

Modellazione di architetture software

introduzione a UML e MDA

Lezione 1 di Ingegneria del software 2

Docente: Giuseppe Scollo

Università di Catania, sede di Comiso (RG)

Facoltà di Scienze Matematiche, Fisiche e Naturali

Corso di Studi in Informatica applicata, AA 2006-7

Indice

1. Modellazione di architetture software
2. l'architettura nel progetto software
3. UML: cosa è, cosa non è
4. cenni preistorici: genealogia essenziale di UML
5. cenni storici: evoluzione di UML
6. perché si usa UML: pragmatica di UML, "prospettive"
7. MDA, Executable UML: linguaggi di programmazione?
8. modelli, metamodello di UML: sintassi, semantica
9. classificazione dei tipi di diagrammi UML
10. valore di modelli: descrittivo, prescrittivo, normativo
11. subito all'opera: quick start

L'architettura nel progetto software

- **progetto dell'architettura:** fase dello sviluppo, ma ...
- ... l'astrazione architetture è **iterabile** a parti del sistema
- **utilità** delle architetture software:
 - riconoscimento di **tratti comuni** a sistemi diversi
 - supporto ad **analisi**, valutazione di alternative, **decisioni**
 - **riusabilità** per implementazioni differenti
 - **documentazione** ad un livello di astrazione appropriato

UML: cosa è, cosa non è

- notazione "**grafica**" per la modellazione di sistemi "**software**" ad "**oggetti**"
"..." = non solo ...
- linguaggio di modellazione **unificato** : uno **standard**
- **modelli UML** : **descrizioni** di sistemi, da realizzare o esistenti
- UML è uno **strumento di comunicazione**
- UML **non è** una metodologia di sviluppo né di documentazione del software
ma può ben essere adoperato in tali **contesti**
- UML + metodologie → strumenti

cenni preistorici: genealogia essenziale di UML

- anni '60: classe, oggetto in *Simula 67*
- anni '70: incapsulamento, *ADT*
- anni '80: linguaggi OO "crescono": *SmallTalk*, *C++*
- anni '90: proliferazione dei "metodi OO"
ci si metta d'accordo almeno sulla notazione!
- **OOPSLA'95**
OOAD (Booch) + OMT (Rumbaugh) + Objectory (Jacobson)
→ UML 0.8 (Rational Inc., ora IBM)

cenni storici: evoluzione di UML

- 1997: **OMG (Object Management Group)**:
UML 1.0 (proposta Rational) →
UML 1.1 (standard, OCL \subseteq UML 1.1)
- uno standard è "buono" quando rende tutti "egualmente infelici" (!?)
- dicembre 1998: UML 1.2 (cosmesi)
- giugno 1999: UML 1.3 (semantica dei diagrammi di attività)
- maggio 2001: UML 1.4 (profili, elaborati (componenti))
- marzo 2003: UML 1.5 (semantica delle azioni)
- 2004→2005: UML 2.0 (molte innovazioni)

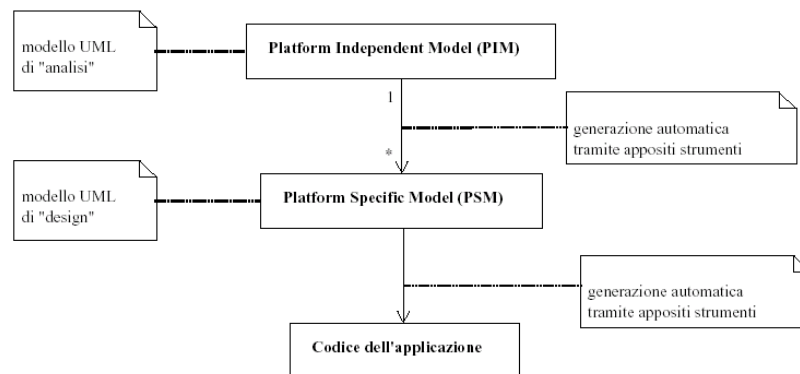
perché si usa UML: pragmatica di UML, "prospettive"

- rappresentazione del sistema a vari livelli di astrazione:
bozza, progetto dettagliato, eseguibile
- "prospettive":
concettuale, specifica, implementazione
- ➤➡ forward engineering (costruzione)
- ➡➤ reverse engineering (documentazione)
- ➡➡➡ round-trip engineering (evoluzione)

MDA, Executable UML: linguaggi di programmazione?

MDA: Model Driven Architecture (standard OMG)

scopo: costruzione automatica del software da modelli UML



tratta da: Adriano Comai (2004) Introduzione a UML, <http://www.analisi-disegno.com>

l'uso di UML non implica quello di MDA

Executable UML: no PSM, sottolinguaggio compilabile di UML

modelli, metamodello di UML: sintassi, semantica

UML definisce:

- una **notazione** (grafica):
elementi di **sintassi** dei modelli
- un **metamodello** di UML stesso:
sintassi astratta del linguaggio
+
regole (semantiche?) in *OCL* che la vincolano
+
rappresentazioni grafiche degli elementi astratti

classificazione dei tipi di diagrammi UML

diagrammi strutturali:

delle classi
degli oggetti
dei componenti
di struttura composita
di deployment
di package

diagrammi comportamentali:

dei casi d'uso
delle attività
di macchina a stati
di interazione:
di sequenza
di comunicazione
di interazione generale
di temporizzazione

valore di modelli: descrittivo, prescrittivo, normativo

nel forward engineering la **prospettiva** semantica di un modello UML ne determina approssimativamente il **valore**:

concettuale ⇒ **descrittivo**

di specifica o implementazione ⇒ **prescrittivo**

nel reverse engineering il valore è **descrittivo**

qual è il valore dello standard UML?

normativo = **prescrittivo & universale**

sì, però con molti contenuti descrittivi e norme non vincolanti, ad esempio vale la regola generale che in un modello UML

qualsiasi informazione può essere soppressa

N. B.: spesso, questa regola può dar luogo ad **ambiguità**, ad es. quando sono definiti dei valori di **default**: l'assenza significa soppressione o presenza del valore di default?

subito all'opera: *quick start*

- perché adoperare un testo tanto "striminzito" che l'autore lo intitola "distillato" dello standard?
 - una risposta non unica: la **legge di Pareto**
 - **esercizio**:
determinare altre buone ragioni per tale scelta

- oltre alla documentazione ufficiale e ai testi consigliati, può essere utile consultare:

Linee Guida UML - Modelli Architetture

<http://www.analisi-disegno.com/LineeGuida.htm>

di Adriano Comai, di orientamento pragmatico, rivolto alla pratica progettuale

- **il laboratorio vi aspetta...**

BOUML è già installato! Per saperne di più: <http://bouml.free.fr/doc>