



UNIFACS

ecossistema
ânima

Engenharia da Computação
TEMAS DISTRIBUÍDOS

Aula 1 – Sistemas Distribuidos



UNIFACS

Ana Cleyge Silva de Azevêdo

- **Bacharel em Sistemas de Informação - UNIFACS;**
- **Pós-Graduado Governança de TI – Católica de Brasília;**
- **Mestra em Computação Aplicada – UEFS.**

Contato: 75 991563683
ana.cleyge@animaeducacao.com.br

O que é um Sistema Distribuído?

**O que é
necessário?**



Introdução aos Sistemas Distribuídos



UNIFACS

Um sistema distribuído é aquele no qual os componentes de *hardware* ou *software* localizados em computadores interligados em rede, se comunicam e coordenam suas ações apenas passando mensagens.



Definição de um Sistema Distribuído



Coulouris et. al:

“Um Sistema Distribuído é aquele no qual os **componentes** de *hardware* ou *software*, localizados em computadores **interligados em rede**, se **comunicam** e **coordenam** suas ações apenas passando **mensagens**”.

Tanenbaum & Van Steen:

“Coleção de computadores autônomos que aparecem para os usuários do sistema como um único computador.”

Leslie Lamport:

“Sistema onde você não consegue trabalhar por causa de uma **falha** em um **computador que você nunca viu**.”

M. Eckhouse:

“Uma coleção de elementos de processamento interconectados, tanto logicamente como fisicamente, **para execução cooperativa de programas** de aplicação com o controle geral dos recursos centralizados.”

ã Ponto comum em todas as definições



UNIFACS

Apesar das definições serem distintas, elas tendem a convergir.

Ponto comum nas definições

- Vários componentes
- Conectados via rede
- Compartilhando recursos
- De forma transparente

O que é **transparência** em Sistemas Distribuídos?

Transparência, consiste em promover acesso a recursos distribuídos de forma oculta, como se fosse um único sistema para o usuário.

ã O que é um Sistema Distribuído?



UNIFACS

Um sistema distribuído é uma coleção de **hosts autônomos**, conectados através de uma **rede** de computadores. Cada *host* executa componentes e opera um **middleware** de distribuição, o qual habilita os componentes a **coordenarem** suas atividades de tal forma que usuários percebam o sistema como um **ambiente computacional único e integrado**.

Obs: A principal motivação para construir e usar sistemas distribuídos é proveniente do desejo de compartilhar recursos.





- Crescente dependência por parte dos usuários;
- Demanda maior que avanços combinados de *hardware* e *software* centralizados;
- Características inexistentes em sistemas centralizados como tolerância a falhas (*fault tolerance*).

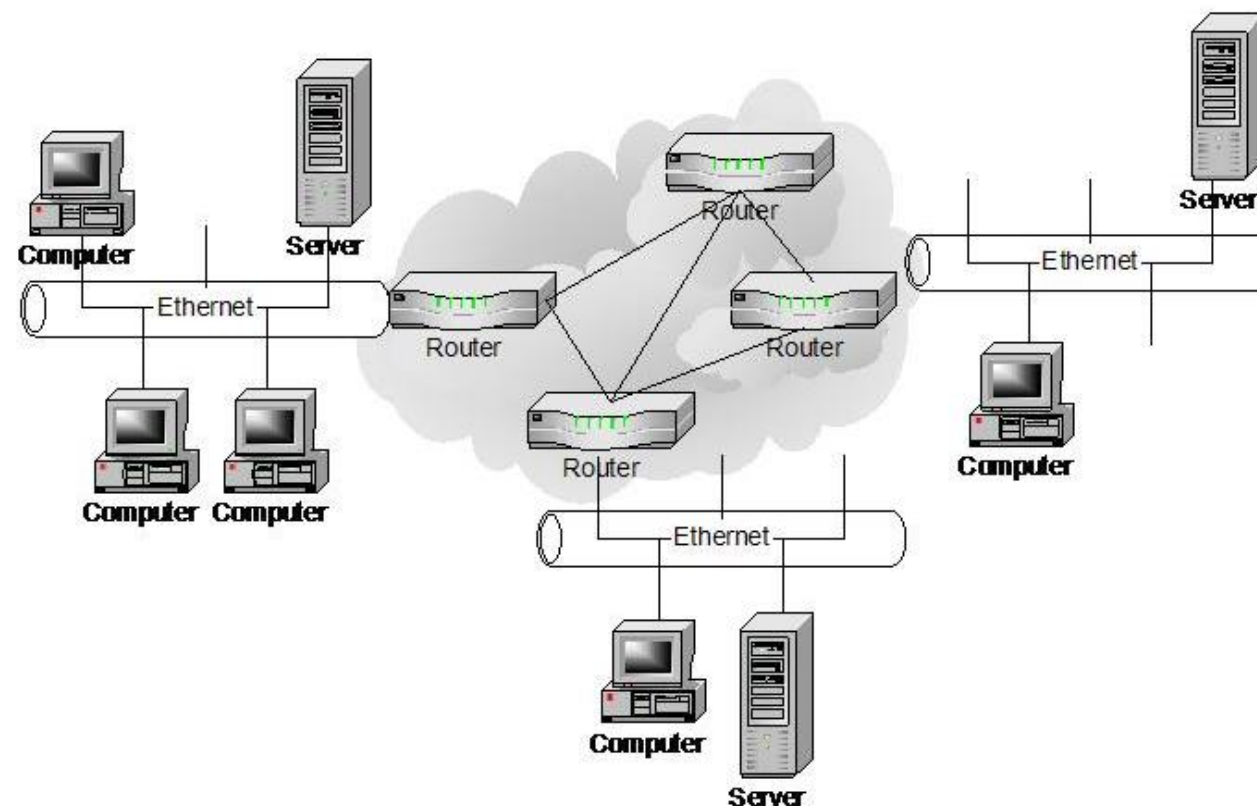
As redes estão em toda parte



UNIFACS

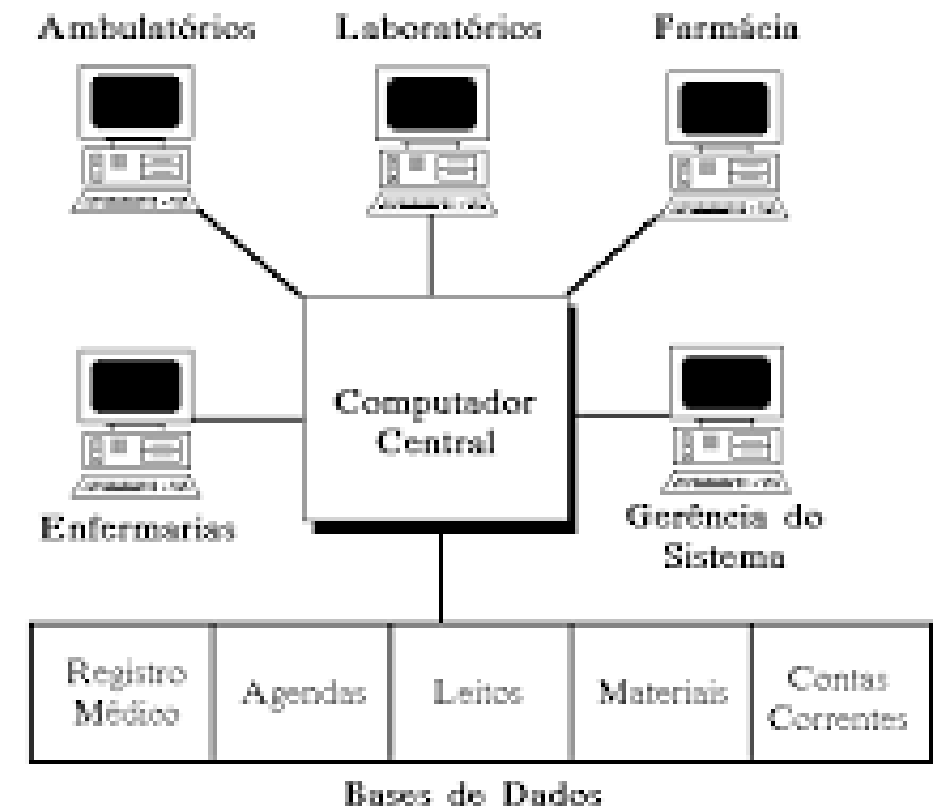


As redes estão em **todo** lugar!



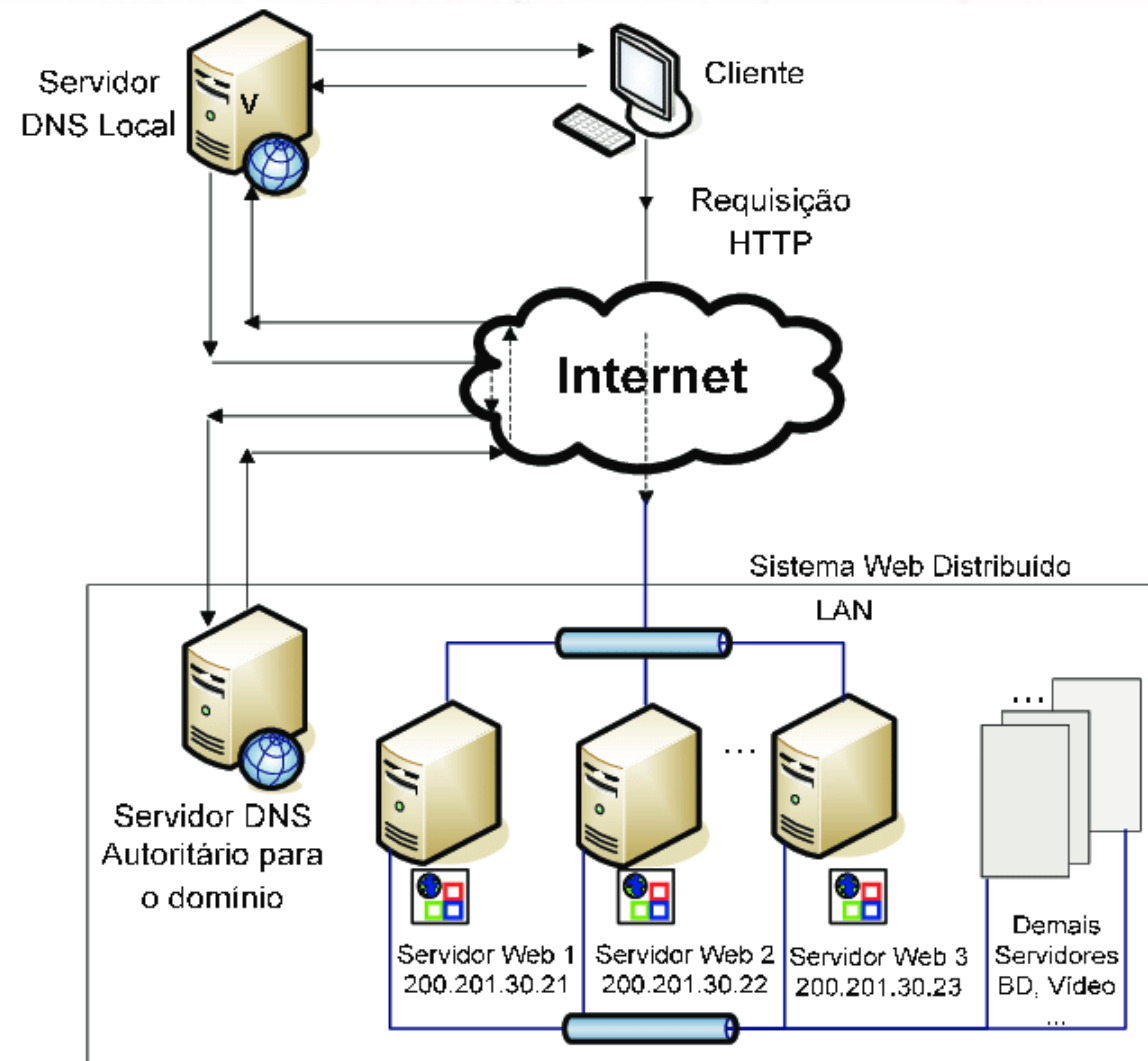
As redes podem estar próximas ou não.

- Um componente, com partes não autônomas – sistema de tráfego aéreo;
- Componentes são compartilhados por todos os usuários durante todo o tempo;
- Todos os recursos acessíveis (tipicamente);
- Software executa em um único processo;
- Ponto de controle único;
- Ponto de falha único;



Características de Sistemas Distribuídos

- Múltiplos componentes autônomos;
- Componentes não são compartilhados por todos os usuários;
- Recursos podem não ser acessíveis;
- Software executa em processos concorrentes e em processadores distintos;
- Múltiplos pontos de controle;
- Múltiplos pontos de falha.



Um Sistemas distribuídos tem as seguintes consequências importantes:

Concorrência:

Execução **concorrente** de programas.



Concorrência: em uma rede de computadores, a execução concorrente de programas é a norma. Posso fazer meu trabalho em meu computador, enquanto você faz o seu em sua máquina, compartilhando recursos como páginas Web ou arquivos, quando necessário.

Um Sistemas distribuídos tem as seguintes consequências importantes:

Inexistência de um relógio global:

Se não existe um relógio global, como os componentes **coordenam** suas ações?



Inexistência de relógio global: quando os programas precisam cooperar, eles coordenam suas ações trocando mensagens. A coordenação frequentemente depende de uma noção compartilhada do tempo em que as ações dos programas ocorrem.

Um Sistemas distribuídos tem as seguintes consequências importantes:



Falhas são independentes!

A rede falhou ou apenas ficou lenta?

Falhas não são imediatamente percebidas.

Falhas resultam no isolamento do componente que falhou.



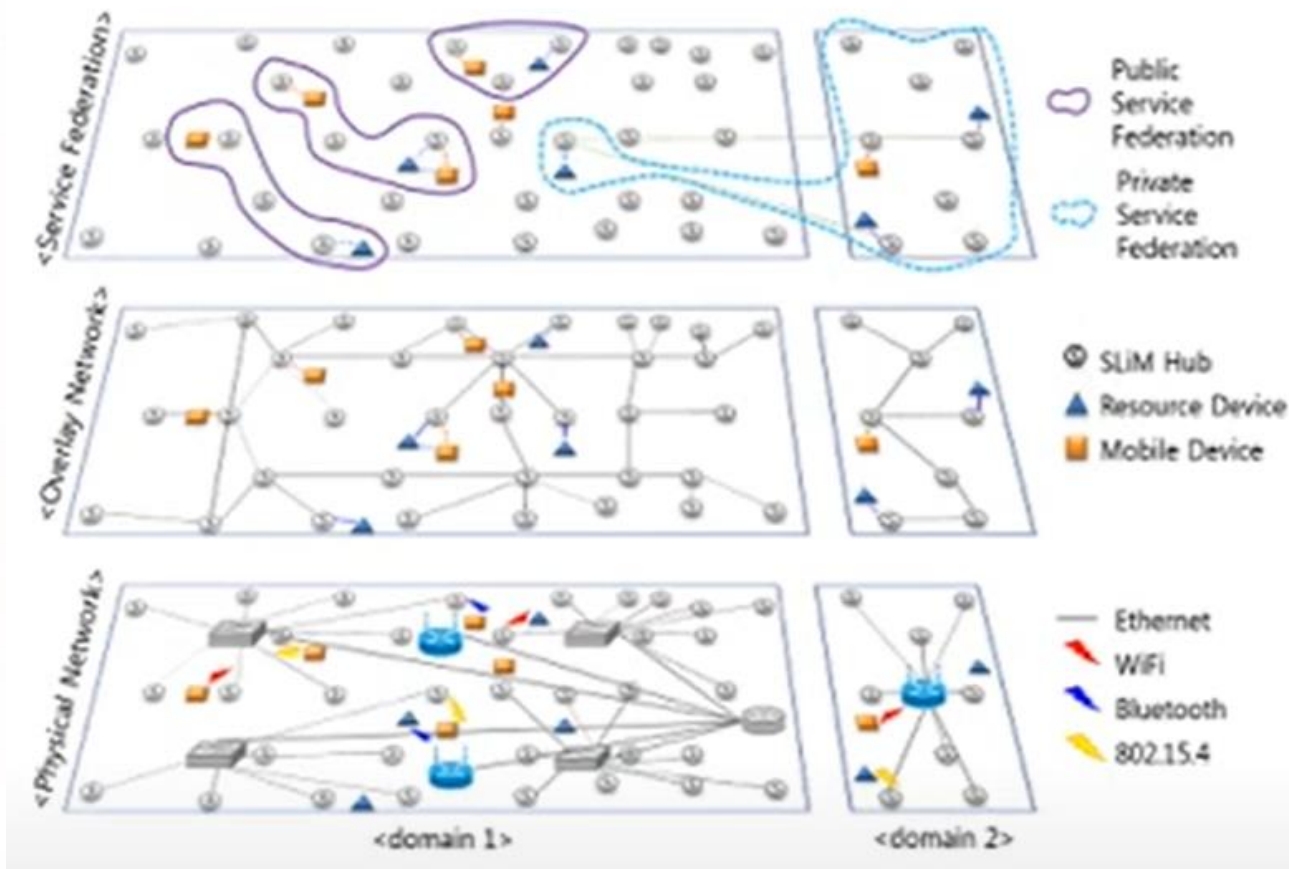
Falhas independentes: todos os sistemas de computador podem falhar, e é responsabilidade dos projetistas de sistema pensar nas consequências das possíveis falhas. Nos sistemas distribuídos, as falhas são diferentes.

Um Sistemas distribuídos tem as seguintes consequências importantes:

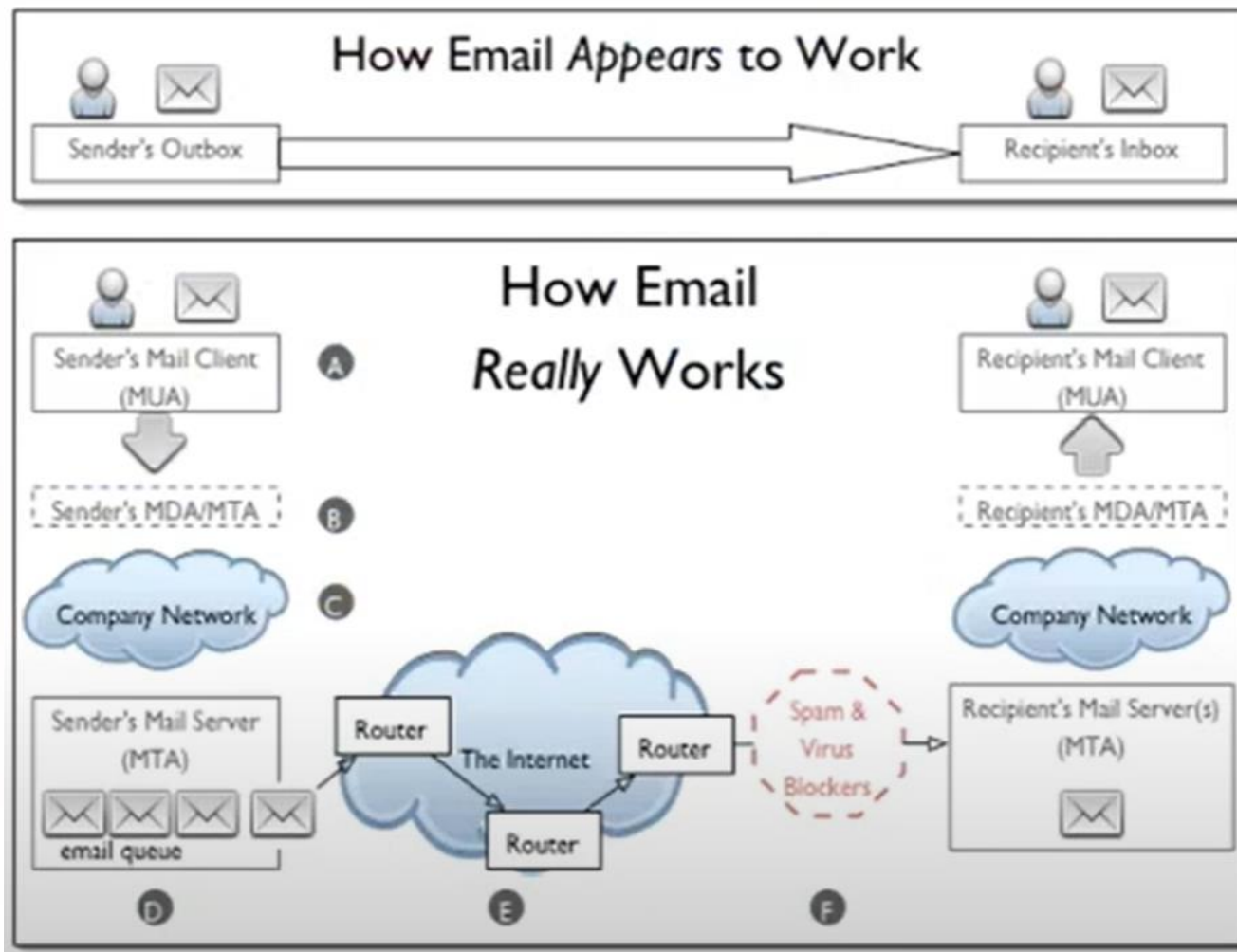
Por que distribuir?



- Maior funcionalidade;
- Maior escalabilidade;
- Maior robustez;
- Maior transparência.



- Vários tipos de computadores;
- Vários tipos de papéis (role);
- Várias formas de comunicação;
- Várias camadas lógicas (abstração).



- Vários tipos de computadores;
- Vários tipos de papéis (role);
- Várias formas de comunicação;
- Várias camadas lógicas (abstração).

Objetivos da Disciplina



UNIFACS

- Estudo de aspectos e técnicas gerais encontradas em muitos sistemas distribuídos;
- Foco em técnicas.





Vantagens de construir Sistemas Distribuídos



UNIFACS

Objetivo → **Compartilhamento de Recursos.**

Recursos:

- **Hardware;**
- **Software.**

Modelos:

- **Cliente-Servidor;**
- **Baseado em Objetos.**



Vantagens de construir Sistemas Distribuídos



UNIFACS

Objetivo → **Extensibilidade**

Extensões de Hardware e Software;
Adição de módulos e componentes;
Interfaces bem definidas;
Arquitetura baseada em pug-ins;
Mecanismo uniforme de comunicação.

Objetivo → **Concorrência**

Mais de um processo em execução a cada instante:

- Atividades separadas de usuários;
- Independência de recursos;
- Localização de processos servidores em computadores distintos.

Acesso concorrente a recursos compartilhados requer sincronização.



Vantagens de construir Sistemas Distribuídos



UNIFACS

Objetivo → **Escalabilidade**

Quantidade de trabalho envolvido no processamento de qualquer requisição de acesso a um recurso compartilhado independe do tamanho da rede.

Objetivo → **Tolerância a Falhas**

Capacidade de um sistema continuar funcionando adequadamente, ou pelo menos de forma aceitável, mesmo quando ocorrem falhas em seus componentes ou em sua infraestrutura. Isso significa que o sistema é projetado para ser resistente a problemas imprevistos, como falhas de hardware, erros de software, falhas de rede ou outras interrupções, sem resultar em uma interrupção total dos serviços oferecidos pelo sistema.



Vantagens de construir Sistemas Distribuídos



UNIFACS

Objetivo → **Transparência**

Esconder do usuário e do programador de aplicações a separação de componentes em um sistema distribuído, tal que este seja visto como um sistema centralizado

Técnicas acesso, localização, concorrência, replicação, falha, migração, desempenho e escala

- Custo com manutenção;
- Falha em rede;
- Vulnerabilidade na segurança;
- Complexidade operacional;
- Escalabilidade complexa.

Modelo Cliente Servidor



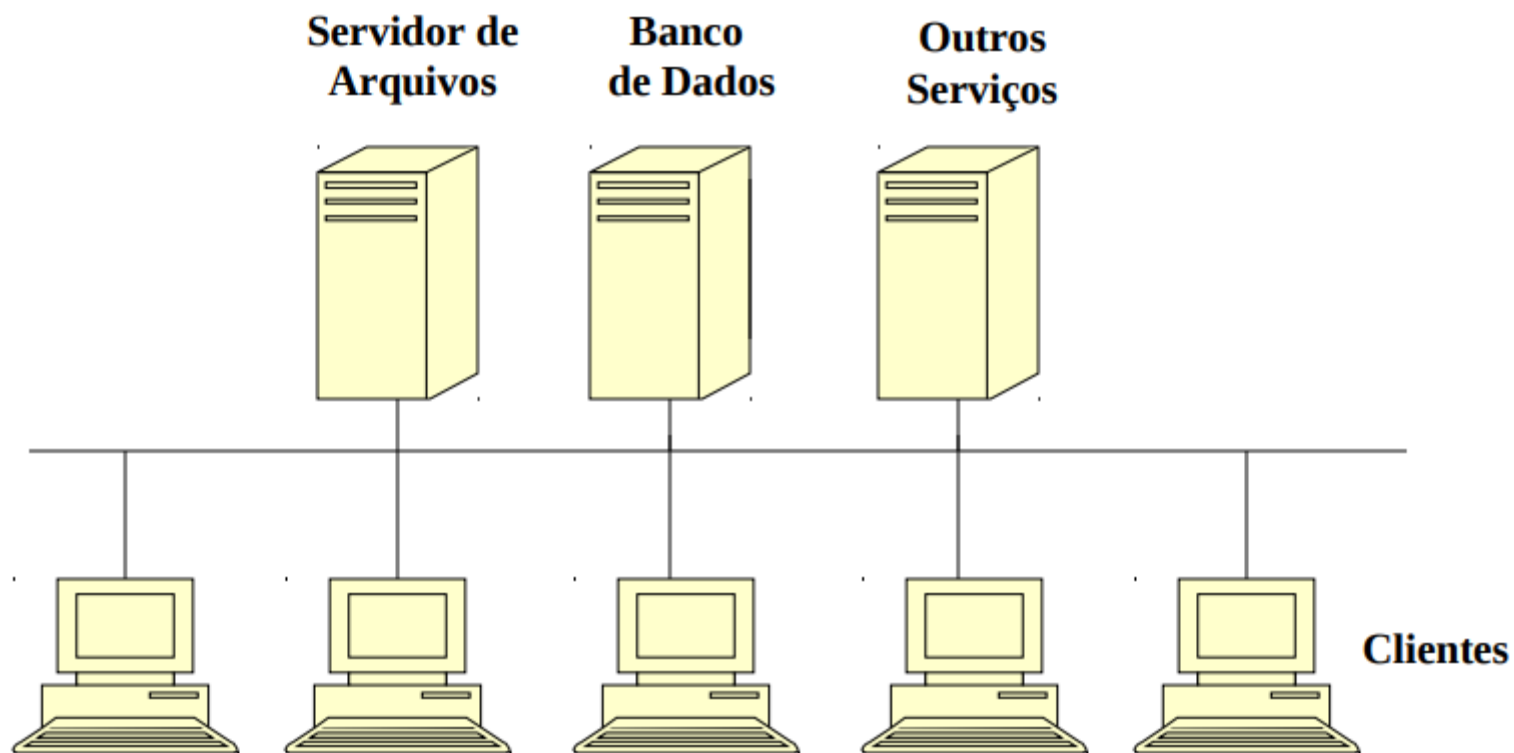
UNIFACS



Modelo Cliente Servidor (Estrutura Básica)



UNIFACS

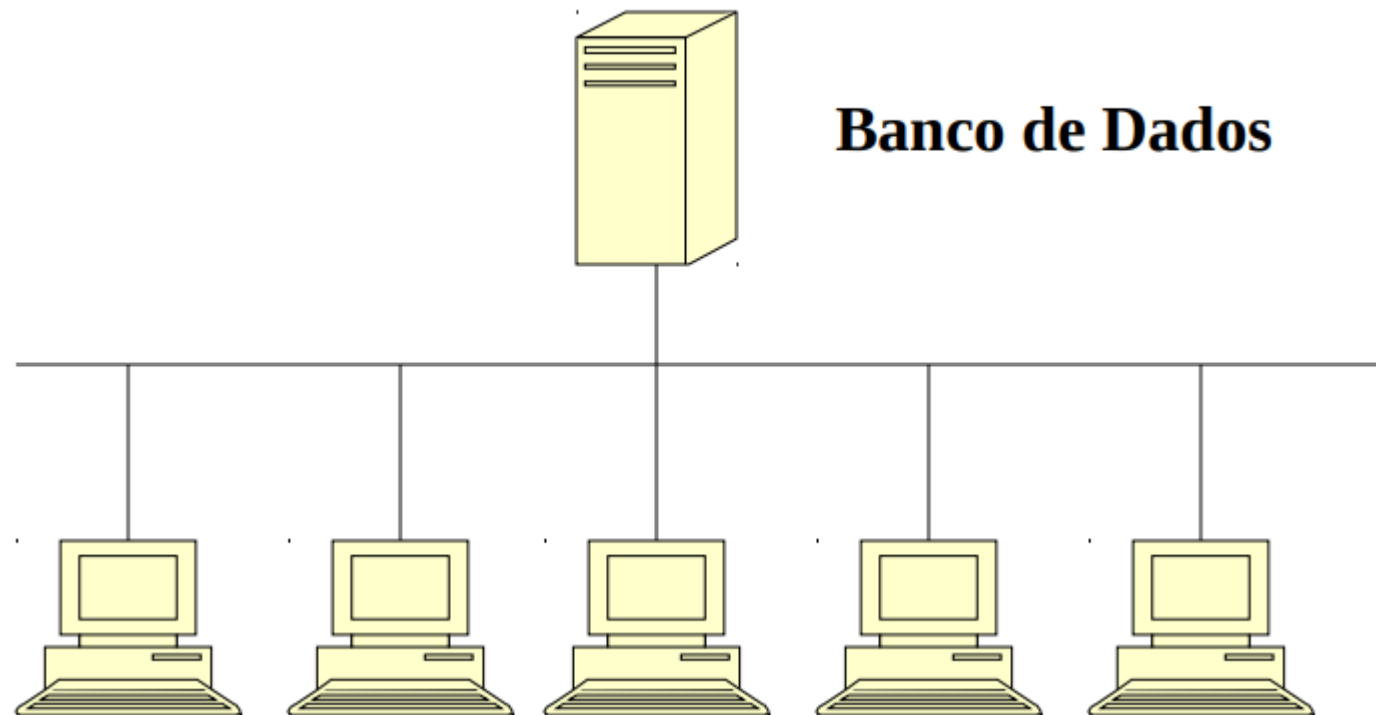


Modelo Cliente Servidor (Duas Camadas)



UNIFACS

Lógica de Negócio no Cliente

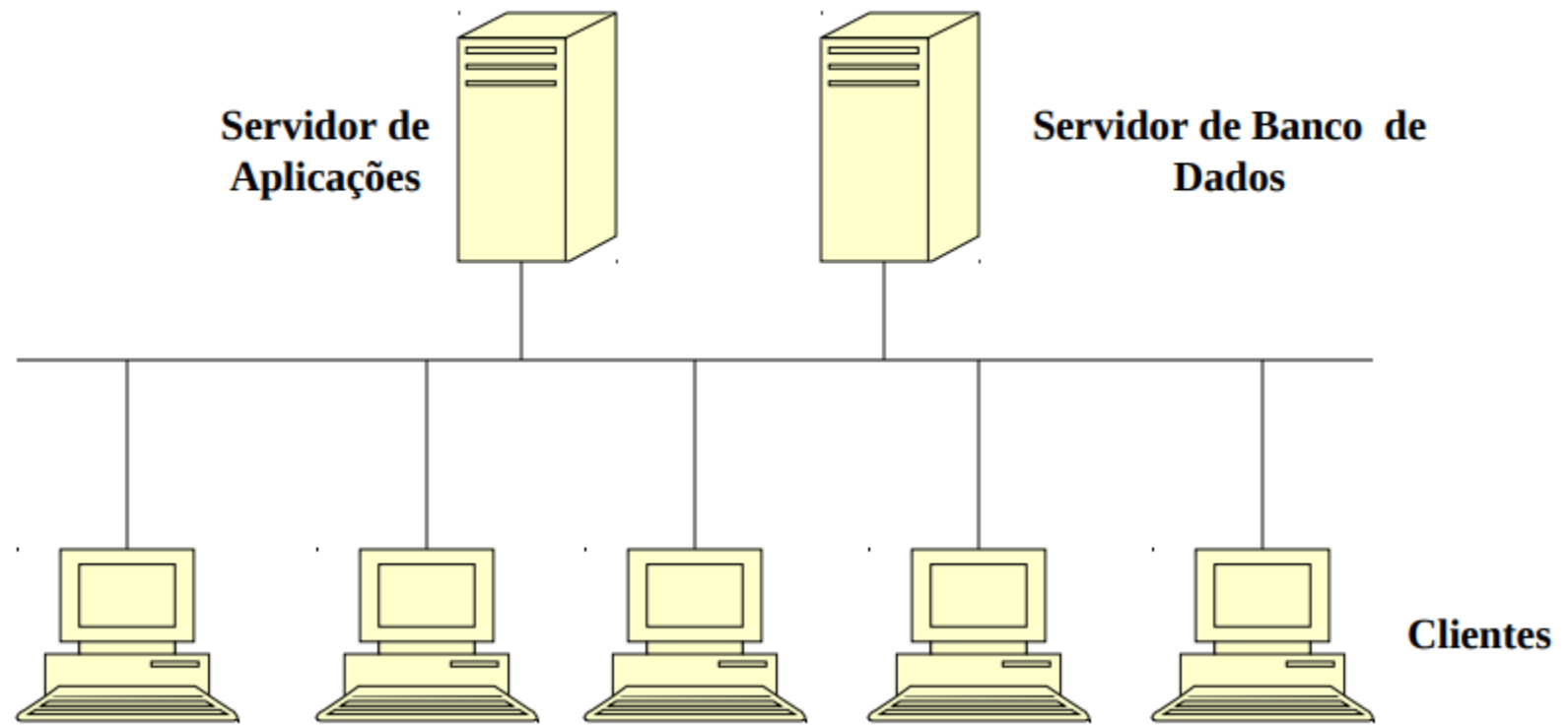


➤ Modelo Cliente Servidor (Três Camadas)



UNIFACS

Remove a Lógica de Negócio no Cliente

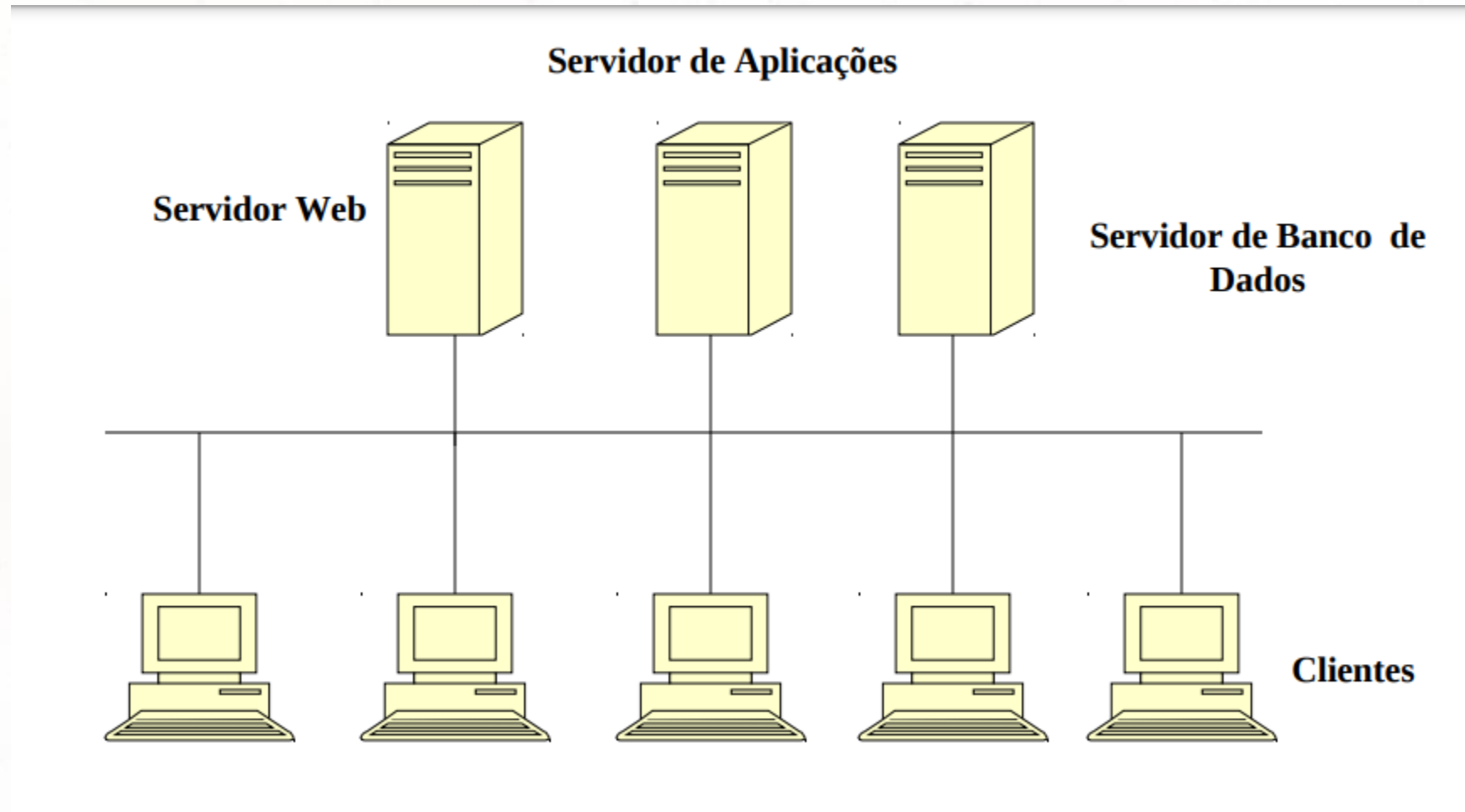


➤ Modelo Cliente Servidor (Quatro Camadas)



UNIFACS

Remove camada de Apresentação do Cliente



Exemplos de Sistemas Distribuídos

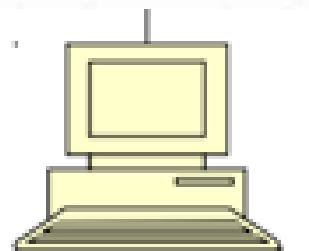


UNIFACS

requisição



www.algumserviço.com



Servidor Web



Servidor de Aplicações



Servidor de Banco de
Dados

Exemplos de Sistemas Distribuídos

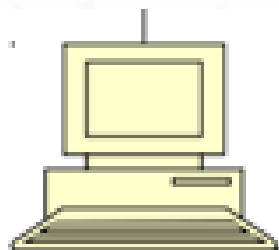


UNIFACS

interface



www.algumserviço.com



Servidor Web



Servidor de Aplicações

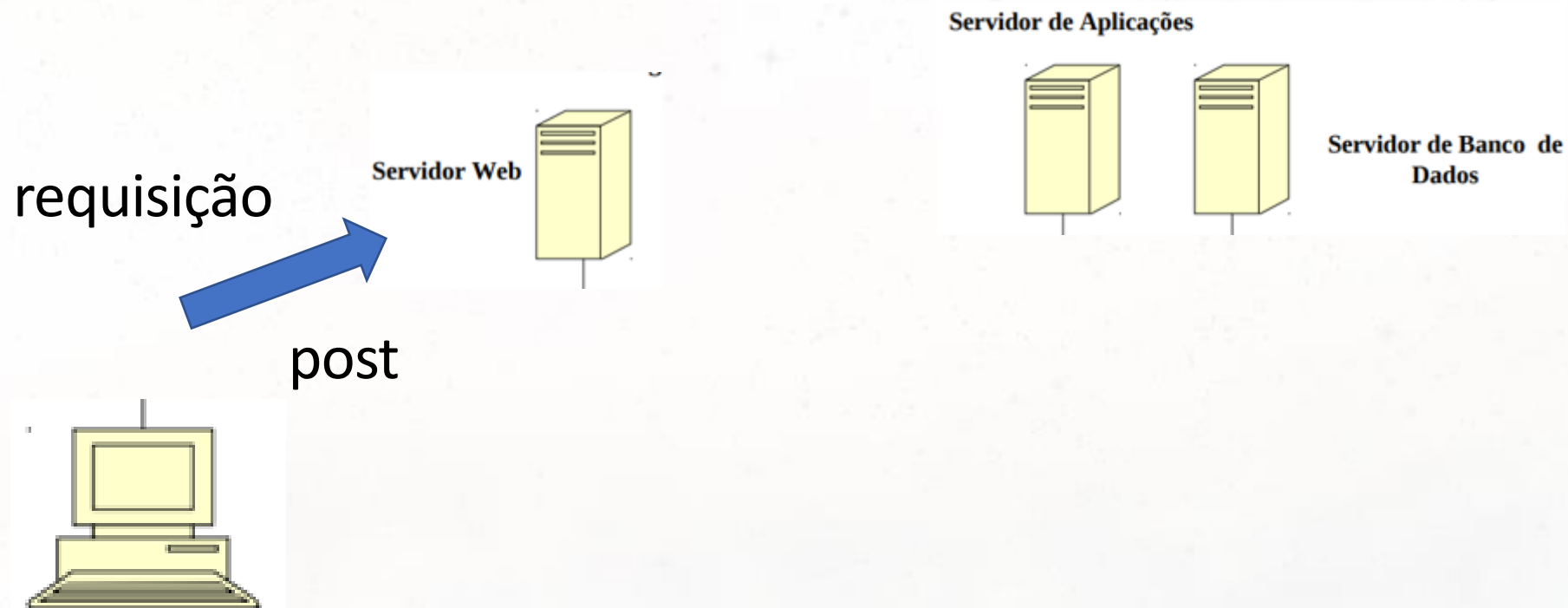


Servidor de Banco de
Dados

Exemplos de Sistemas Distribuídos



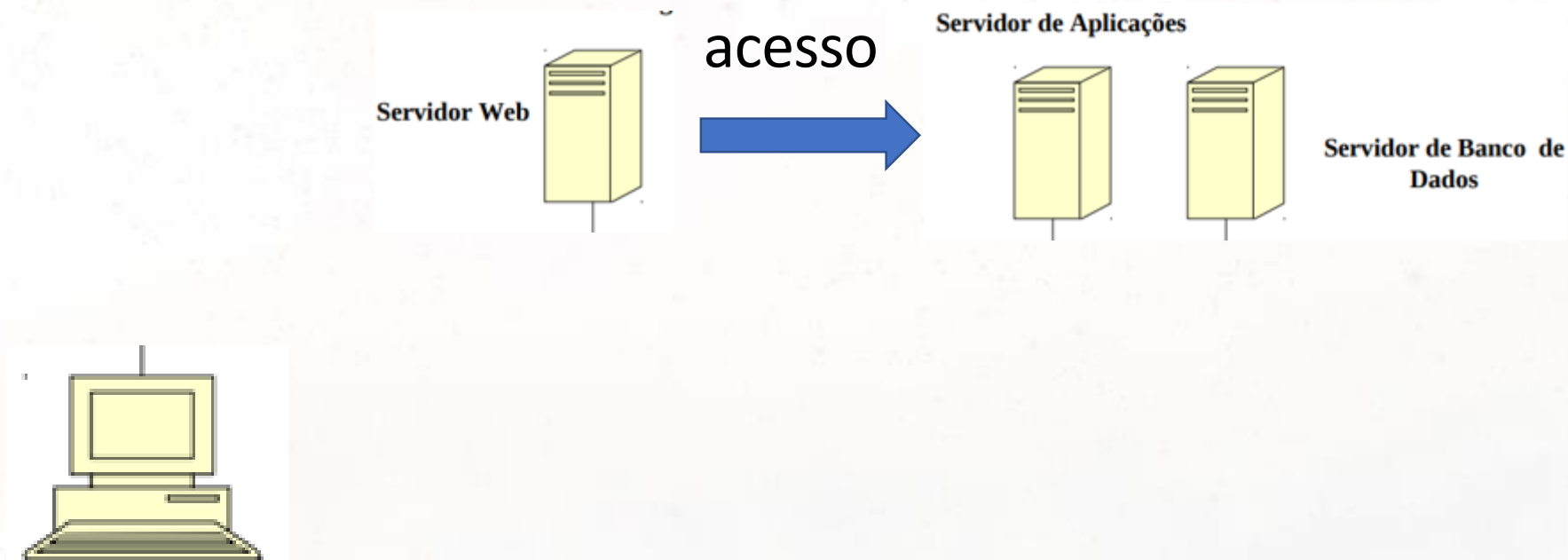
UNIFACS



Exemplos de Sistemas Distribuídos



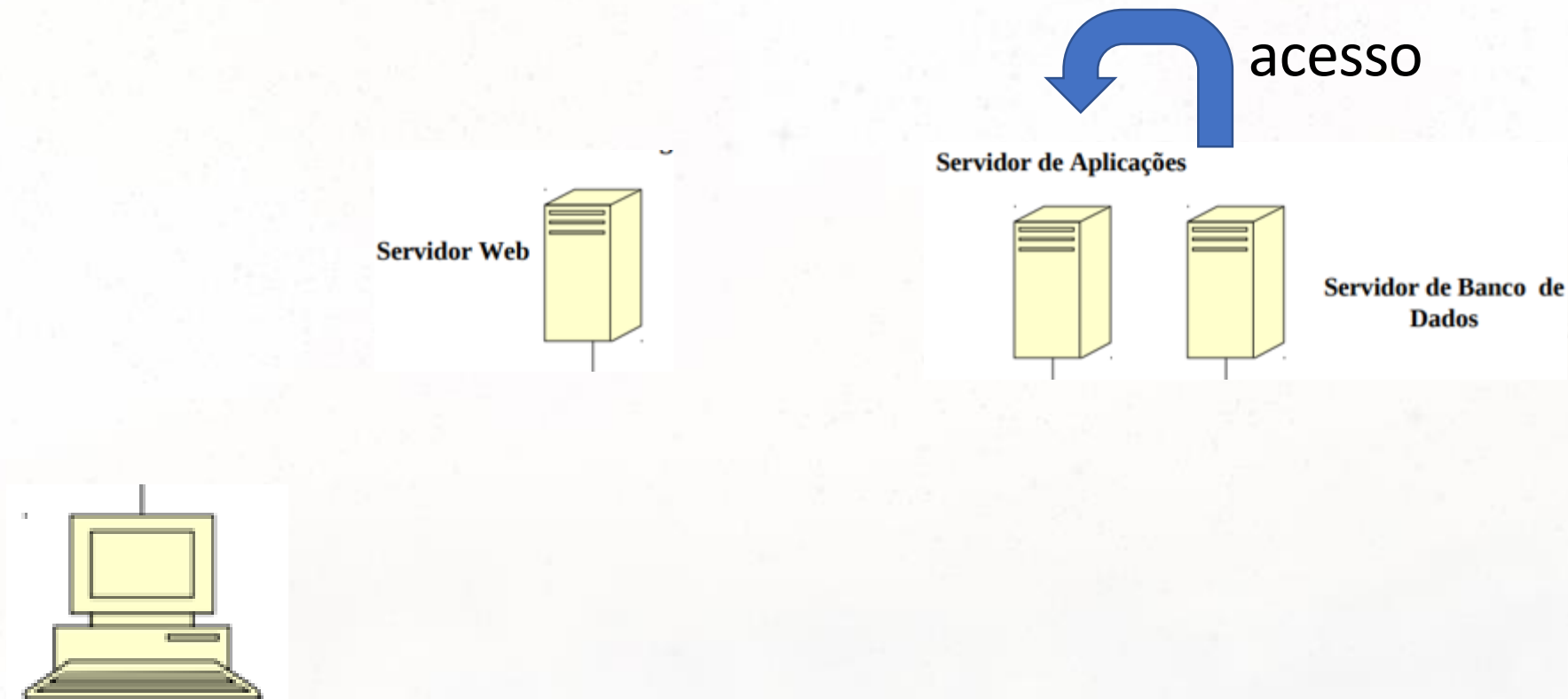
UNIFACS



Exemplos de Sistemas Distribuídos



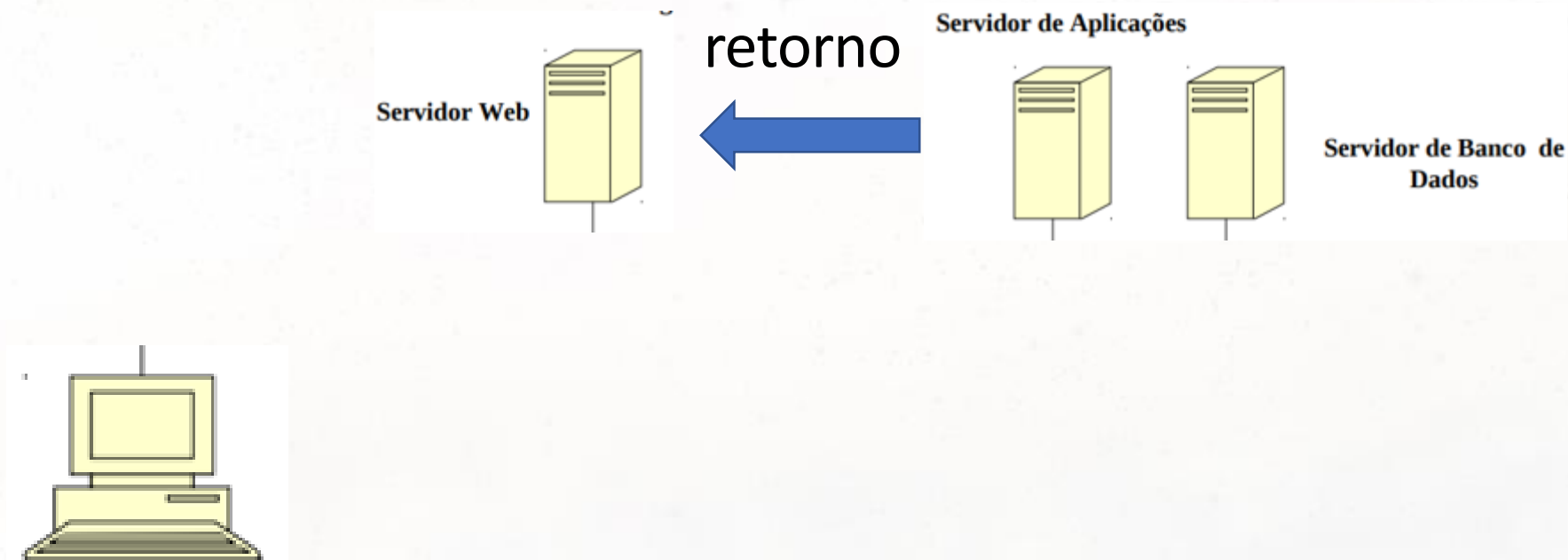
UNIFACS



Exemplos de Sistemas Distribuídos



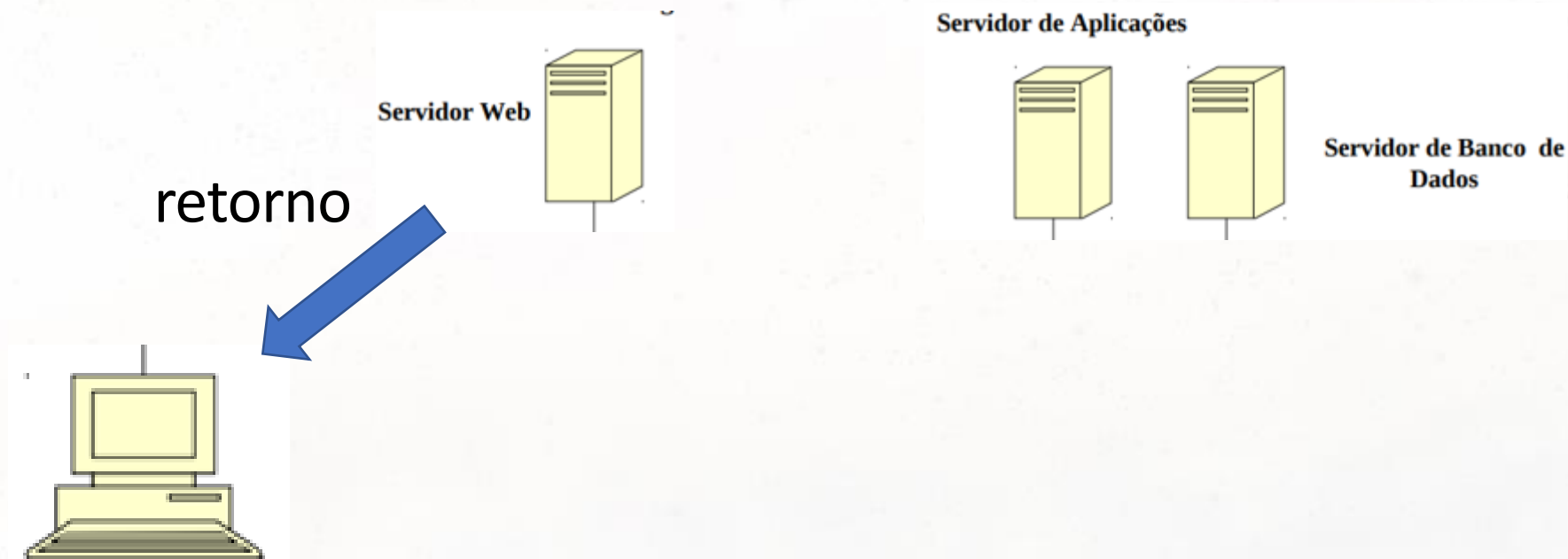
UNIFACS

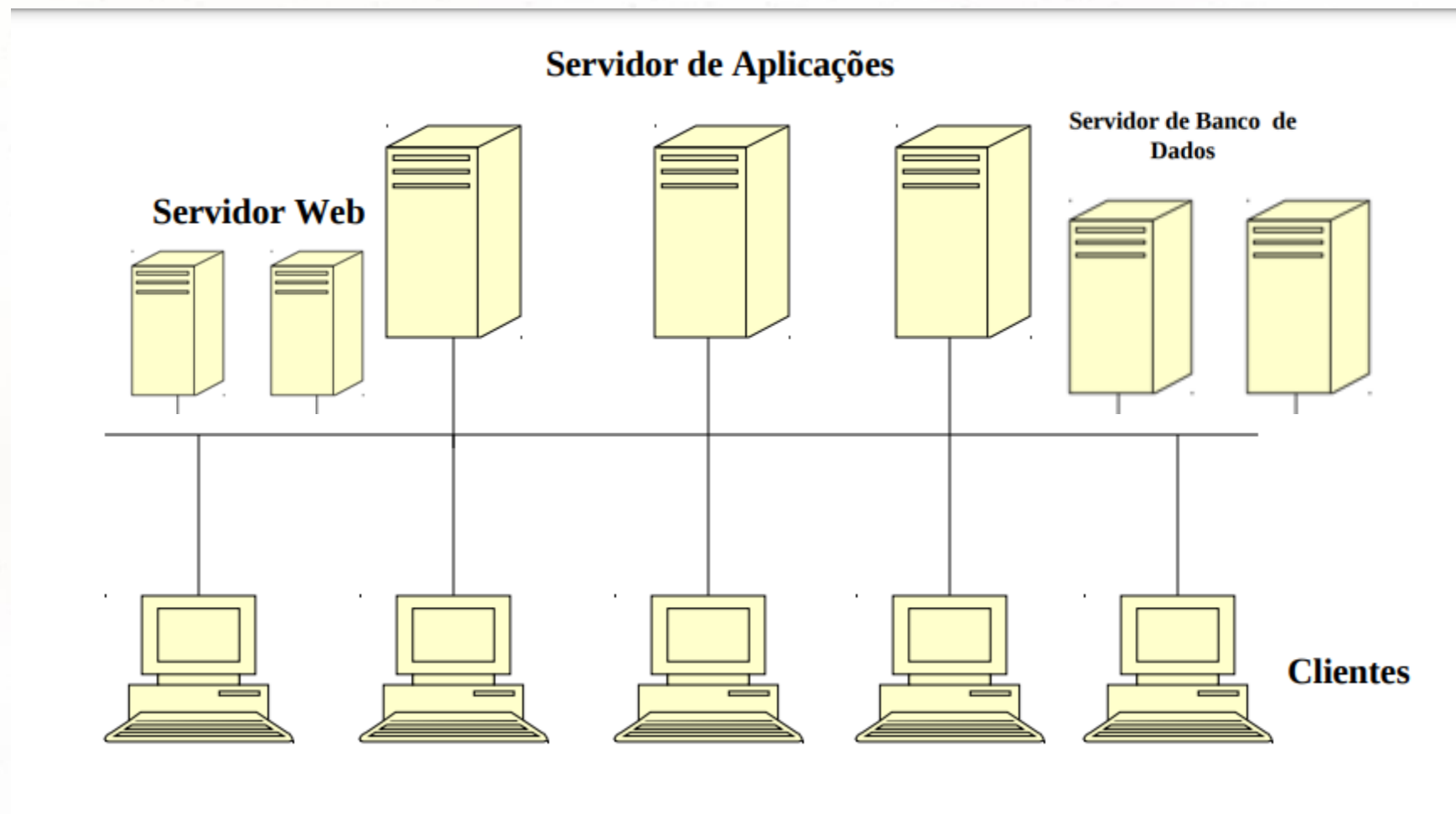


Exemplos de Sistemas Distribuídos



UNIFACS





- Banco de Dados Distribuídos;
- Clusters;
- Grid;
- Aplicações Multimídias.

- Cite e explique três características dos sistemas distribuídos;
- Diferencie o que vem a ser extensibilidade de escalabilidade em relação a sistemas distribuídos;
- Defina sistema cliente-servidor;
- Caracterize o cliente e o servidor;
- Defina e exemplifique uma aplicação em duas camadas
- Defina e exemplifique aplicação em três camadas
- Diferencie uma aplicação em três camadas de outra em quatro camadas.

COULOURIS, George; Dollimore, Jean; Kindberg, Tim. Sistemas Distribuídos: Conceitos e Projeto. 5 ed.. Editora: Bookman, 2013.

DEITEL, Harvey M.; Choffnes, D. R.; Deitel, Paul J. Sistemas operacionais. 3. ed. São Paulo: Pearson Prentice Hall, 2005.

TANENBAUM, Andrew S.; STEEN, Maarte Van. Sistemas Distribuídos: Princípios e Paradigmas. 2 ed.. São Paulo: Pearson Prentice-Hall, 2007.

LECHETA, Ricardo R. Google Android: aprenda a criar aplicações para dispositivos móveis com android SDK. 2. ed. São Paulo: Novatec, 2010. 608 p. ISBN 9788575222447 (broch.)..

Obrigado(a)!

Professor Emerson S. Oliveira
emerson.oliveira@unifacs.br

Professor Luis Gustavo Araujo
luis.araujo@unifacs.br



UNIFACS