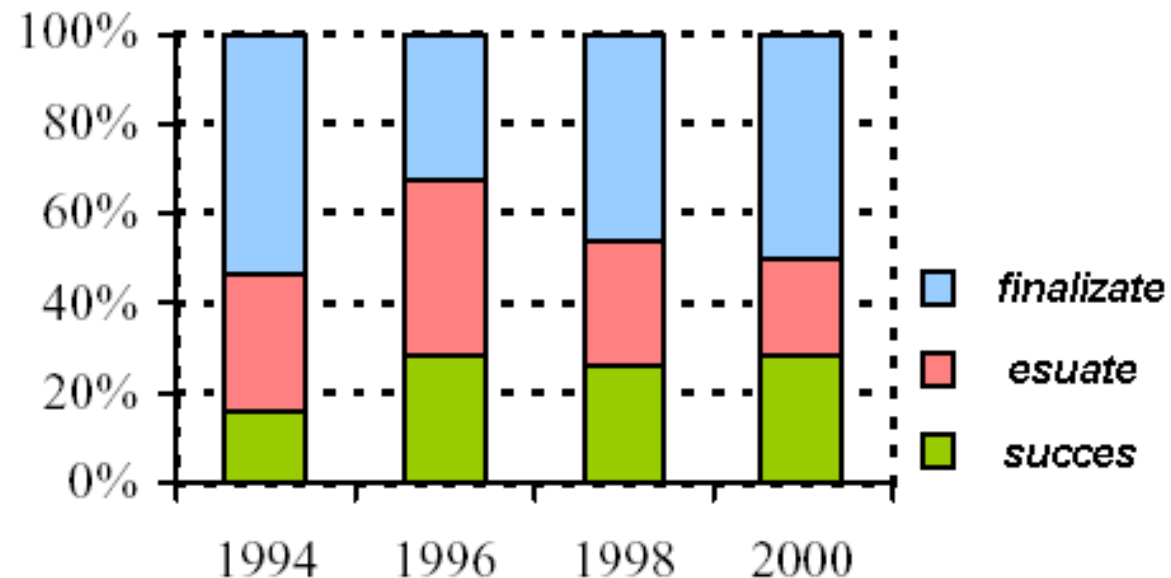


CURS 1

Proiecte Informatice

# Succesul proiectelor informatice

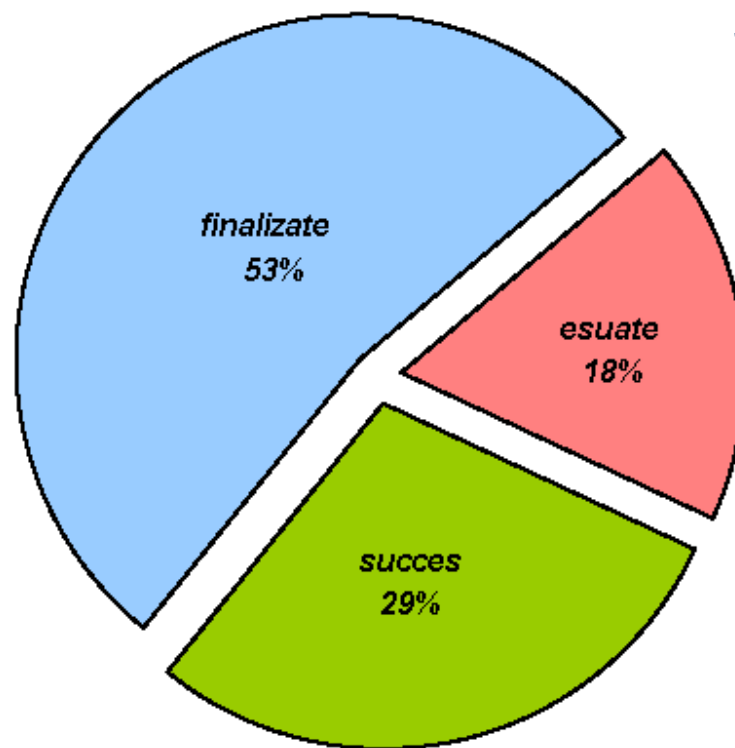
- Studiu *The Standish Group International*
  - 30.000 proiecte informatice americane analizate intre 1994 si 2000



# Succesul proiectelor informatice (cont)

## ■ CHAOS Research

- 50.000 proiecte in 2004 (58% SUA, 27% Europa, 15% rest)



# Succesul proiectelor

---

## ■ *Evaluare succes:*

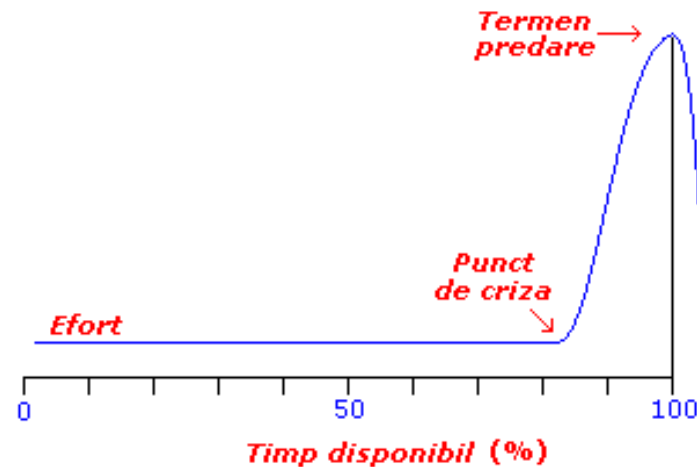
- proiectul a îndeplinit obiectivele referitoare la *termene, costuri și calitate?*
- proiectul satisface *cerințele & nevoile* clientului?
- rezultatele proiectului pot determina *nasterea unor noi proiecte?*
- după încheierea proiectului, este capabilă organizația să-și *continue activitatea?*

## ■ *Garantare succes:*

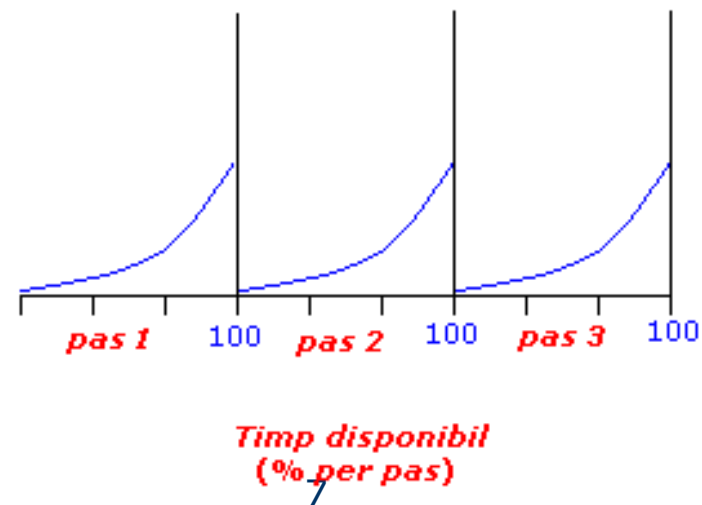
- obiective clare,
- personal capabil,
- susținere managerială,
- resurse eficiente,
- comunicare,
- monitorizare & control

# “Small releases”

- Creșterea exponențială a efortului spre finalizare



- Repartiția efortului (ore/zi) și energiei în cazul abordării iterative și incrementale



# Nemăsurabile, greu de estimat

---

- diferențe importante în estimare pentru persoane diferite, indiferent de experiență
  - nu există un nomenclator
  - consecință: dificil de gestionat schimbările
  - *estimare +20%*
- dificil de monitorizat / controlat progresul
  - în special la analiză și proiectare

# Nemăsurabile, greu de estimat (cont)

			<b>Nr declarații</b>	<b>Nr experți</b>
<code>#define</code>	<code>LOWER</code>	<code>0</code>	o declarație	1
<code>#define</code>	<code>UPPER</code>	<code>300</code>	două declarații	6
<code>#define</code>	<code>STEP</code>	<code>20</code>	trei declarații	5
<code>main()</code>			patru declarații	1
<code>{</code>			cinci declarații	4
<code>  int fahr;</code>			șase declarații	11
<code>  for (fahr=LOWER; fahr&lt;=UPPER;</code>			șapte declarații	4
<code>fahr=fahr+STEP)</code>			opt declarații	1
<code>  printf("%4d %6.1 f\n", fahr,</code>			nouă declarații	11
<code>  (5.0/9.0*(fahr-32))</code>				
<code>}</code>				

---

M. Norris, P. Rigby, M. Payne - *The Healthy Software Project: a guide to successful development*, John Wiley & Sons, Chichester, 1993

# Softul: *invizibil, intangibil*

---

- realizat sub forma unor texte de diferite tipuri:
  - documente de proiectare,
  - cod sursă,
  - manuale de utilizare și operare
  
- nu există ceva concret de arătat clientului în faza de analiză a cerințelor
  - variantă: prototip
  
- cerințele inițiale par ușor de modificat



# Complexitate ridicată

---

- Fazele ciclului de viață a produselor soft:
  - *Specificare funcțională* – document în lb. nat.
  - *Analiză* – model de analiza (grafic + adnotări)
  - *Proiectare* – model de proiectare
  - *Dezvoltare* – cod sursă în limbaj(e) de program.
  - *Compilare/Link-editare* – model executabil
- rezultatele unei faze sunt transferate fazei următoare
  - ⇒ vulnerabilitate mare la erori umane.

## Verificare corectitudine, testare, calitate

- **imposibil** de testat toate ramurile
- refacerea scenariilor de test in cazul modificarilor
- nu există mecanisme de masurare sigură / precisă a calitatii unei aplicatii
- “În cazul proiectelor dezvoltare de softuri așa numitele *proceduri de control al calității* au uneori de-a face mai degrabă cu limitarea defectiunilor decât cu garantarea calității produsului final.” *Norris et. al (1993)*

# Dinamism

---

- atracția noutății tehnologice
  - *stabilitate vs plafonare*
- fluctuații de personal
  - re-evaluare *task-uri*
  - tendința de a respinge codul altora
- cereri de modificare frecvențe în diverse faze ale ciclului de viață

# Decizii pripite în situații extreme

---

- *sporirea resurselor într-un proiect întârziat nu elimină decalajul ci sporește întârzierea*
  - capacitatea de efort a membrilor echipei scade cu o cantitate egală cu efortul de comunicare cu noul membru

... etc

---

- specificațiile sunt prea lungi, “stufosă” și detaliate astfel încât utilizatorii nu identifică ideile principale;
- specificațiile reprezintă mai degrabă dorințe decât o listă de funcționalități cu priorități;
- se descoperă soluții care rezolvă o problemă dar introduc probleme noi;
- funcționalități sub-optimizate neidentificate