



UNIVERSITATEA  
DIN BUCUREȘTI

# Metode de dezvoltare software

---

Diagrame UML introducere

11.03.2020

Alin Ștefănescu



# UML

# UML... în practică



# De ce “modelăm”?

- complexitatea e o problemă în dezvoltarea programelor.
- folosirea unor modele poate înlesni abordarea complexității.
- un **model** este o reprezentare abstractă, de obicei grafică, a unui aspect al unui sistem.
- acesta permite o mai bună înțelegere a sistemului și analiza unor proprietăți ale acestuia.



# O scurtă istorie a modelării

- modele “statice” (structurale): apărute destul de devreme, ca desene, nu obiecte.
- modele “dinamice”: d.ex. flowcharts (Gilbreth 1921), automate finite (McCulloch-Pitts 1943), statecharts (Harel 1980), diagrame de secvențe (1990) etc.
- astfel, diverse tipuri de modele pentru sisteme (software) au fost studiate de-a lungul timpului.
- însă modelarea a devenit foarte vizibilă după introducerea paradigmei “orientate pe obiecte” (OO).

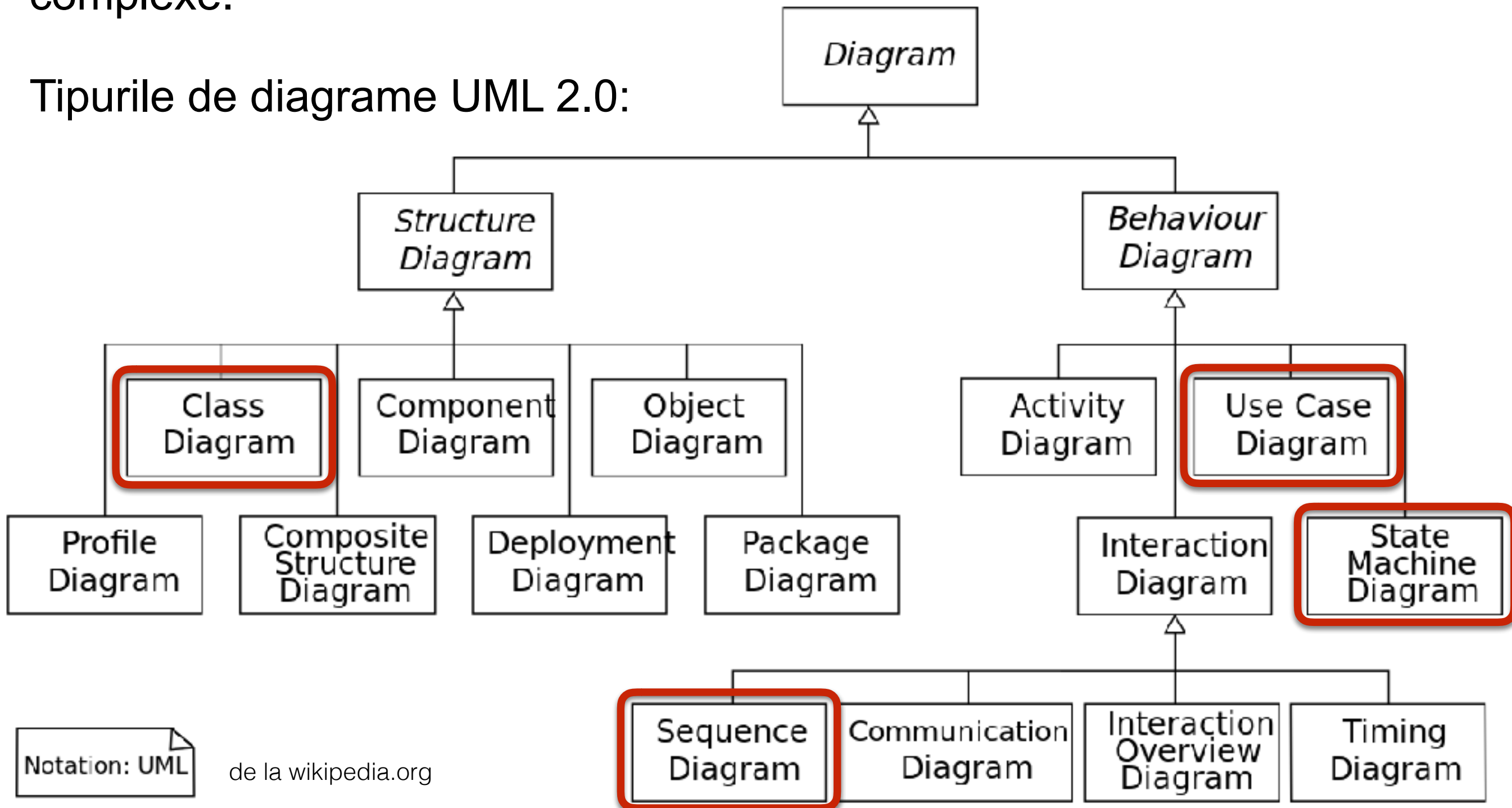
# Modelarea orientată pe obiecte

- În anii '80 și începutul anilor '90 a avut loc o dezvoltare foarte puternică a paradigmei OO.
- o mulțime de experți OO, fiecare cu compania, tool-ul, cartea și modul de modelare propriu
- printre ei și Booch, Rumbauch, Jacobson, denumiți “the three amigos”, cei care au inițiat Unified Modeling Language (UML)
- Consorțiul OMG (Object Management Group) a reușit să standardizeze UML:
  - 1997 - UML 1.0
  - 2015 - UML 2.5

# Cele 14 tipuri de diagrame UML

UML este un **limbaj grafic** pentru vizualizarea, specificarea, construcția și documentația necesare pentru dezvoltarea de sisteme software (OO) complexe.

Tipurile de diagrame UML 2.0:



# Motive pentru care UML nu este folosit

- nu este cunoscută notația UML
- UML e prea complex (14 tipuri de diagrame)
- notațiile informale sunt suficiente
- documentarea arhitecturii nu e considerată importantă



# Motive pentru care UML este folosit

- UML este standardizat
- existența multor tool-uri
- flexibilitate: modelarea se poate adapta la diverse domenii folosind “profiluri” și “stereotipuri”
- portabilitate: modelele pot fi exportate în format XMI (XML Metadata Interchange) și folosite de diverse tool-uri
- se poate folosi doar o submulțime de diagrame
- arhitectura software e importantă

# Tipuri de folosire UML

UML e folosit în diverse moduri în proiecte sau organizații:

- diagrame UML pentru **a schița doar** diverse aspecte ale sistemului
- diagrame UML care **apar în documente** (uneori după ce a fost făcută implementarea)
- diagrame UML **foarte detaliate** sunt descrise în tool-uri înainte de implementare și apoi **cod este generat** pe baza acestor modele

# Tool-uri UML

Există foarte multe tool-uri pentru UML:

`http://en.wikipedia.org/wiki/List\_of\_UML\_tools`

Câteva tool-uri gratuite (care mi s-au părut ok):

- **LucidChart, Creately** (web-based, mobile, etc.) - **recomandate**
- **Microsoft Visio** (gratuit pentru studenți prin Dreamspark)
- **Astah** (community edition, gratuit)  
`http://astah.net/download#community`
- **Visual Paradigm for UML** - (community edition, gratuit)  
`http://www.visual-paradigm.com/download/community.jsp`