

Ingegneria del Software

4. Introduzione a UML

Dipartimento di Informatica
Università di Pisa
A.A. 2014/15

e per i modelli iterativi...



- Necessità di un maggior supporto alle attività di analisi e progettazione
- Strumenti automatici di supporto alla realizzazione

realizzazione vs. analisi

- Realizzazione
 - Centrata sul codice
 - Verifica del codice rispetto alla specifica
 - Revisione del codice per risolvere i problemi
- Analisi e progettaz.
 - Centrata sul modello
 - Verifica del modello rispetto ai requisiti
 - Revisione del modello per risolvere i problemi

cosa è un modello

- Astrazioni del sistema (o del dominio) usati per specificarne struttura e comportamento
 - Auto: motore, sterzo, ruote, acceleratore, freno
 - Elettromagnetismo: equazioni di Maxwell
- Diverso dal modello computazionale: il codice è un modello del calcolo eseguito dalla macchina

uso di un modello

- Contiene la conoscenza sul “cosa” e sul “come” di un sistema
 - Strumento di comunicazione e discussione
 - Documentazione
- Fondamentale per un processo di progettazione collaborativo, al centro del moderno sviluppo sw
- Come si modella un sistema?
- Come si rappresenta un modello?

UML e UP

- Unified Modeling Language: linguaggio di modellazione grafico di sistemi software
 - Generale: non è esclusivamente un linguaggio per definire applicazioni sw OO
 - i diagrammi sono generalmente facilmente comprensibili agli utenti con un minimo di conoscenze nel campo
- Unified Process: specifica un processo di sviluppo del sw
 - indica le attività che devono essere eseguite, i manufatti etc etc
- UP descrive attività e “raccomanda” uso di UML. Vivono bene in simbiosi ma l’uno non implica l’altro

breve cronistoria

- . . . 1994: molti linguaggi e metodi di modellazione OO. Esistevano comunque alcuni primi della classe - Booch e Rumbaugh per i linguaggi e Jacobson per le metodologie
- 1994: primo tentativo di unificazione - Fusion
- 1994: Booch e Rumbaugh si uniscono nella Rational Corporation dando vita a quello che chiamano UML
- 1996: Object Management Group (OMG) lancia una prima proposta di standardizzazione di UML
- 1997: OMG approva lo standard UML 1.0
- 2000: UML 1.4 - linguaggi di azione
- 2006: UML 2.0

SU UML

- Principio base di UML è che un sistema software possa essere visto come un insieme di oggetti che collaborano
- Vengono considerati due aspetti fondamentali del sistema
 - Struttura statica: oggetti necessari e relazioni tra questi
 - Comportamento dinamico: come gli oggetti collaborano per raggiungere lo scopo

generalità

- Modellazione di ogni fase del processo di sviluppo
- Applicabile a più tipi di progetti e domini
- Indipendente
 - dal linguaggio di sviluppo
 - dal modello di processo
- Unificazione a livello di linguaggio, non di **metodo!!**

obiettivi

- Visualizzazione
 - Comunicazione e comprensione
- Specifica e documentazione
 - Descrizione del sistema in tutti i suoi livelli
- Realizzazione
 - Supporto all'automazione della codifica

alcuni concetti di UML

- Modello: astrazione di (parte di) un sistema
 - ESEMPIO: Auto
 - il modellino in scala
 - il progetto della stessa
 - la formula per il calcolo dello spazio di frenata
 - Modello statico: descrive gli elementi del sistema e le loro relazioni
 - Modello dinamico: Descrive il comportamento del sistema nel tempo
- Progetto (o anche disegno)
 - Insieme dei modelli: le diverse dimensioni del sistema
 - modelli statici: DB
 - dinamici: sistemi di controllo, di calcolo
- Vista: descrizione di un aspetto di un modello

caratteristiche dei modelli

- Tolleranza a inconsistenze e incompletezza
 - Supporto al dialogo tra le parti interessate
- Meccanismi di strutturazione (package)
- Personalizzazione mediante stereotipi
- Strumenti di supporto disponibili sul mercato

come usare UML

- Un progetto può essere realizzato come
 - Abbozzo (sketch)
 - Progetto dettagliato (blueprint)
 - Eseguibile (UML come linguaggio di programmazione)
- Secondo due prospettive
 - Software: elementi UML corrispondono a elementi sw
 - Concettuale: elementi UML corrispondono a elementi del dominio

un recap sui linguaggi

- Linguaggio formale = sintassi + semantica
- Sintassi
 - Le regole attraverso le quali gli elementi del linguaggio (ad esempio le classi) sono raggruppati in espressioni (ad esempio i diagrammi)
- Semantica
 - Le regole che assegnano un significato alle espressioni sintattiche

meta-modello di UML

- Per specificare la sintassi e la semantica
- Gli utenti sono i costruttori dei tool e i modellatori esperti
- UML descrive il proprio meta-modello in UML (come alcuni compilatori sono utilizzati per compilare se stessi)
- Stile semi-formale che combina linguaggio naturale e formale
- Esiste una rappresentazione XML del metamodello che si chiama XMI (XML Metadata Interchange)

elementi base di UML

- Entità
 - classi, interfacce, componenti, casi d'uso, ...
- Relazioni
 - associazioni, generalizzazioni, dipendenze, ...
- Diagrammi
 - delle classi, dei casi d'uso, d'interazione, ...

modello statico

- I concetti del dominio
 - modellati mediante classi e relazioni
- Realizzazione del sistema
 - classi di realizzazione, componenti e nodi
- Assenza degli aspetti dipendenti dal tempo

modello dinamico

- Modella il comportamento delle entità descritte nel modello statico

diagrammi

- Rappresentazione grafica
 - di un insieme di elementi del modello
 - secondo una certa vista
- Grafo
 - Vertici = elementi del modello
 - Archi = relazioni fra elementi

diagrammi...

- ...delle classi
- ...delle componenti
- ...di struttura composita
- ...di macchina a stati
- ...dei casi d'uso
- ...di attività
- ...di sequenza
- ...di dislocazione
- ...dei package

diagrammi e modelli

- I diagrammi si usano per costruire i modelli
- I diagrammi non sono i modelli
 - Un modello è un grafo di elementi *semantici*
 - Un diagramma è un grafo di elementi *visuali* che rappresentano gli elementi semantici
- Negli strumenti di supporto
 - Un modello è la struttura sulla quale si opera
 - Un diagramma è una struttura di presentazione
 - Non eliminare un elemento solo da un diagramma

classificatori vs istanze

- Un classificatore modella un concetto che descrive istanze
 - Esempio: una classe modella oggetti
- Alcuni diagrammi possono essere a livello di classificatore o a livello di istanza
- I classificatori sono
 - Classi
 - Attori
 - Casi d'uso
 - Componenti
 - Nodi (ex hw)...

annotazioni: commenti e vincoli

- Alcuni dettagli del modello si esprimono in forma testuale
- Commenti e vincoli sono annotazioni nei diagrammi
 - In OCL (Object Constraint Language), il linguaggio di vincoli di UML
 - In linguaggio naturale
- Attaccati ad un elemento del modello (o a una lista di elementi), con linea tratteggiata
- I vincoli tra parentesi graffe, i commenti senza graffe
- I vincoli sono elementi del modello, i commenti no



Questa è una nota.

stereotipi

- Primitiva di UML comune ad ogni diagramma.
 - Rende un diagramma più informativo arricchendo la semantica dei costrutti UML.
- Uno stereotipo è una parola chiave tra virgolette e abbinata ad un elemento del modello.
 - Esempi: «import», «utility», «interface»

stereotipi cont'd

- Forniscono significato aggiuntivo ai costrutti UML
- Possono essere usati per adattare UML a particolari ambiti e piattaforme di sviluppo
- Sono definiti nei *profili*, che costituiscono uno dei principali meccanismi di estensione di UML
- Alcuni profili disponibili
 - CORBA
 - J2EE
 - SysML