



# UML2

## Concetti base

Andrea Polini

Laboratorio di Ingegneria del Software  
Corso di Laurea in Informatica – L31  
Università di Camerino

# Cos'è l'UML?

- UML è un linguaggio di **modellazione e specifica** che si basa sul paradigma object-oriented.
- UML è standardizzato. La specifica è gestita da OMG.
- La versione corrente dello standard è la 2.5 rilasciata a Marzo 2015  
<http://www.omg.org/spec/UML/2.5> – Versione 2.0 rilasciata nel 2005.
- si tratta di un **linguaggio grafico semi-formale**.
- UML è certamente la notazione più utilizzata per la modellazione del software in un contesto industriale.
- Esistono molti strumenti CASE che permettono di utilizzare il linguaggio UML
- Formato di **interscambio è XMI**. CASE non sono sempre in grado di interoperare
  - Visual Paradigm (consigliato per il progetto)
  - MagicDraw
  - StarUML
  - Eclipse Modeling
  - Papyrus
  - ...

[https://en.wikipedia.org/wiki/List\\_of\\_Unified\\_Modeling\\_Language\\_tools](https://en.wikipedia.org/wiki/List_of_Unified_Modeling_Language_tools)

# Cosa e come utilizzarlo?

UML permette di modellare il sistema (tipicamente software) attraverso un approccio object-oriented i.e. il sistema è visto come un **insieme di entità che possono interagire** UML può essere usato secondo diversi approcci al progetto

- UML per definire la bozza del progetto
- UML come progetto
- UML per derivazione automatica di codice

# Cosa e come utilizzarlo?

È possibile rappresentare due differenti caratteristiche di un sistema:

- **Struttura Statica**: elementi necessari a modellare il sistema e come sono correlati
- **Comportamento Dinamico**: ciclo di vita degli oggetti e specifica delle collaborazioni che intercorrono tra gli oggetti per fornire le funzionalità richieste.

Concetto intuitivo di oggetto: **entità a cui sono associati dei dati e che fornisce funzionalità per l'accesso e la manipolazione dei dati stessi**

# Struttura di UML

La struttura di UML è composta di 3 elementi fondamentali:

- **Costituenti Fondamentali**: entità, relazioni e diagrammi
- **Meccanismi Comuni**: tecniche comuni per raggiungere specifici obiettivi
- **Architettura**: il modo in cui UML esprime l'architettura di un sistema








# Costituenti Fondamentali - entità

Le entità possono essere classificate in:

- strutturali
- comportamentali
- raggruppamento
- informative

# Costituenti Fondamentali - Relazioni

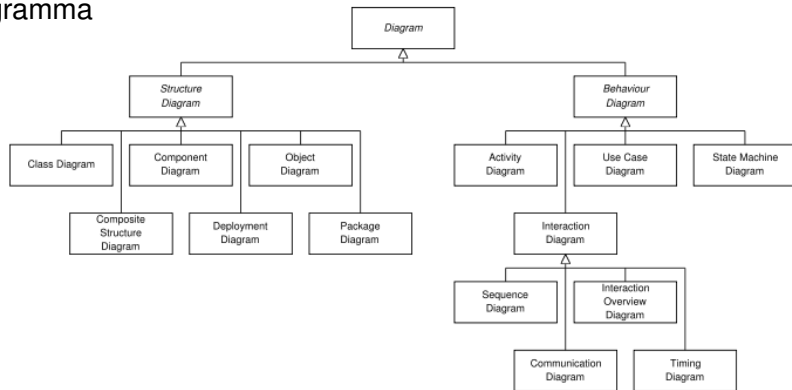
Le relazioni specificano che due o più entità sono correlate:

Tipo di relazione	Sintassi UML sorgente destinazione	Descrizione	Paragrafo
Dipendenza		L'elemento sorgente dipende dall'elemento di destinazione e può essere influenzato dai cambiamenti eseguiti su di esso	9.5
Associazione		La descrizione di un insieme di collegamenti tra oggetti	9.4
Aggregazione		L'elemento di destinazione è parte integrante dell'elemento sorgente	16.4
Composizione		Una forma forte (più vincolata) di aggregazione	16.5
Contenimento		L'elemento sorgente contiene l'elemento di destinazione	11.4
Generalizzazione		L'elemento sorgente è una specializzazione dell'elemento di destinazione più generale e può quindi sostituirlo.	10.2
Realizzazione		L'elemento sorgente garantisce di eseguire il contratto specificato dall'elemento di destinazione	12.3

# Costituenti Fondamentali - Diagrammi

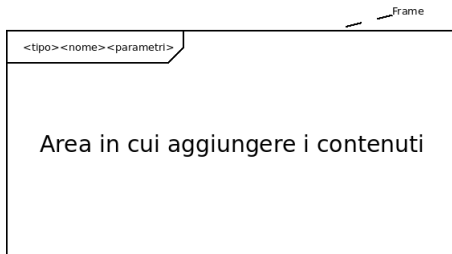
Un diagramma può considerarsi una *vista* che va a specificare una parte del sistema che si vuole modellare. Rimuovere un'entità da una vista non necessariamente la elimina dal modello.

UML fornisce la possibilità di utilizzare 13 diverse tipologie di diagramma





# Come definire un diagramma



# Meccanismi comuni

**Specifiche:** Il diagramma oltre alla parte grafica direttamente visibile è costituito dalle informazioni testuali che lo rappresentano e gli danno semantica. Queste informazioni sono ciò che formano realmente la specifica. La specifica presenta requisiti di *Completezza e Consistenza*

**Ornamenti:** ogni elemento ha una versione base ma può in generale essere arricchito per poter evidenziare informazioni aggiuntive

**Distinzioni comuni:** classificatore/istanza, interfaccia/implementazione

**Meccanismi di estendibilità:** vincoli, stereotipi, valori etichettati, profili

UML

# Meccanismi di estendibilità

- **Vincoli**: frase di testo racchiusa tra graffe che definisce una condizione/regola che deve risultare vera. OCL (Object Constraint Language) è il linguaggio associato ad UML per la definizione formale dei vincoli.
- **Stereotipi**: variazione di un elemento di modellazione esistente
- **Valori etichettati**: sono attributi che vengono definiti per il modello o i suoi elementi ed a cui è possibile assegnare un valore.
- **Profili**: insieme di vincoli, stereotipi e valori etichettati che permettono di personalizzare il linguaggio.

**Architettura:** *“struttura organizzativa di un sistema, inclusa la sua scomposizione in parti, la loro connettività, l’interazione, i meccanismi ed i principi guida che insieme formano il progetto del sistema stesso”*

Philippe Kruchten's 4+1 view model

