

Procesadores de Lenguajes 2

Desarrollo de DSLs usando MDE

Curso 2013-2014

Iván Ruiz Rube

Departamento de Ingeniería Informática

Escuela Superior de Ingeniería

Universidad de Cádiz



Contenidos

- Introducción
- Proceso de desarrollo
- Entorno tecnológico



DESARROLLO DE DSLs USANDO MDE



INTRODUCCIÓN



Introducción

- Los lenguajes específicos de dominio (DSL) son herramientas para abordar problemas concretos.
- Los DSL pueden ser textuales o visuales y pueden implementarse externa o internamente.
- Podemos desarrollar DSLs utilizando el enfoque *model-driven*.



Introducción (II)

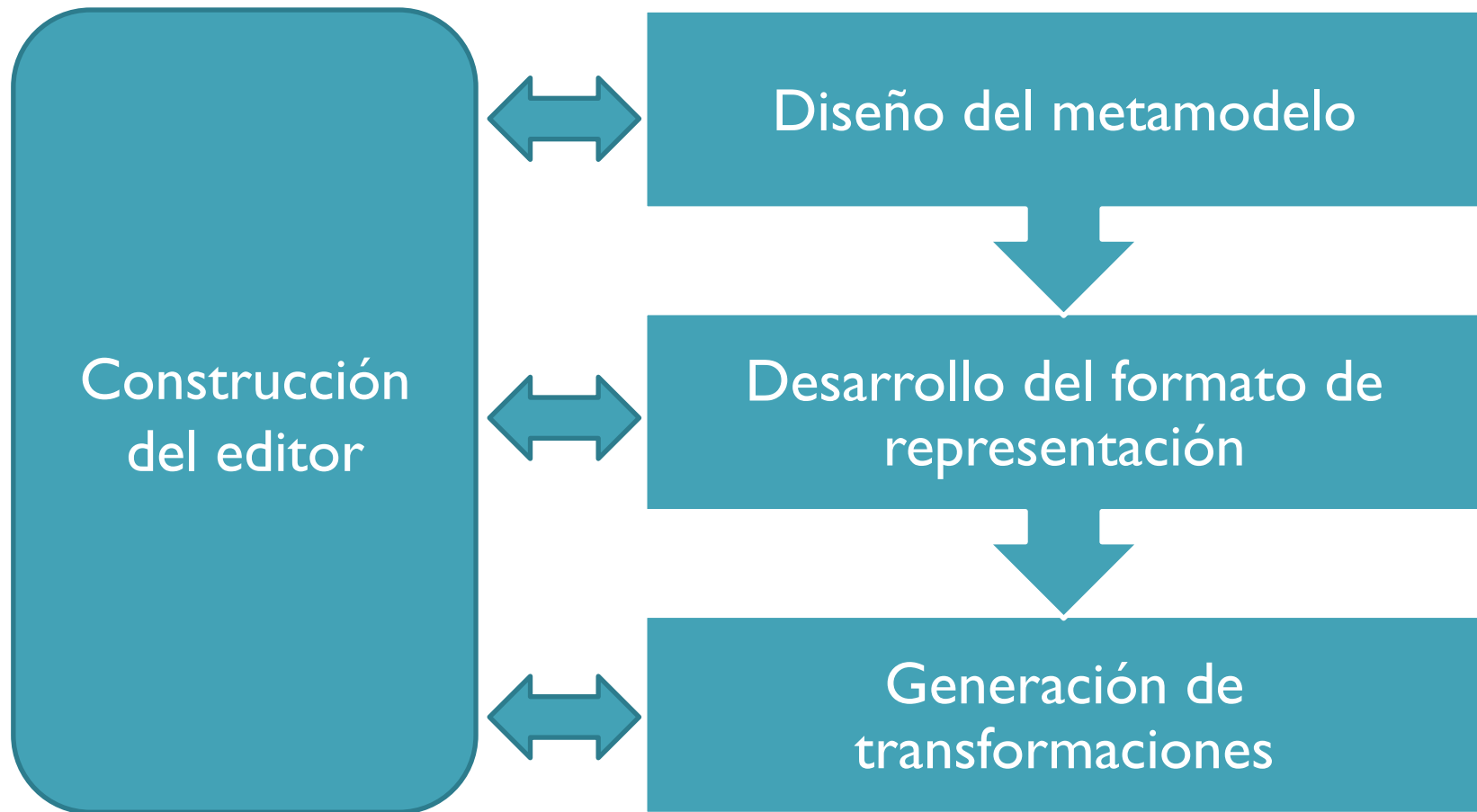
- Los modelos en IS son instrumentos útiles para *representar* sistemas software.
- Model-Driven Engineering promueve utilizar los modelos como *artefactos* software dentro del ciclo de vida.
- A partir de los modelos podemos generar automáticamente código fuente (MDD). Usaremos esto para para construir los DSLs.

DESARROLLO DE DSLs USANDO MDE

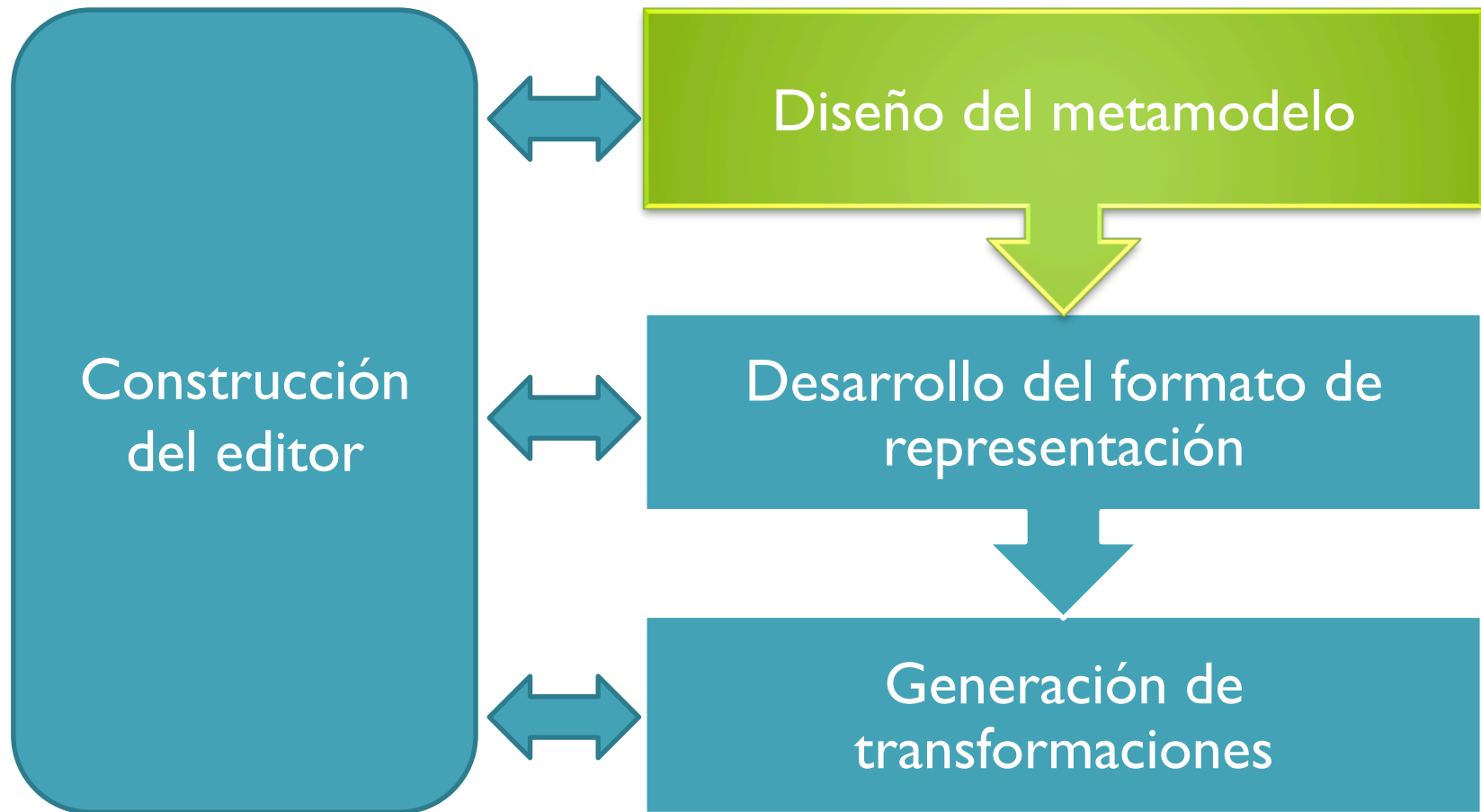


PROCESO DE DESARROLLO

Desarrollo de un procesador de lenguaje (enfoque MDE)



Desarrollo de un procesador de lenguaje (enfoque MDE)



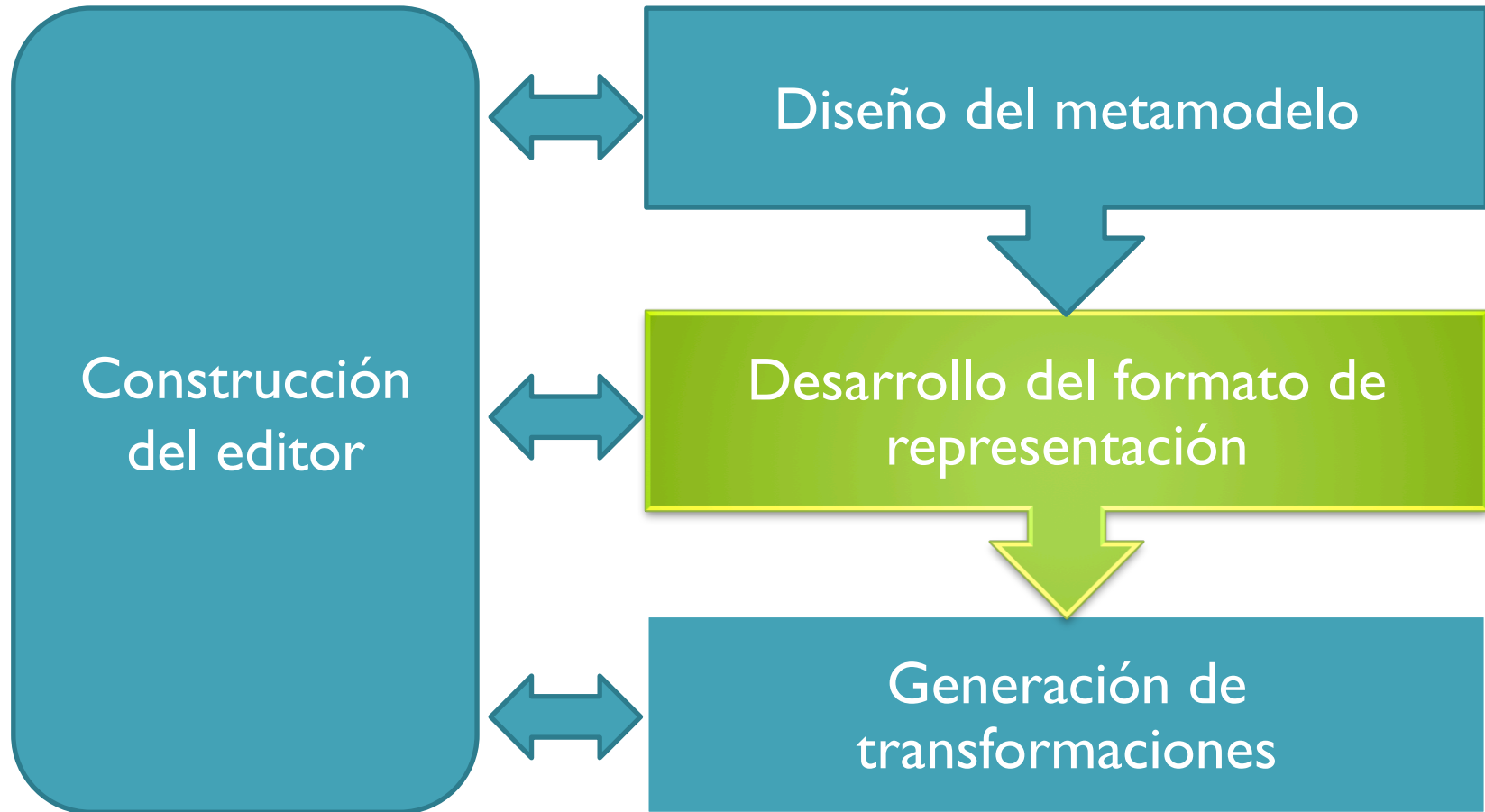
Diseño del metamodelo

- *Metamodelado* es el análisis, diseño y construcción de los metamodelos necesarios para cubrir un determinado tipo de problemas, así como las reglas y restricciones aplicables.
- *Metamodelo*: conjunto de conceptos del dominio a modelar (*metaclases*) y las relaciones entre ellos (*metasociaciones*).

Diseño del metamodelo (II)

- El objetivo es definir los conceptos y relaciones del dominio del problema que queremos abordar, mediante un diagrama de clases.
- Se definen de forma abstracta, independientemente del formato de representación deseado.
- Los metamodelos definirán la *sintaxis abstracta* de nuestro lenguaje.

Desarrollo de un procesador de lenguaje (enfoque MDE)



Desarrollo del formato de representación

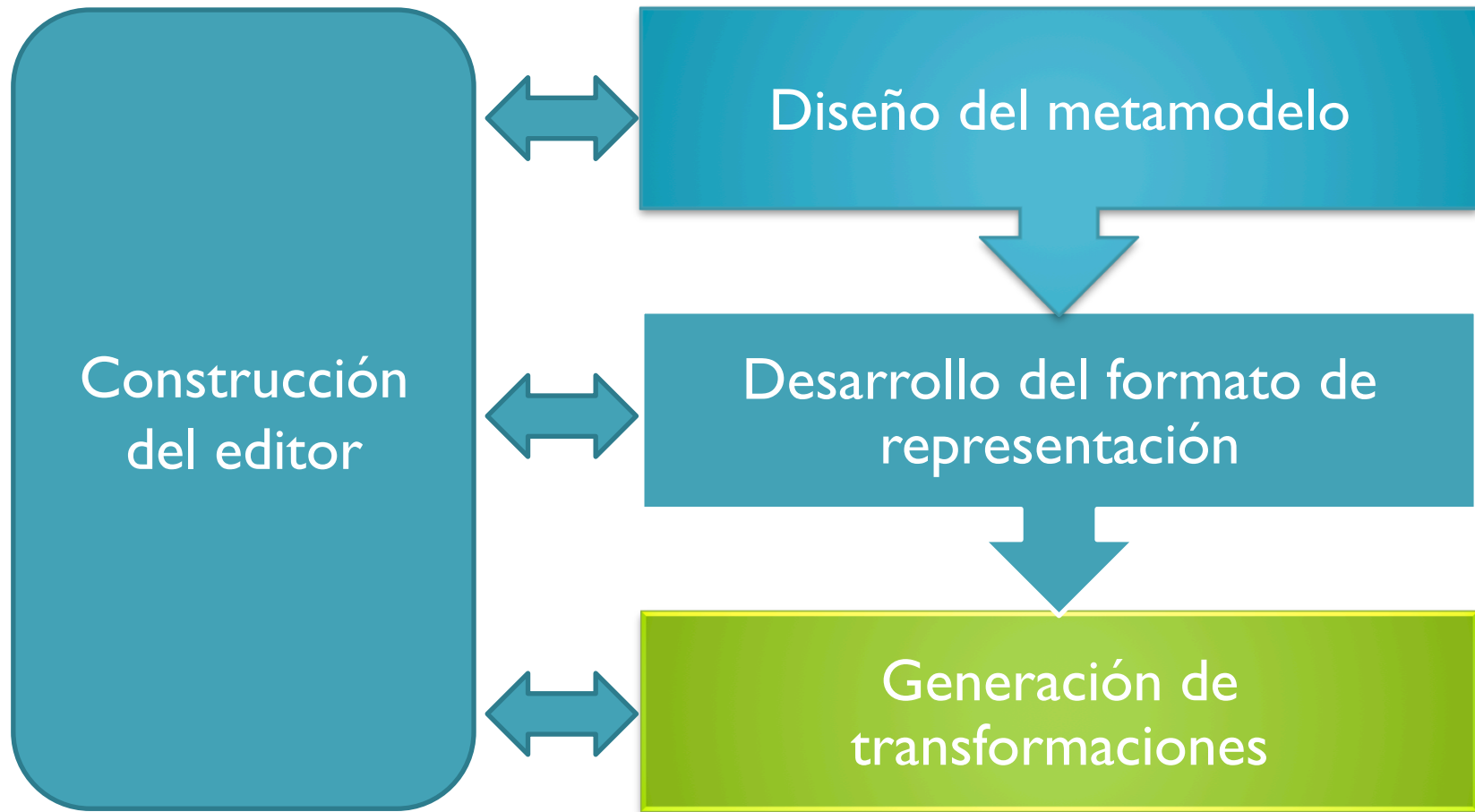
- ¿Cómo se van a representar los elementos del metamodelo? ¿Cual es el formato de representación más adecuado para nuestro DSL?
- ¿Cuál es el tipo de problema que queremos modelar? ¿Quiénes serán los usuarios del lenguaje?
- Los lenguajes visuales son más fáciles de interpretar, pero los textuales son más expresivos.



Desarrollo del formato de representación (II)

- Tenemos que definir la sintaxis concreta para nuestro nuevo lenguaje:
 - **Textual:** basada en una gramática (tipo EBNF)
 - **Visual:** basada en el mapping entre elementos gráficos y elementos de nuestro metamodelo

Desarrollo de un procesador de lenguaje (enfoque MDE)





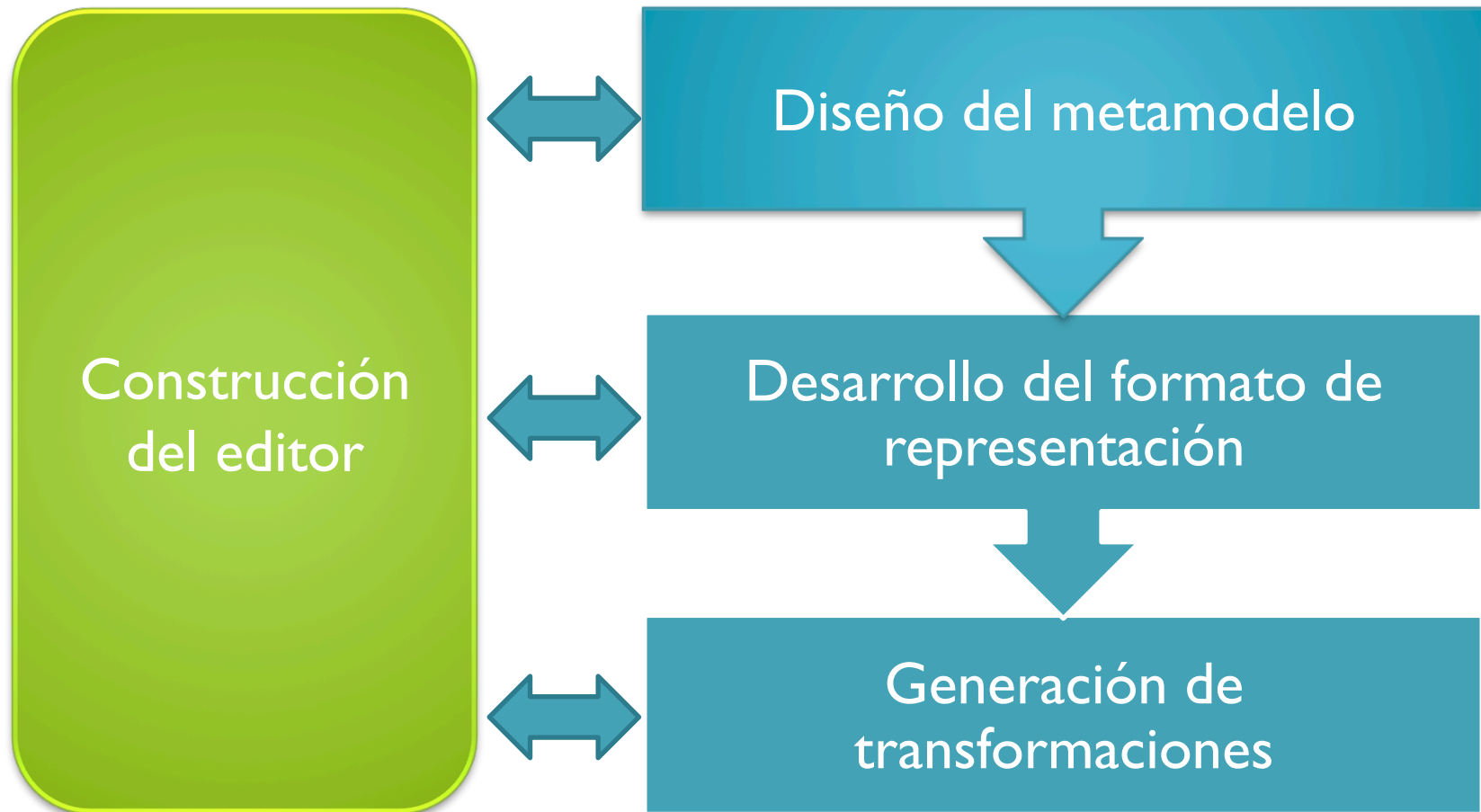
Generación de transformaciones

- Los DSL están concebidos para cubrir una necesidad y automatizar tareas.
- Los modelos creados con nuestros DSL deben ser compilados.
- Los motores de transformación MDE serán nuestros procesadores de lenguajes.
- Las transformaciones se definen mediante un conjunto de reglas.

Generación de transformaciones (II)

- Las transformaciones permiten ‘evolucionar’ modelos.
- Modelo a modelo (M2M)
 - Mismo modelo:
 - Ej: Refactorización o aplicación de un patrón de diseño.
 - Modelo distinto
 - Ej: Derivación de un diagrama de clases para un framework J2EE desde un modelo E/R.
- Modelo a texto (M2T)
 - Ej: Transformación de un diagrama de clases de diseño MVC a un conjunto de clases JAVA.

Desarrollo de un procesador de lenguaje (enfoque MDE)





Construcción del editor

- Necesitamos un entorno integrado que nos permita realizar modelos haciendo uso de nuestro lenguaje.
- Existen diferentes tipos de editores:
 - Editores basados en árbol
 - Editores visuales
 - Editores textuales

Construcción del editor (II)

- ¿Cómo podemos hacer uso de un DSL sin tener en cuenta su formato de representación?
- Un editor basado en árbol puede ser una buena elección para comenzar a desarrollar nuestros DSLs, permitiendo:
 - Creación de instancias de elementos de nuestro lenguaje
 - Asociación de elementos



Construcción del editor (III)

- ¿Cómo utilizar nuestros DSL visuales?
- No nos sirve un programa de dibujo. Es complicado procesar imágenes digitales.
- Necesitamos un entorno de diseño que ofrezca entre otros:
 - Paleta de componentes visuales
 - Área de trazado
 - Validación automática de modelos

Construcción del editor (IV)

- Para utilizar un DSL textual sólo necesitamos conocer su especificación y disponer de un simple editor de texto.
- Sin embargo, sería de gran ayuda disponer de un IDE para nuestro DSL que ofrezca:
 - Coloreado de sintaxis
 - Formato de código
 - Validación instantánea
 - Auto-completado, ...

DESARROLLO DE DSLs USANDO MDE



ENTORNO TECNOLÓGICO

Eclipse Modeling Project (EMP)

The Eclipse Modeling Project focuses on the evolution and promotion of model-based development technologies within the Eclipse community by providing a unified set of modeling frameworks, tooling, and standards implementations.



Subproyectos

- **Abstract Syntax Development**
 - EMF: Eclipse Modeling Framework
- **Concrete Syntax Development**
 - GMP: Graphical Modeling Project
 - TMF: Textual Modeling Framework
- **Model Transformation**
 - M2M: Model-to-Model Transformation
 - M2T: Model To Text Transformation
- **Model Development Tools**
- **Technology and Research**



Procesadores de Lenguajes 2

Desarrollo de DSLs usando MDE

Curso 2013-2014

Iván Ruiz Rube

ivan.ruiz@uca.es