BeTTy: Un Framework de Pruebas para el Análisis Automático de Modelos de Características

Sergio Segura, José A. Galindo, David Benavides and José A. Parejo

Departamento de Lenguajes y Sistemas Informáticos. Universidad de Sevilla.
Av Reina Mercedes S/N, 41012 Sevilla, España

Resumen El análisis automático de modelos de características es un área de investigación activo que ha llamado la atención de numerosos investigadores durante las dos últimas décadas. Durante este tiempo, el número de herramientas y técnicas para el análisis de modelos de características se ha multiplicado y con ellas su complejidad. En este escenario, la falta de mecanismos específicos para validar y evaluar la funcionalidad y el rendimiento de las herramientas de análisis se ha convertido en un gran obstáculo dificultando el desarrollo de herramientas y afectando negativamente a su calidad y fiabilidad. En este artículo, presentamos BeTTy, un framework para la automatización de pruebas en el análisis de modelos de características. Entre otras funcionalidades, BeTTy permite la detección automática de errores en herramientas de análisis de modelos de características. Además, BeTTy permite generar modelos de características tanto aleatorios como computacionalmente duros, útiles para evaluar el rendimiento de las herramientas de análisis. Parte de la funcionalidad del framework es ofrecida a través de una aplicación Web que facilita en gran medida su uso.

1. El Framework BeTTy

La ingeniería de líneas de productos es un paradigma de desarrollo orientado a construir familias de sistemas software que reutilicen características comunes en lugar de construir cada producto desde cero. Un aspecto fundamental para la gestión de una línea de productos es utilizar un modelo que permita representar todos los posibles productos que pueden derivarse de ella. Uno de los modelos más populares usados para tal fin son los denominados modelos de características [3]. Un *modelo de características* se representa visualmente como un árbol donde los nodos representan las características y las aristas representan las posibles relaciones o restricciones entre características.

La extracción automática de información de modelos de características es un tema de investigación activo que ha atraído la atención de numerosos investigadores en los últimos veinte años [1]. Durante este tiempo, se han presentado un gran número de operaciones, técnicas y herramientas y se ha consolidado toda una comunidad alrededor de lo que se ha denominado *análisis automático*

A. Ruíz, L. Iribarne (Eds.): JISBD'2012, pp. 643-646, ISBN:978-84-15487-28-9.
Jornadas SISTEDES'2012, Almería 17-19 sept. 2012, Universidad de Almería.

de modelos de características. Actualmente, los avances en éste área están llevando a una mayor preocupación por la calidad de las herramientas de análisis. Los prototipos de investigación ya no son suficiente y ahora se buscan solidas herramientas de análisis en términos de ausencia de errores y rendimiento. Sin embargo, las pruebas software empleadas en este campo son fundamentalmente aleatorias y guiadas por la intuición de los propios desarrolladores más que por técnicas maduras para el diseño de datos de prueba. Esto hace que las conclusiones de las pruebas sean difícilmente rigurosas y verificables a la vez que limitan el alcance y el valor de las contribuciones científicas en el análisis de modelos de características. De igual forma, la arbitrariedad u omisión de las pruebas en este campo reducen notablemente la confianza de los usuarios sobre el correcto funcionamiento de las herramientas de análisis.

Para abordar el problema de las pruebas en el análisis de modelos de características, hemos presentado una serie de técnicas, algoritmos y herramientas para la realización de pruebas funcionales y de rendimiento en herramientas de análisis de modelos de características. Estas contribuciones son el resultado de la aplicación de varias técnicas clásicas e innovadoras de pruebas software en el contexto de análisis de modelos de características. Para facilitar las pruebas funcionales, hemos presentado un conjunto de casos de prueba [4] así como un generador automático de datos de prueba para la detección de errores en las herramientas de análisis [6,7]. En lo que se refiere a pruebas de rendimiento, hemos trabajado en un algoritmo evolutivo para la generación de modelos de características difíciles de procesar en términos de tiempo y memoria requeridos para su análisis [8]. Estas herramientas han sido evaluadas experimentalmente de forma rigurosa demostrando la viabilidad de la propuesta. Entre otros resultados, nuestras herramientas nos han permitido detectar dos errores en FaMa [2] y otros dos en SPLOT [9], dos herramientas para el análisis de modelos de características ampliamente usadas por la comunidad. Con el objetivo de hacer nuestras contribuciones accesibles a la comunidad y animar a investigadores y profesionales a usar, extender y evaluar nuestro trabajo, hemos integrado todas estas contribuciones en un framework llamado BeTTy (Benchmarking and TesTing on the analYsis of feature models). Las principales funcionalidades del framework son:

- **Generación de modelos de características aleatorios**. BeTTy permite la generación aleatoria de modelos de características. Estos pueden ser usados para evaluar el rendimiento de las herramientas de análisis con modelos de diversos tamaños y formas. Los modelos generados pueden almacenarse en hasta cinco formatos diferentes facilitando la interoperabilidad con otras herramientas.
- **Generación aleatoria de modelos de características atribuidos**. BeTTy también permite la generación de modelos de características atribuidos, también llamados modelos de características extendidos. Estos modelos son ampliamente usados para añadir atributos extra funcionales a las características. Hasta donde sabemos, esta es la primera herramienta disponible para la generación aleatoria de este tipo de modelos.

- **Generación automática de datos de prueba para pruebas funcionales**. BeTTy permite la generación de entradas y salidas esperadas para un gran número de operaciones de análisis sobre modelos de características (18 hasta el momento) facilitando la detección de errores en herramientas de análisis.
- **Generador evolutivo de modelos de características**. BeTTy incluye un algoritmo evolutivo para la generación de modelos de características que maximicen un criterio de optimización dado por el usuario. Esta característica puede ser usada, por ejemplo, para generar modelos de características computacionalmente duros (maximizando el tiempo de análisis) que permitan evaluar el rendimiento de las herramientas de análisis en casos pesimistas.
- **Benchmarking**. BeTTy integra varios componentes para facilitar la comparación del rendimiento de varias herramientas de análisis, esto es, generación de datos de prueba, ejecución y almacenamiento de resultados.

BeTTy está escrito en Java, liberado bajo licencia LGPL3 y distribuido como un fichero jar para facilitar su integración en proyectos externos. Además, parte de la funcionalidad del framework está disponible a través de una aplicación Web (ver Figura 1) permitiendo la generación de modelos de características aleatorios

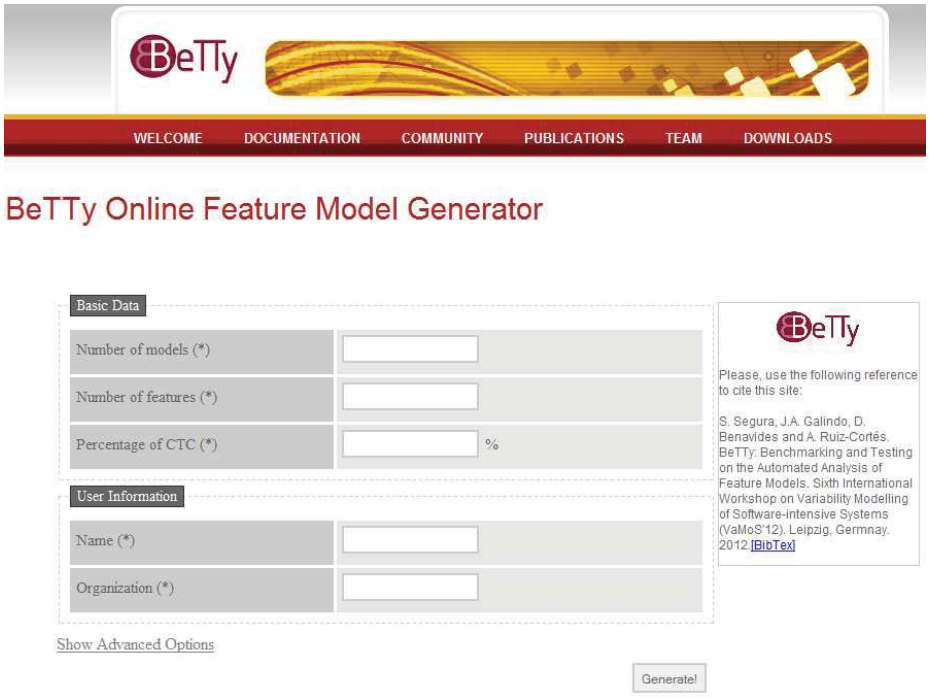

Figura 1. Generador on-line the modelos de características de la Web de BeTTy

con unos pocos clicks. Para más información sobre BeTTy el lector interesado puede consultar [5].

El framework puede descargarse desde la Web de BeTTy: `www.isa.us.es/betty`.

Agradecimientos

Este trabajo ha sido financiado parcialmente por la comisión europea y el gobierno español mediante el proyecto CICYT SETI (TIN2009-07366), y por el gobierno andaluz mediante los proyectos de excelencia ISABEL(TIC-2533) y THEOS (TIC-5906).

Referencias

1. D. Benavides, S. Segura, and A. Ruiz-Cortés. Automated analysis of feature models 20 years later: A literature review. *Information Systems*, 35(6):615 – 636, 2010.
2. FaMa Tool Suite. `http://www.isa.us.es/fama/`, accessed November 2011.
3. K. Kang, S. Cohen, J. Hess, W. Novak, and S. Peterson. Feature–Oriented Domain Analysis (FODA) Feasibility Study. Technical Report CMU/SEI-90-TR-21, SEI, 1990.
4. S. Segura, D. Benavides, and A. Ruiz-Cortés. Functional testing of feature model analysis tools: a test suite. *Software, IET*, 5(1):70 –82, february 2011.
5. S. Segura, J.A. Galindo, D. Benavides, J.A. Parejo, and A. Ruiz-Cortés. BeTTy: Benchmarking and testing on the automated analysis of feature models. In U.W. Eisenecker, S. Apel, and S. Gnesi, editors, *Sixth International Workshop on Variability Modelling of Software-intensive Systems (VaMoS'12)*, pages 63–71, Leipzig, Germany, 2012. ACM.
6. S. Segura, R.M. Hierons, D. Benavides, and A. Ruiz-Cortés. Automated test data generation on the analyses of feature models: A metamorphic testing approach. In *International Conference on Software Testing, Verification and Validation*, pages 35–44, Paris, France, 2010. IEEE press.
7. S. Segura, R.M. Hierons, D. Benavides, and A. Ruiz-Cortés. Automated metamorphic testing on the analyses of feature models. *Information and Software Technology*, 53(3):245 – 258, 2011.
8. S. Segura, JA. Parejo, Robert M. Hierons, D. Benavides, and A. Ruiz-Cortés. ETHOM: An evolutionary algorithm for optimized feature models generation. Tech Report ISA-2011-TR-03 (v. 1.0), Applied Software Engineering Research Group, 2011.
9. S.P.L.O.T.: Software Product Lines Online Tools. `http://www.splot-research.org/`, accessed November 2011.

A Systematic Review of Quality Attributes and Measures for Software Product Lines*

Silvia Abrahão, Sonia Montagud, Emilio Insfran

Grupo ISSI, Departamento de Sistemas Informáticos y Computación
Universitat Politècnica de València
Camino de Vera s/n, 46022, Valencia, España

{sabrahao,smontagud,einsfran}@dsic.upv.es

1 Introduction

While quality is an important factor in the construction of single software products, it becomes crucial in Software Product Lines (SPL) since the quality of all products that can be derived from the product line must be ensured. Software measures provide an appropriate mechanism for understanding, controlling, and predicting the quality of software development projects.

A great number of software measures for assessing the quality of Software Product Lines (SPL) have been proposed over the last few years. However, no studies summarizing the current knowledge about them exist. This paper presents a systematic literature review with the aim of identifying and analyzing the existing quality attributes and measures proposed by researchers from 1996 to 2010 to evaluate the quality of software product lines.

2 Planning the systematic review

To identify the primary studies for our systematic literature review, we used the following digital libraries: IEEEXplore, ACM Digital Library, Science Direct, SpringerLink, and INSPEC. Primary study is the common term to describe the publications contributing to a SLR. We tested several search strings, and the following one retrieved the greatest number of relevant papers: *((attribute* OR factor* OR propert* OR criterion OR criteria OR characteristic*) OR (metric* OR measur*)) AND (software OR engineering OR architectur*) AND ("product line*" OR "product famil*") AND (quality OR non-functional OR "no functional" OR assess* OR assur*)*. In addition, we performed a manual search in relevant conference proceedings.

To extract the data from the selected studies, the following criteria were used: (1) quality characteristic evaluated (those from the ISO/IEC 25010): (2) quality attribute evaluated by the measure (e.g., internal, external; specific for SPL); (3) type of measure (base, derived); (4) type of evaluation (e.g., qualitative, quantitative, probabilis-

* This work has been funded by the MULTIPLE project (TIN2009-13838)

A. Ruíz, L. Iribarne (Eds.): JISBD'2012, pp. 647-648, ISBN:978-84-15487-28-9.
Jornadas SISTEDES'2012, Almería 17-19 sept. 2012, Universidad de Almería.