un procedimiento detallado para la construcción de una taxonomía de componentes COTS en el área de Ingeniería de Requisitos (IR). La taxonomía e información resultante aporta beneficios trascendentales para la selección de herramientas y sistemas que asistan a los actores involucrados en el proceso de IR a simplificar sus diversas labores. La taxonomía es construida mediante un enfoque orientado a metas inspirado en GBRAM (*Goal-Based Requirements Analysis Method*), llamado GBTCM (*Goal-Based Taxonomy Construction Method*), que nos proporciona una guía general para el análisis de las fuentes de información y modelado de requisitos y dominios; así como la recopilación y organización del conocimiento relativo a cualquier dominio del mercado de COTS. GBTCM pretende impulsar la reutilización de requisitos y estándares de Ingeniería de Software, con la finalidad de dar soporte en diferentes procesos de selección e integración de componentes.

**Palabras Clave:** COTS, Taxonomía, Meta, GBTCM, Ingeniería de Requisitos

## 1. Introducción

El uso de componentes comerciales o COTS (*Commercial Off-The-Shelf components*, o simplemente COTS) ha crecido considerablemente y previsiblemente seguirá haciéndolo [1,2]. El proceso de desarrollo de sistemas a partir de COTS es actualmente una necesidad desde el punto de vista económico y estratégico en una amplia variedad de áreas de aplicación. Como resultado, existe una enorme cantidad de COTS asequibles en el mercado, lo que origina un nuevo problema: ¿cómo organizar el conocimiento acerca de esos COTS de modo que el proceso de selección, adquisición e integración de los mismos sea una tarea factible?

---

<sup>‡</sup> Este trabajo se realiza en el marco del proyecto CICYT TIC2001-2165.

<sup>§</sup> Sus estudios doctorales son financiados por el Consejo Nacional de Ciencia y Tecnología (CONACYT, México).

En [3] propusimos el uso de taxonomías como forma de organizar el mercado de COTS (ver fig. 1). La idea general es la construcción de un árbol cuyas hojas representan dominios de COTS; un dominio engloba un grupo de funcionalidades significativas (p.e. el dominio de herramientas anti-virus o sistemas servidores de mails). Los dominios están agrupados en categorías (p.e. la categoría de sistemas de infraestructura para comunicaciones o paquetes financieros) y éstas a su vez pueden agruparse en otras categorías. El medio de clasificación entre diferentes categorías y dominios son los *atributos de caracterización* [4] y además se adjuntan preguntas y respuestas a esos atributos para facilitar la amigabilidad y navegabilidad en la taxonomía. Por otro lado, las dependencias entre dominios que pertenecen a la taxonomía también son representadas e incluidas en la jerarquía (p.e., los sistemas servidores de mails dependen de las herramientas anti-virus para favorecer la integridad de los datos). Como aspecto adicional, se añaden modelos de calidad a los nodos en la taxonomía para describir los factores de calidad que son de interés para una categoría o dominio en particular.

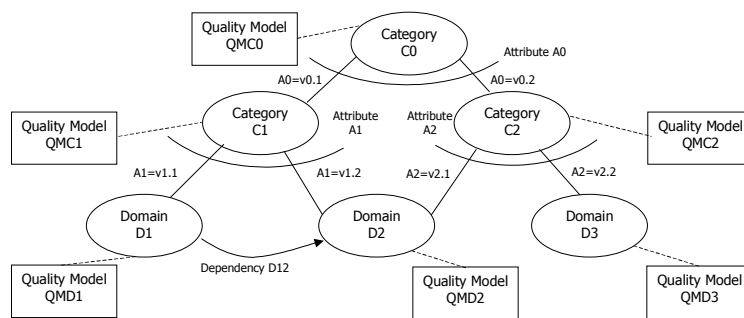


Fig. 1. Elementos fundamentales de la taxonomía de COTS

Subsecuentemente, en [5] expusimos la idea general de un enfoque basado en metas para generalizar, formalizar, mejorar y clarificar el proceso de identificación y evaluación de atributos de caracterización. Analizamos específicamente GBRAM (*Goal-Based Requirements Analysis Method*), propuesto por Annie I. Antón en el campo del análisis de requisitos de software [6].

En [7], partimos de la idea general de las actividades de GBRAM y lo adecuamos al ámbito de COTS, obteniendo el método GBTCM (*Goal-Based Taxonomy Construction Method* -Método de Construcción de Taxonomías basado en Metas-) que nos proporciona una guía para el análisis de las fuentes de información y el modelado de requisitos y dominios para la construcción de taxonomías; así como la recopilación y organización del conocimiento relativo a COTS. Hacemos énfasis en que la noción de meta usada está basada en el contexto de ingeniería de requisitos [8,9]. Las metas son el razonamiento para identificar los atributos de caracterización y por consiguiente las categorías y dominios de COTS. En general, las metas son más estables respecto a los cambios, y su refinamiento provee un mecanismo natural para estructurar y explorar varias alternativas en el mercado de COTS. Por otro lado,

GBTCM contribuye y fomenta la reutilización de los requisitos y del conocimiento en diferentes procesos de selección e integración.

El objetivo de este artículo consiste en describir la aplicación del GBTCM a un segmento particular del mercado de COTS: los sistemas y herramientas de ayuda a las variadas actividades relacionadas con la fase de Ingeniería de Requisitos de Software, obteniendo como resultado una taxonomía y organización del conocimiento en el área.

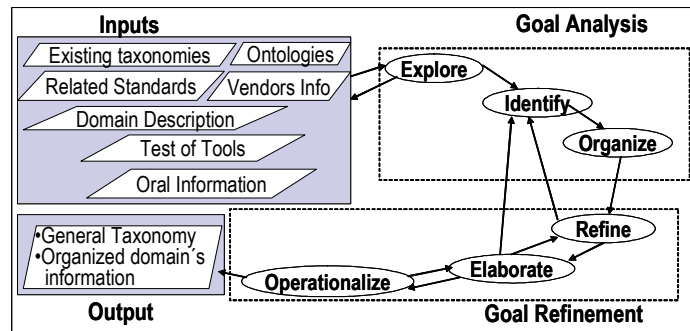


Fig. 2. Actividades generales de GBTCM

## 2. GBTCM: Construcción de Taxonomías de COTS

En la Fig. 2 mostramos las actividades (óvalos) y artefactos (rectángulos inclinados) de GBTCM. Las dos fases de alto nivel del método son: *Análisis de Metas* (referente a la exploración de las fuentes de información disponibles para la identificación y posteriormente la organización y clasificación de metas) y *Refinamiento de Metas* (referente a la evolución de las metas desde el momento que son identificadas hasta que son traducidas a requerimientos operacionales).

Las actividades son llevadas a cabo desde el punto de vista de *atributos de caracterización* y no como requisitos operacionales de un sistema de software. Las actividades generales son: *Exploración* (se refiere al análisis de la información disponible), *Identificación* (indica la extracción de metas aplicando heurísticas), *Organización* (involucra la clasificación y organización de metas de acuerdo a relaciones de dependencia), *Refinamiento* (se refiere al refinamiento o poda del conjunto de metas), *Elaboración* (se refiere a un proceso de mayor refinamiento en el cuál se analizan las metas considerando posibles obstáculos y la construcción de escenarios para descubrir metas y requisitos que de otra forma hubieran sido soslayados) y *Operacionalización* (es el proceso de traducir las metas en requisitos para producir finalmente la taxonomía de COTS). La tabla 1 muestra las entradas y salidas de cada actividad.

Las fuentes de información (inputs), son la base para la obtención de metas. El resultado final de GBTCM no es simplemente una taxonomía de COTS, sino un conjunto de información o conocimiento detallado (repositorio) que contiene además las actividades del área de COTS analizada (expresada en forma de metas) y las relaciones de dependencia entre ellas y entre dominios de COTS (expresada a través de artefactos y modelos que facilitan el mantenimiento ante la evolución y cambios en

el dominio). Esta información es la base para la construcción de una taxonomía general del dominio.

Básicamente, GBTCM guía para recopilar, gestionar y generalizar la información referente a un dominio (tomando en cuenta especialmente las taxonomías existentes, estándares en el área e información de vendedores de productos), generando artefactos y modelos auxiliares que permitan mantener la consistencia y evolución de un repositorio de conocimiento en el área de COTS, para su uso repetido en diferentes organizaciones y éstas puedan a partir de la taxonomía general y la información del repositorio, crear la taxonomía más adecuada a sus procesos.

Actividad	Salidas
Exploración	Fuentes de Información Cualificadas; Algunas Metas
Identificación	Conjunto de Metas, <i>Stakeholders</i> y Agentes; Modelos y Artefactos Auxiliares
Organización	Correspondencia de Metas entre las diferentes fuentes de Información; Relaciones de Dependencia entre Metas; Jerarquía de Metas
Refinamiento	Conjunto Refinado de Metas; Dependencias ente metas
Elaboración	Escenarios; Restricciones
Operacionalización	Estructura Jerárquica de Metas; Información Asociada y Modelos y Artefactos; Atributos de Caracterización para la Construcción de la Taxonomía del Dominio

**Tabla 1.** Actividades del GBTCM y sus salidas

### 3. Una Taxonomía para el Área de Ingeniería de Requisitos

Es ampliamente reconocido que dentro de la industria del software, los proyectos pasan a ser extremadamente vulnerables cuando las actividades de Ingeniería de Requisitos de Software (IR) son desarrolladas inadecuadamente [10]. En este contexto, la tecnología de COTS ayuda a mejorar la eficiencia de las actividades desarrolladas en el área auxiliando a los actores relacionados con la IR a facilitar y simplificar su trabajo. Nuestro objetivo en esta sección es aplicar GBTCM al área de la IR y como resultado, proponer una taxonomía general de COTS para la selección e integración de estas herramientas y componentes.

#### 3.1 Fuentes de Información

Cómo es lógico al hablar de un tópico como éste, las fuentes de información existentes son muchas y muy variadas. La tabla 2 resume varias de las fuentes que hemos considerado más relevantes para este caso de estudio [7], y detallamos de manera general el tipo de información contenida que ha sido analizada. Se observa que la mayoría de la información es textual, disponible en forma impresa o en la web, firmada por diferentes organizaciones o personas, sin embargo, información tal como opiniones de expertos y demostraciones de herramientas también es contemplada de forma secundaria.

Es importante destacar que se pretende representar información que no depende de un proyecto o sistema software concreto, sino hacer un recopilatorio general con toda la información posible de esta área de conocimiento, que permita su posterior adecuación y aplicabilidad en diferentes organizaciones.

### 3.2 Análisis de las Fuentes e Identificación de Metas y Objetivos

El uso de una u otra fuente de información es determinada por varias cualidades, entre las cuales podemos mencionar: confiabilidad de la información, disponibilidad de la fuente, costo de adquisición, tiempo disponible, objetivos cubiertos y tiempo necesario para procesar la información contenida. Estas cualidades dependen de 3 factores: tipo de la fuente de información; organización o personas que distribuyen la información; y la información en sí misma.

Tipo de Fuente	Organismo	Información Contenida	Comentarios
<b>Taxonomías Existentes</b>	INCOSE	Clasificación de Herramientas de Ingeniería de Software	Sección disponible sin costo
<b>Estándares Relacionados</b>	SWEBOK	Principales áreas de IR; tipos de <i>Stakeholders</i> relacionados con las áreas	Disponible sin costo, ampliamente aceptada
	IEEE std 830-1998	Actividades de Software relacionadas con IR	Se necesita suscripción/pago
	IEEE/EIA 12207.1-1997		
	ISO/IEC 12207		
<b>Información de Vendedores</b>	IBM-Rational	Capacidades de Productos y tendencias	Descripción exhaustiva de productos
	ComponentSource	Capacidades de Productos y tendencias enfocada a plataformas	Disponible sin costo, ampliamente aceptada
<b>Herramientas</b>	RequisitePro	Capacidades reales de Herramientas de Administración de Requisitos	Incluida en IBM-Rational
	IRqA		Herramientas usadas en nuestros proyectos
	EasyWinWin	Capacidades de una herramienta para la negociación de requisitos	Algunos tutoriales y contactos con los autores
<b>Sitios Académicos</b>	eCOTS	Información sobre tipos de componentes y evaluaciones de los mismos	Disponible sin costo
<b>Items Científicos</b>	Conferencias relacionadas con IR e IS	Estado del arte	Se necesita suscripción/pago
	Libros sobre IR e IS	Conocimiento consolidado	
<b>Revistas</b>	Requirements Engineering Journal	Tendencias y estado del arte	
<b>WebSites</b>	Volere	RE resources	Disponible sin costo
<b>Técnica</b>	INCOSE	Tendencias y conceptos en IR	Se necesita suscripción/pago
	Gartner		
<b>Experiencias Propias</b>	Universidad Politécnica de Catalunya	Uso de herramientas de IR en proyectos reales	Gestión de requisitos acorde a CMM-2

**Tabla 2.** Principales fuentes de información utilizadas

Es recomendable como buena práctica el basar mayormente el análisis de metas de alto nivel en las más sólida y confiable de las fuentes y subsecuentemente extraer sub-metas de las otras fuentes.

Por ello, debido a la naturaleza de estándar de SWEBOK en el área de IR, iniciamos el análisis a partir de éste (aun considerando que SWEBOK no es orientado a herramientas, contrariamente a otras fuentes) para obtener las metas de alto nivel.

Por ejemplo, considerando el siguiente fragmento en lenguaje natural contenido en SWEBOK: “The next topics breakdowns for RE discipline are generally accepted in that they cover areas typically in texts and standards: activities such as Requirements Engineering Process, Requirements Elicitation, and Requirements Analysis, along

with Requirements engineering-specific descriptions. Hence, we identify Requirements Validation and Requirements Management as separate topics”. Al examinar la sentencia a través de heurísticas o cuestionamientos guía como: “¿Qué metas ejemplifica este fragmento?”, algunas metas parecen evidentes.

Presentamos algunas de las metas de alto nivel en la primera columna de la tabla 3. Estas metas se irán descomponiendo en sub-metas al ir analizando las otras fuentes de información y posteriormente aplicando técnicas de refinamiento, como advertiremos en las subsecciones siguientes.

Metas	Agentes	Stakeholders
M1: Proceso de Ingeniería de Software Definido	(IR)	IR, AP,RAC
M2: Elicitación de Requisitos Realizada	IR	IR, <i>Stakeholders</i>
M3: Análisis de Requisitos Realizado	IR	IR, <i>Stakeholders</i>
M4: Especificación de Requisitos Hecha	IR	IR, usuarios/clientes, RAC
M5: Validación de Requisitos Realizada	IR	IR, usuarios/clientes, <i>Tester</i>
M6: Administración de Requisitos Controlada	IR	IR, RCS
M6.1: Administración de Cambios en Requisitos Controlada.	IR	IR
M6.2: Atributos de Requisitos Definidos	IR	IR, RCS
M6.3: Trazo de Requisitos Controlada	IR	IR, RCS

**Tabla 3.** Algunas metas obtenidas del análisis de SWEBOK

### 3.3 Identificación de *Stakeholders* y Agentes

El objetivo es determinar cuáles son los *stakeholders* involucrados en el logro de metas, una vez que las metas y *stakeholders* son especificados, las metas pueden ser asignadas a su(s) agente(s) responsable(s).

Un *stakeholder* es cualquier ente afectado o que puede afectar el logro o prevención de una meta particular. Múltiples *stakeholders* pueden estar asociados con una meta.

Los agentes son los responsables del cumplimiento y/o satisfacción de metas.

La identificación de *stakeholders* y agentes es crucial para entender el dominio e identificar fuentes de información adicionales que se deben considerar, por ejemplo personas que deben ser entrevistadas.

Los *stakeholders* para cada meta son determinados aplicando cuestionamientos heurísticos generales tales como: “¿quién o qué manifiesta un interés en esta meta?” y “¿quién o qué es afectado por la satisfacción o no de esta meta?”. Para identificar agentes: “¿quién o qué agente debe o podría ser responsable de esta meta?”.

Algunos *stakeholders* identificados son (ver tabla 3): Ingeniero de Requisitos (IR), Administrador del Proyecto (AP), Responsable de Aseguramiento de la Calidad (RAC), Responsable de la Configuración de Software (RCS), Testers, Usuarios Finales, Clientes y *Stakeholders* No Técnicos tales como regulaciones, analistas del mercado, desarrolladores de sistemas (SNT). El único agente en este caso es el Ingeniero de Requisitos.

Algunas relaciones y dependencias entre los *stakeholders* son presentadas en el modelo i\* SD de la fig. 3.

### 3.4 Modelos y Artefactos Auxiliares

GBTM considera esencial la generación de algunos artefactos y modelos a partir de las fuentes de información, para comprender, manejar, formalizar y sobre todo garantizar la consistencia e integridad de la información respecto a cambios y evolución. Por ello, sugiere la creación de glosarios para homogeneizar términos, la creación de diagramas de clases de UML [23] para representar el modelo conceptual del dominio y definir la ontología base [24]; y como parte fundamental del método, se requiere la construcción de modelos orientados a metas: específicamente i\* como notación [25] (sin embargo otras opciones son válidas también). Lógicamente todos esos artefactos y modelos deben concordar. Por ejemplo, los términos del glosario, de los diagramas i\* y los atributos de UML deben tener el mismo nombre.

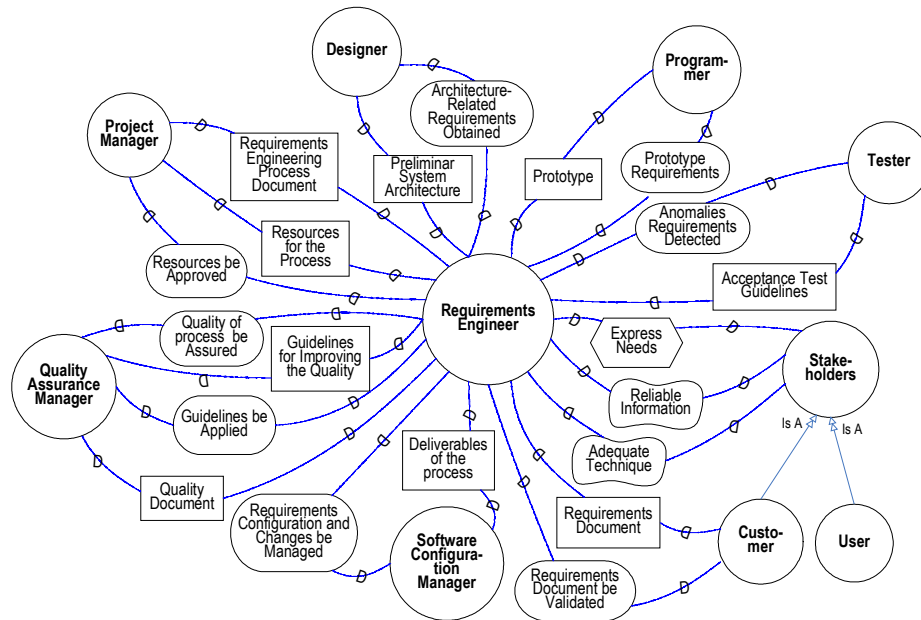


Fig. 3. Extracto del modelo i\* SD que muestra las relaciones entre actores del sistema

Además, como técnica de refinamiento de metas, GMTM sugiere la construcción de escenarios, que se refieren a la recreación de las distintas situaciones y condiciones en las que ejecutamos o podríamos ejecutar una meta, facilitando así la identificación y refinamiento de metas en circunstancias especiales o extraordinarias.

La aplicabilidad y precisión con que sean construidos los escenarios dependerá de las características del dominio de COTS que se analice y el grado de detalle que se persiga; es decir, el análisis y refinamiento de metas de un dominio con alto nivel de criticidad requerirá un análisis más exhaustivo y la construcción de escenarios muy detallados.

En este caso de estudio, elaboramos glosarios de términos que han sido sumamente útiles para uniformizar la información, facilitar la comprensión y evitar confusiones.

Por ejemplo, el mismo concepto fue nombrado como “captura” y “extracción” en fuentes distintas, por lo que reconciamos el término como “extracción” y lo definimos claramente en el glosario del dominio.

En algunos casos elaboramos ciertos escenarios con el fin de ser exhaustivos en el refinamiento de metas, y a la vez abarcar el mayor número de actividades que suelen llevarse a cabo en la fase de IR en los proyectos de desarrollo de software. Por ejemplo, detectamos que para el logro de la meta *Análisis de Requisitos Realizado*, en algunos casos se precisa la identificación de los subsistemas y componentes que serán los responsables de la satisfacción de los requisitos, por lo que consideramos e identificamos la meta *Diseño Preliminar de la Arquitectura Realizado* como una sub-meta, que sin la ayuda de la construcción de escenarios hubiera sido a primera instancia pasada por alto. La técnica de construcción de escenarios nos resultó muy útil para descubrir, reconciliar y validar la completitud de la meta y los conflictos, así como para facilitar la comunicación con los *stakeholders*.

### **3.5 Uso y Creación de Heurísticas**

Las heurísticas nos auxilian proporcionando una guía descriptiva para el manejo de varios niveles de detalle en la información disponible. Las secciones 3.2 y 3.3 muestran ejemplos de aplicación de heurísticas que guían la búsqueda de información concreta mediante preguntas específicas, éstas heurísticas son generales y no dependen de ningún dominio particular. Sin embargo, un objetivo crucial de GBTCM es la reutilización del conocimiento adquirido en los diferentes casos de estudio; por ello, se contempla la creación de nuevas heurísticas específicas del dominio analizado para su uso posterior en diversos casos de estudio, facilitando así el análisis de la información. Este mecanismo evolutivo facilita el crecimiento y mantenimiento de la información sobre el dominio. El caso de estudio presentado en este artículo, se ha enfocado inicialmente a la aplicación general del método, por lo que no presentamos heurísticas específicas para el dominio de IR, sin embargo como trabajo futuro se contempla su creación a partir del desarrollo de otros casos particulares que aporten las bases para su fundamento.

### **3.6 Correspondencia y Organización de Metas**

Una vez analizadas todas las fuentes, organizamos la información obtenida estableciendo una correspondencia de metas entre las fuentes. El resultado de este proceso es representado por medio de tablas. La tabla 4 es un extracto de ese proceso, en ella podemos observar la correspondencia de metas entre algunas fuentes de información; como ejemplo mostramos la correspondencia con los nodos de la taxonomía de INCOSE. Al final de este proceso, obtenemos un conjunto de metas más completo (que más adelante será refinado).

Partiendo de este conjunto de metas, se especifican dependencias jerárquicas (por medio de tablas, referidas como topografía de metas). Luego del proceso de refinamiento se especifican dependencias entre metas (por medio de diagramas i\*), ya sean dependencias entre metas del mismo dominio o con otros dominios tratados previamente o ya sea que dependan de otra meta, agente o recurso para su



cumplimiento o habilitación; esto es, la representación explícita de dependencias potenciales entre dominios de COTS, asegurando la trazabilidad del impacto de los cambios entre dominios. Como resultado, tenemos una representación dual de los modelos de metas, uno más técnico (Fig 4, izquierda) y otro más comprensible visualmente, fácil de indexar y leer (Fig. 4, derecha).

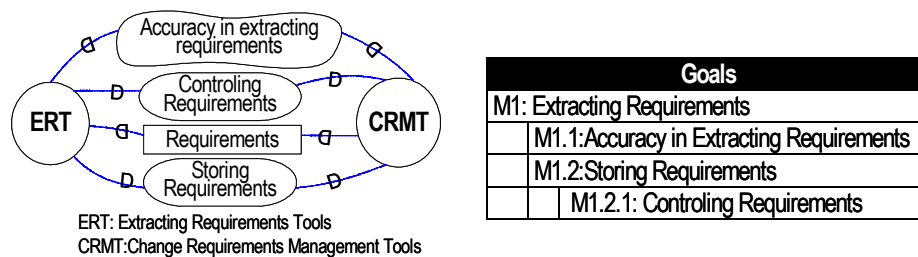
Metas		Herramientas	Categoría de la Taxonomía de INCOSE
M2: Elicitación de Requisitos Realizada			
M2.1:	Fuentes de Requisitos Definidas y Analizadas		
M2.2:Técnicas de Elicitación Elegidas			
M2.2.1 :	Extracción de Requisitos	Sí	Requirements Engineering/Requirements Management/Requirements Capture&Identification/Tools For Elicitation Of Requirements Design Domain
	M2.2.1.1: Entrevistas	Sí	
	M2.2.1.2: Escenarios	Sí	

**Tabla 4.** Un extracto de organización de metas

### 3.7 Refinamiento del Conjunto de Metas

GBTM se basa en la construcción y organización del conocimiento a partir de refinamientos sucesivos. Así, una vez obtenido un conjunto de metas más completo y especificado sus dependencias jerárquicas, las metas son analizadas para su refinamiento; para ello usamos 3 mecanismos: eliminación de metas duplicadas, refinamiento de metas basado en entidades del sistema y consolidación de metas que, aunque se llaman diferente, hagan referencia al mismo hecho.

El uso de glosarios y diagramas de clases construidos durante todo el proceso, fue crucial en esta etapa de reconciliación de metas. Además, en esta fase, se construyeron escenarios sencillos para elevar el grado de detalle con que son refinadas las metas y lograr reflejar la mayor cantidad de actividades llevadas a cabo en la mayoría de los proyectos de IR.



**Fig. 4.** Ejemplo del modelo i\* SD relacionado con las herramientas de IR (izquierda). Organización de metas por medio de tablas jerárquicas (derecha)

### 3.8 Operacionalización de Metas

Finalmente, la información de las metas obtenidas debe ser operacionalizada (es decir relacionarla con acciones). Esto se hace consolidando toda la información recopilada

de la meta aplicando el *Inquiry-Cycle* [26], que consiste en una serie de preguntas y respuestas diseñadas para ayudar a comprender y organizar dónde y cuándo debe aparecer la información. Hemos aplicado este mecanismo partiendo de la jerarquía de metas y los diagramas *i\** para obtener las preguntas-respuestas generales vinculadas a cada atributo de caracterización y para organizar apropiadamente la información resultante.

Nivel 2	Nivel 3	Pregunta	Nivel 4
1.2 Elicitation de Requisitos			
	1.2.1 Generación	¿Aplica técnicas de simulación para generar requisitos?	1.2.1.1 Simulation Tools
	1.2.1 Extracción		
		¿Aplica técnicas de entrevistas?	1.2.2.1 Interview Tools
		¿Aplica técnicas de escenarios?	1.2.2.2 Scenario Tools
		¿Aplica técnicas de prototipos?	1.2.2.3 Prototype Tools
		¿Aplica técnicas de reuniones facilitadas?	1.2.2.4 Facilitate Meetings Tools
		¿Aplica técnica de observación?	1.2.2.5 Observation Tools
		¿Aplica otras técnicas para la extracción de requisitos?	1.2.2.6 Other Extraction Techniques Tools

**Tabla 5.** Extracto de la taxonomía de IR

La tabla 5 muestra solo un extracto del resultado. Fundamentalmente, cada respuesta a una pregunta da lugar a un nivel (por ejemplo, la respuesta a la pregunta de la columna 3, da lugar al nivel 4 de la tabla). Por falta de espacio no mostramos las preguntas-respuestas relacionadas con cada nivel de la tabla, solo presentamos un pequeño esquema con el fin de dar una idea general de la taxonomía y la organización resultante.

#### 4. Conclusiones y Trabajo Futuro

En nuestra experiencia, podemos afirmar que aplicando GBTCM, logramos una alta probabilidad de éxito, en la construcción y organización del conocimiento sobre el dominio de IR, que es la base para la construcción de taxonomías de COTS.

Debemos destacar que: la representación de las relaciones entre dominios (es decir, el conocimiento preciso de las implicaciones de uso de un componente en particular) así como la organización de la información y los artefactos y modelos producidos, favorecen no sólo la reutilización del conocimiento conseguido, sino también su mantenimiento y su evolución, que es un aspecto básico a partir de la cual pretendemos que diversas organizaciones pueden basar sus casos particulares para construir la taxonomía que mejor se adapte a sus procesos [27].

En general, GBTCM representa una aproximación práctica de ayuda para elicitar, especificar, seleccionar e integrar componentes COTS, basado en la reutilización de requisitos y conocimientos, y en estándares de Ingeniería de Software.

El conocimiento y taxonomía obtenida apoya a los ingenieros de software en los procesos de selección e integración de componentes de IR, asegurando el retorno de inversión principalmente en tiempo y esfuerzo requerido.

Nuestro trabajo futuro se concentra en fortalecer y mejorar este caso de estudio, con miras entre otros aspectos a: formalizar un dominio de conocimiento de IR con heurísticas establecidas; y la construcción de un programa más exhaustivo de análisis de las fuentes de información, que permita tomar en cuenta más formalmente características de calidad de las fuentes para una mejor clasificación de acuerdo a su relevancia.

Asimismo, nos proponemos estudiar el uso de otras técnicas existentes para la construcción de taxonomías y jerarquías tales como *laddering* o redes neuronales para su eventual incorporación en GBTCM.

## References

- [1] Carney D., Long F., "What Do You Mean by COTS? Finally a Useful Answer". *IEEE Software*, 17 (2), March/April 2000.
- [2] Craig Meyers, B., Oberndorf, P., "Managing Software Acquisition", *SEI Series in Software Engineering*, 2002.
- [3] Carvallo, J.P., Franch, X., Quer, C., Torchiano, M., "Characterization of a Taxonomy for Business Applications and the Relationships Among Them", *Proceedings 3rd International Conference on COTS-Based Software Systems (ICCBSS)*, 2004, Lecture Notes in Computer Science Vol. 2959.
- [4] Morisio, M., Torchiano, M., "Definition and Classification of COTS: A Proposal", *Proceedings 1st. International Conference on COTS-Based Software Systems (ICCBSS)*, Orlando Florida, 2002.
- [5] Ayala, C.P., Botella, P., Franch, X., "Goal-Based Reasoned Construction of Taxonomies for the Selection of COTS Products", *Proceedings 8th Multi-Conference on Systemics, Cybernetics and Informatics (SCI)*, Orlando Florida, July 2004. ISBN:980-6560-13-2.
- [6] Antón, A.I., "Goal-Based Requirements Analysis", *Proceedings 2nd IEEE International Conference on Requirements Engineering (ICRE)*, 1996.
- [7] Ayala, C.P., Botella, P., Franch, X., "On Goal-Oriented COTS Taxonomies Construction", *Proceedings 4th International Conference on COTS-Based Software Systems (ICCBSS)*, Bilbao, Spain, Febrero 2005.
- [8] Mylopoulos, J., Chung, L., Yu, E., "From Object-Oriented to Goal-Oriented Requirements Analysis", *Communications ACM* 42(1), January 1999.
- [9] Van Lamsweerde, A., "Goal-Oriented Requirements Engineering: A Guided Tour", *Proceedings 5th IEEE International Symposium on Requirements Engineering (ISRE)*, 2001.
- [10] Standish Group Report <http://www.standishgroup.com>. *CHAOS Survey* 1994.
- [11] INCOSE. "Software Engineering Tools Taxonomy" <http://www.incose.org>
- [12] Guide to the Software Engineering Body of Knowledge, SWEBOK, [www.swebok.org](http://www.swebok.org)
- [13] International Standard IEEE Std 830-1998 IEEE, "Recommended Practice for Software Requirements Specifications", June 1998.
- [14] IEEE/EIA Guide. IEEE/EIA12207.1-1997, "Standard for Information Technology Software Life Cycle Processes", April 1998.
- [15] International Standard ISO/IEC 12207 "Software Life Cycle Processes", 1995.
- [16] ComponenSource <http://www.componentsource.com>
- [17] eCots. Software Components Open Directory Project. <http://ecots.org>

- [18] The Gartner Group, available on-line at <http://www4.gartner.com>
- [19] Infrastructure Software Market Definitions for Application Development. Gartner, Dataquest Guide. 4<sup>th</sup> June, 2003. <http://www.gartner.com>
- [20] Software Market Research Methodology and Definitions 2003-2004, Gartner Dataquest Guide, January, 2004. <http://www.gartner.com>
- [21] CBSE Net. "Application Domain Taxonomy". Available on-line (previous registration) at: [http://www.cbse.net/pls/CBSEnet/eco\\_ricerca\\_documenti.concept\\_search\\_frame](http://www.cbse.net/pls/CBSEnet/eco_ricerca_documenti.concept_search_frame)
- [22] International Data Corporation <http://www.idc.com>
- [23] UML 2.0 Specifications <http://www.uml.org/>, last accessed July 2004.
- [24] Carvalho, J.P., Franch, X., Quer, C., "Defining a Quality Model for Mail Servers", *Proceedings 2<sup>nd</sup> International Conference on COTS-Based Software Systems (ICCBSS)*, 2003, Lecture Notes in Computer Science 2580.
- [25] Yu, E., "Towards Modeling and Reasoning Support for Early-Phase Requirements Engineering", *Proceedings 3<sup>rd</sup> IEEE International Symposium on Requirements Engineering (ISRE)*, 1997.
- [26] Potts, C., Takahashi, K., Antón, A., "Inquiry-Based Requirements Analysis", *IEEE Software*, 11(2), March 1994.
- [27] Basili, V.R., Caldera, G., Rombach, H.D., "The Experience Factory", J.J Marciniak (ed), *Encyclopedia of Software Engineering, vol 1*. pp 469-476. John Wiley & Sons, 1994.