Tabla de contenido

1. Model-Driven Software Engineering in Practice 1

1.1. Principios básicos del MDSE 1

1.2. Visión general 1

# Model-Driven Software Engineering in Practice

## Principios básicos del MDSE

La ingeniería de software orientado por modelos (MDSE) puede ser definido como una metodología [1] para aplicar las ventajas del modelamiento de las actividades de la ingeniería de x software. La metodología del MDSE se puede resumier en los siguientes aspectos:

1. **Conceptos** son los componentes que construyen la metodlogía (artefactos del lenguaje, actores, etc)
2. **Notaciones** Forma de representación de los conceptos (lenguajes utilizados en la metodología)
3. **Procesos y reglas** Toda actividad que resulta en la producción del producto final; reglas para su coordinación y control, y la validación de las propiedades deseadas la finalización del productode los productos o el proceso (consistencia, correctitud, etc)
4. **Herramientas** Aplicaciones que facilitan la ejecución y coordinación de actividades, asegurandose del proceso de producción y apoyo al desarrollador por medio de notaciones.

En el contexto del MDSE, los conceptos básicos son: modelos y transformaciones. Para ver como dichos conceptos interactuan, se revisará la analogía de Niklus Wirth:

Algoritmos + Estructuras de Datos = Programas

En el contexto del MDSE, está fórmula se puede leer de la siguiente forma:

Modelos + Transformaciones = Software.

Los modelos y las transformaciones deben ser expresadas en alguna *notación*, dicha notación es llamada lenguaje de modelamiento (equivalente al lenguaje de programación en la ecuación de Wirth).

El proceso orientado a modelos define cuales son los tipos de modelos, el orden de los mismos y el nivel de abstracción a definir de acuerdo al tipo de software a producir.

Finalmente necesitamos un conjunto de herramientas: \* Entorno integrado de desarrollo (IDE), para definir los modelos y transformaciones \* Compiladores o interpretes para ejecutar los modelos definidos y producir los artefactos finales del software.

En MDSE, todo es un modelo. Por tanto, MDSE tiene un metamodelo (**buscar la definición en otra parte**)

## Visión general

La Figura 2.2 resume los diferentes problemas que resuelve el MDSE, por medio de dos conceptos ortogonáles: conceptualización (columnas), e implementación (filas).

Los problemas de *implementación*, resuelven el problema de correspondencia de los modelos hacia un sistema existe (o uno por construir). Por tanto, define los siguientes conceptos básicos: \* Nivel de modelado: lugar en donde los modelos son definidos. \* Nivel de realización: espacio para definir la implementación de las soluciones a través de artefactos que son utilizados en los sistemas de ejecución. \* Nivel de automatización: Aquí el “mapeo” definido en el modelamiento y en los niveles de realización son puestos en marcha.

Los problemas de *conceptualización* está orientados a definir modelos conceptuales para describir la realidad. Esto puede ser aplicado en tres niveles: \* Nivel de aplicación: Los modelos de la aplicación son definidos, las reglas de transformación son aplicados y los componentes de ejecución son generados. \* Nivel del dominio de la aplicación: La definición del lenguaje de modelamiento, transformaciones y las plataformas de implementación son definidas.

El flujo central del MDSE es partir desde los modelos de alicación hasta la realización de ejecución, a través de subsecuentes transformaciones de modelos. Esto permite la reutilización de modelos y ejecución de sistemas en diferentes plataformas. El niver de realización del software en ejecución, se apoya de la plataforma específica para esta ejecución=

Los modelos son especificados de acuedo al lenguaje de modelamiento, que a su vez son definidos de acuerdo al lenguaje de modelamiento. Las ejecuciones de transformación son definidas en base a un conjunto de reglas de transformación, definidas por medio de un lenguaje específico de transformación.