Caracterización de los riesgos inherentes a la Ingeniería Reversa

Herramientas para la Ingeniería Reversa

Inés Gimenez Zens, Juan Carlos Cuevas, Franco Mana

Universidad Tecnológica Nacional

Facultad Regional Córdoba

Departamento de Ingeniería de Sistemas de Información

Grupo de Investigación en Ingeniería Reversa

Maestro López Esq. Cruz Roja Argentina – Ciudad Universitaria
C.P 5016 – Córdoba – Argentina

RESUMEN

Se presenta en este documento el resultado de la línea de investigación, "Herramientas de Ingeniería Reversa", en el marco del proyecto "Caracterización de los riesgos inherentes a la Ingeniería Reversa". Este proyecto lo desarrolla el grupo de investigación de la Cátedra de Servicios del Software de la Universidad Tecnológica Nacional Córdoba. Se describe aquí el proceso de investigación y el desarrollo de un gráfico que tiene como objetivo informar las posibles herramientas de ingeniería de software a utilizar según las variables del proyecto que se haya encarado.

PALABRAS CLAVES

Ingeniería reversa, Herramientas de Ingeniería reversa, Ingeniería de Software.

CONTEXTO

La Cátedra de Servicios del Software de la Universidad Tecnológica Nacional Córdoba desarrolla desde el año 2008 investigaciones tendientes a arribar a un Framework para la Gestión de la Calidad en Ingeniería Reversa. Como primer proyecto de investigación se presentó "Caracterización de los riesgos inherentes a la Ingeniería Reversa" ante la Secretaría de Ciencia y Tecnología de la UTN, el cual fue acreditado por la mencionada Secretaría y se encuentra en su fase final de desarrollo. Este proyecto tiene, entre sus objetivos aportar a una gestión más eficiente del software en aspectos tales como funcionalidad, confiabilidad, usabilidad, mantenibilidad, portabilidad, seguridad, costos y satisfacción de los clientes-usuarios.

INTRODUCCION

En la actualidad el hombre tiene una alta dependencia de los sistemas de software para la realización de todo tipo de tareas. Estos sistemas permiten automatizar procesos, simplificar tareas y un sin fin de objetivos que mejoran la calidad de vida de las personas. Esta dependencia junto con las necesidades empresariales y sumadas a los avances tecnológicos, han ocasionado que constantemente se necesiten crear o adaptar sistemas de software. Si el sistema de software es creado sin tener un precedente, se debe realizar un análisis profundo de los requerimientos para que este cumpla con el objetivo y resuelva las necesidades. En el caso de que el sistema sea heredado[1], o que se necesite un nuevo sistema pero con un precedente, la tarea a realizar es la investigación de la información existente que permita comprender y posteriormente actuar de

[1] Un sistema heredado (o sistema legacy) es un sistema informático (equipos informáticos o aplicaciones) que ha quedado anticuado pero continúa siendo utilizado por el usuario (típicamente una organización o empresa) y no se quiere o no se puede reemplazar o actualizar de forma sencilla.

acuerdo a los objetivos prefijados. Este último caso se denomina ingeniería reversa de software.

La ingeniería reversa de software se enmarca dentro del mantenimiento de software y es definida por Chikofsky y Cross como el proceso de analizar un sistema para crear una representación del mismo, pero a un nivel más elevado de abstracción[2]. Esto se logra a través de procesos de compresión y no a través de la modificación del sistema. La recopilación de la información es fundamental para elevar los niveles de conocimiento del programa y es una tarea que debe realizar el ingeniero de software a través la identificación de artefactos, el descubrimiento de relaciones y la generación de abstracciones. La recuperación del entendimiento del sistema no es tarea fácil, se deben tener habilidades específicas para realizar este trabajo y conocimiento sobre herramientas de ingeniería reversa específicas que permitan simplificar el trabajo.

Las herramientas clásicas de ingeniería reversa son:

El Decompilador: es un programa que realiza la operación inversa a un compilador. Esto es, traducir código o información de bajo nivel de abstracción (sólo diseñado para ser leído por un ordenador, por ejemplo el código máquina) a un lenguaje o medio de mayor nivel de abstracción.

El depurador (en inglés, debugger): es un programa usado para probar y depurar (eliminar los errores) de otros programas (el programa "objetivo").

El desensamblador: es un programa que traduce el lenguaje de máquina a lenguaje ensamblador, la operación inversa de la que hace el ensamblador.

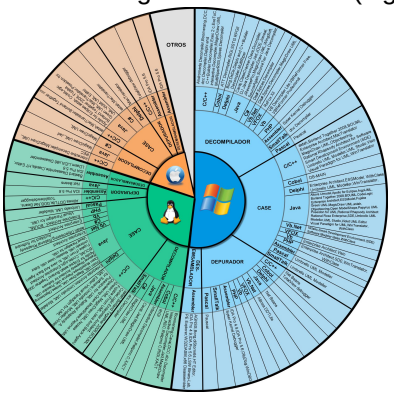
Las herramientas CASE (Computer Aided Software Engineering, Ingeniería de Software Asistida por Computadora): son diversas aplicaciones informáticas destinadas a la recuperación de los modelos en los que fue originalmente desarrollado el software heredado, en el caso de la ingeniería reversa. Debemos destacar que es muy escasa la información y bibliografía que el ingeniero de software dispone para hacer uso de una en herramienta en particular. Esto se debe, en gran medida, a que la elección de la herramienta dependerá de diversas variables que complejizan la búsqueda. Las principales variables limitantes en el uso de una herramienta de ingeniería reversa son el sistema operativo, el tipo de herramienta y su salida, el tipo de lenguaje de programación, y la funcionalidad de la herramienta, entre otros. Cuando el ingeniero de software necesita una herramienta en particular, la definición de estas variables en que opera limitará significativamente el universo de posibilidades. Por ejemplo, un caso común podría ser la búsqueda de un decompilador que funcione para el lenguaje Java y que se opere bajo la plataforma Windows. Pero muy distinta será la misma búsqueda de herramientas bajo la plataforma MacOS, en donde el universo es significativamente menor.

LINEAS DE INVESTIGACION Y DESARROLLO

En el marco del proyecto “Caracterización de los riesgos inherentes a la Ingeniería Reversa” se abrió una línea de investigación en el tema “Herramientas para la ingeniería reversa” con la intención de aportar el conocimiento necesario sobre la temática para poder avanzar hacia la definición de los riesgos en la ingeniería reversa.

RESULTADOS Y OBJETIVOS

Como resultado de este relevamiento se obtuvo una lista de herramientas que fueron categorizas y puestas a disposición de un diseñador que transformó esta lista en un gráfico denominado “Herramientas de Ingeniería Reversa” (Fig.1).



- Fig.1 Herramientas de Ingeniería Reversa

Este gráfico (expuesto en sus reales dimensiones en el anexo 1) presenta la información de una

[2] La abstracción, en la ingeniería de software, es entendida como las diferentes cualidades aisladas del sistema de software heredado que permiten mejorar la información para comprenderlo más fácilmente.

manera que el ingeniero de software puede acceder rápidamente, seleccionando las variables limitantes, a un listado de los nombres de las herramientas de ingeniería reversa disponibles. Esta información minimiza el esfuerzo de investigación de herramientas específicas necesarias al encarar un proyecto de ingeniería reversa de un software heredado. El gráfico aporta datos concretos según el sistema operativo, el tipo de herramienta, y lo que es fundamental, el tipo de lenguaje de programación.

FORMACION DE RECURSOS HUMANOS

El proyecto de investigación está destinado a contribuir a la formación de recursos humanos como docentes de la Cátedra, Docentes de otras asignaturas, Estudiantes, Graduados, Especializandos, Maestrandos y Doctorandos y Profesionales de la Industria del Software.

Transferencia de conocimiento

La especialización adquirida en el estado del arte de la temática objeto de estudio como así también el eventual nuevo conocimiento generado, está destinado a ser transferido, en primera instancia a los estudiante de la cátedra de Servicios del Software y como segundo paso a los distintos estamentos académicos, gobierno, industria y sociedad.

Metodología

La base de ejecución del programa de investigación es el método científico. La metodología propuesta para el abordaje de dicho programa se fundamenta en desarrollar una serie de actividades que se basan en el método que permite relacionar el estudio y evaluación de las relaciones entre los cuerpos teóricos disponibles y la evidencia empírica de los fenómenos estudiados, como por ejemplo: los estudios de índole exploratorios, descriptivos y explicativos, los principios, métodos y herramientas estadísticas y teorías probabilísticas, diseños experimentales, etc.

Para el caso particular de la investigación en las herramientas para la ingeniería reversa, se analizaron los tipos de herramientas disponibles en la web y se categorizaron. Con esta información se lanzó un relevamiento de un año de duración en busca de las herramientas para las plataformas Windows, Linux y MacOS. Los resultados fueron tabulados por tipo de lenguaje de programación y luego se diseñó el gráfico para simplificar las búsquedas.

Impacto del Proyecto

El proyecto apunta a reducir la brecha entre el amplio conocimiento existente para el desarrollo del software y el escaso conocimiento para el mantenimiento y la ingeniería reversa. Para esto el gráfico presentado “Herramientas de ingeniería reversa” aporta la información necesaria para que el ingeniero de software pueda reducir significativamente el esfuerzo en la búsqueda de herramientas que funcionen en entornos particulares, facilitando así la tarea de ingeniería reversa.

BIBLIOGRAFIA

1. [Chikofsky:1990] Chikofsky, E. and Cross, J. (1990). Reverse engineering and design recovery: A Taxonomy. IEEE Software.
2. [Jarzabek:2007] Stanislaw Jarzabek; "Effective Software Maintenance and Evolution. A Reuse-Based Approach". Auerbach Publications. NY - EEUU - 2007.
3. [Grubb-Takang:2005] Penny Grubb and Armstrong A. Takang; "Software Maintenance - Concepts and Practice" Second edition. World Scientific - Singapore - 2005.
4. [Tonella-Potrich:2005] Paolo Tonella and Alessandra Potrich; "Reverse Engineering of Object Oriented Code". Springer Science + Business Media, Inc. - Boston - EEUU - 2005
5. [Eilam:2005] Eldad Eilam "Reversing - Secrets of Reverse Engineering". Wiley Publishing, Inc - IN - EEUU - 2005.
6. [Davis:2005] Kathi Hogshead Davis " August-11: A Tool for Step- by-Step Data Model Reverse Engineering" IEEE - 0-8186-7111-4/95 - 1995
7. [Gannod-Cheng:1996] Gerald C. Gannod and Betty H. C. Cheng " Using Informal and Formal Techniques for the Reverse Engineering of C Programs" - 0-8186-7674-4/9 - IEEE - 1996.
8. [Bellay-Gall: 1997] Bemdt Bellay and Harald Gall; "A Comparison of four Reverse Engineering Tools" 0-8186-8162497 - IEEE - 1997.
9. [Davis:2001] Kathi Hogshead Davis; "Lessons Learned in Data Reverse Engineering". 1095-1350/01 - IEEE - 2001.
10. [Jahnke-Valenstein:2000] Jens H. Jahnke and Andrew Walenstein "Reverse Engineering Tools as Media for Imperfect Knowledge" 1095-1350/00 - IEEE - 2000.
11. [Tilley:1998] Scott Tilley; "A Reverse-Engineering Environment Framework". TECHNICAL REPORT
12. CMU/SEI-98-TR-005. Carnegie Mellon University - Software Engineering Institute - PA - EEUU. 1998.
13. [Zhou et al:1999] Shikun Zhou, Hongji Yang, Paul Luker and Xudong He; "A Useful Approach to Developing Reverse Engineering Metrics". 0-7695-0368-3199 - IEEE - 1999.
14. [Chiang-Barron:1995] Roger H. L. Chiang and Terence M. Barron "Quality Issues in Database Reverse Engineering: An Overview". 95 ENGINEERING MANAGEMENT CONFERENCE - 0-7803-2799-3/95/ - IEEE - 1995.
15. [Garcia et al:2004] Vinicius C. Garcia, Daniel Lucr´edio, Antonio F. do Prado, Alexandre Alvaro and Eduardo S. de Almeida "Towards an effective approach for Reverse Engineering". Proceedings of the 11th Working Conference on Reverse Engineering (WCRE'04) - 1095-1350/04 - IEEE - Computer Society - 2004.
16. [D´Ambros-Lanza:2006] Marco D'Ambros and Michele Lanza "Reverse Engineering with Logical Coupling". Proceedings of the 13th Working Conference on Reverse Engineering (WCRE'06) - 0-7695-2719-1/06 - IEEE - Computer Society - 2006.
17. [Fahmi-Choi:2007] Syed Ahsan Fahmi and Ho-Jin Choi; "Software Reverse Engineering to Requirements". 2007 International Conference on Convergence Information Technology - 0-7695-3038-9/07 - IEEE - Computer Society - 2007.

ANEXO I

Gráfico - Herramientas de Ingeniería Reversa