

INDICE

- 1- El desarrollo de software dirigido por modelos (MDD)**
 - . Los modelos de MDD**
- 2- La Arquitectura dirigida por modelos (MDA)**
 - . El proceso de transformación de Modelos**
- 3- El lenguaje MOF**
 - . El lenguaje QVT**
 - . El lenguaje OCL**
- 4- El proyecto Eclipse**

El ciclo de vida Basado en Modelos



Ciclo de vida del desarrollo de software basado en modelos

El ciclo de vida Basado en Modelos

-En las primeras fases se construyen distintos modelos, tales como:

- . **Modelos de los requisitos** (escritos en lenguaje natural)
- . **Modelos de Análisis**
- . **Modelos de Diseño** (Expresados mediante diagramas)

-Las Fases del ciclo de vida pueden realizarse en forma Iterativa –incremental o en forma de cascada.

-En la práctica, **los modelos quedan rápidamente desactualizados debido al atajo que suelen tomar los desarrolladores en el ciclo de vida de este proceso.**

- Existen **problemas** con la productividad, el mantenimiento, la documentación y la falta de flexibilidad a los cambios tecnológicos.

El desarrollo de software dirigido por modelos (MDD)

“El **Model Driven Development (MDD)** mejora el proceso de construcción de software basándose en un proceso guiado por modelos y soportado por potentes herramientas de desarrollo”.

-El **MDD** asigna a los modelos un rol central **los cuales son tan importantes como el código fuente**.

-**Los modelos se van generando** desde los más abstractos a los más concretos, a través de pasos de **Transformación y/o refinamientos**, hasta llegar al código aplicando una última transformación.

-La **Transformación, entre modelos** constituye el motor de MDD.

El desarrollo de software dirigido por modelos (MDD)

- Una de las mayores diferencias entre los ciclos de vida de MDD y el desarrollo clásico está en el **”Tipo de Artefactos”** que se crean durante el proceso de desarrollo.

-En MDD **”los Artefactos son modelos formales”**, es decir modelos que pueden ser comprendidos por una computadora.

.

El desarrollo de software dirigido por modelos (MDD)

-**MDD** identifica distintos tipos de modelos:

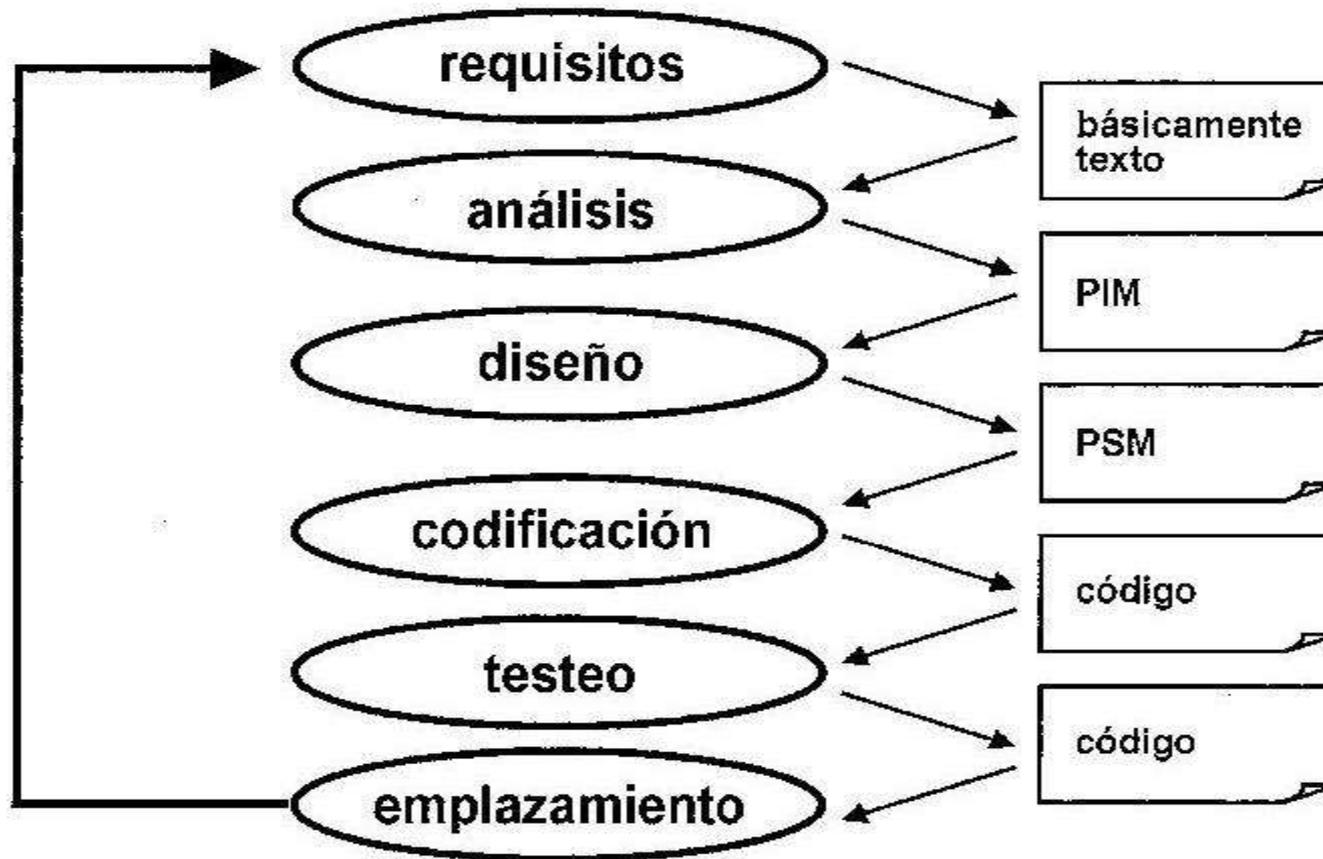
1. Modelos de Alto nivel de abstracción independientes de cualquier metodología computacional, llamados **CIM** (Computational Independent Model)
2. Modelos independientes de cualquier tecnología de implementación llamados **PIM** (Platform Independent Model)
3. Modelos que especifican el sistema en términos de construcciones de implementación disponibles en alguna tecnología específica, conocidos como **PSM** (Platform Specific Model)
4. Modelos que representan el código fuente en sí mismo, identificados como **IM** (Implementation Model).

El desarrollo de software dirigido por modelos (MDD)

-MDD identifica distintos **tipos de modelos**:

1. Los **CIM's** (Computational Independent Model), son una vista del sistema que no muestra detalles de la estructura del mismo. Se lo denomina "Modelo del dominio", y en su construcción se utiliza vocabulario utilizado por los expertos del dominio.
2. Los **PIMs** (Platform Independent Model) son modelos que se independizan de tecnologías y lenguajes de implementación,. El PIM modela el sistema, pensando en cómo se soporta mejor al negocio sin tener en cuenta cómo va a ser implementado, ignorando sistemas operativos, lenguajes de programación, hardware, topología de red, etc.
Un PIM puede ser implementado en cualquier plataforma específica.
3. Los **PSM** (Platform Specific Model) representan la proyección del PIM en una plataforma específica. Por ejemplo un PSM para Java contiene términos como clase. Un PIM puede generar múltiples PSM's que deben colaborar entre sí para lograr una solución completa y consistente interfaz, etc. Un PSM para base de datos relacional contiene términos como tabla, columna, clave foránea, etc.
4. Los Modelos de implementación **IM (Implementation Model)**, representan la transformación directa de cada PSM a código fuente.

Arquitectura dirigida por modelos (MDA)



Ciclo de vida del desarrollo de software dirigido por modelos

El desarrollo de software dirigido por modelos (MDD)

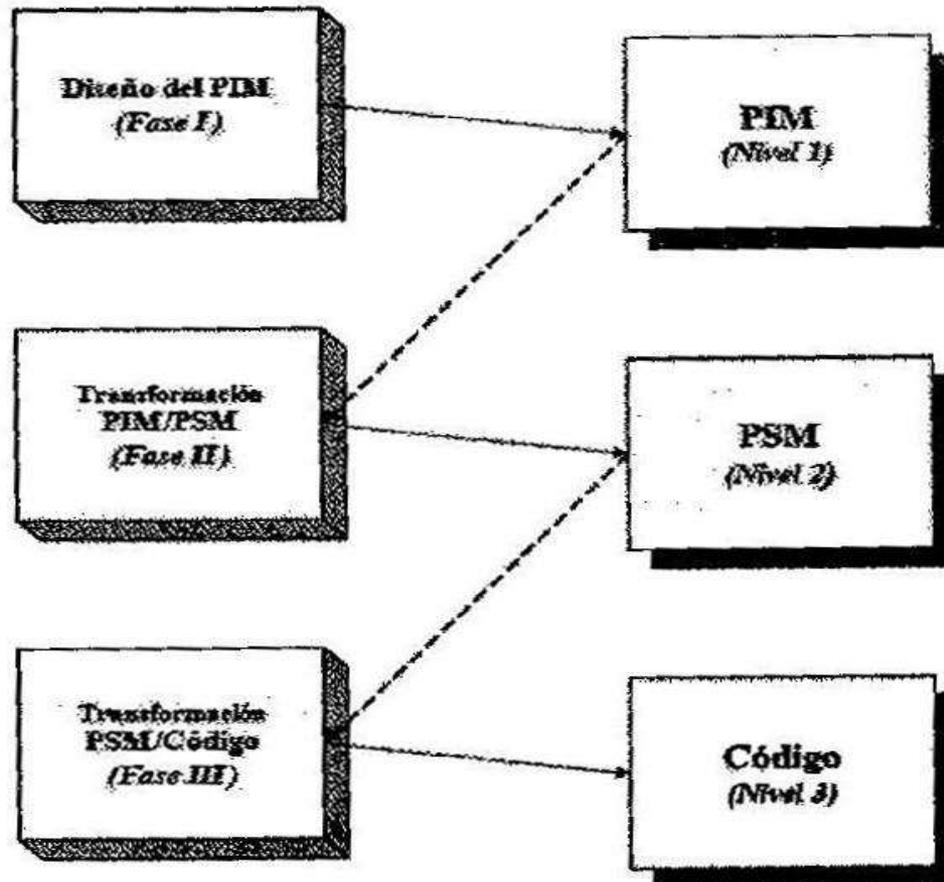
- En el proceso de desarrollo de software tradicional, las transformaciones de modelo a modelo, o de modelo a código son hechas generalmente con la intervención humana.

-En **MDD** las transformaciones son siempre ejecutadas por herramientas.

-Muchas herramientas pueden transformar un PSM a código, lo cual no representa una dificultad debido a la cercanía de ambos, lo novedoso de MDD es la “transformación automatizada entre modelos”, por ejemplo PIM a PSM

.

Arquitectura dirigida por modelos (MDA)



Los tres pasos principales en el proceso de desarrollo MDD.

El desarrollo de software dirigido por modelos (MDD)

-Dos de las propuestas mas conocidas y utilizadas en el ámbito de MDD son:

- 1) **Arquitectura dirigida por Modelos o Driven Model Architecture (MDA)**
- 2) **Modelado específico del dominio o Domain Specific Modeling (DSM)**

Arquitectura Dirigida por Modelos (MDA)

Introducción

- La Arquitectura Dirigida por Modelos (MDA, *Model-Driven Architecture*) es una aproximación definida por el OMG (*Object Management Group*) que promueve el desarrollo de sistemas software basado en modelo de diseño

Arquitectura Dirigida por Modelos (MDA)

Introducción

- Los términos MDA y MDD (*Model-Driven Development*) se suelen utilizar de manera indistinta:

– MDD se suele referir a las actividades llevadas a cabo por los desarrolladores de software

– MDA es la definición formal OMG centrada en crear un marco en el que se pueda llevar a cabo el MDD

Arquitectura Dirigida por Modelos (MDA)

Introducción

- Uno de los principales objetivos de MDA es hacer la especificación de un sistema de manera independiente de los detalles de implementación
- MDA proporciona un marco que permite:
 - 1- Dar una especificación de un sistema con independencia de los detalles de implementación
 - 2- Especificar plataformas de implementación

Arquitectura Dirigida por Modelos (MDA)

Introducción

- 3- Elegir una de estas plataformas
- 4- Transformar la especificación del sistema en una plataforma de implementación

- La independencia se logra utilizando distintos tipos de modelos
- Estos modelos permiten proporcionar distintas especificaciones de sistemas desde distintos puntos de vista

Arquitectura Dirigida por Modelos (MDA)

Conceptos fundamentales

- **Sistema:** colección de componentes organizados para llevar a cabo una función o conjunto de funciones. Dicho sistema puede incluir casi cualquier cosa:
 - Un programa
 - Un sistema basado en un único ordenador
 - Combinación de distintos sistemas
 - Personas

Arquitectura Dirigida por Modelos (MDA)

Conceptos fundamentales

- *Modelo* (de un sistema):
 - Abstracción semánticamente completa de un sistema
 - Representación abstracta de un sistema
 - Descripción o especificación de ese sistema y su entorno para cierto propósito

Arquitectura Dirigida por Modelos (MDA)

Conceptos fundamentales

- *Arquitectura* (de un sistema): estructura organizativa de un sistema que incluye su descomposición en partes, conectividad, mecanismos de interacción y principios de guía que proporcionan información sobre el diseño del mismo

Arquitectura Dirigida por Modelos (MDA)

Conceptos fundamentales

- **Plataforma:** conjunto de subsistemas y tecnologías que proporcionan un conjunto coherente de funcionalidad mediante interfaces y patrones de uso, y que cualquier aplicación soportada por esa plataforma puede utilizar sin preocuparse por los detalles específicos de cómo está implementada la funcionalidad proporcionada por la plataforma

Arquitectura Dirigida por Modelos (MDA)

Conceptos fundamentales

- **Aplicación:** funcionalidad que es desarrollada.
Así un sistema es una o más aplicaciones soportadas por una o más plataformas
- **Metamodelo:** modelo de un lenguaje de modelo. Define la estructura, semántica y restricciones para una familia de modelos (modelos que comparten una sintaxis y semántica común)

Arquitectura Dirigida por Modelos (MDA)

Conceptos fundamentales



Modelos, metamodelos y plataformas

Arquitectura Dirigida por Modelos (MDA)

Conceptos fundamentales

- *Transformación de modelo:* proceso de convertir un modelo en otro modelo del mismo sistema
- *Mapping:* especificación de una transformación de un modelo en otro (normalmente un PIM en un PSM)

Arquitectura Dirigida por Modelos (MDA)

Conceptos fundamentales

- Las transformaciones pueden estar definidas:
 - Entre metamodelos (tipos de modelos)
 - Entre modelos (normalmente utilizando marcas)
 - Combinadas
 - Basadas en plantillas
- Proviengan de metamodelos, o de modelos decorados específicamente, las transformaciones pueden aceptar marcas

Arquitectura Dirigida por Modelos (MDA)

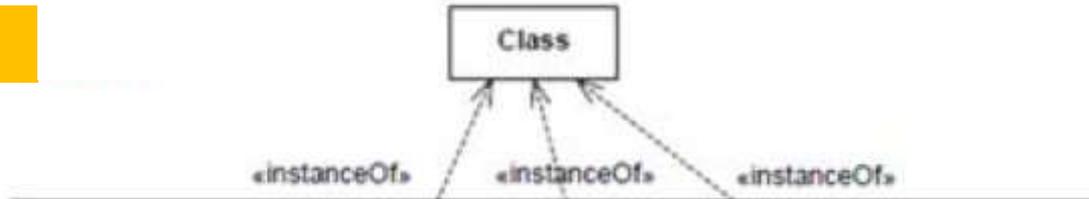
Arquitectura OMG

- OMG ha definido una arquitectura de cuatro niveles para organizar estos conceptos

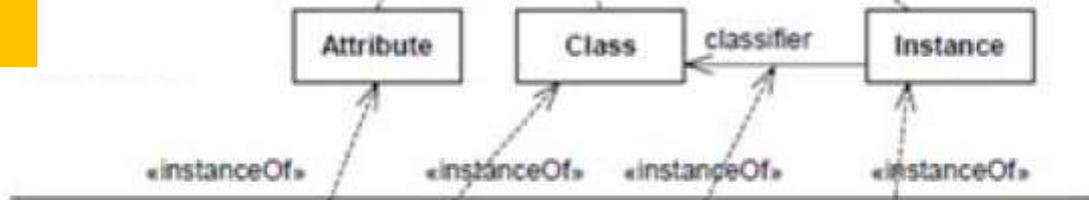
Arquitectura dirigida por modelos (MDA)

Arquitectura OMG

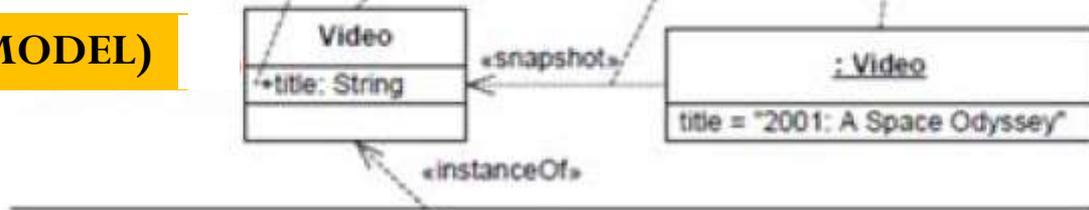
M3 (MOF)



M2 (UML)



M1 (USER MODEL)



M0 (RUN-TIME INSTANCES)



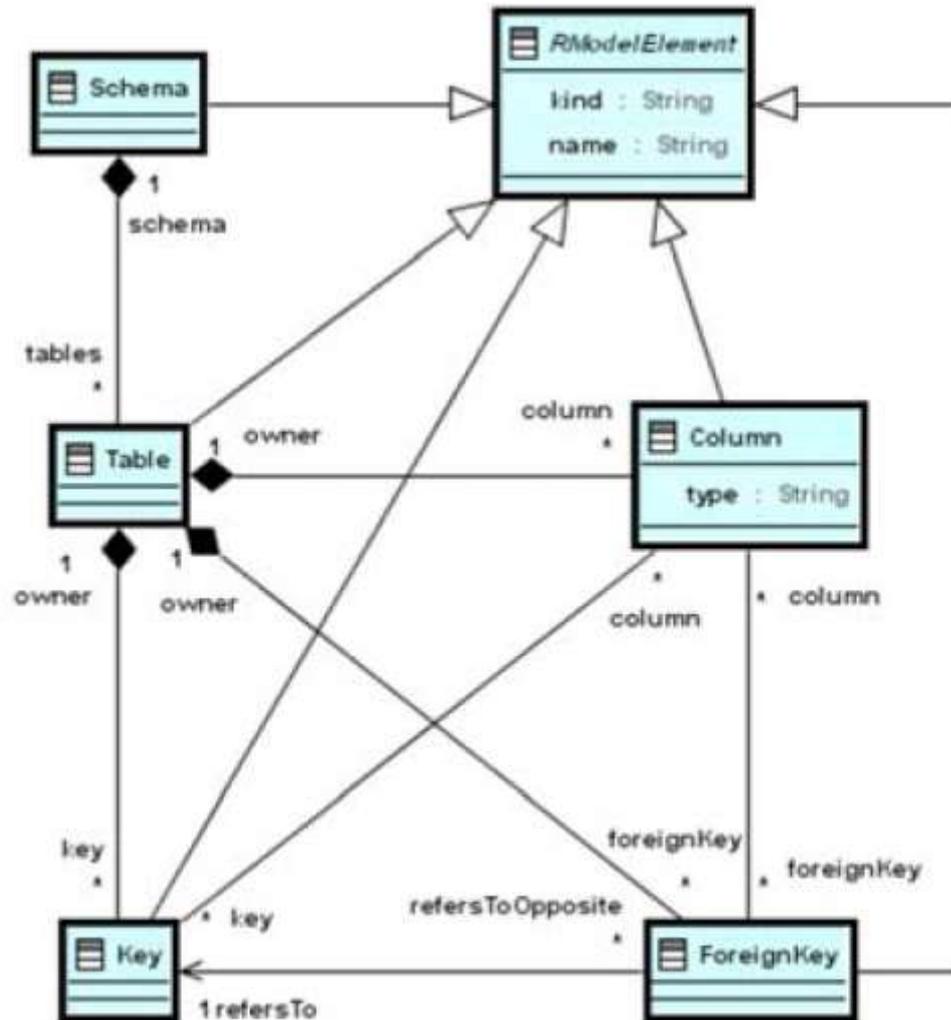
Ejemplo de arquitectura de cuatro niveles de metamodelado OMG

Arquitectura OMG

- M3: meta-metamodelo. Describe las propiedades que pueden tener los metamodelos
- M2: metamodelo. Describen la propiedades que pueden tener los modelos
- M1: Modelo. Describe un modelo de un sistema
- M0: Instancias. Valores del modelo ejecución

Arquitectura dirigida por modelos (MDA)

Arquitectura OMG



Metamodelo básico para el modelo relacional

Arquitectura Dirigida por Modelos (MDA)

Camino a seguir

- Vamos a definir un mecanismo de meta-
metamodelado ↓
Lo utilizaremos para definir metamodelos
- Definiremos también transformaciones entre
estos metamodelos ↓
Así, podremos traducir modelos concretos en
modelos concretos

Arquitectura dirigida por modelos (MDA)

Arquitectura Dirigida por Modelos (MDA)

- En **MDA** la Funcionalidad del sistema es definida en primer lugar como un modelo independiente (**PIM-Platform Independent Model**), a través de un lenguaje específico para el dominio del que se trate.
- En **MDA** surge el concepto de **Platform Definition Model (PDM)**, o **Modelo de Definición de la Plataforma**.
- A partir de un **PDM** (que puede ser CORBA, .NET, Web, etc.) el modelo PIM puede traducirse a uno o más modelos específicos de la plataforma (PSM) para la implementación correspondiente, **usando diferentes lenguajes específicos del dominio**, o también lenguajes de propósito general como Java, C#, Python, etc.

Camino a seguir

