

# TRIPTICO DE LA INGENIERÍA DEL SOFTWARE. ANÁLISIS DESARROLLO Y APLICACIONES DE METODOLOGÍA RAISE

José H. Paganini, Sebastián M. Figueroa Juan C. Rodriguez, Hector P. Liberatori, José Eduardo Huanca, Nancy Cholele, Delia C. Cruz. Centro de Investigaciones Básicas y Aplicadas (CIBA) Facultad de Ingeniería Universidad Nacional de Jujuy UNJu

## CONTEXTO

Este proyecto constituye un avance en la Argentina de la aplicación de la metodología RAISE, a tal punto que casi no hay antecedentes de aplicaciones similares en el país

A su vez es la investigación en aplicaciones y desarrollos que puedan enriquecer la Metodología RAISE con aportes desde lógicas deónticas, modales y paraconsistentes junto con otros provenientes del análisis funcional, para aumentar el espectro de formalización de dominios, entidades y procedimientos en el campo de las Ciencias de la Computación, genera una contribución al desarrollo de la informática, en su finalidad de obtener sistemas más confiables. En ese sentido la Facultad de Ingeniería de la UNJu, a través del CIBA, con este grupo de I/D contribuye a la investigación en temas de la Ingeniería del Software.

## RESUMEN

El Tríptico de la Ingeniería del Software, compuesto por la Ingeniería del Dominio, Ingeniería de Requisitos y el Diseño del Software, tiene como fase más importante a la primera de ellas, que especifica el espacio donde será eventualmente aplicado un producto software particular.

Especificar según la metodología RAISE es expresar mediante formulaciones lógico matemáticas las entidades y sus relaciones en un dominio; logrando una sintaxis precisa y una semántica única para ello se emplea el lenguaje formal RSL (RAISE Specification Language)

El presente proyecto encara, por un lado, el análisis de la metodología RAISE, su vinculación a los procesos de Verificación y Validación del software y eventuales extensiones de la metodología a campos de la Lógica Modal y teoría de Probabilidades. Por otro se realizarán aplicaciones a casos concretos de diseños efectuados bajo el paradigma del Tríptico, con énfasis en la Ingeniería del Dominio

**Palabras clave:** RAISE, RSL, LÓGICA TRIPTICO DEL SOFTWARE

## 1. INTRODUCCION

El diseño de un producto software debe ser acorde a un conjunto de requisitos que debe satisfacer; estos requisitos a su vez están enmarcados, por decirlo así, en un dominio que se estableció para fijar su alcance

y visión, más específicamente el dominio en que el software debe servir. Entonces el proceso que desemboca en el diseño posee tres componentes, el establecer el dominio, que se lo denota con Ingeniería del Dominio; el establecimiento de los requisitos, conocido como Ingeniería de requisitos, y por último, el diseño del software emergente.

De esta forma, estos tres elementos, Ingeniería del Dominio, Ingeniería de Requisitos y el Diseño del Software constituyen un Tríptico. Concepto este, establecido por Dines Bjørner; y expuesto en [7], [8], [12], [13] y [14].

Este enfoque constituye un nuevo paradigma en la manera de producir productos software, bajo el cual se están elaborando proyectos software, principalmente, en la Comunidad Europea como el AMORE, que organiza la gestión de las distintas líneas férreas de los países de la Comunidad, y también aplicaciones a sistemas de control de vías férreas en China; junto con las diversas aplicaciones generadas en el UNU/ IIST. (United Nations University International Institute of Software Technology) sito en Macao, República Popular China.

Este nuevo paradigma, establece como necesario, como paso inicial y fundamental, la descripción formal del Dominio y de los Requerimientos, para poner en claro, sin las ambigüedades del lenguaje coloquial, de que se habla, a que se refiere cuando se designa algo, y el establecimiento de los requerimientos. Entonces, para la descripción formal del dominio y los requerimientos se emplea el Método RAISE, con un lenguaje de especificación, riguroso basado en la Lógica y Matemática, que es el RSL (RAISE Specification Language).

Los lineamientos básicos, del método formal y del lenguaje, se encuentran en los trabajos del investigador británico Christopher William George y un grupo de colaboradores (conocido como RAISE Group) [6] y [9].

Esto también supone para el grupo de I/D el estudio y análisis de sistemas formales, leyes de la lógica y matemática [1], [2], [3], [4] y [5].

Para estructurar la fase de Diseño del Software con las fases anteriores del Tríptico, se acude al empleo de herramientas software (EMACS y otras), desarrolladas en el UNU/IIST, que por un lado depuran las formalizaciones y establecen condiciones de confiabilidad de las mismas; y por otro avanzan a una etapa de implementación

produciendo código en lenguajes aplicativos cómo consecuencia lógica de las formalizaciones en RSL. Con esta fase culmina la aplicación del paradigma del Tríptico a través de esta metodología que parte de una especificación formal y termina en un código en un lenguaje aplicativo

Esta metodología constituye un aporte más, entre muchos otros, a la Ingeniería del Software, constituyéndose esta última en una de las ramas de la Informática que mayor crecimiento e importancia ha adquirido en los últimos años.

En efecto siendo que el desarrollo de sistemas es uno de los aspectos económicamente más importante de la Informática y que dicho desarrollo es crucial para las aplicaciones informáticas que tiende a cerrar la brecha entre las crecientes necesidades de desarrollo y las prestaciones que implican velocidad en producir y llegar a tiempo al mercado con nuevos productos, garantizando la seguridad y confiabilidad de los desarrollos, ajustándose a las previsiones de costo, y que la Ingeniería del Software es la rama de la Informática que trata con las herramientas, métodos y principios necesarios para el desarrollo de sistemas, cuya importancia no puede negarse.

#### 1.4.1. Objetivos Generales y Particulares

El proyecto tiene como objetivos generales, en el marco de la Ingeniería del Software, los siguientes:

- I. El estudio, investigación y desarrollo de diversas técnicas y herramientas, dentro de la metodología RAISE que sean útiles en el campo del desarrollo de sistemas.
- II. La investigación básica en Lógica y Matemática para el enriquecimiento de la Metodología RAISE y de los Métodos Formales.
- III. La preparación de estudiantes de grado y tesis de postgrado en estas técnicas.

Los objetivos particulares son los siguientes:

- I. El ensayo de herramientas software para la metodología RAISE.
- II. El eventual adaptación y/o diseño de herramientas software para la metodología RAISE
- III. La aplicación concreta de diseño de un software complejo con la metodología RAISE y bajo el paradigma del Tríptico para el Control de Gestión de la Logística en el Corredor Bioceánico (Atlántico Pacífico) del Mercosur, que pasa por la Provincia de Jujuy en la Argentina.

#### 1.4.2. Metodología.

Como el presente es un proyecto de investigación que presenta los aspectos fundamentales siguientes:

1. Estudio y Análisis del Método RAISE y de su aplicación en el Paradigma del Tríptico.
2. Análisis e Investigación de modificaciones y/o herramientas de aplicación al Método RAISE.

3. Aplicaciones de Diseño Software a casos concretos con metodología RAISE y bajo el Paradigma del Tríptico,

Entonces dado que los dos primeros aspectos tienen un sesgo teórico, la metodología a que se emplea es la de estudiar bibliografía, asistir a cursos sobre la temática y el tratar de producir resultados y publicaciones que contribuyan a difundir y/o perfeccionar esta técnica,

Respecto al tercer aspecto, el grupo de I/D decidió establecer que la aplicación se realizaría sobre la Logística del Corredor Bioceánico (Atlántico Pacífico) del Mercosur; y que esta etapa empezaría en Julio de 2009. Por lo consiguiente en esa fecha se empezaría recopilando información e iniciando los contactos con entidades gubernamentales que ejercen el control en la logística de esa área geográfica.

## 2. LINEAS DE INVESTIGACION y DESARROLLO

Esta línea de investigación pretende moverse sobre el eje de hallar parámetros formales que establezcan condiciones cada vez más confiables para el diseño de productos software.

Dado que el software es una creación humana, es una aproximación a la realidad bajo una visión particular; el desarrollo de productos es acorde a una cultura modelada de acuerdo a un conjunto de paradigmas.

Entonces software de una forma particular, es un problema cultural, la implementación de un diseño de un producto software con un proceso riguroso metódico validable y verificable, con toda la documentación pertinente y necesaria, con el dominio de aplicación y los requerimientos estrictamente formalizados, constituye la adopción de nuevos paradigmas que producen un cambio cultural.

Este cambio cultural, reviste caracteres de importancia si es aplicado como norma de adquisición de software en reparticiones del estado provincial; elevando, en consecuencia el estándar de calidad de desempeño de estos productos, y contribuye a fijar criterios para elaboración de disposiciones y leyes regulatorias en el mercado de los sistemas informáticos.

## 3. RESULTADOS OBTENIDOS/ESPERADOS

Los resultados obtenidos concuerdan con los esperados y a esto hay que sumarle el hecho de que en la Facultad de Ingeniería de la UNJu se dicta un Maestría en Ingeniería del Software de la UNSL, en donde docentes de este proyecto colaboran en el dictado contribuyendo a la formación de alumnos de maestría y especialidad. Además se tramitan becas internacionales para estudiar esta temática en el International Insntitute for Software Technology, de las Naciones Unidas UNU/IIST, sito en Macao

República Popular China. Exponer los resultados alcanzados y los objetivos en curso.

#### 4. FORMACION DE RECURSOS HUMANOS

En la formación de recursos humanos, con trabajos contributivos a la investigación dentro de la temática del proyecto, se destacan los siguientes:

1. Tesis de Maestría, en la Universidad Nacional de San Luis UNSL “Dominio de un Sistema de Viaje” de José Humberto Paganini
2. Tesis de Maestría, en la Universidad Nacional de San Luis UNSL “Especificación de requerimientos en un Sistema de Viaje” de Sebastián Marcos Figueroa.
3. Proyecto final de carrera de Ingeniería Informática de la Universidad Nacional de Jujuy, de José Eduardo Huanca.
4. Proyecto final de carrera de Ingeniería Informática de la Universidad Nacional de Jujuy, de Nancy Cholele.
5. Proyecto final de carrera de Ingeniería Informática de la Universidad Nacional de Jujuy, de Delia Cristina Cruz

#### 5. BIBLIOGRAFIA

1. Lógica para Matemáticos. Hamilton, A. G.. Paraninfo. Madrid. 1981.
2. Foundations of Logic Programming. Second extend Edition. Lloyd, J. W. Springer Verlag New York. 1987
3. Fundamentos de Lógica Computacional. Juan Fausto Solís. Gildardo Sánchez Ante. Editorial Trillas. México. ISBN 968-24-6100-6, año de edición 2000.
4. Introducción a la Lógica Matemática. P. Suppes; S. Hill. Editorial Reverté. México. ISBN 968-6708-01-4, año de edición 1999.
5. Introducción a la Lógica. Inving M. Copi. Eudeba manuales. Argentina. ISBN 950-23-0040-8, año de edición 1984.
6. The RAISE Specification Language. Chris George; P. Haff; K. Havenlund; A. E. Haxthausen; R. Milne; C. B. Nielsen; S. Prehn; K. R. Wagner. Prentice Hall. UK., año de edición 1996.
7. Software Engineering A New Approach. Dines Bjøner.. Technical University of Denmark. Dinamarca., año de edición 2002.
8. The SE Book (Software Engineering Book) Dines Bjøner. Technical University of Denmark. Dinamarca., año de edición 2004.
9. The RAISE Development Method. Chris George. Springer Verlag. London 2000
10. Decision Support Systems in the Twenty-First Century. George Marakas. Prentice Hall. USA ISBN 0-13-744186-X, año de edición 2000.
11. Fundamentals of Software Engineering. C. Ghezzi; M. Jazayeri. D. Mandrioli. Prentice Hall. USA. ISBN -0-13-820432-2, año de edición 1999
12. Software Engineering 1 Abstraction and Modeling. Dines Bjøner. SPRINGER 2006 ISBN 10-3540211497
13. Software Engineering 2 Specification System and Languages. Dines Bjøner. SPRINGER 2006 ISBN 10-3540211300
14. Software Engineering 3 Domain Requirements and Software Design. Dines Bjøner. SPRINGER 2006 ISBN 10-3540211518