

Clasificación de Componentes OTS (Off-The-Shelf) para Sistemas de Información Geográficos

Proyecto de Investigación Área Ingeniería de Software
Unidad Académica Caleta Olivia
Universidad Nacional de la Patagonia Austral
Acceso Norte Ruta 3 – Caleta Olivia – Santa Cruz

Grupo de Investigación en Ingeniería de Software del Comahue (GIISCO)
<http://giisco.uncoma.edu.ar>
Departamento de Ciencias de la Computación
Universidad Nacional del Comahue

1. Motivación

El Desarrollo de Software Basado en Componentes (DSBC) [14, 15] procura la reducción del tiempo de desarrollo, de los costos y del esfuerzo, y también la mejora de la calidad del producto final debido a la reutilización de composiciones de componentes software ya desarrollados, probados y validados. Técnicas modernas de construcción sugieren reducir riesgos y maximizar calidad vía incorporación componentes OTS (off-the-shelf) pre-existentes; por lo tanto, el antiguo problema de programar nuevo software ha sido reemplazado por el de identificar y adaptar componentes. Igualmente, surge otro problema, ya que generalmente los desarrolladores gastan mucho tiempo y esfuerzo para encontrar los OTS que satisfacen la funcionalidad que pretenden implementar. Es sabido que resulta dificultoso identificar ciertos componentes en uno o varios repositorios.

Con respecto a sistemas de información con un dominio específico, como son los Sistemas de Información Geográficos (SIG), surgen las mismas inquietudes. Los componentes deben poseer la suficiente especificidad, en cuanto a la descripción de la tarea que realizan, de manera de provocar la menor cantidad de modificaciones y acortar así el tiempo de desarrollo. Las características particulares de los SIG provocan que las composiciones de componentes no puedan construirse sólo como simples piezas de un producto, sino como un conjunto de elementos pre-ensamblados en un marco de aplicación específico. Una documentación de alta calidad facilita que los estándares de diseño e implementación se reflejen en el contenido de la aplicación construida con componentes. Estándares aplicables a un dominio deberían facilitar el proceso de clasificación y modelado de la información (Ej. familia de estándares ISO 19100 [5, 6, 7, 8] para el caso de SIG).

2. Descripción del Problema

Según la definición de Szyperski [14], un componente es una unidad binaria de composición de aplicaciones software, que posee un conjunto de interfaces y un conjunto de requisitos. El mismo puede ser desarrollado, adquirido e incorporado al sistema y a su vez compuesto con otros componentes de forma independiente, en tiempo y espacio.

Las propuestas de procesos de DSBC se componen por diferentes etapas. Según [12] se clasifican en, *identificación, evaluación, selección, integración y actualización de componentes*. La Figura 1 presenta la propuesta del proceso de selección de COTS (Commercial Off-The-Shelf) presentado en [11]. Como podemos observar, el éxito del DSBC debe considerar dos aspectos: el punto de vista del desarrollador del componente (vendedor) y el punto de vista del desarrollador de la aplicación (comprador). Los desarrolladores de componentes desarrollan el artefacto final, y luego de testear su funcionalidad, lo almacenan en un repositorio para ser usado luego. Junto con el componente, se debe almacenar cierta información que luego será necesaria para encontrar, comprender y decidir sobre la utilización del mismo. Este “*Proceso de publicación*” está formado por tres pasos: (1) la

clasificación del componente, (2) la documentación del componente y (3) el almacenamiento de toda la información en el repositorio.

Por su parte, los desarrolladores de aplicaciones basados en COTS, en base a los requerimientos del sistema, analizan los componentes disponibles, en el “*Proceso de selección*”, formado por otros 3 pasos: (1) la búsqueda de la funcionalidad y otros atributos de calidad deseada, (2) el entendimiento técnico de los componentes candidatos encontrados y (3) la toma de decisiones sobre su utilización.

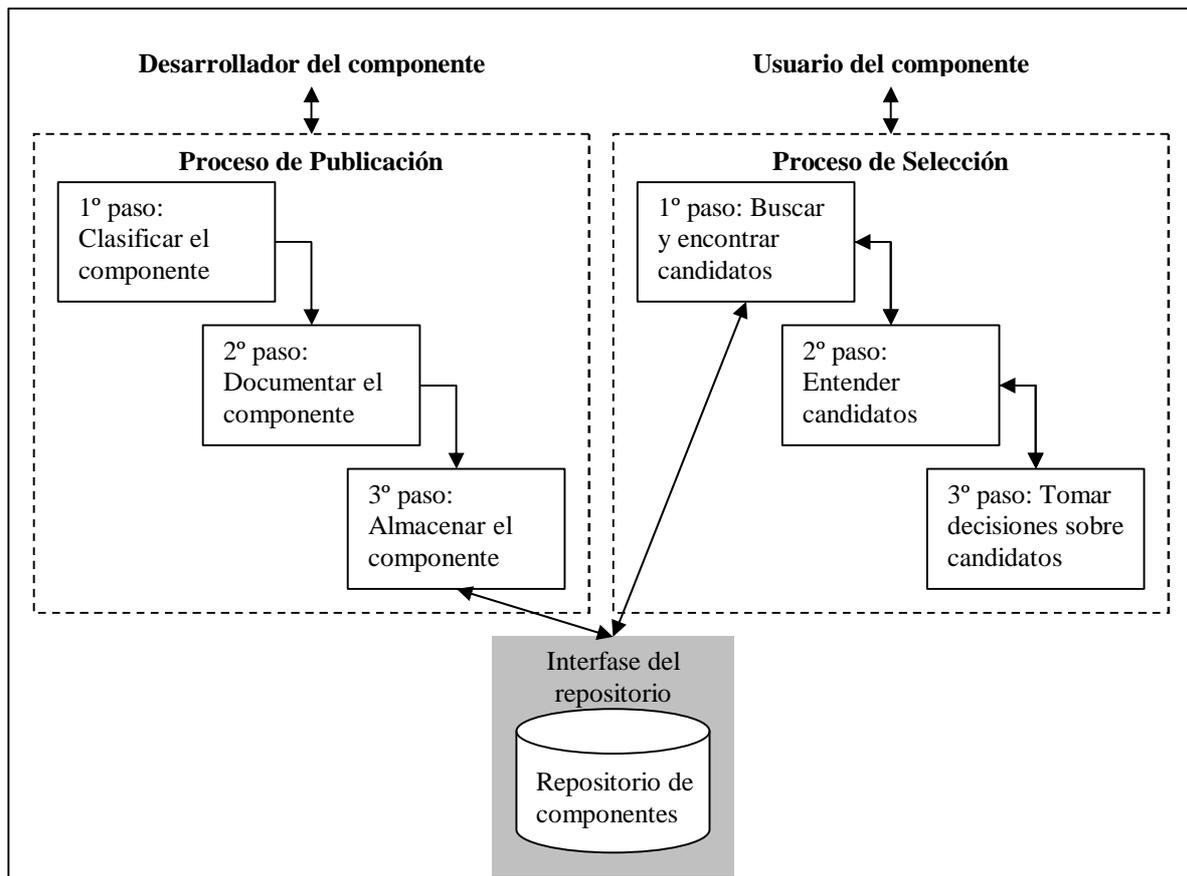


Figura 1. Proceso de selección de componentes

Como ya vimos, uno de los aspectos clave en el DSBC es la disponibilidad de especificaciones de componentes adecuadas y completas para llevar a cabo de forma efectiva los procesos de selección de componentes. Para mejorar la calidad de los componentes liberados, el desarrollador y el usuario deben cooperar en los procesos de documentación de los mismos.

En [1] se analizaron los problemas principales que surgen del proceso de selección de COTS. Entre ellos podemos enunciar:

- *Falta de un proceso definido:* el proceso de selección de COTS se implementa de manera “ad-hoc”, dificultando la planificación, desaprovechando métodos y herramientas de evaluación.
- *Criterios de evaluación:* los evaluadores incluyen atributos no apropiados, conduciendo a incompatibilidades.
- *Naturaleza “black box” de los componentes:* la falta de acceso al interior de los COTS hace difícil la evaluación. Muchas veces el soporte de documentación es incompleto o erróneo.
- *Rápidos cambios del mercado:* por ejemplo, una nueva liberación de un COTS puede tener características que no están disponibles en el componente que se está evaluando actualmente.

En cuanto a la documentación, los componentes OTS no están documentados en una forma estándar, y por lo tanto los desarrolladores tienen problemas para encontrar y evaluar la calidad de los componentes. Estos usuarios necesitan información mas precisa sobre los componentes que tienen que satisfacer requerimientos técnicos (funcionales y no funcionales) y requerimientos no-técnicos. Kallio & Niemelä en [10] proponen una plantilla para documentar un componente de software, en la que participan distintos roles del desarrollo: arquitecto (describe requerimientos funcionales y de calidad), diseñador (describe detalles técnicos), reusador (agrega comentarios de errores y cambios realizados) y mantenedor (actualiza componentes y su historia con información de calidad). Dong et al. [3] sostiene que la información que acompaña a los componentes es escasa, y proponen otra plantilla de especificación de componentes COTS. En [9] Iribarne et al. proponen un modelo de documentación de componentes soportado por el lenguaje XML.

2.1 Clasificación facetada de componentes

En general, los catálogos de componentes basados en la web (eCots, SourceForge, ComponentSource, Flashline, ESRI) contienen información relativamente uniforme, utilizando enfoques de clasificación y/o caracterización de componentes, basados en taxonomías u ontologías. Sin embargo, como se señala en [2], no existe un entendimiento común sobre la información que se hace disponible sobre los componentes; por un lado, los modelos de descripción de estos catálogos no están estandarizados, y por otro existen propuestas muy diferentes en cuanto a su nivel de caracterización, punto de vista o intención y significado.

Los esquemas de clasificación pueden ser *enumerativos* o *facetados*. El método de clasificación enumerativa postula un universo de discurso dividido sucesivamente en clases que incluyen todas las clases compuestas posibles. La clasificación facetada consiste en descomponer el campo científico o disciplina en sus facetas correspondientes a puntos de vista, perspectivas o dimensiones de un dominio conocido.

Prieto-Diaz & Freeman [13] proponen un esquema de clasificación facetada orientado al reuso de software y organizado en seis facetas. Cada faceta tiene un conjunto de valores fijos (vocabulario controlado) y un conjunto extensible de términos de usuario. Un enfoque de clasificación para componentes del dominio de Automatización Industrial basado en un esquema facetado se presenta en [11]. Esta propuesta pretende por un lado, atender la necesidad de los desarrolladores de componentes para documentar sus productos apropiadamente y almacenarlos en un repositorio electrónico; y por otro, dar soporte a los usuarios de componentes para encontrar los componentes almacenados previamente (y para entender correctamente su uso). Las facetas adoptadas para esta taxonomía de componentes se basan en conocimiento adquirido sobre componentes y sobre el dominio de automatización industrial. Tres aspectos principales son cubiertos por el conjunto de facetas elegidas: la funcionalidad implementada por los componentes (QUE hacen), la forma en que realizan esto junto con detalles de implementación (COMO lo hacen), y la descripción del dominio donde los componentes pueden ser usados (DONDE lo hacen). La información ofrecida sobre los componentes se completa con información no técnica (identificación del componente, información de la compañía, precio, versión, desarrollador, etc.).

2.2 Clasificación de Componentes SIG

Componentsource¹ es un sitio web para la adquisición de componentes, en donde los componentes SIG se presentan en la categoría de “*Mapping & GIS Components*”, con la siguiente información:

- Una lista de características en lenguaje natural, en manera informal (overview).
- Precio y licencia, detalles legales, usos y responsabilidades.
- Evaluaciones y downloads (si existen versiones de evaluación).

1 <http://www.componentsource.com>

- Información de compatibilidad con respecto a sistema operativo, arquitectura del producto, tipo de componente (COM, javabeans, ActiveX, etc), herramientas de creación (Visual Basic, C++, Java), etc.
- Detalle de información sobre el vendedor.
- Revisiones de clientes (si hay disponibles)
- Foros de soporte (si hay disponibles)

El proyecto FreeGIS² promueve el uso y difusión de software libre para SIG. En el sitio web que mantiene, organiza la información en categorías algunas veces diferentes a las de Componentsource. En el ámbito de la información geográfica, existen organizaciones como Open Geospatial Consortium (OGC)³ y International Organization for Standardization (ISO) dedicadas a la creación de especificaciones de estándares que permitan a los desarrolladores crear implementaciones que puedan interoperar sin problemas. Por ejemplo, en el estándar ISO 19119 (Services) [4] se presenta una taxonomía de servicios geográficos categorizados.

La diversidad de categorizaciones es por lo tanto común al problema de identificación de componentes OTS.

3. Investigación Propuesta

Dado que las aplicaciones para SIG son utilizadas en dominios muy diferentes, cada aplicación posee un punto de vista distinto sobre la información geográfica, cada desarrollador ha definido modelos conceptuales, modelos de datos geográficos, formatos de almacenamiento, operaciones de análisis o procedimientos de representación especialmente adaptados a los requerimientos de la aplicación. Como consecuencia, se ha generado un gran problema de interoperabilidad entre las herramientas.

Por lo tanto, en el presente trabajo nos concentramos en el proceso de Publicación (Oferta) y Selección (Demanda), proponiendo un enfoque basado en el desarrollo de catálogos de componentes SIG organizados en un esquema de clasificación facetado que permita mejorar el proceso de selección de componentes OTS para estos sistemas.

Para la realización de nuestra propuesta se requiere el análisis de varios aspectos que se relacionan con la selección de los componentes. Debemos analizar aspectos generales como la documentación de los diferentes componentes tanto para SIG como para sistemas convencionales y los diferentes esquemas de facetado analizados en algunas de las propuestas citadas en este trabajo. Luego, dentro de los aspectos específicos, debemos basarnos en algunos de los procesos de selección de componentes para analizar posibles extensiones cuando estamos trabajando con SIG. Las particularidades de los SIG (respecto a los servicios que proveen), extraídas desde las normas ISO 19100 y desde la OGC nos ayudarán a definir un lenguaje estandarizado de descripciones de servicios geográficos. Así, se verán beneficiados tanto los proveedores como los consumidores de componentes SIG.

Los resultados obtenidos a partir de las líneas citadas serán integrados en herramientas de software que faciliten las actividades de verificación de propiedades y búsqueda de componentes.

4. Investigadores

Proyecto conjunto UNPA-UACO/UNComa

Integrantes Proyecto de Investigación Área Ingeniería de Software UNPA-UACO:

Gabriela Gaetan (ggaetan@uaco.unpa.edu.ar); Viviana Saldaño (vivianas@uaco.unpa.edu.ar);

Miriam Diaz (mdiaz@unpa.edu.ar); Bruno Gelman (bgelman@uaco.unpa.edu.ar); Claudio Mayorga

(cmayorga@uaco.unpa.edu.ar); Gabriel Pejcich (gpejcich@uaco.unpa.edu.ar)

² <http://freegis.org/>

³ <http://www.opengeospatial.org/>

Investigadores del grupo GIISCo (Universidad Nacional del Comahue): 2 (dos)
Contacto UNComa: Dra. Alejandra Cechich (acechich@uncoma.edu.ar)
Contacto UNPA-UACO: Gabriela Gaetan (ggaetan@uaco.unpa.edu.ar) Tel/Fax: 0297
4854888 (int. 122)

5. Referencias

1. C. Alves and J. Castro. CRE: A systematic method for cots component selection. Brazilian Symposium on Software Engineering, Rio De Janeiro, Brazil, October 2001.
2. Cechich, A. Réquile, J. Aguirre, and J. Luzuriaga. Trends on COTS Component Identification. 5th International Conference on COTS-Based Software Systems, ICCBSS 2006, 13-17 Febrero 2006, Orlando, USA. IEEE Computer Science Press.
3. Dong, P. S. C., Alencar, D. D., Cowan. "A Component Specification Template for COTS-based Software Development." First Workshop on Ensuring Successful COTS Development. May 1999.
4. Geographic Information – Services. Draft International standard 19119, ISO/IEC, 2002.
5. Geographic information. Geographic Information Metadata. Draft International standard 19115, ISO/IEC, 2003.
6. Geographic information. Geographic Information Methodology for Feature Cataloguing. Draft International standard 19110, ISO/IEC, 2005.
7. Geographic information. Rules for Application Schema. Draft International Standard 19109, ISO/IEC, 2005.
8. Geographic information. Spatial Schema. International standard 19107, ISO/IEC, 2003.
9. Iribarne, J. M. Troya, and A. Vallecillo. Trading for COTS Components in Open Environments. In 27th Euromicro Conference, Warsaw, Poland, Sept. 2001. IEEE Computer Society Press.
10. Kallio and E. Niemelä. Documented Quality of COTS and OCM Components. In Proceedings of the 4th ICSE Workshop on Component-Based Software Engineering, 2001, available at <http://www.sei.cmu.edu/pacc/CBSE4-Proceedings.htm>
11. Lucena Jr. Flexible Web-based Management of Components for Industrial Automation. Stuttgart Univ., Diss., 2002.
12. Oberndorf. Facilitating Component-Based Software Engineering: COTS and Open System. In Proceedings of the
13. Prieto-Díaz, P.; Freeman. Classifying software for reusability. IEEE Software, 4(1):6-16, January 1987.
14. Szyperski. Component Software – Beyond Object-Oriented Programming. Addison-Wesley, 1998.
15. Wallnau, S. Hissam, and R. Seacord. Building Systems from Commercial Components. Addison-Wesley, 2002.