

INTRODUCCIÓN A UML

¿ Qué es UML ?

UML es un lenguaje gráfico que permite modelar, visualizar y documentar sistemas. Está compuesto por distintos diagramas que permiten ir representando las distintas vistas de un sistema, cada diagrama tiene un objetivo bien definido.

UML significa Unified Modeling Language o Lenguaje Unificado de Modelado, y está basa en tres principios fundamentales:

Es un Lenguaje: está formado por elementos y reglas bien definidas, que poseen su propia sintaxis y semántica

Está Unificado: unifica los distintos criterios utilizados antes de su creación, es decir que toma las mejores propuestas de herramientas previas para presentar una propuesta sumamente abarcativa e integradora

Permite Modelar: está basado en la construcción de modelos que permite representar abstracciones de la realidad.

UML está estrechamente ligado con el paradigma de objetos, lo que permite construir sistemas de información de una forma mucho más intuitiva, integrada y sencilla con el proceso de desarrollo.

UML no es una metodología que presenta los pasos a seguir para realizar un desarrollo, sino que es un lenguaje gráfico de modelado.

¿ Cómo nace UML ?

La historia cuenta que UML da sus primeros pasos con la unión de los “tres amigos”: Booch, Rumbaugh y Jacobson.

En los años 80 cada uno utilizaba un lenguaje propietario aunque como denominador común tenían como objetivo el desarrollo de sistemas, con previa modelización. A partir de los años 90 comienzan a intercambiar ideas para intentar unificar criterios.

Booch es el fundador de Rational Software Corp, y recluta en el año 1995 a Rumbaugh y Jacobson para comenzar a determinar una especificación genérica, sencilla y abarcativa. Así es como las grandes empresas de tecnologías de información – entre ellas Rational - deciden formar un consorcio para la construcción de un Lenguaje Unificado de Modelado: UML. La primer versión de UML, la 1.0, sale a la luz en el año 1997, y a partir de 1998 un organización llamada OMG (Object Management Group) se encargó de generar nuevas revisiones. En el año 1998 UML se establece como Standard de facto en la industria del Software.

¿ Dónde se utiliza ?

UML se utiliza dentro del marco de IT, aunque puede utilizarse en proyectos que no son de tecnología de la información, como ser el modelado de un motor o de una turbina.

En el campo de IT, se utiliza tanto para sistemas monolíticos como para sistemas distribuidos, abarca desde proyectos pequeños hasta grandes proyectos. Permite realizar la integración del software, donde representa el correcto enlace de los roles para lograr el éxito de la construcción del sistema.

En proyectos de software, es utilizado desde la gestación hasta la instalación y el testing.

Si bien para utilizar UML es posible realizar los distintos diagramas con papel y lápiz, es conveniente contar con alguna herramienta del tipo IDE que facilite su construcción, corrección e integración entre diagramas.

INTRODUCCIÓN A LOS DIAGRAMAS DE UML

Introducción

UML está organizado en una serie de diagramas que tienen objetivos bien definidos, con una sintaxis y semántica determinada, que intentan representar / modelar distintas vistas de un sistema.

Diagrama de Clases

El Diagrama de Clases tiene como objetivo describir las clases del dominio y sus relaciones. Permite modelar la estructura del sistema desde un punto de vista estático, modelando las clases desde distintos enfoques de acuerdo a la etapa del proyecto.

Está compuesto por clases, relaciones entre clases y opcionalmente los paquetes que agrupan a las clases.

Clases

Una clase describe un conjunto de objetos con propiedades (atributos) similares y un comportamiento común. Los objetos son instancias de las clases.

No existe un procedimiento inmediato que permita localizar las clases del diagrama de clases. Éstas suelen corresponderse con sustantivos que hacen referencia al ámbito del sistema de información y que se encuentran en los documentos de las especificaciones de requisitos y los casos de uso.

Dentro de la estructura de una clase se definen los atributos y las operaciones o métodos.

Los atributos de una clase representan los datos asociados a los objetos instanciados por esa clase.

Las operaciones o métodos representan las funciones o procesos propios de los objetos de una clase, caracterizando a dichos objetos.

El diagrama de clases permite representar clases abstractas. Una Clase abstracta es una clase que no puede existir en la realidad, pero que es útil conceptualmente para el diseño del modelo orientado a objetos. Las clases abstractas no son instanciables directamente sino en sus descendientes. Una clase abstracta suele ser situada en la jerarquía de clases en una posición que le permita ser un depósito de métodos y atributos para ser compartidos o heredados por las subclases de nivel inferior.

Las clases y en general todos los elementos de los diagramas, pueden estar clasificados de acuerdo a varios criterios, como por ejemplo su objetivo dentro de un programa. Esta clasificación adicional se expresa mediante un Estereotipo. Algunos de los autores de métodos OO, establecen una clasificación de todos los objetos que pueden aparecer en un modelo.

Los tipos son:

- Objetos Entidad
- Objetos límite o interfaz
- Objetos de control.

Éstos son estereotipos de clases. Un estereotipo representa una la meta-clasificación de un elemento. Dependiendo de la herramienta utilizada, también se puede añadir información adicional a las clases para mostrar otras propiedades de las mismas, como son las reglas de negocio, responsabilidades, manejo de eventos, excepciones, etc.

Relaciones

Los tipos más importantes de relaciones estáticas entre clases son los siguientes:

Asociación

Las relaciones de asociación representan un conjunto de enlaces entre objetos o instancias de clases. Es el tipo de relación más general, y denota básicamente una dependencia semántica. Por ejemplo, una Persona trabaja para una Empresa. Cada asociación puede presentar elementos adicionales que doten de mayor detalle al tipo de relación:

Rol, o nombre de la asociación, que describe la semántica de la relación en el sentido indicado. Por ejemplo, la asociación entre Persona y Empresa recibe el nombre de trabaja para, como rol en ese sentido.

Multiplicidad, que describe la cardinalidad de la relación, es decir, especifica cuántas instancias de una clase están asociadas a una instancia de la otra clase. Los tipos de multiplicidad son:

- Uno a uno.
- Uno a muchos.

- Muchos a muchos.

Herencia

Las jerarquías de generalización/especialización se conocen como herencia. Herencia es el mecanismo que permite a una clase de objetos incorporar atributos y métodos de otra clase, añadiéndolos a los que ya posee. Con la herencia se refleja una relación "es_un" entre clases. La clase de la cual se hereda se denomina superclase, y la que hereda subclase. La generalización define una superclase a partir de otras. Por ejemplo, de las clases profesor y estudiante se obtiene la superclase persona. La especialización o especificación es la operación inversa, y en ella una clase se descompone en una o varias subclases. Por ejemplo, de la clase empleado se pueden obtener las subclases secretaria, técnico e ingeniero.

Agregación. La agregación es un tipo de relación jerárquica entre un objeto que representa la totalidad de ese objeto y las partes que lo componen. Permite el agrupamiento físico de estructuras relacionadas lógicamente. Los objetos "son-parte-de" otro objeto completo. Por ejemplo, motor, ruedas, carrocería son parte de automóvil.

Composición. La composición es una forma de agregación donde la relación de propiedad es más fuerte, e incluso coinciden los tiempos de vida del objeto completo y las partes que lo componen. Por ejemplo, en un sistema de Máquina de café, las relaciones entre la clase máquina y producto, o entre máquina y depósito de monedas, son de composición.

Dependencia

Una relación de dependencia se utiliza entre dos clases o entre una clase y una interfaz, e indica que una clase requiere de otra para proporcionar alguno de sus servicios.

Diagrama de Objetos

El Diagrama de Objetos tiene como objetivo describir los objetos del dominio y sus relaciones. Permite representar al sistema en un momento determinado del tiempo, es proporcional a obtener una fotografía o snapshot del sistema en un momento determinado.

Está compuesto por objetos y relaciones de enlace. También es posible pensarlo como una instancia de un Diagrama de Clases.

Diagrama de Casos de Uso

El Diagrama de Casos de Uso tiene como objetivo describir las acciones del sistema desde el punto de vista del usuario. Representa las formas que tiene un usuario de utilizar un sistema, y se puede

utilizar como un "contrato" entre cliente y proveedor de software para determinar la funcionalidad del sistema, es decir los requisitos funcionales.

Está compuesto por actores (agentes externos al sistemas, pueden ser usuarios u otros sistemas), casos de uso y distintos tipos de relaciones. Es posible construir diagramas con diferentes niveles de detalle.

Diagrama de Estados

El Diagrama de Estados tiene como objetivo describir los estados por los cuales puede pasar un objeto durante su ciclo de vida. Permite modelar tanto estas simples como compuestas y concurrentes. Está compuesto por estados, pseudo-estados y transiciones entre estados.

Diagrama de Actividades

El Diagrama de Actividades tiene como objetivo describir las acciones que ocurren dentro de un proceso. Se utiliza principalmente para modelar flujo de trabajo o workflow, con lo cual visualiza las acciones de manera ordenada.

Está compuesto por acciones simples y concurrentes, y transiciones entre las acciones.

Diagrama de Comunicación

El Diagrama de Comunicación tiene como objetivo describir cómo colaboran o se comunican los distintos objetos entre sí para conseguir un objetivo. Se lo suele llamar también Diagrama de Colaboración. Es posible verlo como una extensión del Diagrama de Objetos, ya que es muy parecido pero tiene como valor agregado los mensajes que se envían entre los objetos.

Está compuesto por objetos, relaciones de enlace y relaciones del tipo llamadas, representando qué objeto se comunica con que otro. Es semánticamente equivalente al Diagrama de Secuencia.

Diagrama de Secuencia

El Diagrama de Secuencia tiene como objetivo describir cómo colaboran los distintos objetos entre sí para conseguir un objetivo a lo largo del tiempo. Está directamente relacionado con el Diagrama de Comunicación ya que el objetivo es el mismo, pero tiene la particularidad de estar obligatoriamente ordenado en el tiempo.

Está compuesto por objetos y relaciones del tipo llamadas, representando qué objeto se comunica con que otro.

Es semánticamente equivalente al Diagrama de Comunicación.

Diagrama de Componentes

El Diagrama de Componentes tiene como objetivo describir la relación que existe entre los distintos componentes del sistema. Está directamente vinculado con el diseño del sistema, permitiendo modelar las relaciones e interfaces que existen entre los componentes. Está orientado a la implementación del sistema.

Está compuesto por componentes, interfaces y sus relaciones.

Diagrama de Despliegue

El Diagrama de Despliegue tiene como objetivo describir la arquitectura de un sistema. Es posible representar la arquitectura desde el punto de vista lógico, basándose en la organización del software, o desde un punto de vista físico, representando directamente cada unidad de hardware.

Está compuesto por nodos, componentes y sus relaciones.

CLASIFICACIÓN DE DIAGRAMAS

Categorías

Es posible clasificar a los diagramas según si tienen alguna relación con el tiempo o no. Existen dos posibles categorías, los diagramas estáticos y los diagramas dinámicos.

Diagramas Estáticos

Los Diagramas Estáticos son aquellos que no tienen ninguna relación con el tiempo. Generalmente representan a estructuras o a posibles acciones pero sin una relación directa con el tiempo.

Estos diagramas son:

- Diagrama de Clases
- Diagrama de Objetos
- Diagrama de Casos de Uso
- Diagrama de Componentes
- Diagrama de Despliegue

Diagramas Dinámicos

Los Diagramas Dinámicos son aquellos que tienen relación con el tiempo. Están directamente vinculados con acciones que ocurren bajo cierta secuencia, es decir primero una acción, luego otra, y así sucesivamente.

Estos diagramas son:

- Diagrama de Actividades
- Diagramas de Interacción (es una subcategoría, que incluye a los Diagramas de Secuencia y Comunicación)
- Diagrama de Estado

Diagramas Estructurales

Los Diagramas Estructurales son aquellos que reflejan relaciones estáticas de una estructura.

Estos diagramas son:

- Diagrama de Clases
- Diagrama de Objetos
- Diagrama de Componentes
- Diagrama de Despliegue

Diagramas de Comportamiento

Los Diagramas de Comportamiento son aquellos que reflejan características de comportamiento del sistema o modelan procesos de negocios.

Estos diagramas son:

- Diagrama de Casos de Uso
- Diagrama de Comunicación
- Diagrama de Secuencia
- Diagrama de Actividades
- Diagrama de Estados