

Desarrollo Orientado a Objetos con UML

Xavier Ferré Grau, María Isabel Sánchez Segura
Facultad de Informática – UPM

Índice

I UML	1
I.1 Introducción	1
II NOTACIÓN BÁSICA UML	3
II.1 Modelos	3
II.2 Elementos Comunes a Todos los Diagramas	3
II.2.1 Notas.....	3
II.2.2 Dependencias	3
II.3 Diagramas de Estructura Estática.....	4
II.3.1 Clases	4
II.3.2 Objetos	5
II.3.3 Asociaciones	5
II.3.3.1 Nombre de la Asociación y Dirección.....	5
II.3.3.2 Multiplicidad.....	6
II.3.3.3 Roles.....	6
II.3.3.4 Agregación	7
II.3.3.5 Clases Asociación	7
II.3.3.6 Asociaciones N-Arias.....	7
II.3.3.7 Navegabilidad.....	8
II.3.4 Herencia.....	8
II.3.5 Elementos Derivados.....	9
II.4 Diagrama de Casos de Uso	9
II.4.1 Elementos	9
II.4.1.1 Actores	9
II.4.1.2 Casos de Uso.....	9
II.4.1.3 Relaciones entre Casos de Uso.....	9
II.5 Diagramas de Interacción.....	10
II.5.1 Diagrama de Secuencia.....	10
II.5.2 Diagrama de Colaboración	11
II.6 Diagrama de Estados.....	12
III NOTACIÓN AVANZADA UML	14
III.1 Modelado Dinámico.....	14
III.1.1 Diagramas De Actividades.....	14
III.1.2 Contenido del diagrama de actividades	14
III.1.2.1 Estados de actividad y estados de acción	14
III.1.2.2 Transiciones.....	15
III.1.2.3 Bifurcaciones.....	16
III.1.2.4 División y unión.....	16
III.1.2.5 Calles	16
III.2 Modelado Físico De Un Sistema OO.....	17

III.2.1 Componentes	17
III.2.1.1 Interfaces	18
III.2.1.2 Tipos de componentes	19
III.2.1.3 Organización de componentes.....	19
III.2.1.4 Estereotipos de componentes.....	19
III.2.2 Despliegue	19
III.2.2.1 Nodos.....	19
III.2.2.2 Nodos y componentes.....	20
III.2.3 Diagramas de Componentes	21
III.2.3.1 Algunos conceptos	21
III.2.3.2 Usos más comunes	21
III.2.4 Diagramas de Despliegue.....	22
III.2.4.1 Técnicas más comunes de modelado	22
III.2.5 Arquitectura del Sistema	23
III.2.5.1 Arquitectura de tres niveles	23
III.2.5.2 Arquitectura de tres niveles orientadas a objetos.....	23
III.2.5.3 Arquitectura MULTI-nivel	23
III.2.5.4 Paquetes.....	24
III.2.5.5 Identificación de Paquetes.....	24
IV DESARROLLO ORIENTADO A OBJETOS	25
IV.1 Proceso de Desarrollo.....	25
IV.1.1 Visión General	25
IV.2 Fase de Planificación y Especificación de Requisitos.....	27
IV.2.1 Actividades.....	27
IV.2.2 Requisitos	27
IV.2.3 Casos de Uso	28
IV.2.3.1 Casos de Uso de Alto Nivel.....	28
IV.2.3.2 Casos de Uso Expandidos	28
IV.2.3.3 Identificación de Casos de Uso.....	30
IV.2.3.4 Identificación de los Límites del Sistema	30
IV.2.3.5 Tipos de Casos de Uso	30
IV.2.3.6 Consejos Relativos a Casos de Uso.....	31
IV.2.4 Construcción del Modelo de Casos de Uso	32
IV.2.5 Planificación de Casos de Uso según Ciclos de Desarrollo.....	33
IV.2.5.1 Caso de Uso <i>Inicialización</i>	34
IV.3 Fase de Construcción: Análisis	34
IV.3.1 Actividades.....	34
IV.3.2 Modelo Conceptual.....	35
IV.3.2.1 Identificación de Conceptos	35
IV.3.2.2 Creación del Modelo Conceptual	36
IV.3.2.3 Identificación de Asociaciones	36
IV.3.2.4 Identificación de Atributos	37
IV.3.3 Glosario.....	38
IV.3.4 Diagramas de Secuencia del Sistema.....	38
IV.3.4.1 Construcción de un Diagrama de Secuencia del Sistema.....	39
IV.3.5 Contratos de Operaciones	39
IV.3.5.1 Construcción de un Contrato	40
IV.3.5.2 Post-condiciones.....	41
IV.3.6 Diagramas de Estados.....	41
IV.4 Fase de Construcción: Diseño	41
IV.4.1 Actividades.....	41
IV.4.2 Casos de Uso Reales.....	42
IV.4.3 Diagramas de Colaboración.....	42
IV.4.3.1 Creación de Diagramas de Colaboración.....	42

IV.4.4 Diagrama de Clases de Diseño.....	43
IV.4.4.1 Relaciones de Dependencia para Representar Visibilidad entre Clases	44
IV.4.4.2 Construcción de un Diagrama de Clases de Diseño.....	45
IV.4.4.3 Navegabilidad	46
IV.4.4.4 Visibilidad de Atributos y Métodos	46
IV.4.5 Otros Aspectos en el Diseño del Sistema.....	46
IV.5 Fases de Implementación y Pruebas.....	46
V BIBLIOGRAFÍA.....	47

I UML

I.1 Introducción

UML (*Unified Modeling Language*) es un lenguaje que permite modelar, construir y documentar los elementos que forman un sistema software orientado a objetos. Se ha convertido en el estándar de facto de la industria, debido a que ha sido concebido por los autores de los tres métodos más usados de orientación a objetos: Grady Booch, Ivar Jacobson y Jim Rumbaugh. Estos autores fueron contratados por la empresa Rational Software Co. para crear una notación unificada en la que basar la construcción de sus herramientas CASE. En el proceso de creación de UML han participado, no obstante, otras empresas de gran peso en la industria como Microsoft, Hewlett-Packard, Oracle o IBM, así como grupos de analistas y desarrolladores.



Figura 1 Logo de UML

Esta notación ha sido ampliamente aceptada debido al prestigio de sus creadores y debido a que incorpora las principales ventajas de cada uno de los métodos particulares en los que se basa: Booch, OMT y OOSE. UML ha puesto fin a las llamadas “guerras de métodos” que se han mantenido a lo largo de los 90, en las que los principales métodos sacaban nuevas versiones que incorporaban las técnicas de los demás. Con UML se fusiona la notación de estas técnicas para formar una herramienta compartida entre todos los ingenieros software que trabajan en el desarrollo orientado a objetos.

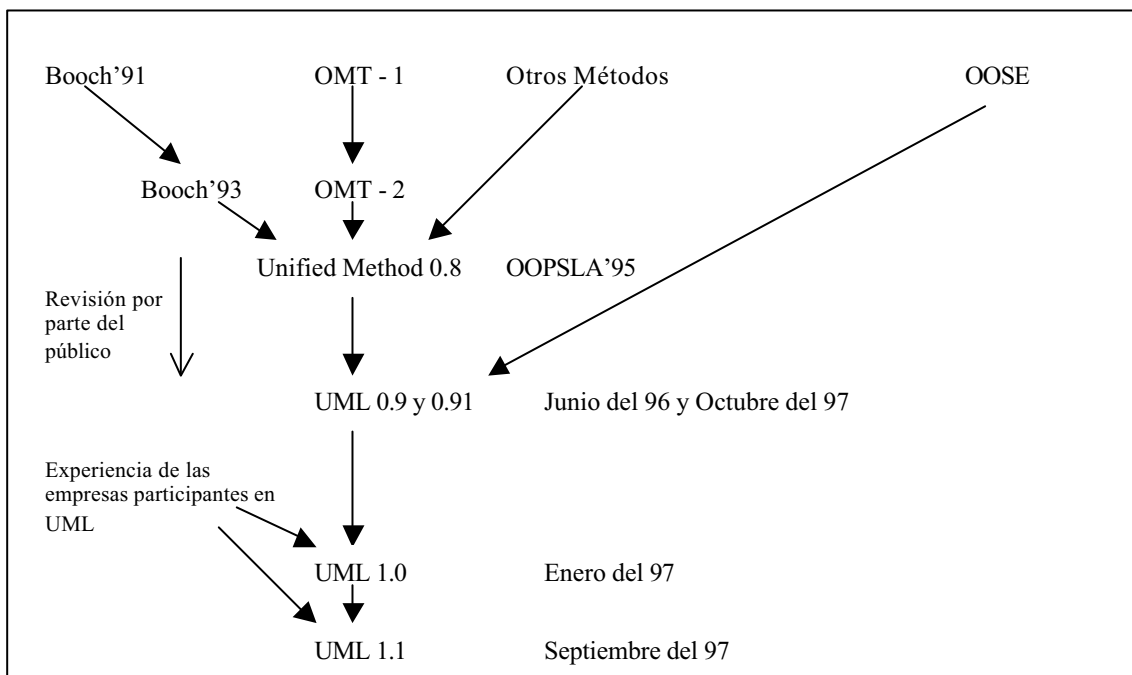


Figura 2 Historia de UML