

# MONARC

## SIMULAREA SISTEMELOR DISTRIBUITE

### **TEHNICI DE IMBUNATATIREA PERFORMANTELOR. CAZURI DE TEST.**

Autor: Andrei Dragos  
Indrumatori: Prof. Dr. Ing. Valentin Cristea  
Prof. Iosif Legrand  
ing. Corina Stratan  
ing. Ciprian Dobre

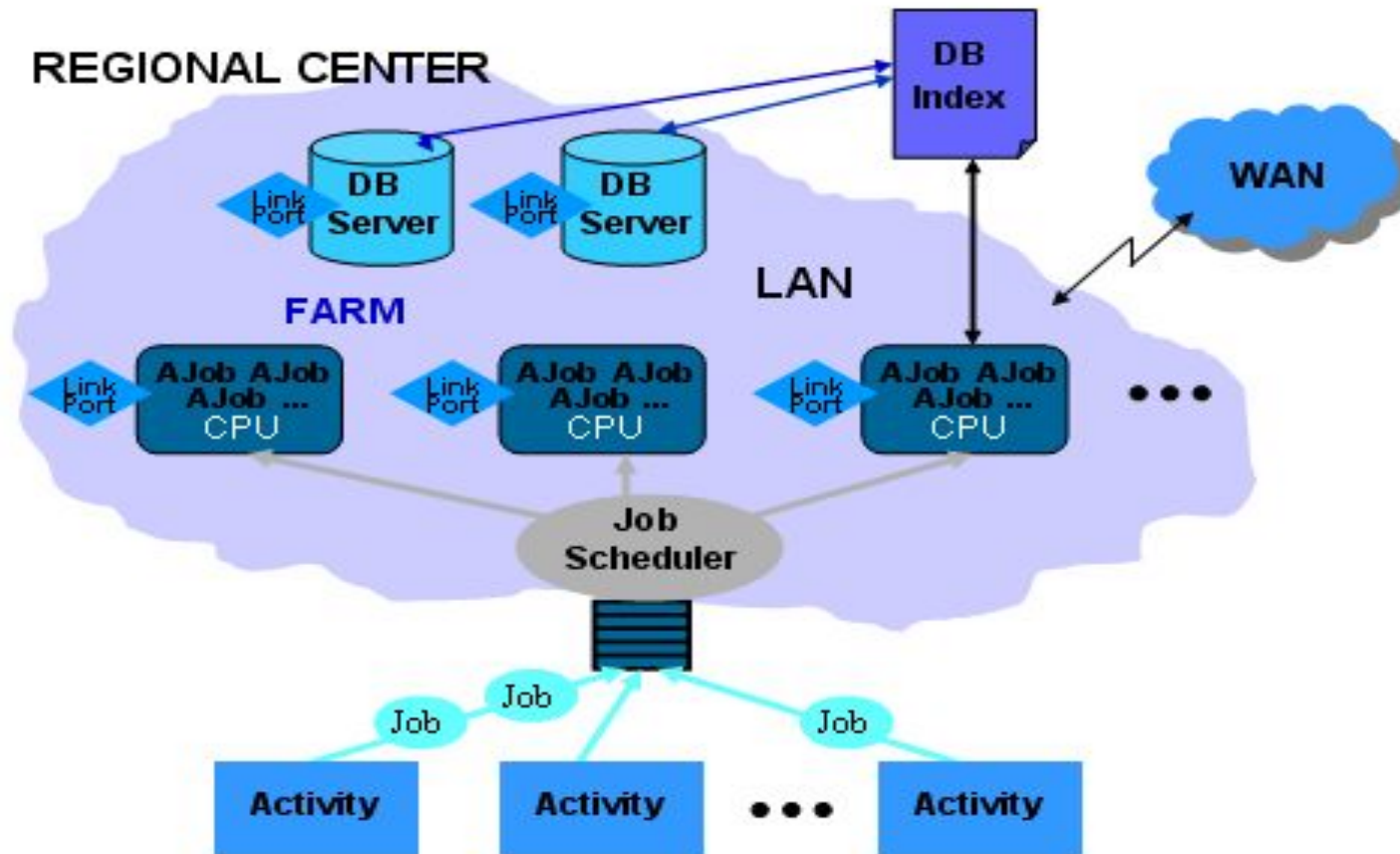
# Continutul prezentarii

- Introducere
- Prezentare generala a simulatorului MONARC
- Simularea clusterului de Proof
- Optimizarea pachetului de retea si a procesarii de date ale MONARC-ului
- CernTier – simulare pentru activitati de replicare si productie de date
- Concluzii

# Introducere

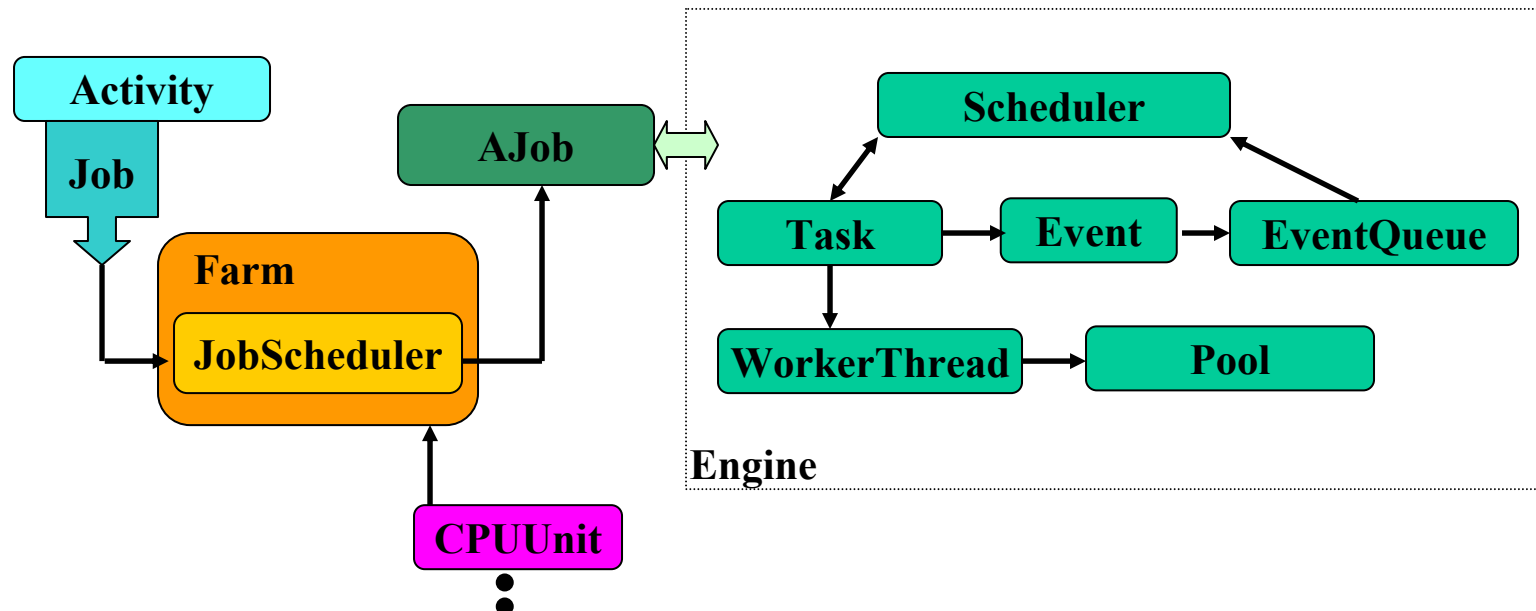
- Monarc
  - simulator ce serveste la evaluarea performantelor generale ale SD
  - modelarea si simularea SD in scopul stabilirii performantelor aplicatiilor care ruleaza pe acestea
  - scopul de a alege:
    - o structura si arhitectura optima
    - algoritimi de scheduling si replicare de date
- Tipuri de simulari
  - Simularea cu timp continuu
  - Simularea cu timp discret
    - simularea discreta orientata pe timp
    - simularea orientata pe evenimente discrete (DES)
- Monarc: DES
  - activitatile relevante din sistem
  - ceas intern

# Structura unui sistem simulat



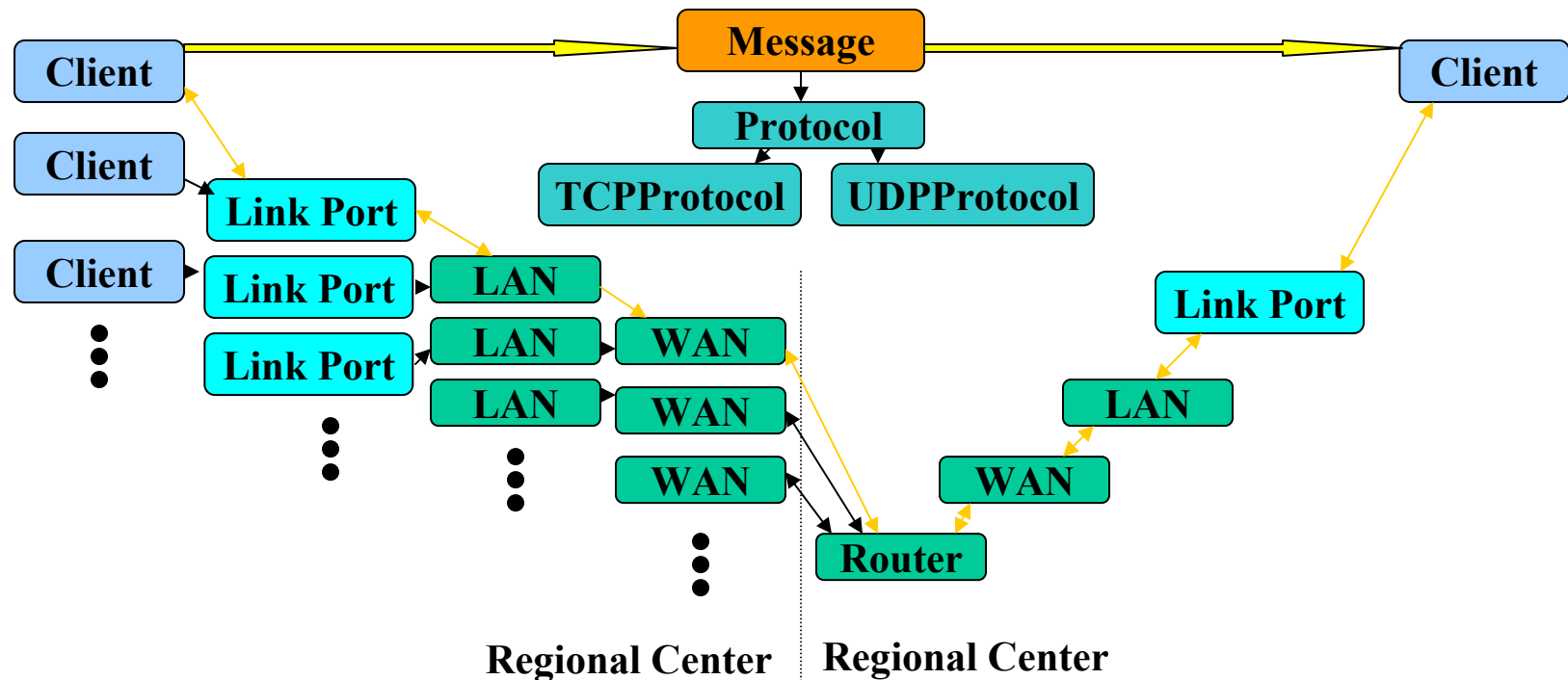
# Motorul simulatorului

- Gestioneaza obiectele active si evenimentele
- Furnizeaza mecanismul de simulare multithreading
- Entitatile din sistem cu comportament dependent de timp (active) sunt simulate prin fire de executie
- Reutilizabilitatea firelor de executie (thread pool)



# Simularea retelelor

- **realistic in timp - parametri reali din teoria retelelor**
- **mecanism bazat pe intreruperi**
- **Permite simularea unor transferuri concurente avand parametri/protocoale diferite.**



# Simularea Clusterului de Proof

- facilitate pentru date distribuite sub o structura Root, dezvoltata la CERN
- analiza a multimii vaste de fisiere Root in paralel pe clustere de calculatoare aflate distribuit
- Root - sistem de procesare si stocare masiva de date
- principalele scopuri pentru sistemele Proof
  - transparenta
  - scalabilitatea
  - adaptibilitatea
- procesarea datelor cu Proof contine fazele:

# Arhitectura unui Proof Cluster

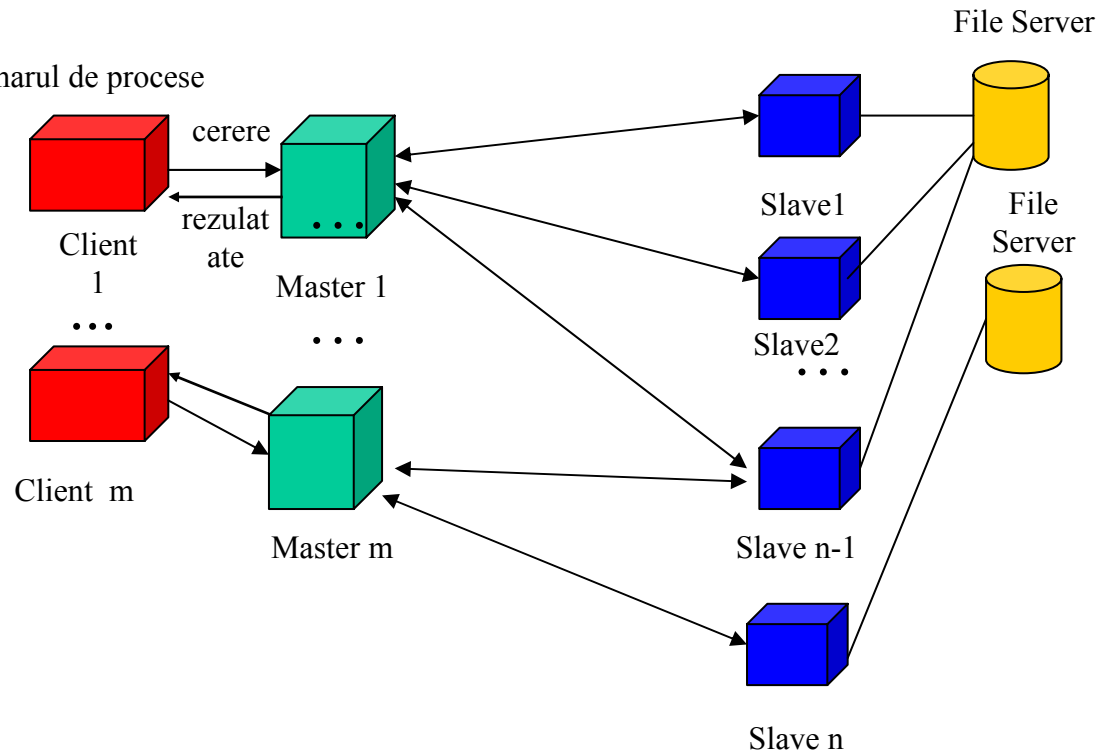
Configuratia simulata

- m clienti, masteri, n slaves

-datele procesate de un slave sunt disponibile pe discul local cu o anume probabilitate; daca nu sunt obtinute de pe un server de fisiere

-numar fixat de slaves

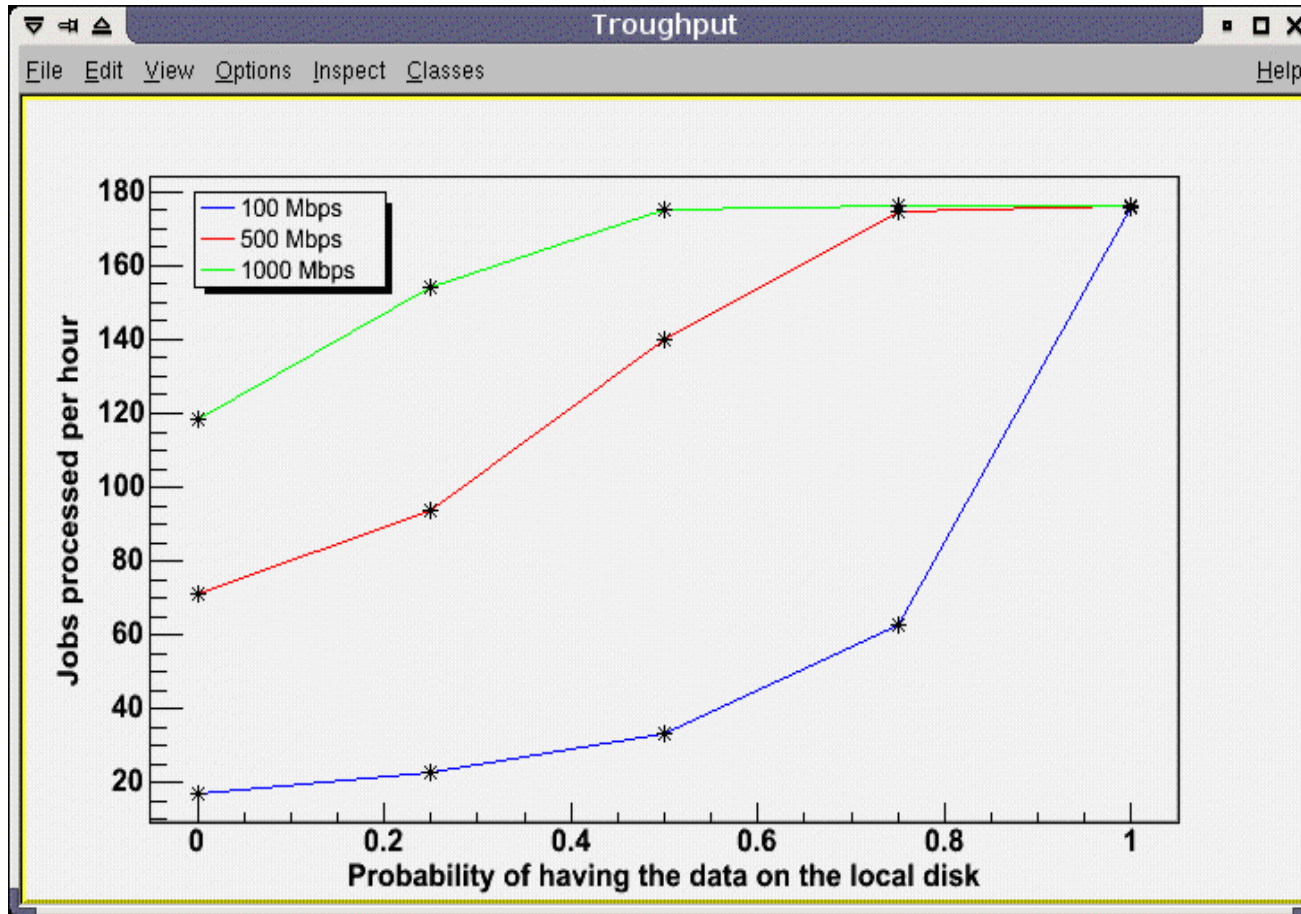
-scopul: sa evaluam performantele cand numarul de procese slave variaza



- 2 variante de scheduling



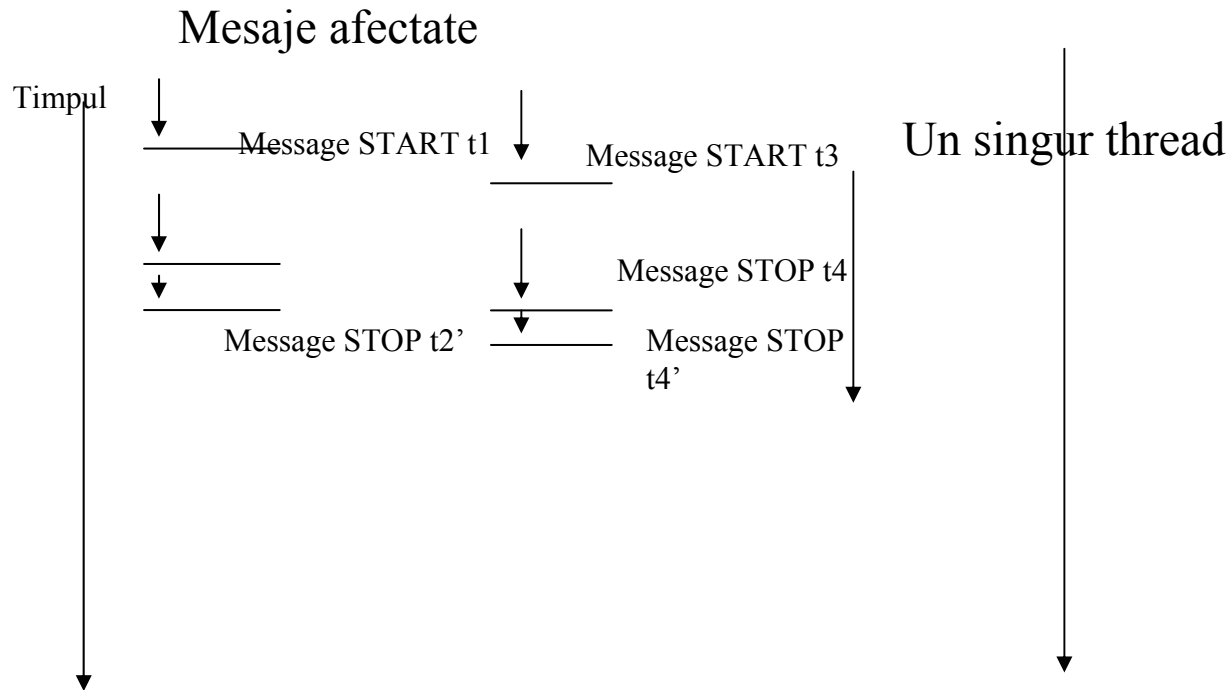
# Influenta bandwidth-ului LAN-ului



- 20 master
- 500 slave
- 2 servere de date
- 50 procese slave pe master
- retea de 100Mbps, 500Mbps, 1000Mbps

# Optimizarea partii de retea si a joburilor de procesare ale Monarc

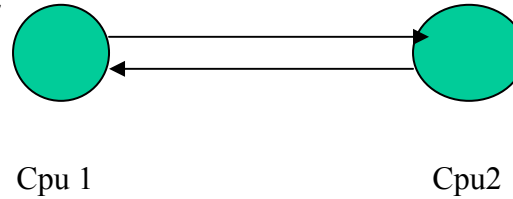
- un thread care se ocupa de toate mesajele care au acelasi Cpu sursa/destinatie; joburi
- functionarea precedenta: pentru fiecare mesaj exista un thread de trimitere / de primire; pentru fiecare job cate un thread
- probleme de context switching, sincronizare a threadurilor
- cu cat creste numarul de evenimente de intrerupere cu atat este mai indicata varianta optimizata (schimbari context)



# Performantele variantelor optimizate fata de cele neoptimizate

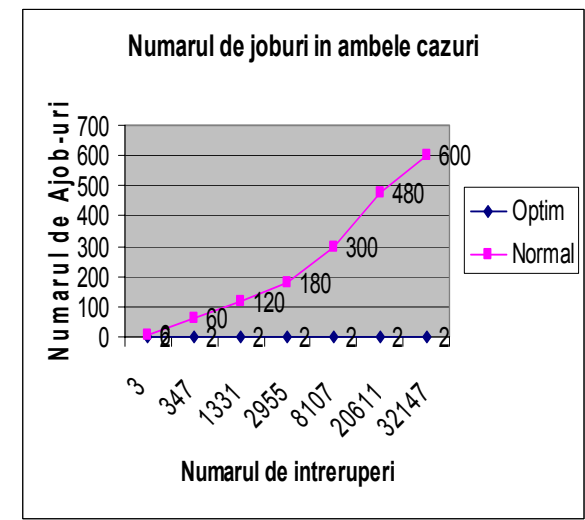
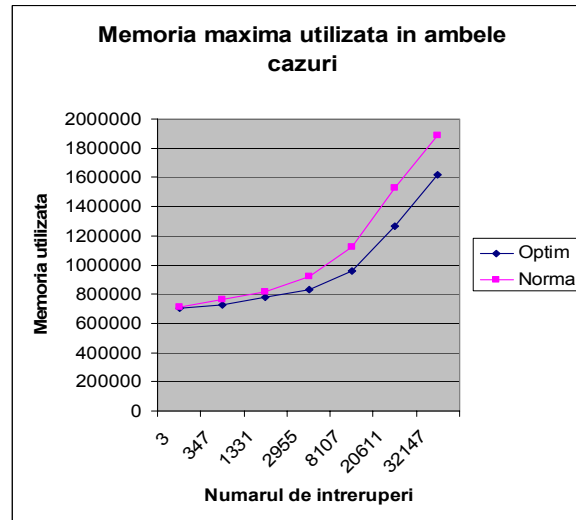
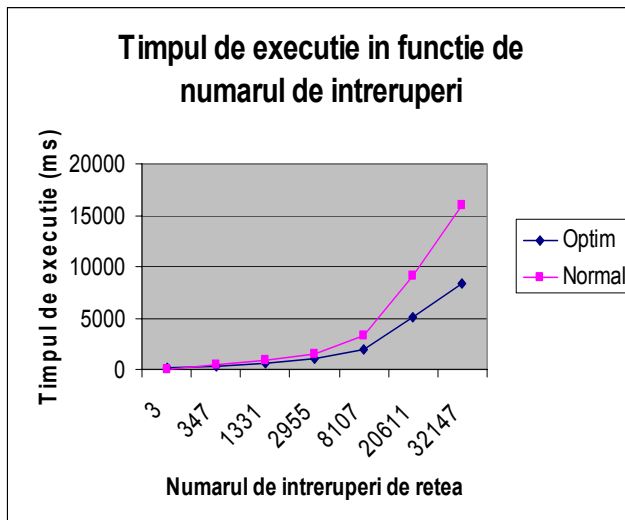
- Simulari cu numar mare de intreruperi varianta optimizata este superioara:

- memorei maxime utilizate
- timpului de executie
- numarului de threaduri



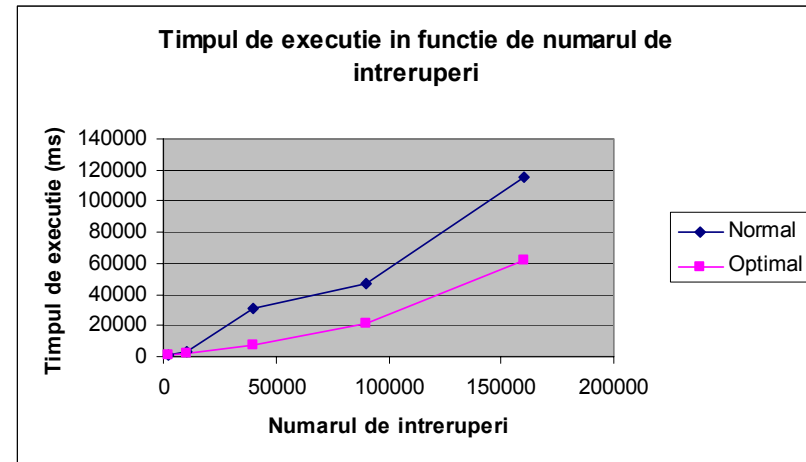
- Bandwidth 100Mbps
- Dimensiune pachet: 25 B
- timp intre mesaje: 2.0 s-queuing
- variem numarul de iteratii

Partea de retea:

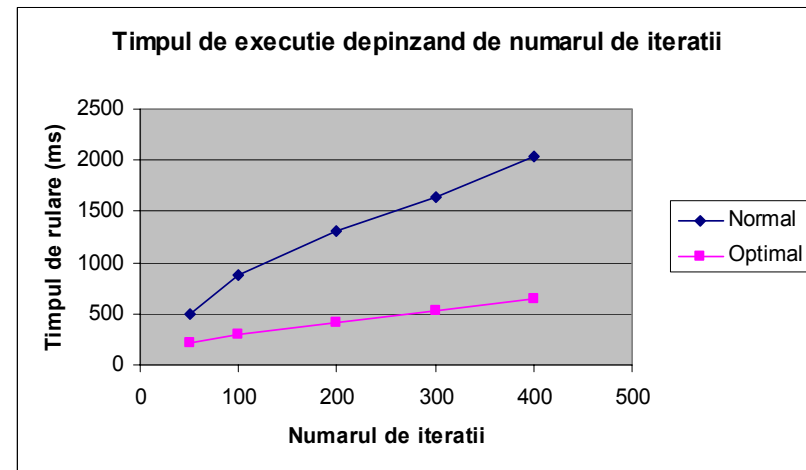


# Simulare pentru partea de retea dar si pentru cea de procesare de joburi

- un cpu care are un numar de job-uri de rulat (procesare)

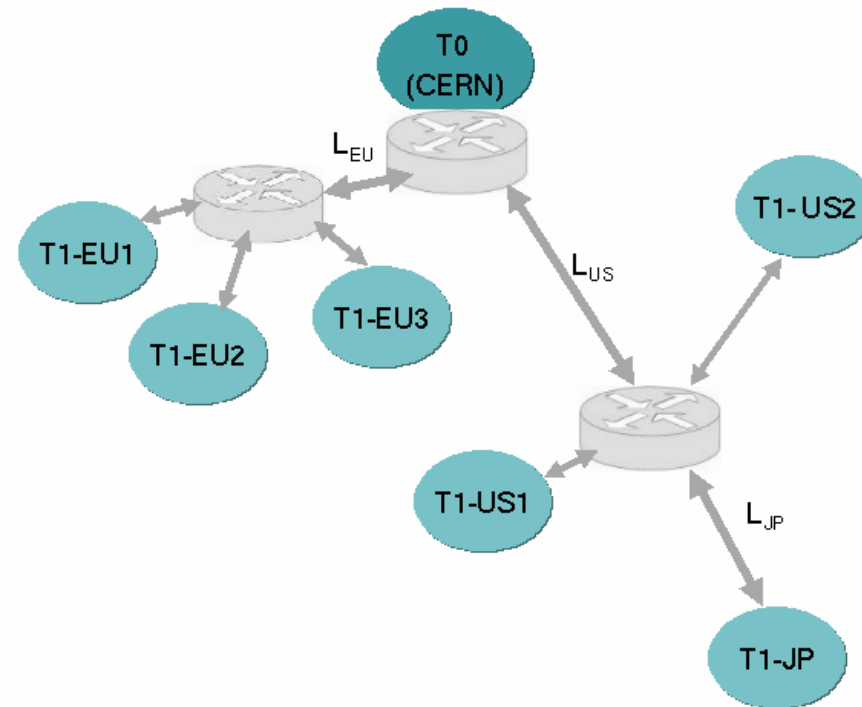


- acelasi cpu care isi trimite singur mesaje (retea)



# CernTier – Simulare pentru activitati de replicare si producere de date T0/T1

- descrie procese reale de la CERN si cerintele de stocare, acces si calcule ale analizei de date LHC
- arhitectura reala



**Aplicatiile** legate de fizica **simulate** sunt:

- Replicarea de date RAW
- Producerea si distributia DST
- Re-producerea si distributia DST
- Analiza detectiei

# Concluzii

- Monarc-ul a depasit de mult stadiul de aplicatie pentru reprezentarea fenomenelor fizice
- Cadru mult mai larg – sisteme distribuite de mari dimensiuni
- Proiectul este in faza de optimizare
- Se urmareste folosirea claselor deja existente in scopuri cat mai apropiate de realitate si cu o importanta practica cat mai pronuntata (ex: simularile CernTier, Proof)
- Pentru mai multe informatii: <http://monarc.cacr.caltech.edu/>