

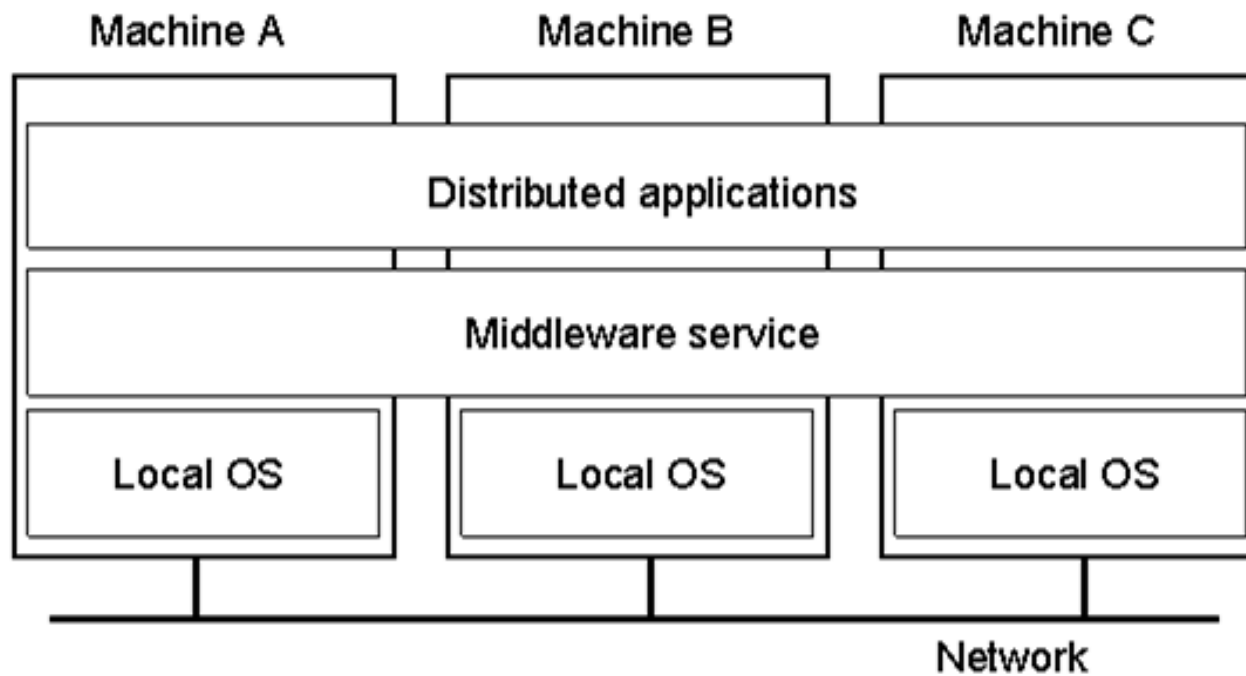


BCC 362 – Sistemas Distribuídos

Joubert de Castro Lima – joubertlima@gmail.com
Professor Adjunto – DECOM

UFOP

SISTEMAS DISTRIBUÍDOS



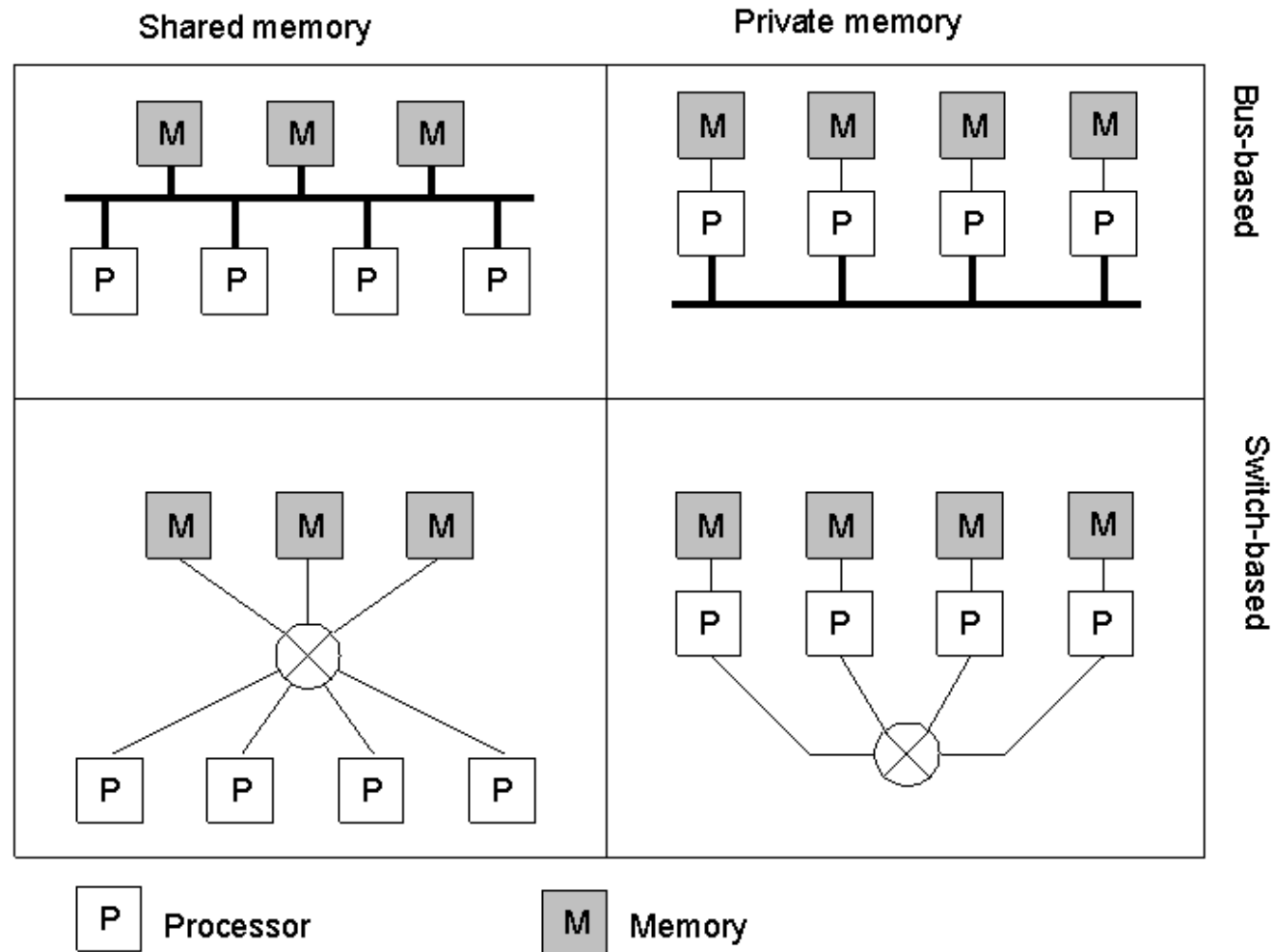
Facilitar aos usuários e as aplicações o acesso a recursos remotos

Vamos falar mais de hardware.....

Vamos falar mais de sistemas operacionais



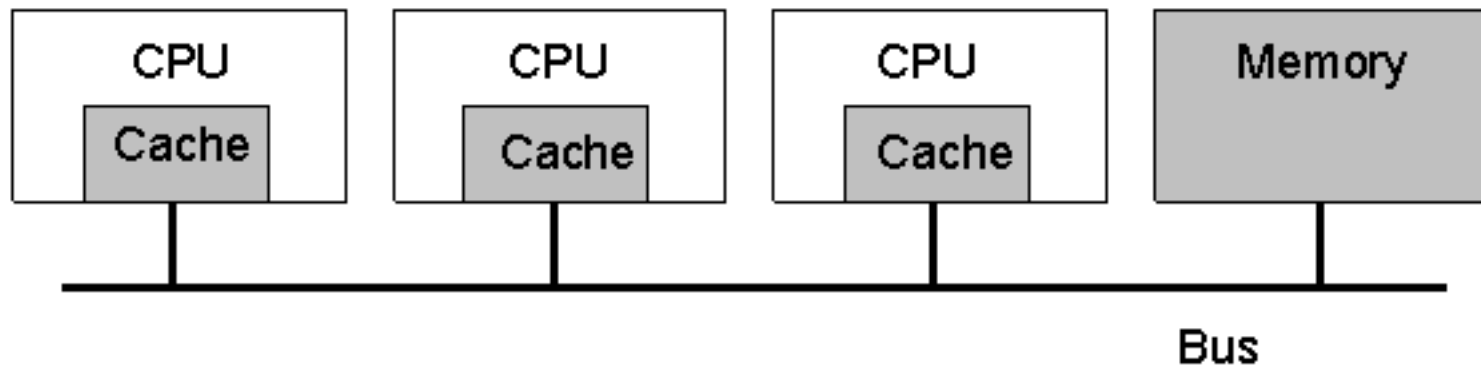
.Hardware Concepts



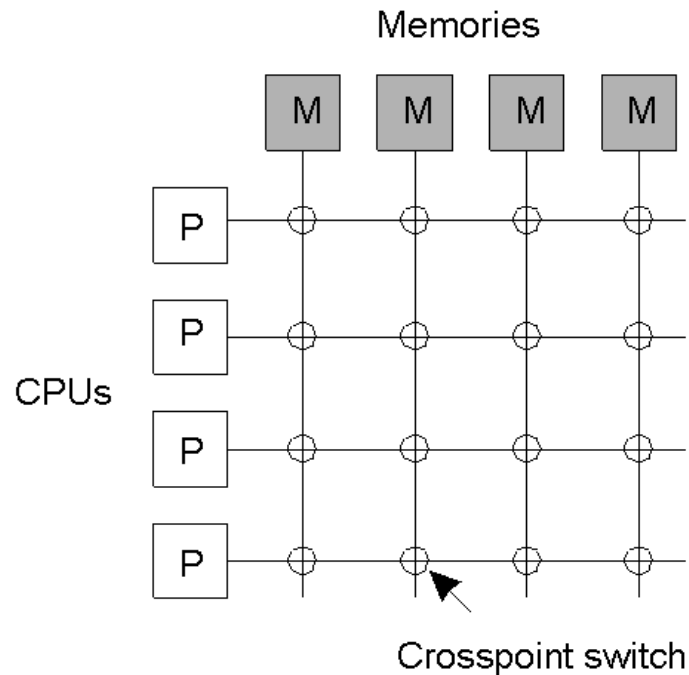
Different basic organizations and memories in distributed computer systems

•Multiprocessors (1)

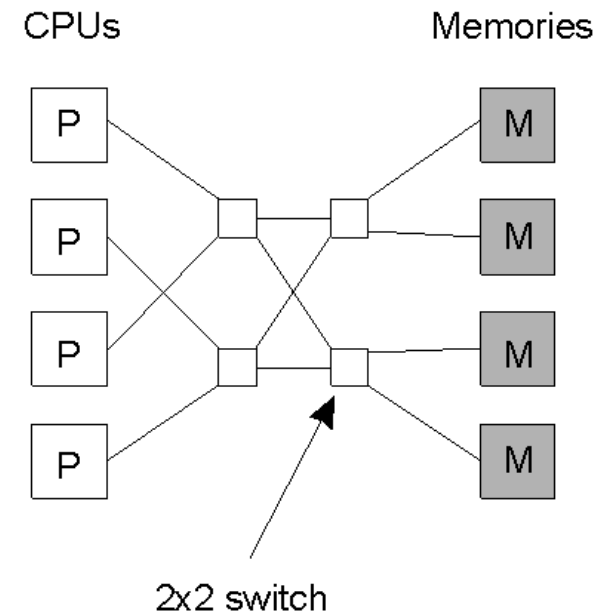
- A bus-based multiprocessor.



•Multiprocessors (2)



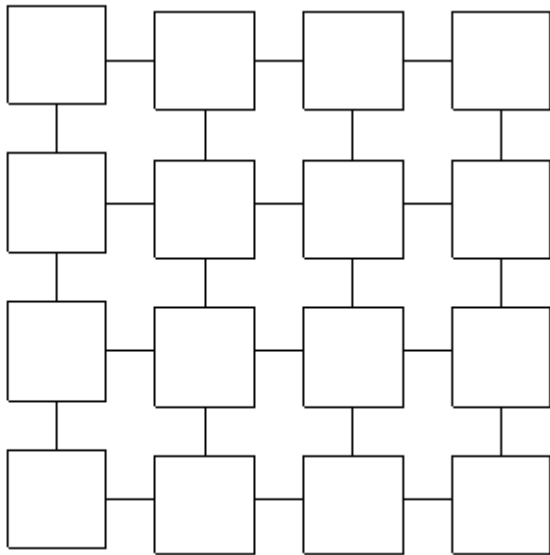
(a)



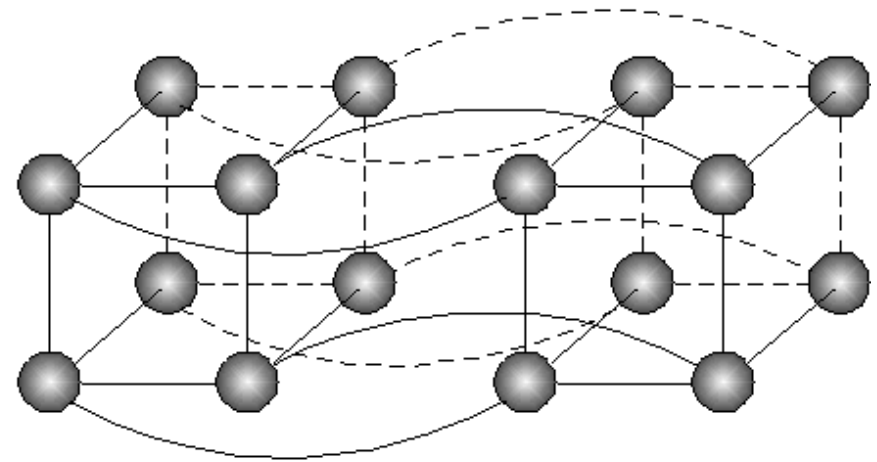
(b)

- a) A crossbar switch
- b) An omega switching network

•Homogeneous Multicomputer Systems



(a)



(b)

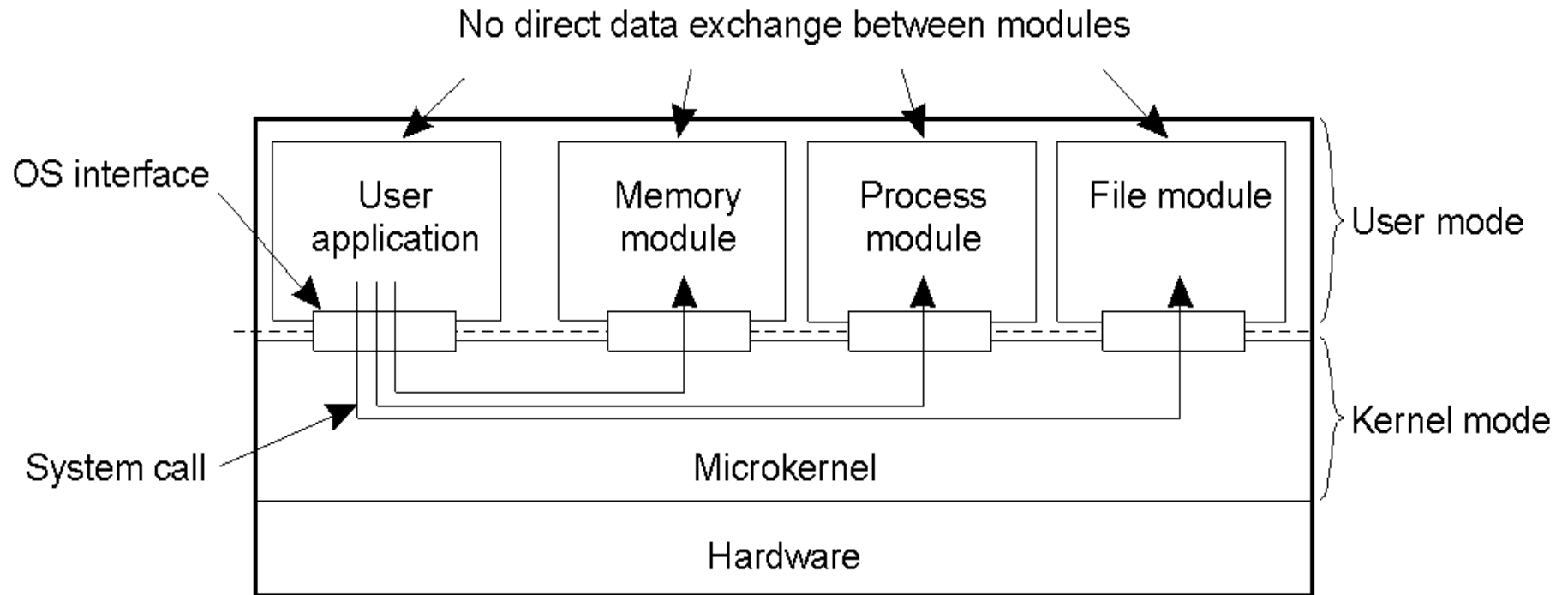
- a) Grid
- b) Hypercube

.Software Concepts

System	Description	Main Goal
DOS	Tightly-coupled operating system for multi-processors and homogeneous multicomputers	Hide and manage hardware resources
NOS	Loosely-coupled operating system for heterogeneous multicomputers (LAN and WAN)	Offer local services to remote clients
Middleware	Additional layer atop of NOS implementing general-purpose services	Provide distribution transparency

- An overview of
- DOS (Distributed Operating Systems)
- NOS (Network Operating Systems)
- Middleware

.Uniprocessor Operating Systems

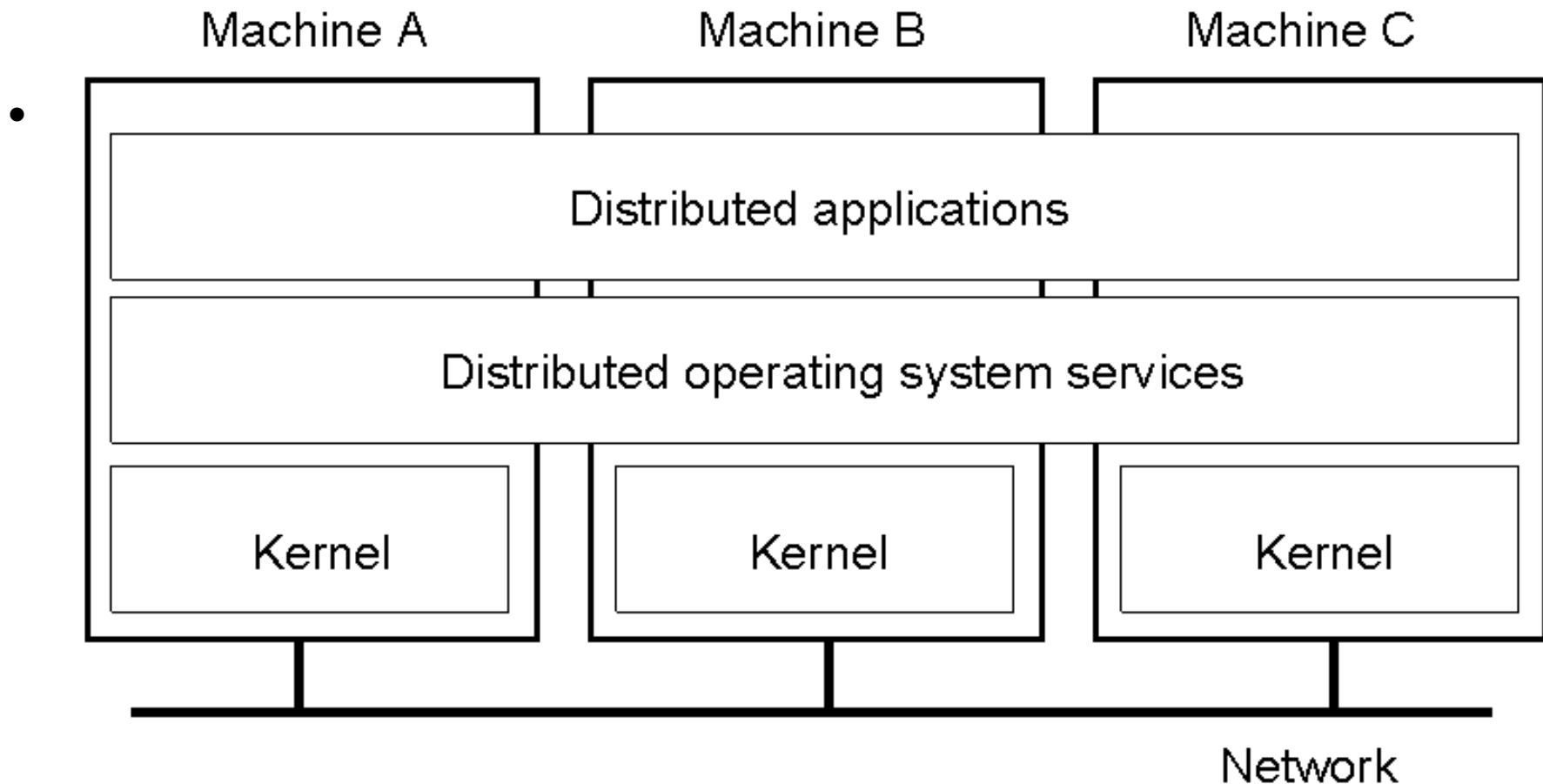


- Separating applications from operating system code through a microkernel.

Multiprocessor Operating Systems

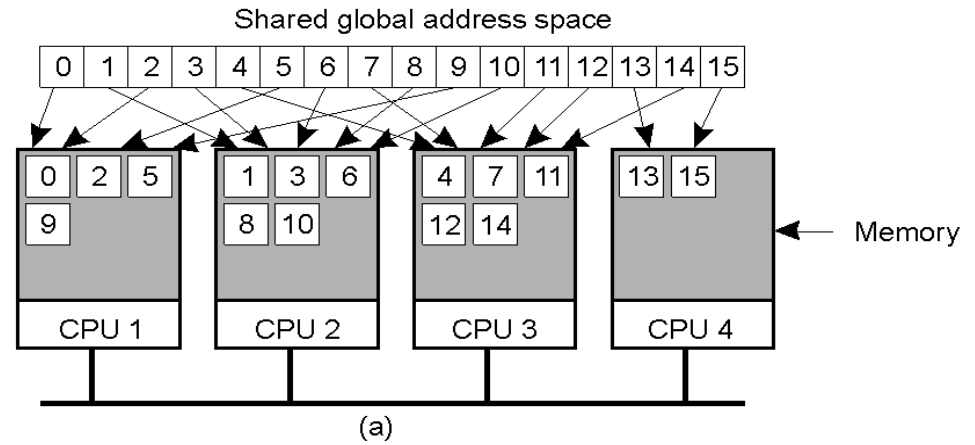
- Multiprocessor operating systems aim to support high performance through multiple CPU.
- The main goal is to make the number of CPUs transparent to the application.
- The idea is that all communication is done by manipulating data at shared memory location, and that we only have to protect that against simultaneous access.

• Multicomputer Operating Systems

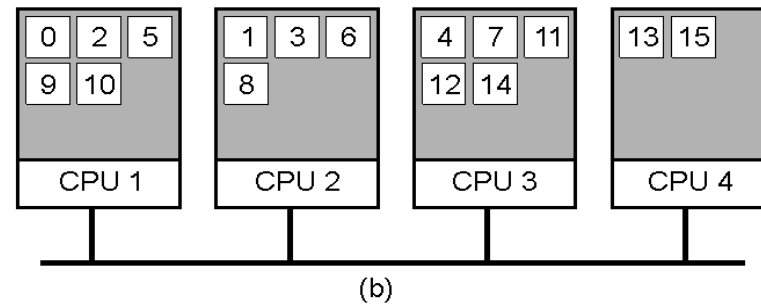


.Distributed Shared Memory Systems

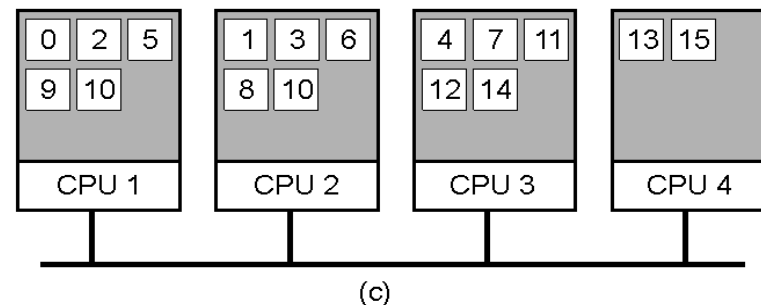
- a) Pages of address space distributed among four machines



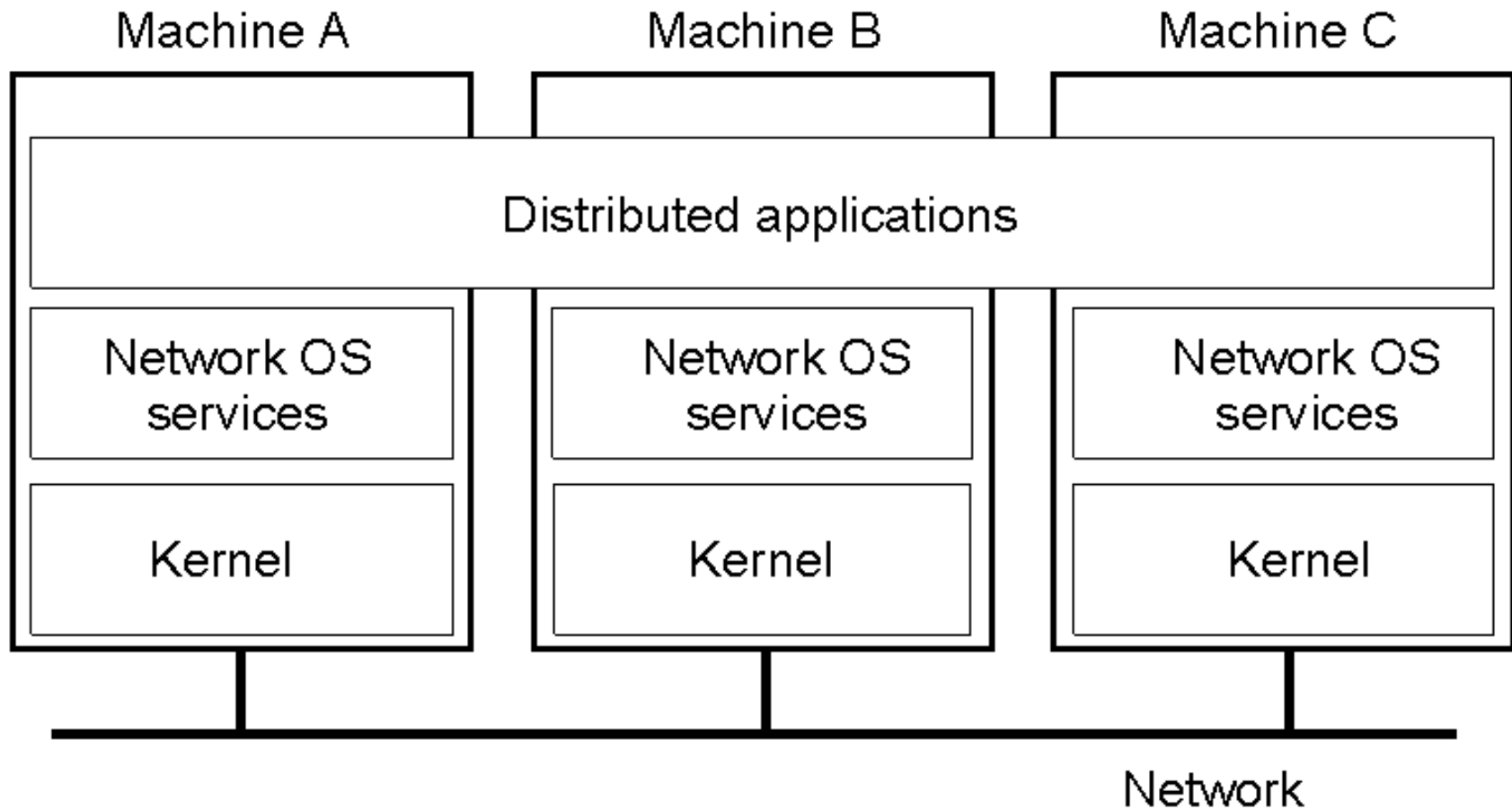
- b) Situation after CPU 1 references page 10



- c) Situation if page 10 is read only and replication is used

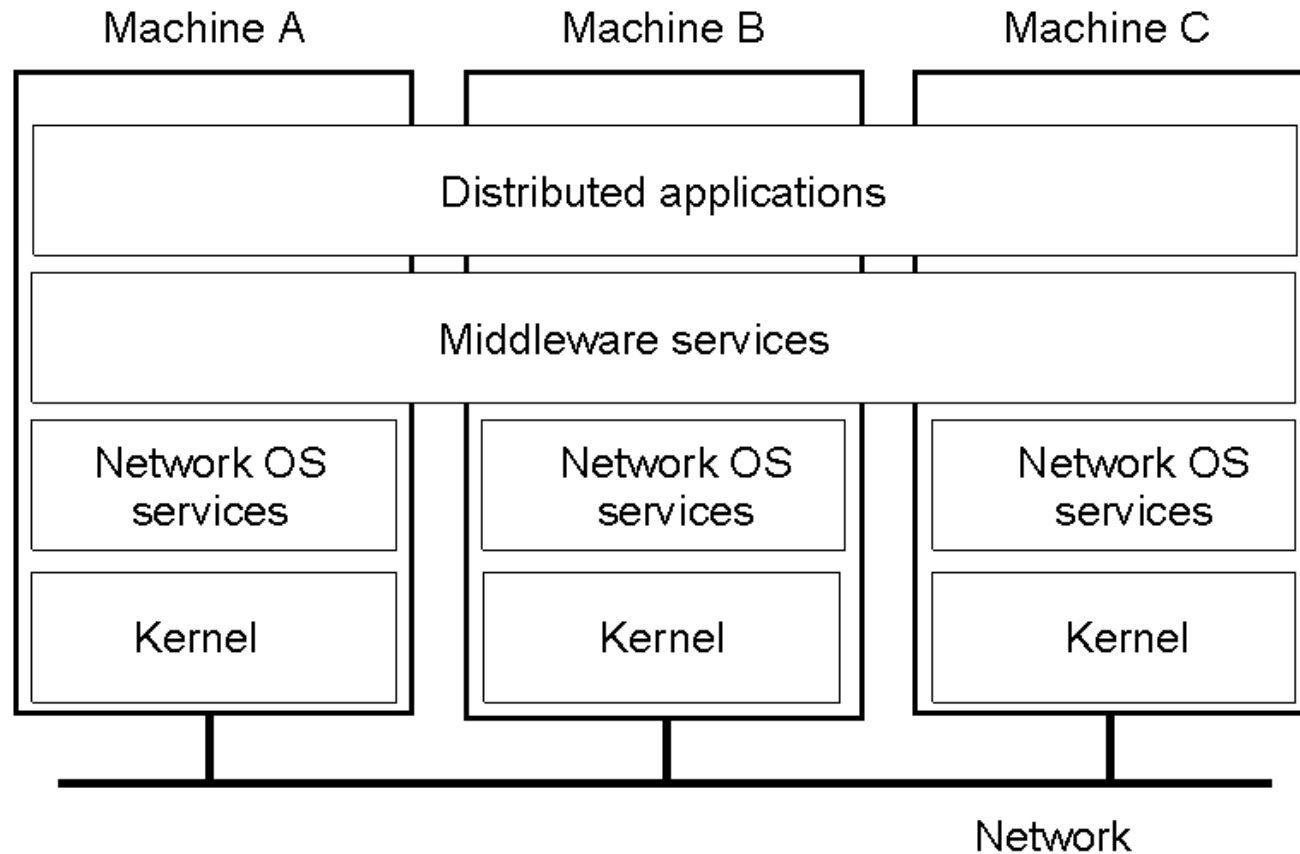


•Network Operating System



- General structure of a network operating system. It provides facilities to allow users to make use of services available in a specific machine.

Positioning Middleware



General structure of a distributed system as middleware offering a higher level of abstraction. A important goal is to hide heterogeneity of the underlying platforms applications.

•Comparison between Systems

Item	Distributed OS		Network OS	Middleware-based OS
	Multiproc.	Multicomp.		
Degree of transparency	Very High	High	Low	High
Same OS on all nodes	Yes	Yes	No	No
Number of copies of OS	1	N	N	N
Basis for communication	Shared memory	Messages	Files	Model specific
Resource management	Global, central	Global, distributed	Per node	Per node
Scalability	No	Moderately	Yes	Varies
Openness	Closed	Closed	Open	Open

- A comparison between multiprocessor operating systems, multicomputer operating systems, network operating systems, and middleware based distributed systems.

FRISANDO NOVAMENTE.....

Main goals of a distributed system:

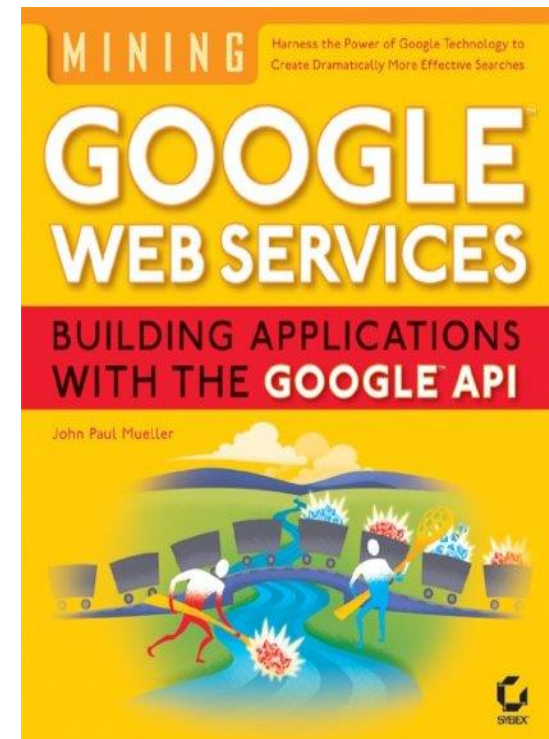
to connect users to resources;

to be open; to hide the distribution;

and to be scalable

**Seja implementando novos SOs ou construindo
Middlewares devemos satisfazer os objetivos impostos.....**

RECURSOS ???



Um sistema distribuído TRANSPARENTE

Transparência de acesso

Transparência de localização

Transparência de migração

Transparência de relocação

Transparência de replicação

Transparência de concorrência

Transparência à falha

UFA !!!!!

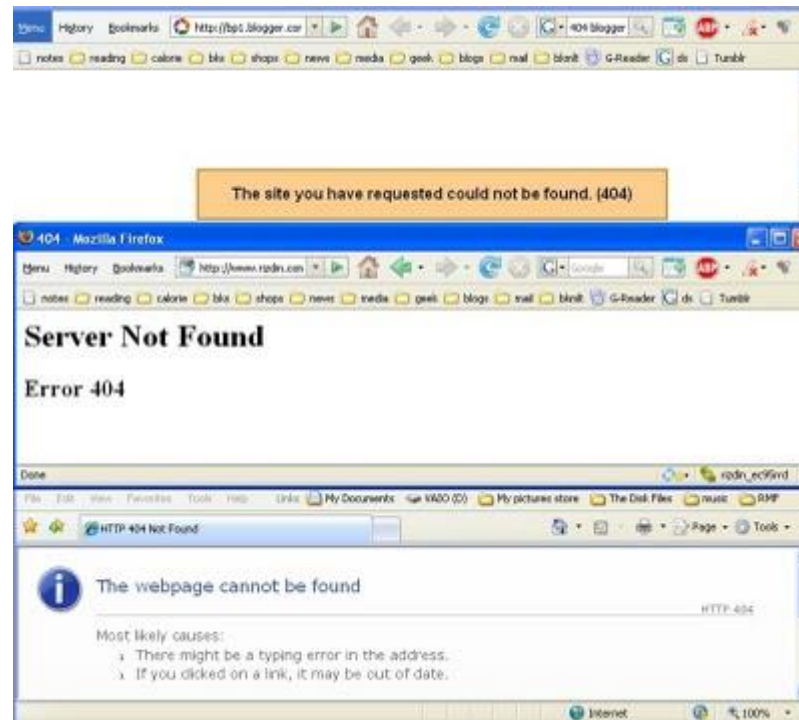


GRAU DE TRANSPARÊNCIA

Dependendo do problema, não é interessante ou nem possível ocultar que o sistema é transparente.

O grau de transparência está ligado ao desempenho do sistema.

Um exemplo:



Nem sempre compensa manter a transparência na distribuição

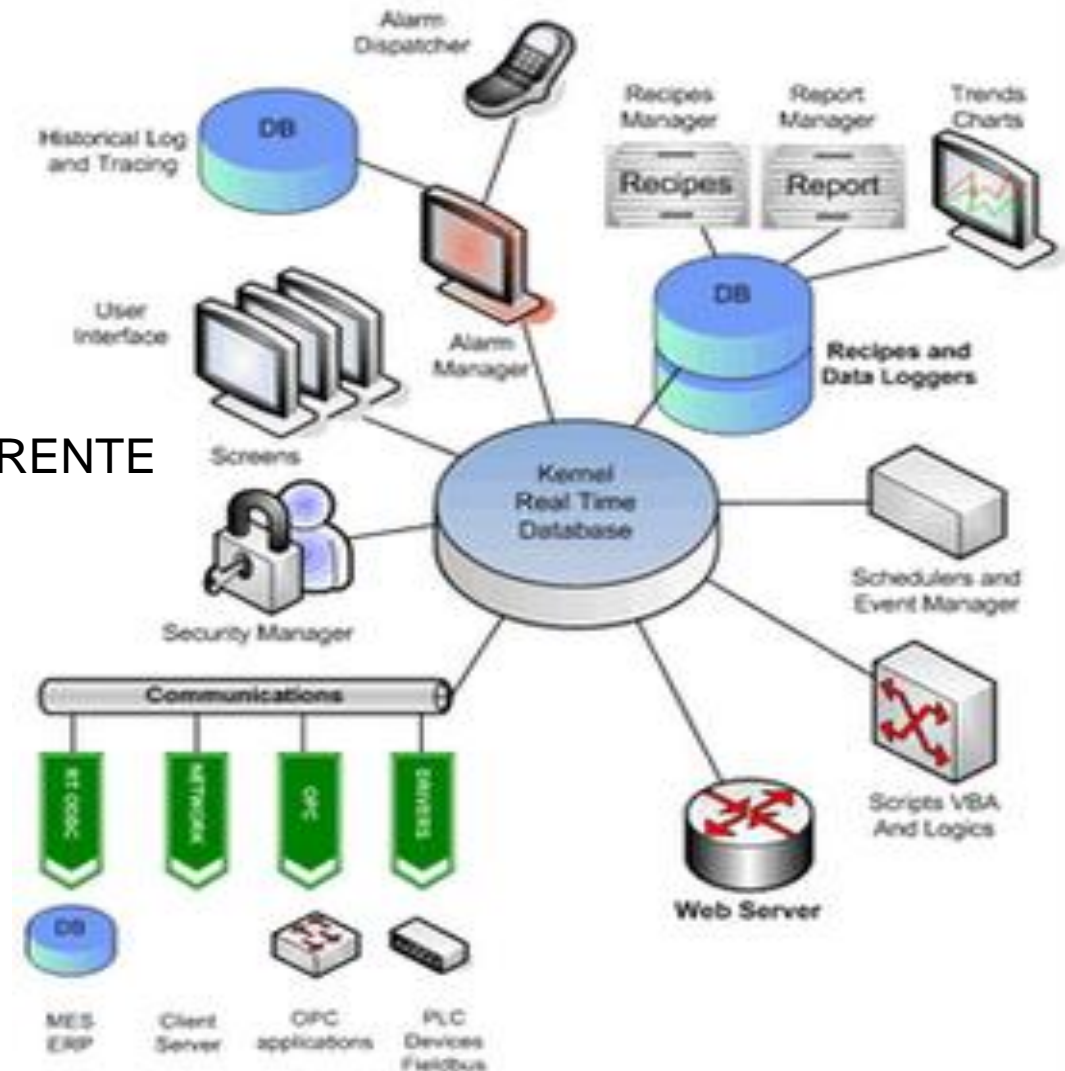
ABERTURA

O MUNDO É HETEROGÊNEO

O SETOR DE TI NÃO É DIFERENTE

Interoperabilidade

Portabilidade



Escalabilidade

Em relação ao seu tamanho, em relação a sua geografia e, por fim, em relação a sua administração

Ser escalável em uma ou mais dimensões acima pode implicar em perda de desempenho também!!!



Vamos aos detalhes....

Ser escalável em relação ao tamanho

É fácil adicionarmos mais usuários ou recursos ao sistema

Serviços centralizados
Dados centralizados
Algoritmos centralizados



Nenhuma máquina tem informações completas do sistema

As decisões tomadas usam informações locais

A falha de uma máquina não arruína o algoritmo

Não existe um relógio global

Escalabilidade geográfica

Mensagens síncronas e longas distâncias um problemão!!!!

Atrapalha a escalabilidade de tamanho

Longas distâncias => comunicação não confiável e quase sempre ponto a ponto

Localização de um serviço: **Imagine um broadcast num sistema distribuído mundialmente**

Técnicas de escalabilidade

Ocultar a latência de comunicação

Distribuição

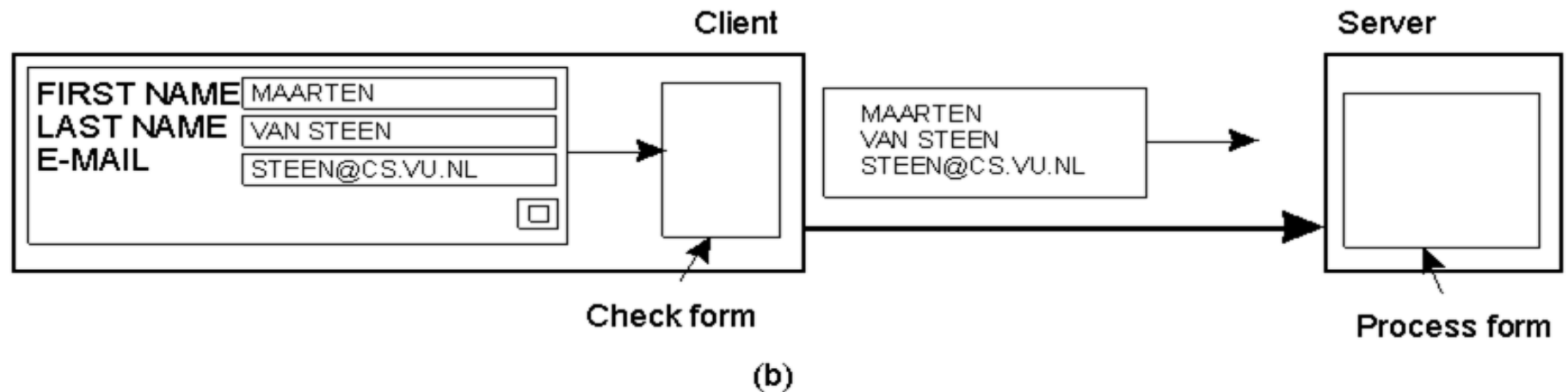
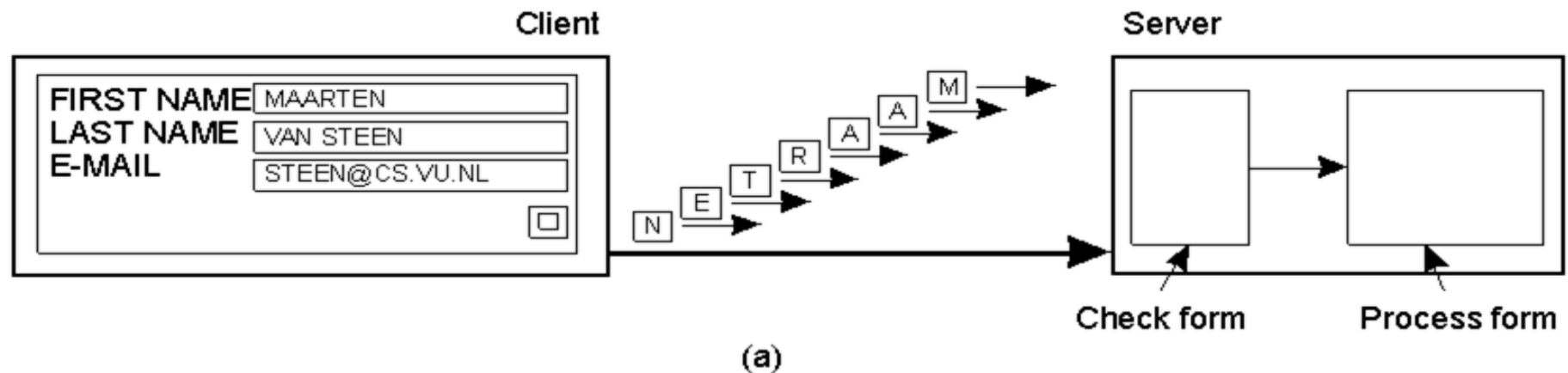
Replicação



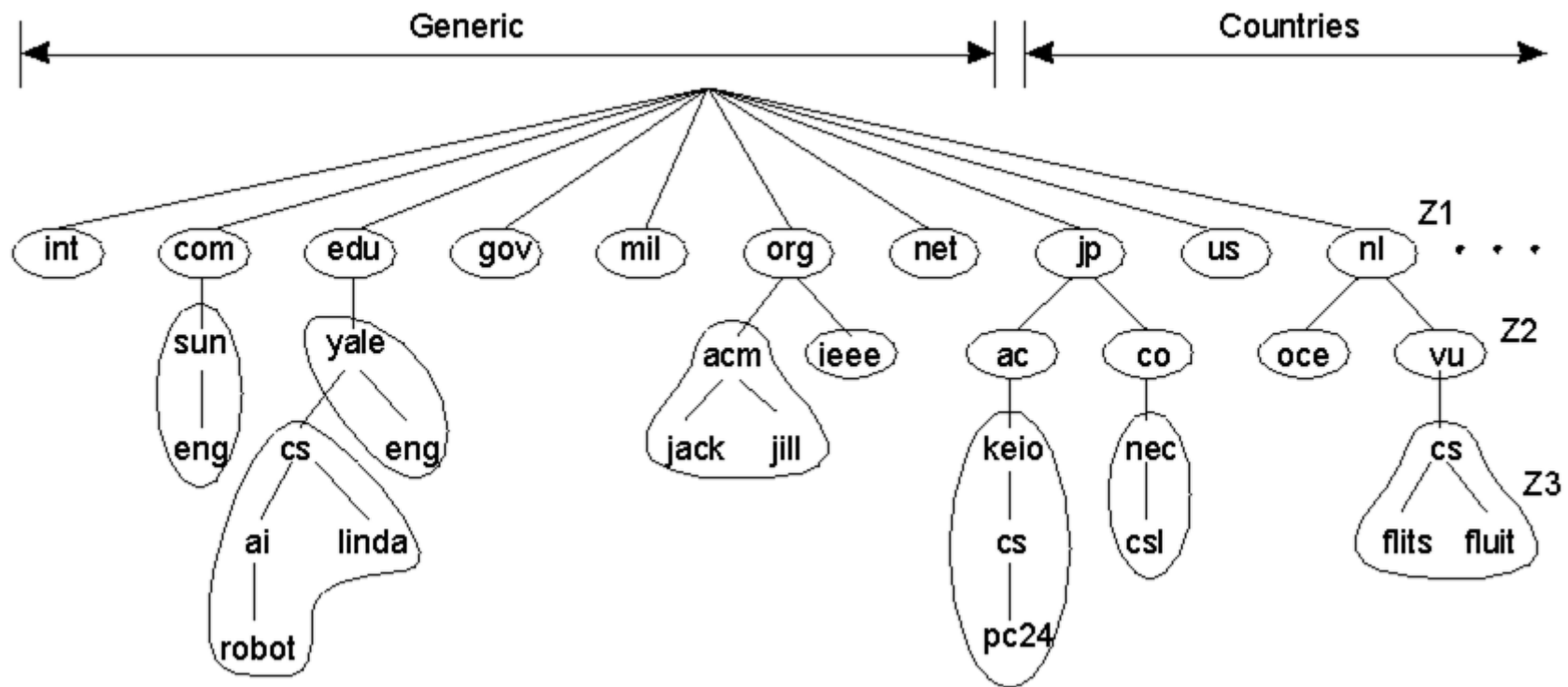
Ocultar a latência de comunicação

Usar quando possível mensagens assíncronas

AGRUPAR QUANDO POSSÍVEL



Distribuição



Distribuição

Model= "civic" AND Year= "2001" AND (Color= "green" OR Color= "white")

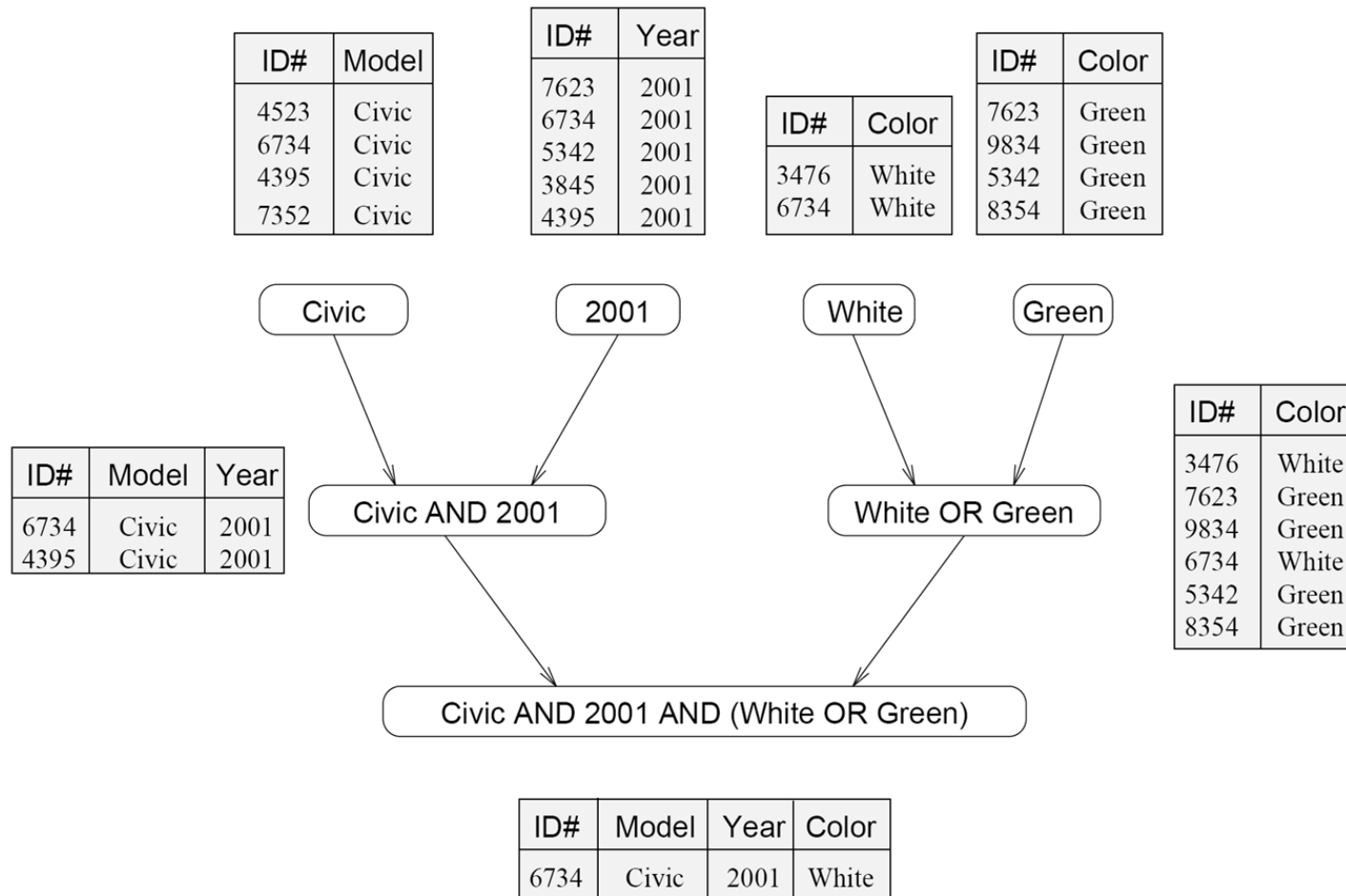
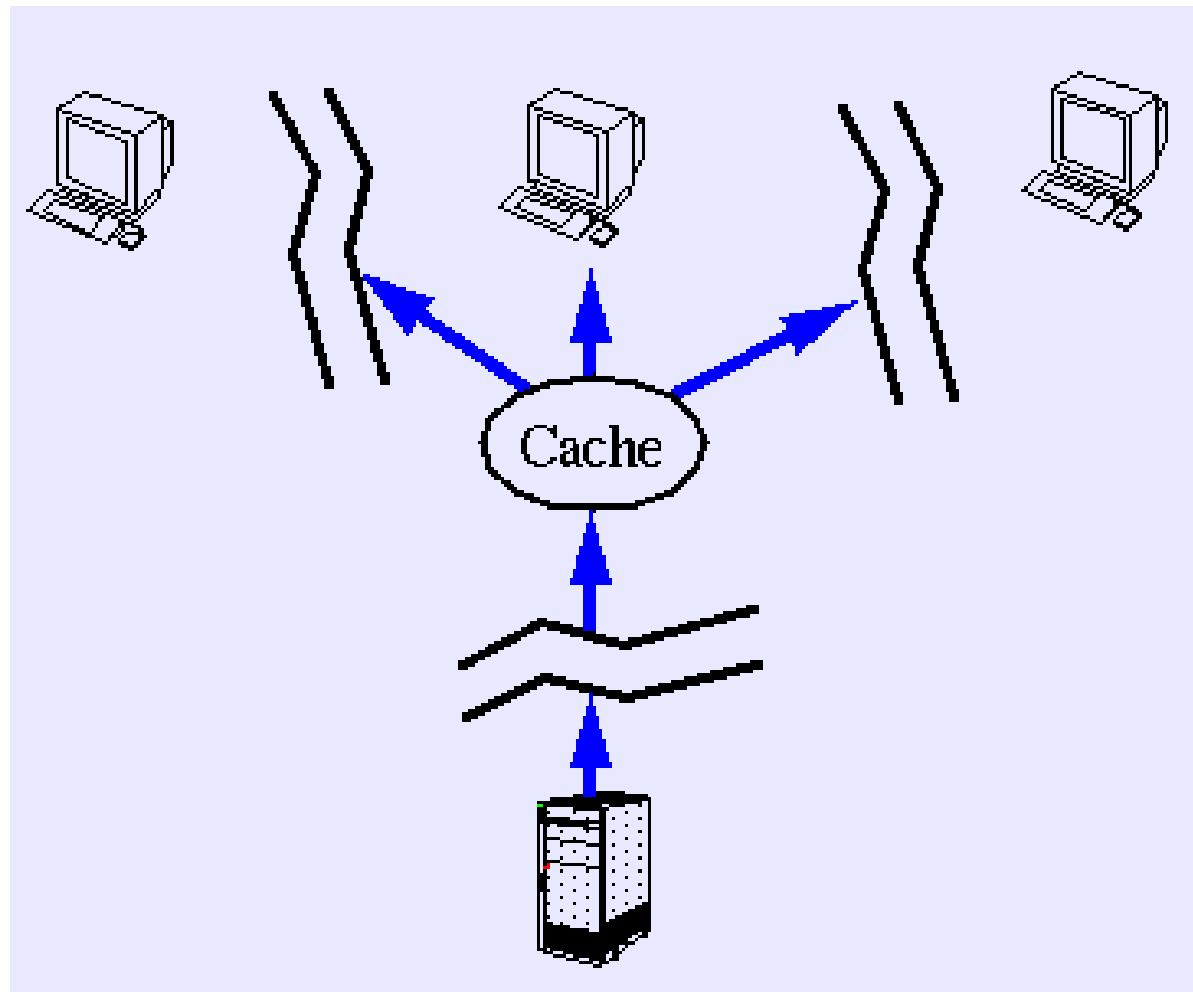


Figure 3.2 The different tables and their dependencies in a query processing operation.

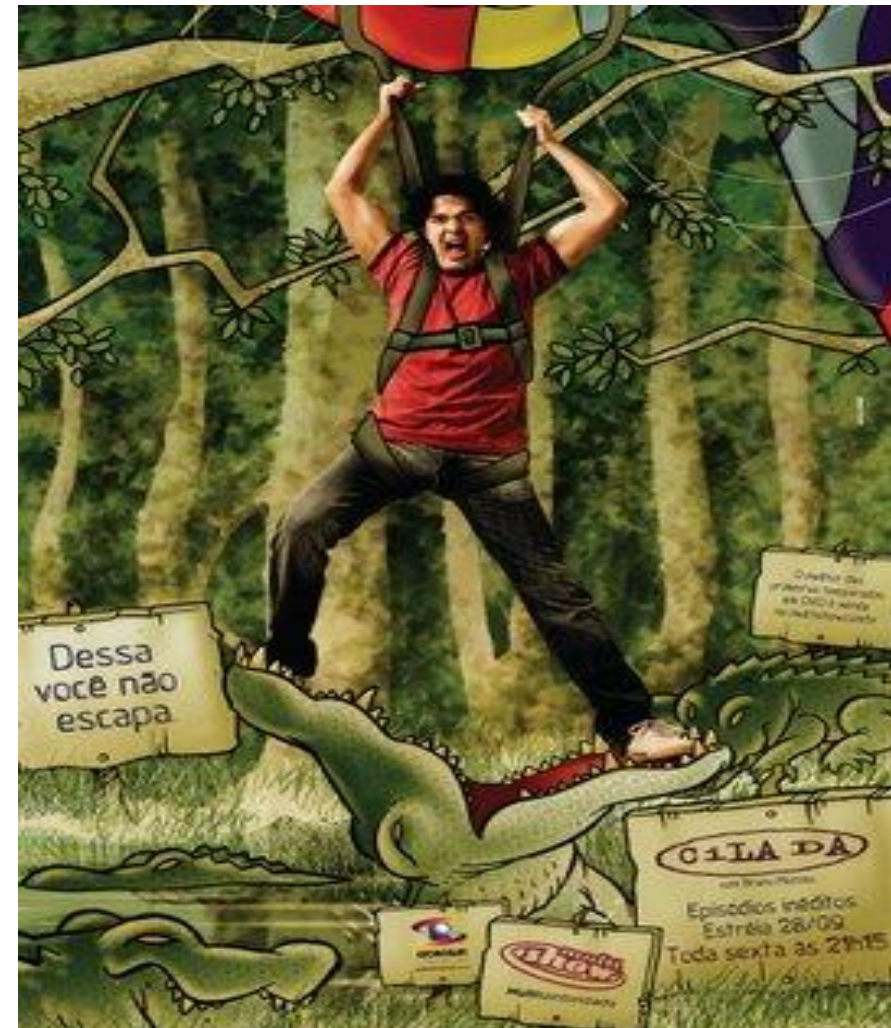
Replicação



SINCRONIZAÇÃO DE CACHE EM SISTEMAS DISTRIBUÍDOS !!!! %\$#@..%%

Ciladas

- I. A rede é confiável
- II. A rede é segura
- III. A rede é homogênea
- IV. A topologia não muda
- V. A latência é zero
- VI. A largura de banda é infinita
- VII. O custo de transporte é zero
- VIII. Há só um administrador



Cluster versus Grid Computing

Sistemas de computação de cluster (algumas APIs já fazem parte do serviço). Normalmente, arquitetura mestre-escravo, podendo garantir serviços como balanceamento, segurança, Tolerância a falhas, migração de código, entre outros....

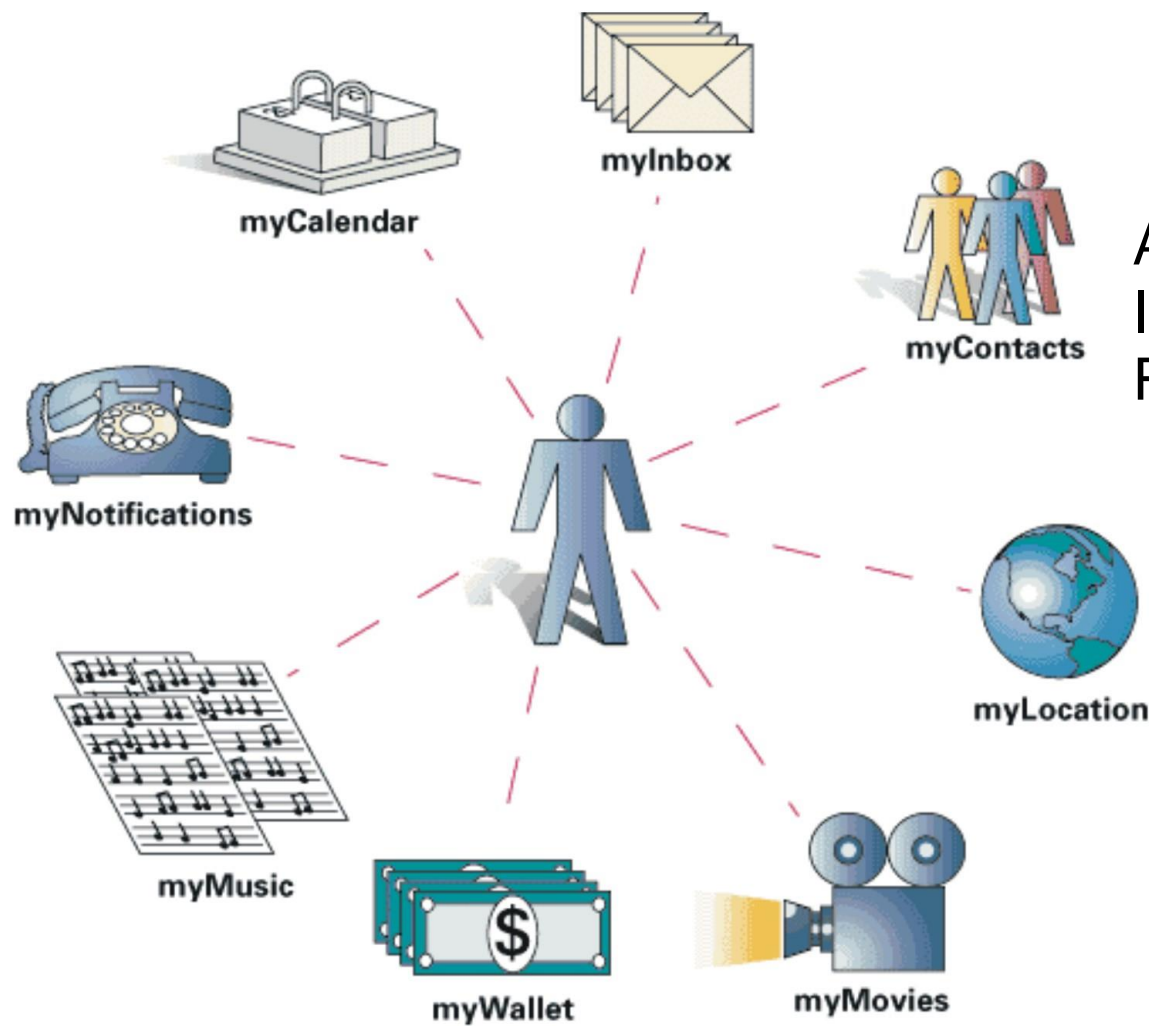
Sistemas de computação em grade: normalmente, os computadores são organizados em federações com domínios distintos, equipamento distintos, política de segurança distintas, entre outras...

Note que

CLUSTER = HOMOGENEIDADE

GRADE = HETEROGENEIDADE

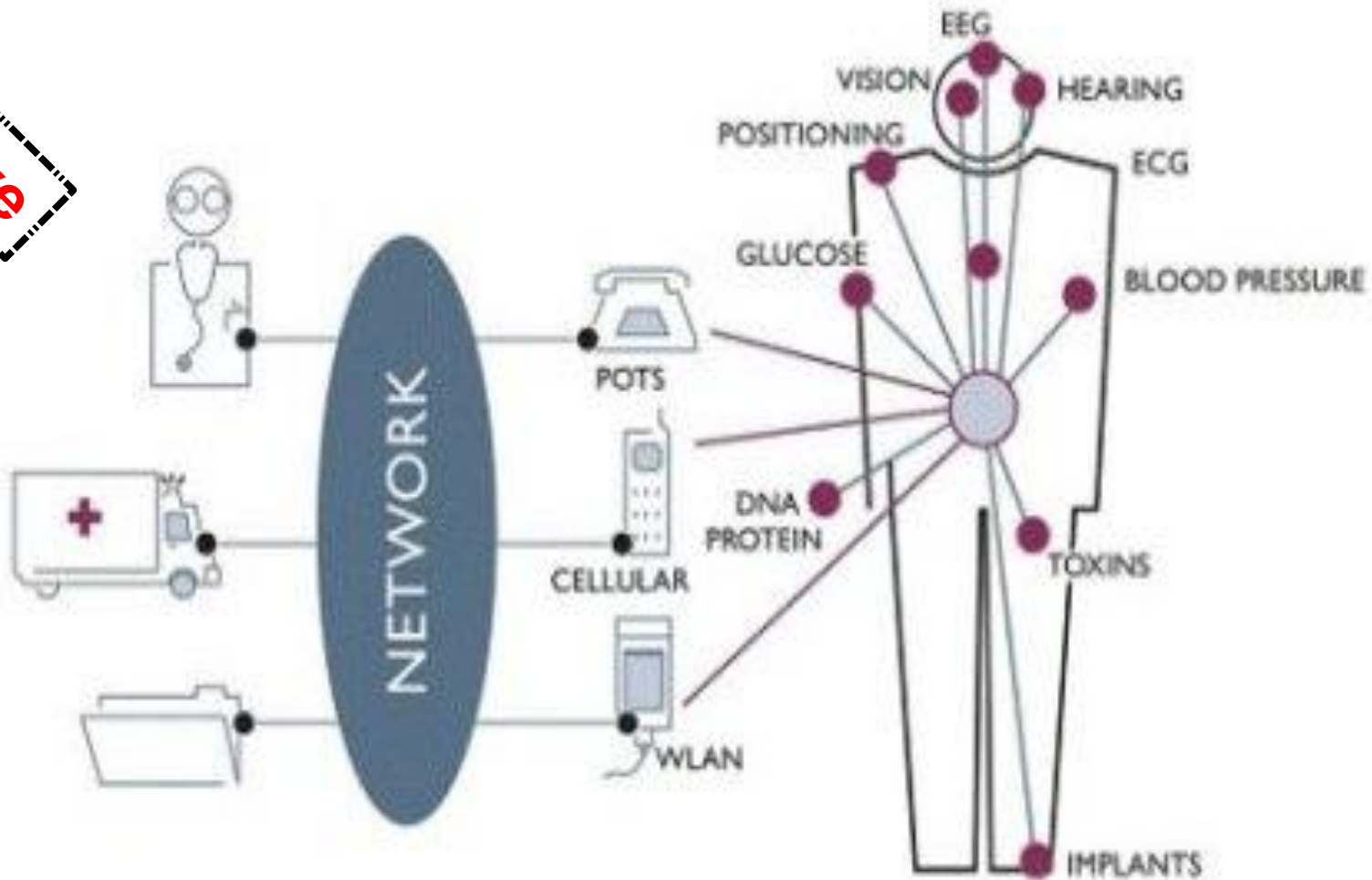
Sistemas distribuídos pervasivos



Adotar mudanças contextuais
Incentivar a composição ad hoc
Reconhecer compartilhamento
como padrão

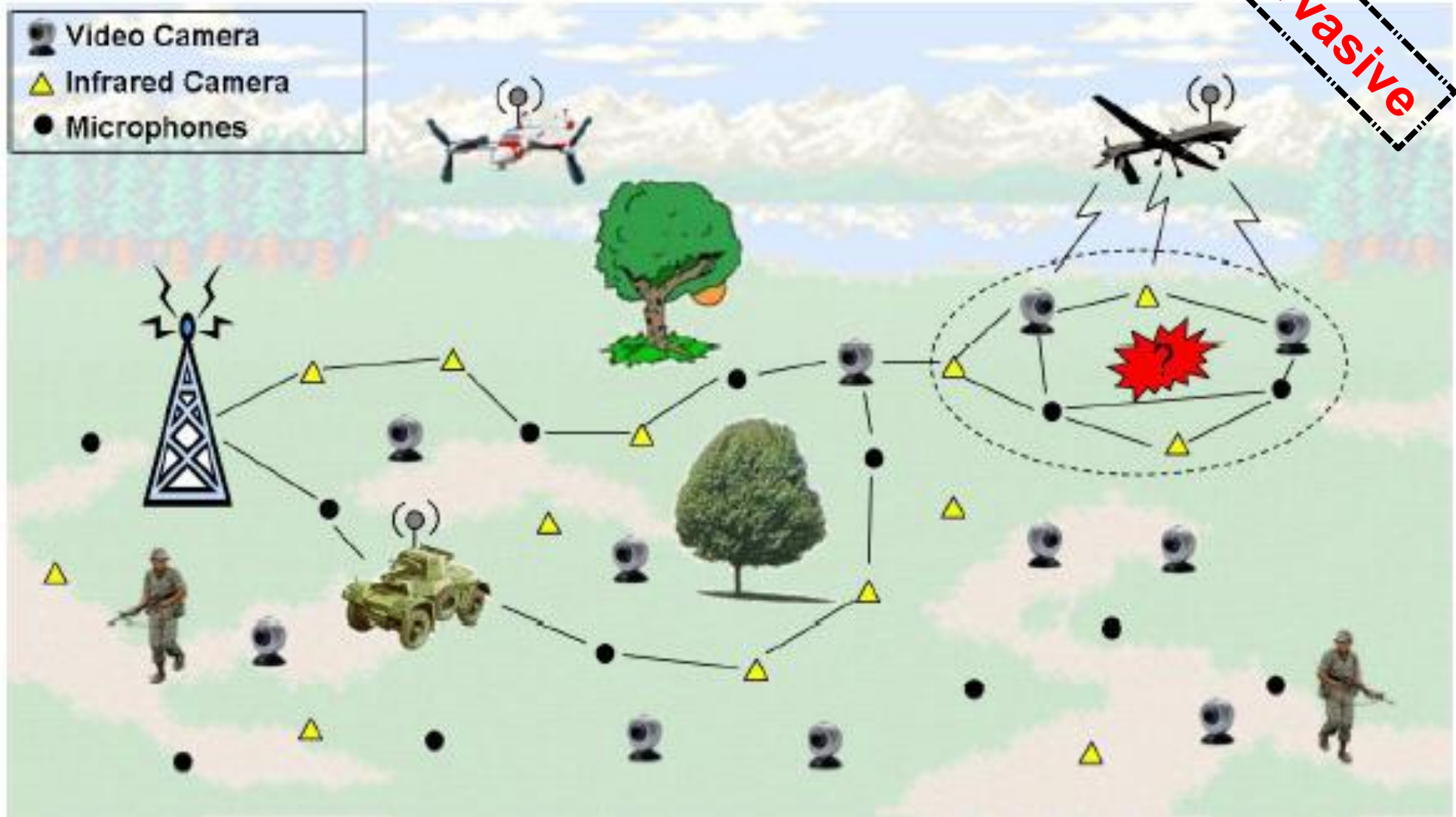
BODY AREA NETWORK

pervasive



SENSORES

pervasive



Cloud Computing

