

MODELE DE DEZVOLTARE SOFTWARE - 1

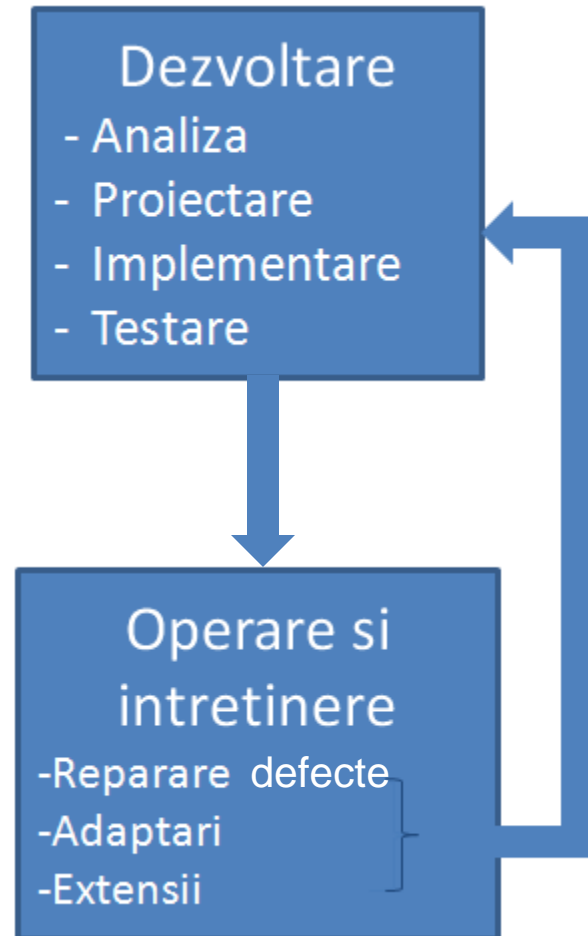
Prof. univ. dr. ing. Florica Moldoveanu

Ingineria programelor
UPB, Automatică și Calculatoare
2020-2021

Ciclul de viata al unui program

(Software life cycle)

➤ O secvență ciclică de etape în existența programului, care include toate activitățile necesare pentru dezvoltarea, operarea și întreținerea sa.



Modele ale procesului de dezvoltare software

Software development life cycle (SDLC) models

Model de dezvoltare: descriere generală a procesului de dezvoltare (ghid)

Scopuri:

- Optimizarea procesului de dezvoltare, pentru **obținerea de produse de calitate, în mod reproductibil**, în limitele costurilor și termenelor de livrare prevăzute.
- Minimizarea riscurilor de eșec al proiectului.

Doua abordări distincte:

- Dezvoltarea liniară (într-un singur ciclu de dezvoltare):

- Modelul cascadă (Waterfall), modelul în V, dezvoltarea pe baza de prototip.

Dezvoltarea iterativă și incrementală (în mai multe cicluri de dezvoltare):

- RUP (Rational Unified Process), modele agile (SCRUM, XP, AUP și altele).

Modelul “In cascada” (Waterfall)

- Clasic si cel iterativ - (1)

- Procesul de dezvoltare: succesiune de faze ce se înlanțuie linar, de la analiza cerintelor pana la livrarea la client.

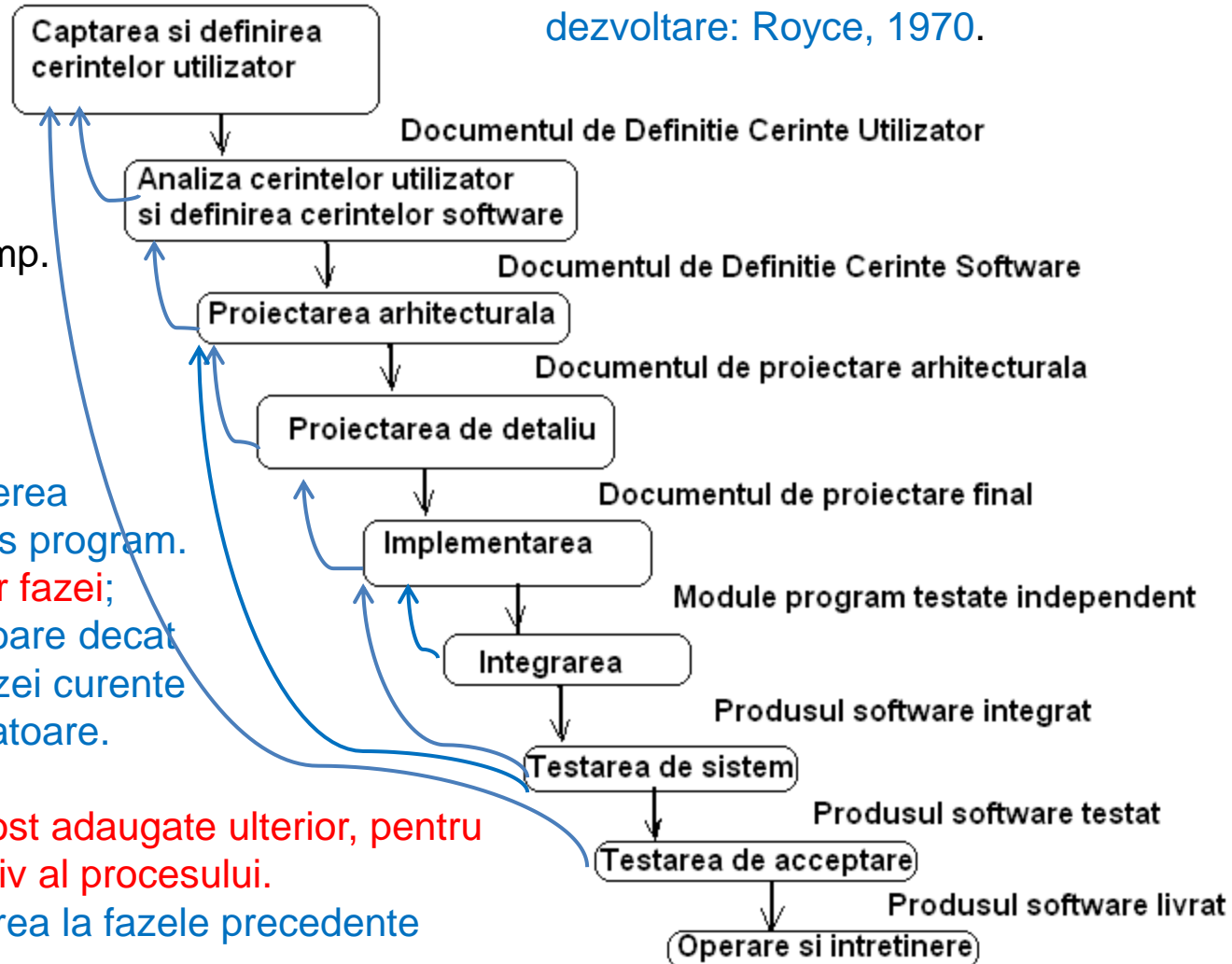
- Fazele sunt planificate în timp.

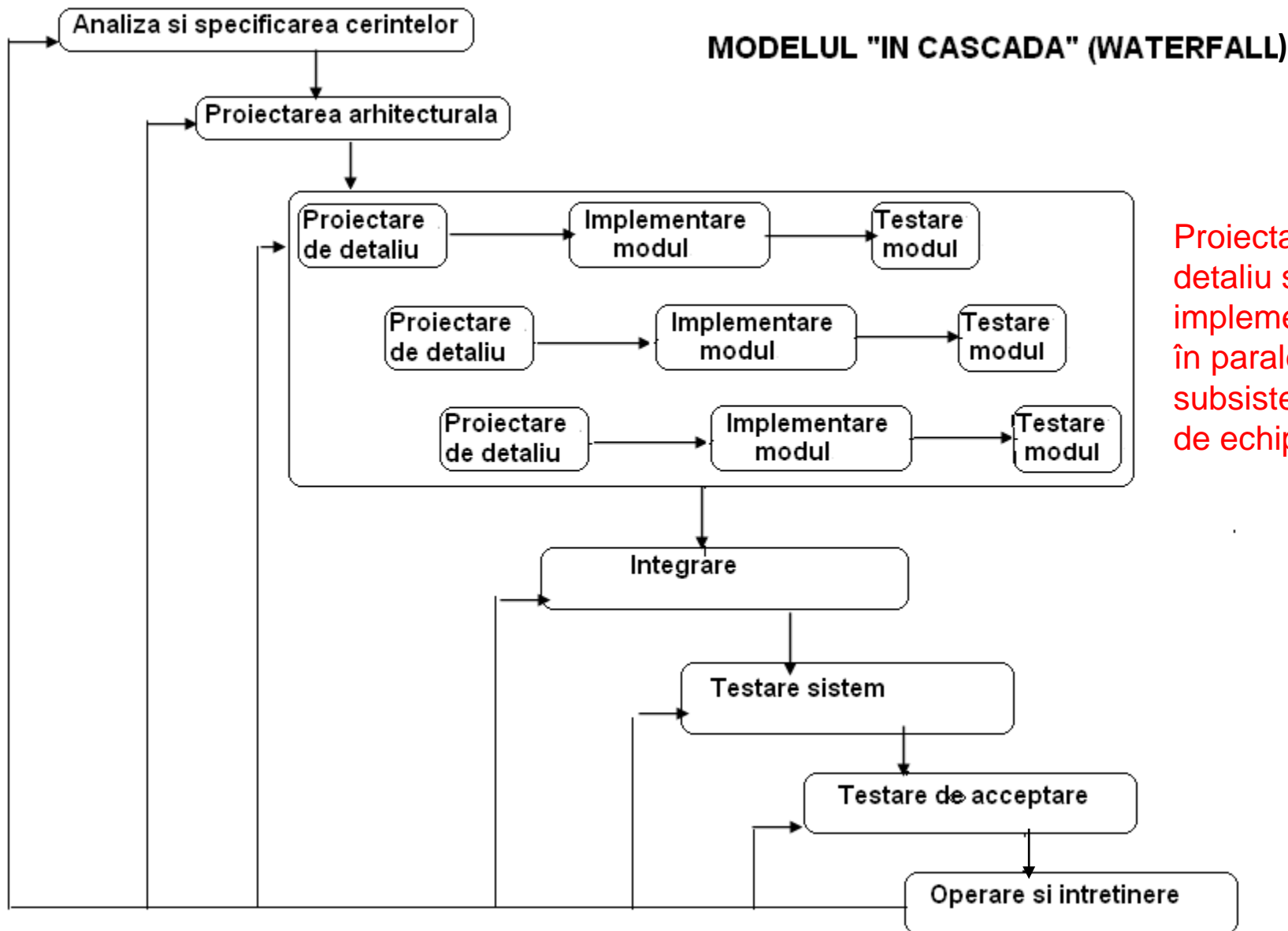
- Fiecare faza:

- corespunde unei activitati
- trebuie sa se termine la o anumita data prin producerea unui document sau produs program.
- include revizia rezultatelor fazei;
- nu se trece la faza urmatoare decat atunci cand rezultatele fazei curente sunt considerate satisfacatoare.

- Sagetile ascendente au fost adaugate ulterior, pentru a reflecta caracterul iterativ al procesului.
- NU se incurajeaza revenirea la fazele precedente

- Primul model de dezvoltare: Royce, 1970.





Proiectarea de detaliu si implementarea în paralel a subsistemelor, de echipe diferite

Modelul “In cascada” (Waterfall) (3)

Avantaje:

- Produsul software este bine documentat.
- Permite un bun management al proiectului:
 - planificarea resurselor umane pe etape
 - estimari de cost destul de exacte

Dezavantaje:

- Un produs executabil, care sa demonstreze functionarea sistemului este disponibil tarziu → risc crescut de neacceptare a produsului
- Multe defecte sunt descoperite tarziu; in special cele de proiectare → cost crescut (sunt reluate activitatile de proiectare, implementare, testare).
- **Inflexibil la modificarea cerintelor pe parcursul dezvoltarii**
- **Toate riscurile sunt incluse intr-un singur ciclu de dezvoltare**

Modelul “In cascada” (Waterfall) (4)

Nu se recomanda pentru:

- Proiectele în care cerintele sunt vagi, neclare
- Proiectele în care cerintele se pot schimba pe parcursul dezvoltării
- Proiectele de lungă durată

Adecvat pentru:

- Proiectele în care cerintele sunt bine înțelese de la început și nu se modifică pe parcursul procesului de dezvoltare.
- Experiența ultimelor decenii a demonstrat că modelul este valoros.
- Este utilizat și în prezent de multe companii mari pentru anumite tipuri de produse software.