



# FACULTAD DE INFORMÁTICA

## TESINA DE LICENCIATURA

**Título:** Mecanismos de rastreo en el desarrollo de software dirigido por modelos

**Autores:** Mariano Gabriel Gili

**Director:** Prof. Dra. Claudia Pons

**Carrera:** Licenciatura en Informática – Plan 90

### Resumen

En la Ingeniería de Software Dirigida por Modelos (MDE de Model-Driven Engineering), se define al modelo como artefacto principal de todas las tareas que conforman dicha ingeniería, ésto es, el análisis, el diseño, el desarrollo, las pruebas, el mantenimiento, etc. Una característica muy importante de todo proceso de Desarrollo Dirigido por Modelos (MDD de Model-Driven Development), es lo que se conoce como posibilidad de rastreo (en inglés traceability), que ayuda y toma parte en todo lo que respecta a las relaciones que existen entre cada uno de los artefactos productos del proceso de desarrollo.

Traceability nos ayuda a identificar las relaciones que existen entre los artefactos de software, es crucial entre los requerimientos y su representación en los modelos para asegurar que un conjunto relevante de requerimientos fueron debidamente implementados, y entre otras cosas además, puede facilitar el análisis de impactos de cambios durante el desarrollo de software.

Este trabajo de tesis introduce al lector en el tema de traceability, expone los principales problemas y desafíos que aún se encuentran abiertos en el ámbito de MDE y presenta un esquema de traceability como propuesta. Finalmente, detalla el diseño y la implementación de la herramienta integrada a Eclipse que se construyó basada en dicho esquema.

### Palabras Claves

MDE – Ingeniería de Software Dirigida por Modelos  
Traceability – Posibilidad de rastreo  
Esquema de traceability – Meta-modelo de traceability  
Tracelinks  
Transformaciones de modelo

### Conclusiones

Todo proceso de desarrollo debería hacer uso de una metodología de traceability, que muestre de forma precisa la relación existente entre todos los artefactos que surgen y forman parte del mismo durante el ciclo de vida completo del software. Como se ha visto en el presente trabajo, la información que se obtenga de un buen y actualizado servicio de traceability es muy valiosa y hará más difícil que un proyecto falle. Los problemas y desafíos aún están abiertos, y existen muchas propuestas de solución con varias metodologías e implementaciones.

### Trabajos Realizados

- Se introdujo en el concepto de Traceability y en el conjunto de temas importantes de la teoría que la definen.
- Se presentaron los principales problemas y desafíos.
- Se elaboró un esquema de Traceability.
- Se definió e implementó una transformación QVT que traduce un modelo de tracelinks QVT al modelo del esquema propuesto.
- Se diseñó e implementó una herramienta integrada en el entorno Eclipse como plug-in basada en el esquema de Traceability propuesto.

### Trabajos Futuros

- Extender el esquema de Traceability para ofrecer funcionalidad de versionado, detección de errores y representación de cadenas de transformaciones.
- Definir e implementar una solución para que la herramienta también trabaje sobre transformaciones ATL.
- Ofrecer en la herramienta TraceEditor mejores opciones de filtrado o vistas parciales para facilitar la navegación sobre modelos de tracelinks grandes y también, por ejemplo en el análisis de cambios, mostrando qué artefactos se verían afectados si se modificaran un conjunto seleccionado.