

Investigación UML

Luis Ángel Serrano Catalá | Prof. Ricardo Vegas Morales

Programación Orientada a Objetos, Unidad 1.

LIDTS 2ºP, Universidad Autónoma de Chiapas.

Tabla de contenidos:

- [Investigación UML](#)
 - [Historia](#)
 - [Metamodelismo](#)
 - [Tipos](#)
 - [Diagrama de clases](#)
-

Historia

Bajo el nombre de "Socios UML", los desarrolladores comenzaron a trabajar con un equipo para completar UML en 1996. Luego se lo entregaron a la Object Management Group (OMG), que introdujo la versión 1.1 de Unified Modeling Language como estándar en 1997. Se necesitaba un lenguaje no sólo para comunicar las ideas a otros desarrolladores sino también para servir de apoyo en los procesos de análisis de un problema. Con este objetivo se creó el Lenguaje Unificado de Modelado (UML: Unified Modeling Language).

Metamodelismo

El metamodelo de UML describe un dominio particular: el propio UML. Por lo tanto, el modelo describe el conjunto de elementos de UML. Se trata de una descripción estática basada en elementos de los diagramas de clases y de objetos. El metamodelo forma parte de UML y se describe a sí mismo.

Tipos

Existen dos tipos principales de diagramas UML: diagramas de estructura y diagramas de comportamiento (y dentro de esas categorías se encuentran varios otros). Estas variaciones existen para representar los numerosos tipos de escenarios y diagramas que usan los diferentes tipos de personas.

Desde clientes y gestores de proyectos hasta autores técnicos, diseñadores, analistas, codificadores y encargados de pruebas y control de calidad, cada rol utilizará un diagrama específico que se adapte a sus necesidades. Eso significa que cada disposición requiere un enfoque y nivel de detalle diferente. El objetivo es que el UML exprese visualmente diagramas que sean fáciles de entender para todos.

Diagrama de clases

Los diagramas de componentes UML representan las relaciones entre los componentes individuales del sistema mediante una vista de diseño estática. Pueden ilustrar aspectos de modelado lógico y físico.

En el contexto del UML, los componentes son partes modulares de un sistema independientes entre sí, que pueden reemplazarse con componentes equivalentes. Son autocontenidos y encapsulan estructuras de

cualquier grado de complejidad. Los elementos encapsulados solo se comunican con los otros a través de interfaces. Los componentes no solo pueden proporcionar sus propias interfaces, sino que también pueden utilizar las interfaces de otros componentes, por ejemplo, para acceder a sus funciones y servicios. A su vez, las interfaces de un diagrama de componentes documentan las relaciones y dependencias en una arquitectura de software.