



UNIDADE I

Sistemas Operacionais

Prof. Me. Michel Fernandes

Apresentação da disciplina

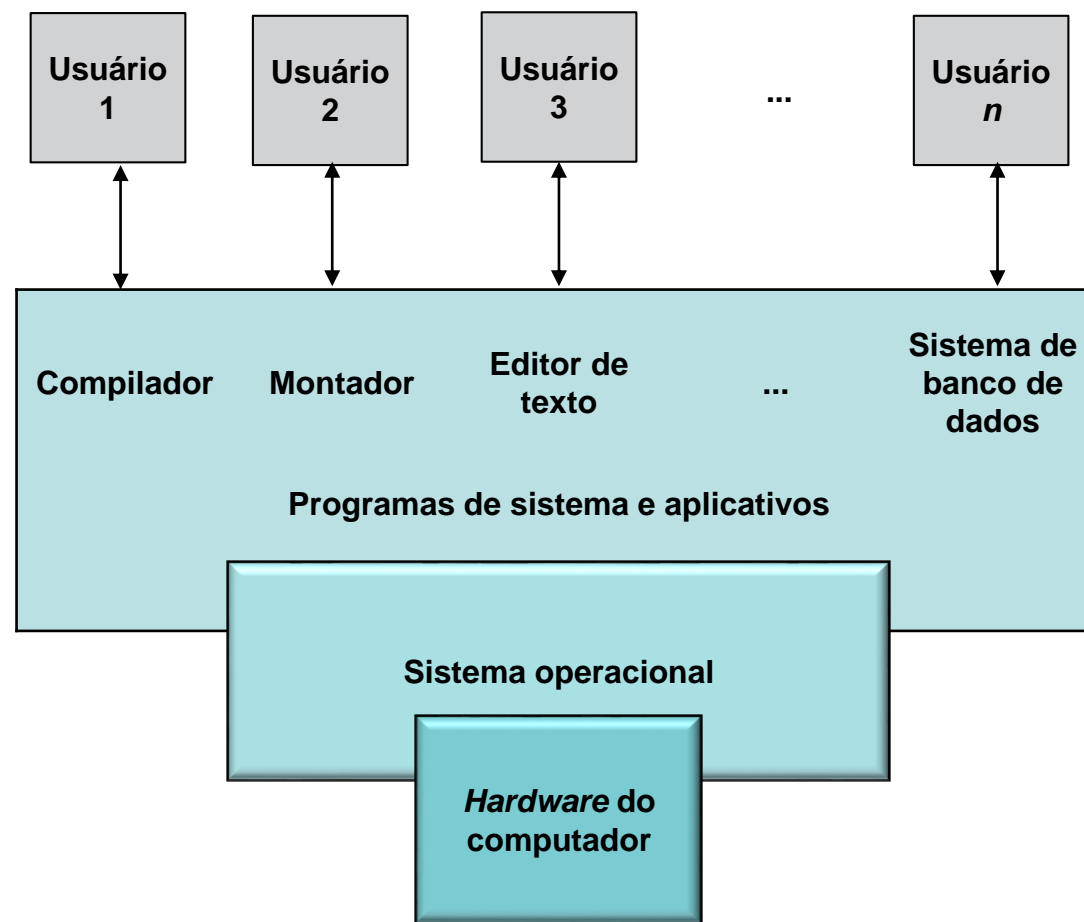
Entender, analisar e recomendar a evolução dos sistemas operacionais, gerenciamento de:

- Processos.
- Memória (hierarquia de memória, modos de endereçamento, estruturas de controle, memória virtual).
- Arquivos.
- Entrada e saída.
- E própria interface gráfica.



Visão geral do sistema operacional

- Camada de *software* que opera entre o *hardware* e os programas aplicativos voltados ao usuário final.
- Incorpora aspectos de baixo nível (como *drivers* de dispositivos e gerência de memória física) e de alto nível (como programas utilitários e a própria interface gráfica).



Visão abstrata de um sistema computacional.

Funções essenciais do sistema operacional

Pode-se sumarizar em duas palavras-chave os objetivos básicos de um sistema operacional:

Abstração:

- Os sistemas operacionais devem prover uma camada de abstração para usuários e aplicativos no uso dos recursos físicos.

Gerência:

- Cabe ao sistema operacional baseado nas políticas de sua estrutura gerenciar o uso dos recursos de *hardware* e administrar disputas e conflitos.

Abstração

- Cada *hardware* tem sua particularidade.
- Cabe ao sistema operacional gerenciar as particularidades do *hardware*.
- Um processador de texto não deve deter o conhecimento de como gravar um arquivo (disquetes, *pen drives*, HDs).



Abstração

Cabe ao sistema operacional:

- Prover interfaces de acesso aos dispositivos mais simples de usar que as interface de baixo nível.
- Tornar os aplicativos independentes do *hardware*.
- Definir interfaces de acesso homogêneas para dispositivos com tecnologias distintas.

Gerência

- Os programas, por meio do sistema operacional, usam o *hardware* para atingir seus objetivos. Cabe ao sistema operacional definir políticas para gerenciar o uso dos recursos de *hardware* pelos aplicativos e resolver eventuais disputas e conflitos.
 - Uso de processador.
 - Acesso a disco.
 - Memória.

Exemplos de gerência:

- O uso dos processadores deve ser distribuído entre os aplicativos ativos no sistema, de forma que cada um deles possa executar no tempo, sequência e velocidade adequada para cumprir suas funções sem prejudicar os outros.
- Utilização de memória RAM deve ser distribuída de forma justa entre as aplicações.

Estrutura do ambiente computacional

- Estrutura do ambiente computacional típico.

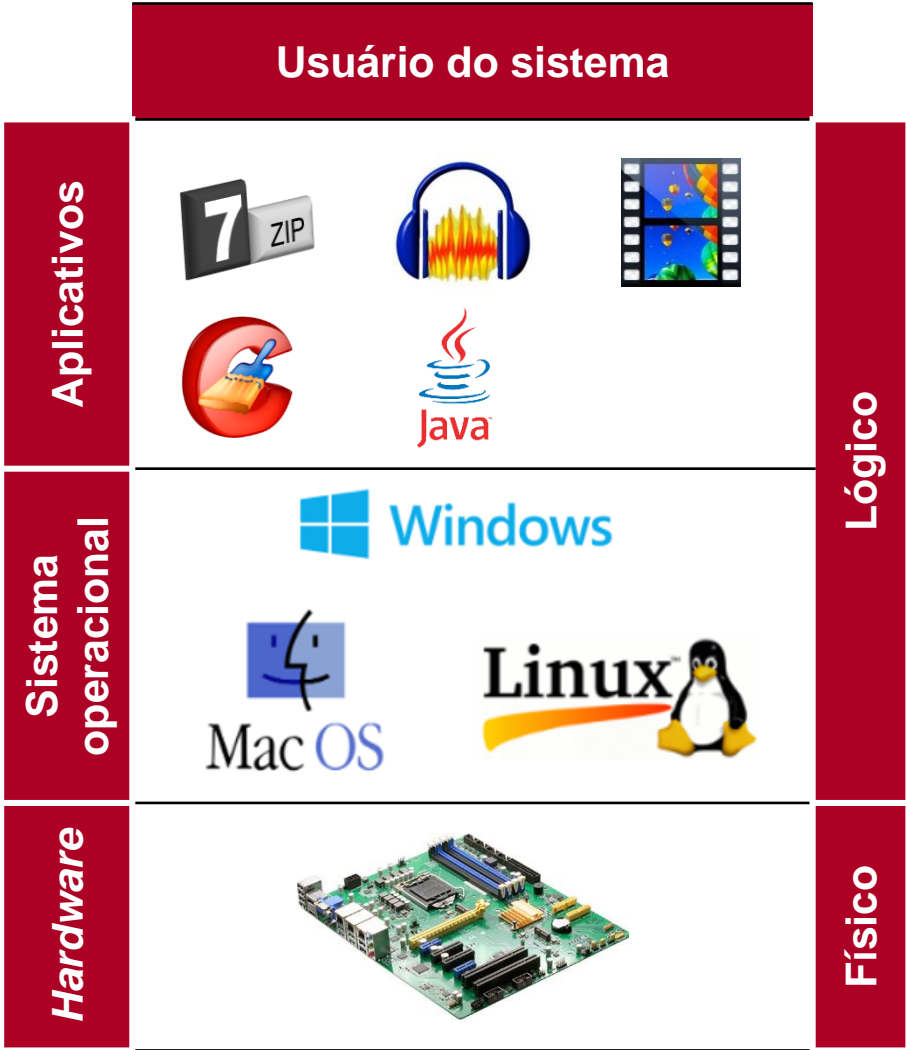
Fonte:

https://images.tcdn.com.br/img/img_prod/167552/180_cpu_intel_celeron_4gb_hd_320gb_win_10_13885_1_0b61554571707c04bc9447f63ac997ef.png

<https://adictoalcodigo.blogspot.com/2016/07/que-es-el-software.html>

<https://www.linuxpro.com.br/2017/06/a-historia-dos-sistemas-operacionais/>

<https://www.iperiusbackup.net/pt-br/breve-introducao-e-importancia-da-placa-mae-no-mundo-da-informatica/>

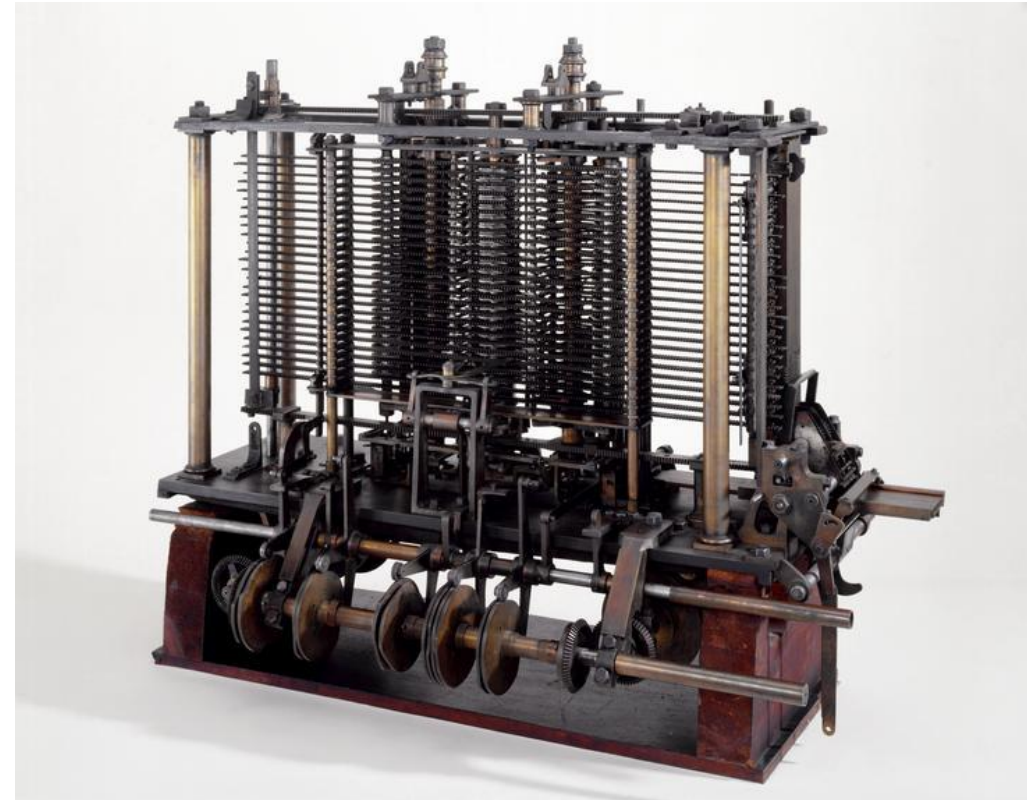


Histórico de sistemas operacionais – Época pré-computadores

- O inglês Charles Babbage projetou o primeiro computador de uso geral. Esta máquina exclusivamente mecânica era conhecida como “a máquina analítica”.
- Em seu projeto, Charles vislumbra partes como rodas, engrenagens e correias de alta precisão que não eram compatíveis com a tecnologia disponível em sua época, portanto, a máquina foi construída na época do inventor.

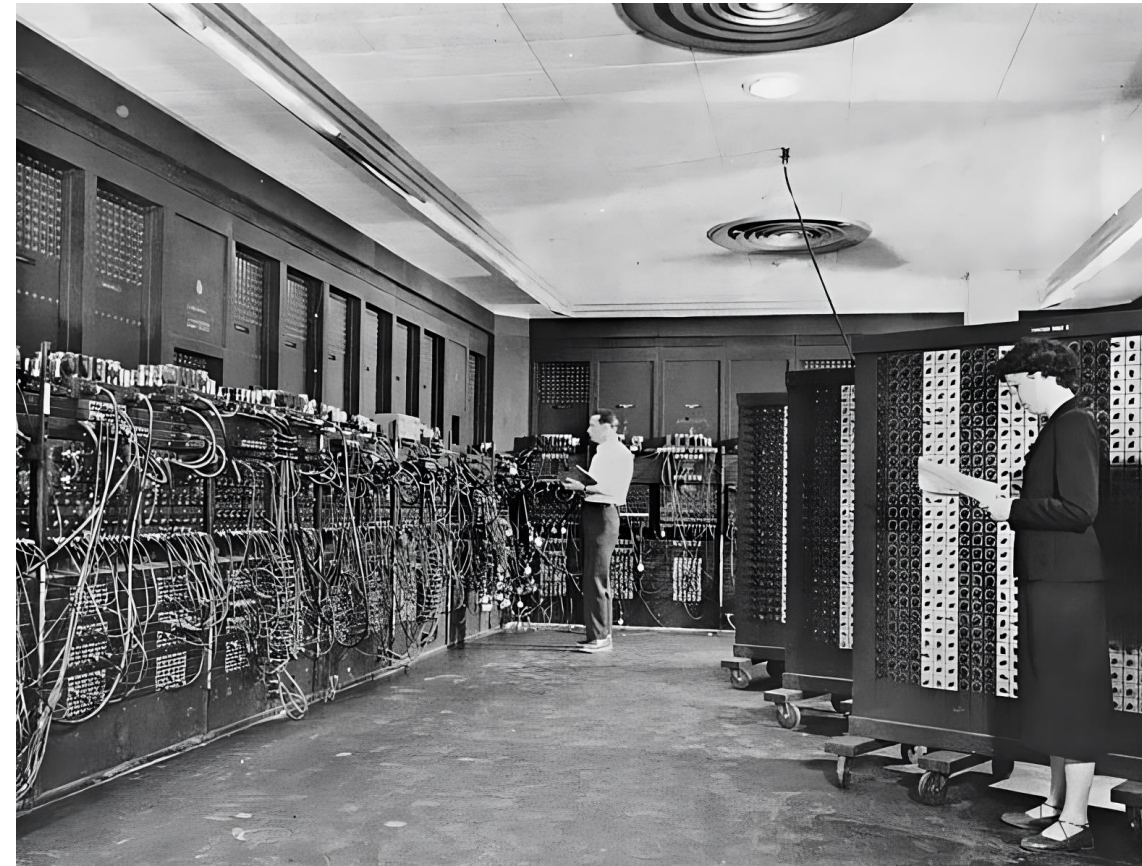
Máquina de Babbage.

Fonte: [https://commons.wikimedia.org/wiki/File:Babbages_Analytical_Engine,_1834-1871._\(9660574685\).jpg](https://commons.wikimedia.org/wiki/File:Babbages_Analytical_Engine,_1834-1871._(9660574685).jpg)



1º geração de computadores (1945-1955) – Válvulas

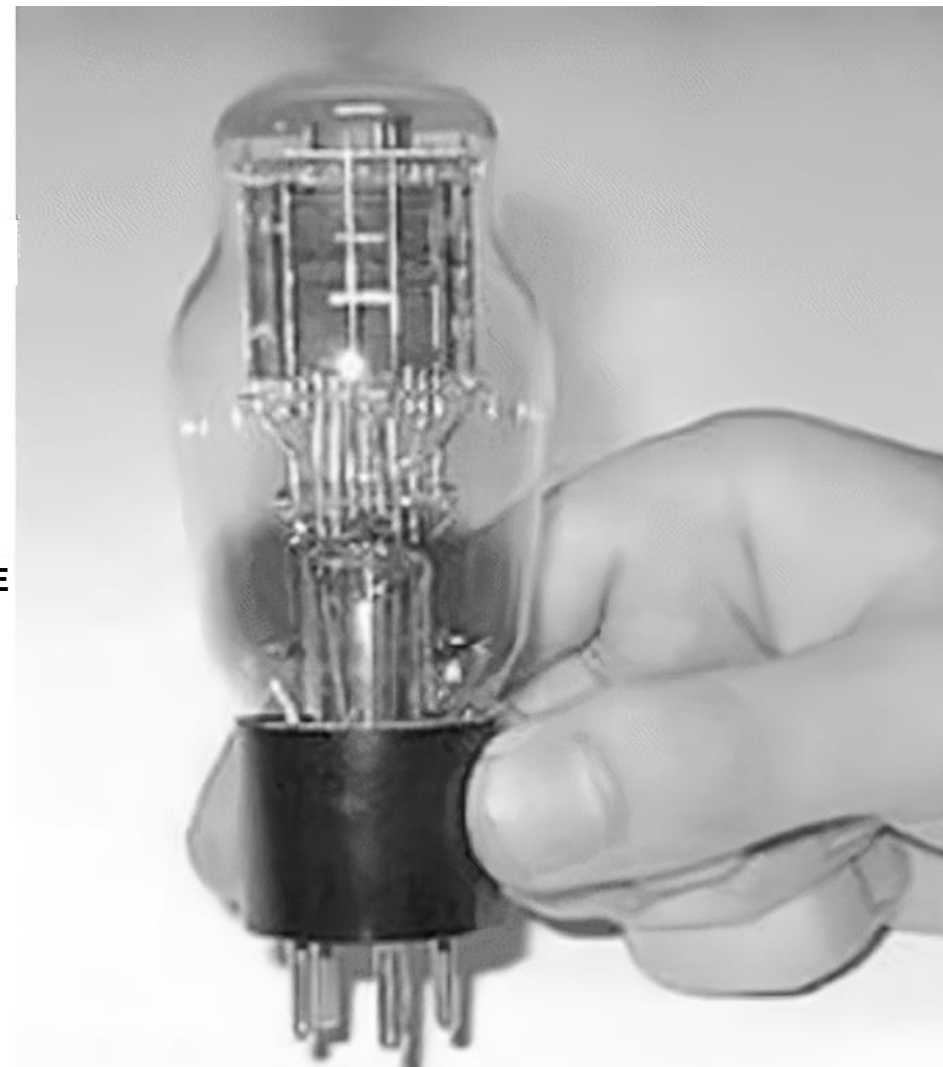
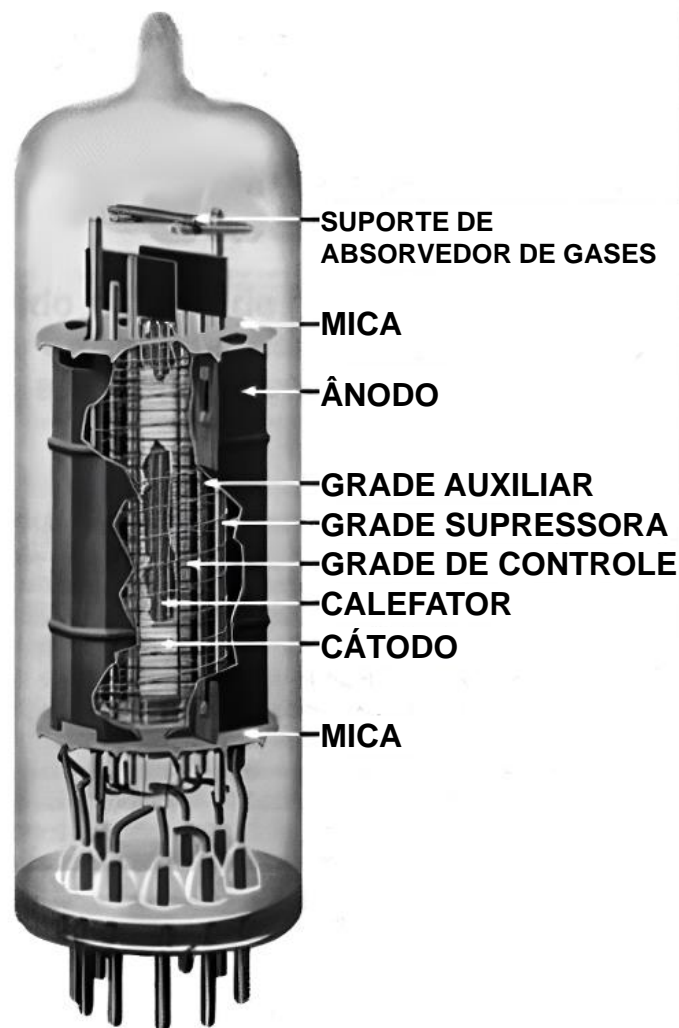
- John William Mauchly e seu aluno John Presper Eckert, em parceria com o governo dos Estados Unidos, construíram, na Universidade da Pensilvânia, o primeiro computador eletrônico, conhecido como ENIAC (*Electronic Numerical Integrator and Computer*).



Fonte:
<https://tecnoblog.net/especiais/eniac-primeiro-computador-do-mundo-completa-65-anos/>

1º geração de computadores (1945-1955) – Válvulas

- Válvulas tinham tamanho semelhante à lâmpada.



1º geração de computadores (1945-1955) – Válvulas

- ENIAC possuía 17 mil válvulas, 10 mil capacitores, 70 mil resistores, pesava 30 toneladas, consumia 140 quilowatts e era capaz de executar 5 mil adições por segundo.
- Seu painel de programação continha 6 mil conectores.
- É considerado o primeiro computador digital e eletrônico.
- Realizou cálculos balísticos e foi utilizado no projeto da bomba de hidrogênio.
 - Cada computador era programado para uma tarefa específica e tinha uma linguagem binária diferente (código máquina).

1º geração de computadores (1945-1955) – Válvulas

- O mesmo grupo de profissionais projetava, construía, programava, operava e realizava a manutenção de cada equipamento.
- Nesta geração, não existia linguagem de programação nem sistema operacional.

A operação era realizada de acordo com os seguintes passos:

- O programador reservava antecipadamente o tempo de máquina.
- O programador encaminhava-se até a sala da máquina, inseria um painel de programação no computador.
- Aguardava horas monitorando esperando que nenhuma das milhares de válvulas queimasse durante a execução.

Interatividade

O sistema operacional deve definir interfaces para os recursos do *hardware*, visando atender aos seguintes objetivos de prover interfaces de acesso aos dispositivos mais simples de usar que as interfaces de baixo nível e tornar os aplicativos independentes do *hardware*.

Essa função do sistema operacional é denominada:

- a) Write-Back.
- b) Abstração.
- c) Concorrência.
- d) Gerência.
- e) Kernel.

Resposta

O sistema operacional deve definir interfaces para os recursos do *hardware*, visando atender aos seguintes objetivos de prover interfaces de acesso aos dispositivos mais simples de usar que as interfaces de baixo nível e tornar os aplicativos independentes do *hardware*.

Essa função do sistema operacional é denominada:

- a) Write-Back.
- b) Abstração.**
- c) Concorrência.
- d) Gerência.
- e) Kernel.

2º geração de computadores (1955-1965) – Transistores

- Em meados dos anos de 1950, foram introduzidos os transistores no mercado computacional, permitindo assim uma verdadeira revolução, elevando a confiabilidade em nível desejado para que pudessem ser fabricados e comercializados.
- Neste período, também houve a criação da organização estruturada por funções, em que havia separação entre projetistas, fabricantes, programadores e técnicos de manutenção.
- Esses computadores denominados computadores de grande porte (*mainframes*) eram instalados em salas especialmente planejadas e operados por equipes de profissionais altamente capacitados para a época.

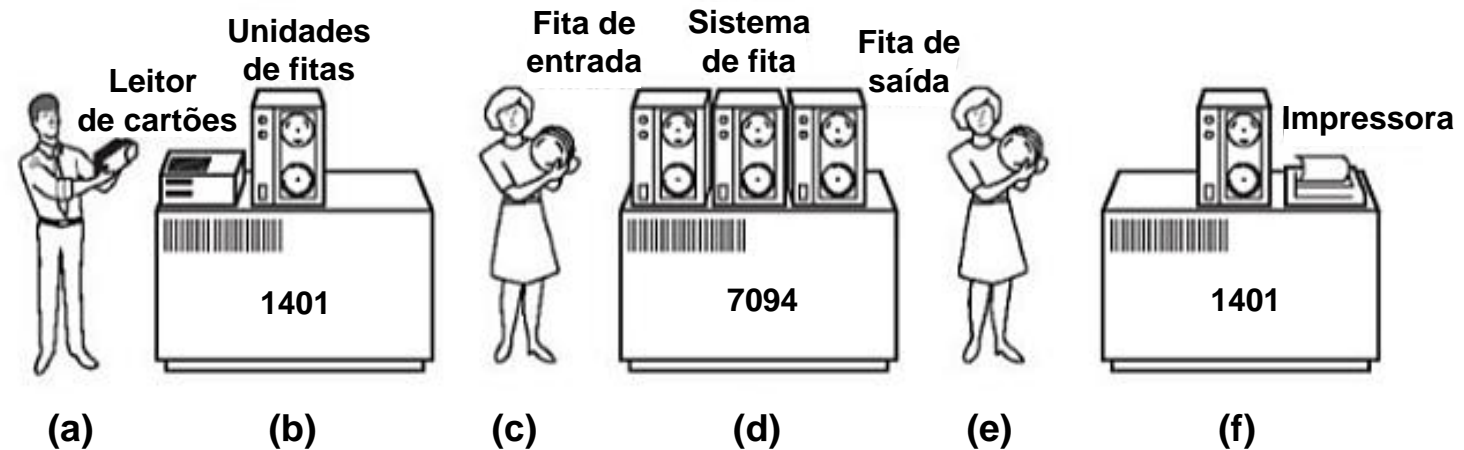
2º geração de computadores (1955-1965) – Transistores

Principais características:

- Transistores.
- 1/200 do tamanho das válvulas.
- Consumo de menos de 1/100 da energia de uma válvula.
- Redução do tamanho das máquinas.
- Processamento ainda era lento, mas já na ordem de milissegundos.

2º geração de computadores (1955-1965) – Transistores

- Houve a organização estruturada por funções, em que havia separação entre projetistas, fabricantes, programadores e técnicos de manutenção. As máquinas desta época ainda permaneciam com tamanhos elevados se comparadas às máquinas dos dias atuais.



- (a) Programadores levavam cartões para o 1401.
- (b) O 1401 lia o lote de tarefas em uma fila.
- (c) O operador levava a fita de entrada para o 7094.
- (d) O 7094 executava o processamento.
- (e) O operador levava a fita de saída para o 1401.
- (f) O 1401 imprimia as saídas.

2º geração de computadores (1955-1965) – Transistores

- IBM 7094 (versão de maior sucesso dessa segunda geração) pesava apenas 890 kg.
- Vendeu mais de 10 mil unidades.



Fonte: IBM.
https://www.ibm.com/ibm/history/exhibits/mainframe/mainframe_PP7094.html

3º geração de computadores (1965-1980) – CIs e multiprogramação

- Em 1964, a IBM lançou a linha de equipamentos da série IBM 360, que foi a primeira linha de computadores a usar circuitos integrados (CIs) em pequena escala, proporcionando melhor custo-benefício em comparação à geração anterior.
- A multiprogramação consistiu no melhor aproveitamento de CPU e memória, repartindo-as e alocando-as para outras tarefas.
- Enquanto uma tarefa estivesse esperando uma operação de E/S (entrada e saída) terminar, a outra tarefa poderia usar a CPU.

3º geração de computadores (1965-1980) – CIs e multiprogramação



Foto do Mainframe
IBM System 360.

Fonte:
https://commons.wikimedia.org/wiki/File:IBM_System360_Mainframe.jpg

Principais características:

- Introdução dos circuitos integrados.
- Grande contribuição no processo de miniaturização.
- Início da produção de minicomputadores.
- Sistema operacional (orientado à arquitetura).
- Tempo compartilhado.
- Programação em *assembly*.
- Criação dos primeiros *softwares*.

4º geração de computadores (1980-1990) – Computadores pessoais

- O início da quarta geração teve como marco o desenvolvimento dos circuitos integrados contendo milhares de transistores em alguns centímetros quadrados de silício, dando origem à era dos computadores pessoais.
- Os computadores eram mais confiáveis, mais rápidos, menores e com maior capacidade de armazenamento. Esta geração é marcada pela venda de computadores pessoais.

4º geração de computadores (1980-1990) – Computadores pessoais

Principais características:

- Introdução dos microprocessadores.
- Desenvolvimento dos computadores.
- Pessoais (*Personal Computer* ou PC).

Escala de integração:

- VLSI: *Very Large Scale Integration*.

4º geração de computadores (1980-1990) – Computadores pessoais

- Esta geração foi marcada pelo surgimento dos processadores, os sistemas operacionais como o MS-DOS, Unix e Macintosh, linguagens de programação orientadas a objetos como o C++ e o *smalltalk*.
- Impressoras matriciais e os teclados com os *layouts* atuais também foram criados nessa época.

```
Starting MS-DOS...
```

```
Microsoft(R) MS-DOS(R) Version 6.22
```

```
(C)Copyright Microsoft Corp 1981-1994.
```

```
A:\>dir
```

```
Volume in drive A has no label
```

```
Volume Serial Number is 0016-2244
```

```
Directory of A:\
```

COMMAND	COM	54,869	08-19-94	12:00p
AUTOEXEC	BAT	1,320	03-28-02	9:39p
CONFIG	SYS	99	08-29-03	3:11p
DRIVERS	<DIR>		08-29-03	4:08p
SYSTEM	<DIR>		08-29-03	4:08p
UTILS	<DIR>		08-29-03	4:08p
6 file(s)			56,288 bytes	
			774,144 bytes free	

```
A:\>_
```

Fonte: Livro-texto.

4º geração de computadores (1980-1990) – Computadores pessoais

- Surgimento das interfaces gráficas (GUI).



Fonte:
https://commons.wikimedia.org/wiki/File:Apple_Macintosh_512K_Desktop.jpg

5º geração de computadores (1990 em diante) – Mobilidade

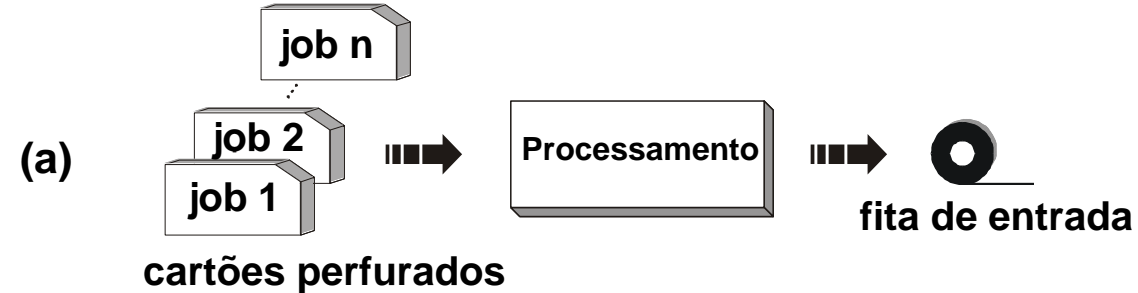
- Expansão dos recursos computacionais para *notebooks*, *netbooks* e também para os celulares.
- Surgimento do *Software Livre* – GNU/Linux.
- Difusão da internet.
- Após 2010: *Cloud Computing*.

Tipos de sistemas operacionais

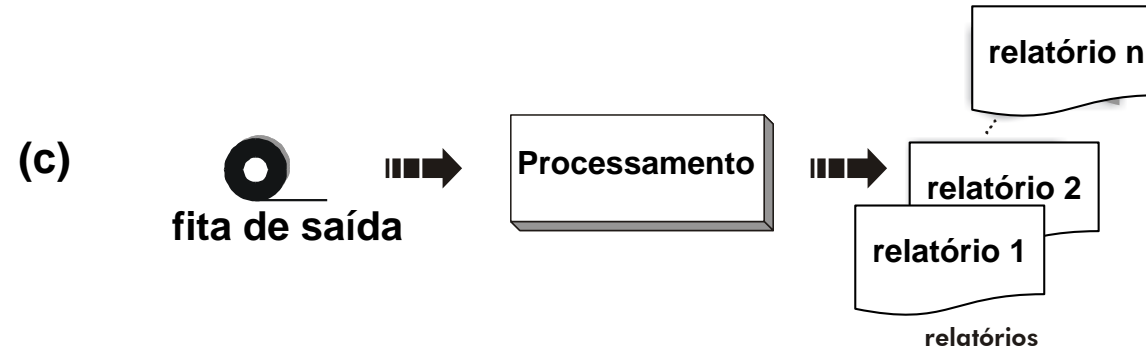
- Podemos classificar os sistemas operacionais com base referencial em diversos parâmetros e perspectivas, velocidade, suporte a recursos específicos, acesso à rede etc.
- Muitos sistemas operacionais estão enquadrados em mais de uma das categorias apresentadas.

Tipos de sistemas operacionais – Sistemas em Batch ou em Lote

- Todos os programas a executar eram colocados em uma fila.
- O processador recebia um programa após o outro, processando-os em sequência, o que permitia um alto grau de utilização do sistema.
- O termo lote ainda é usado para definir um conjunto de comandos que rodam sem interferência do usuário.



Fonte: Autoria própria.



Multiusuário X Monousuário

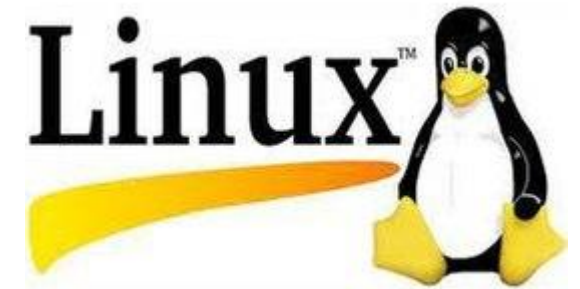
- Múltiplos usuários simultâneos.
 - Maioria dos sistemas operacionais atuais.
- Deve suportar a identificação do “dono” de cada recurso dentro do sistema
 - Arquivos, processos, conexões de rede.
- Imposição de regras de controle de acesso para impedir o uso desses recursos por usuários não autorizados.

Monotarefa X Multitarefa

- Sistema Monotarefa: executa somente 1 tarefa por vez, que será executada até a sua finalização.
- Sistemas Multitarefas ou multiprogramados permitem repartir a utilização do processador entre várias tarefas aparentemente simultâneas.
- Sistema operacional controla qual tarefa será executada e por quanto tempo.

Desktop

- Sistema operacional “de mesa”.
- Usuários domésticos e corporativos
 - Atividades corriqueiras como editar textos, elaborar planilhas e gráficos, navegar na internet.
- Ambiente gráfico, interatividade com usuário e suporte à rede.



Fonte:
<https://olhardigital.com.br/2018/08/13/dicas-e-tutoriais/dicionario-linux-entenda-10-termos-importantes-do-sistema-operacional/>



Fonte: https://commons.wikimedia.org/wiki/File:Windows_10_Logo.svg

Servidor

- Gestão de grandes quantidades de recursos
 - Discos, memórias e processadores.
- Multiusuários.
- Suporte à rede.

Servidor

Têm-se alguns tipos diferentes de servidores de rede:

- **Servidor de arquivos** – distribui/compartilha arquivos em uma rede.
- **Servidor de banco de dados** – consulta e/ou cadastro de dados.
- **Servidor de impressão** – gerencia impressoras/impressões em uma rede.
 - **Servidor de comunicação** – serviços *web*, *e-mail* etc.
 - **Servidor de gerenciamento** – *firewalls*, *proxies*, diretório de usuários.

Embarcados

- Do inglês *Embedded*.
- *Hardwares* com capacidade limitada de processamento
 - Celulares, calculadoras, tocadores de MP3.
- Função específica.

Móveis

- Evolução dos dispositivos como celulares/*smartphones* e *tablets*.
- Sistema Operacional inclui um *kernel* básico e uma outra estrutura de *software* denominada *middleware* com objetivo de fornecer serviços adicionais para desenvolvedores de aplicativos.
- Por exemplo, os dois sistemas operacionais de dispositivos móveis mais utilizados, Android da Google e iOS da Apple, utilizam um *kernel* básico e o *middleware* que possui capacidade de suportar operação em bancos de dados, elementos gráficos e ambientes multimídia.



Fonte: https://commons.wikimedia.org/wiki/File:Android_phones.jpg



Fonte: https://www.flaticon.com/br/icone-gratis/maca_2175370

Tempo real

- “Sistemas operacionais de tempo real são uma subclasse de sistemas operacionais destinados à concepção de sistemas computacionais, geralmente embarcados, em que o tempo de resposta a um evento é fixo e deve ser respeitado sempre que possível” (DENARDIN; BARRIQUELLO, 2019, p. 37).
 - Tempo de resposta conhecido no melhor caso e pior caso de operação.
- *Soft real-time systems* (não críticos)
 - Perda de prazo implica em degradação do serviço prestado (gravação de CD).
- *Hard real-time systems* (críticos)
 - Perda de prazo pode causar grandes prejuízos econômicos ou ambientais (usina nuclear, caldeiras industriais, linha de produção).

Sistema operacional de rede

- Suporte à operação em rede
 - Maioria dos sistemas operacionais atuais.
- Compartilhamento de recursos de vários computadores.
- Disponibilizar os próprios recursos.
- São independentes e, caso a conexão entre um dos nós sofra qualquer problema, os demais continuam operando normalmente, apesar de alguns recursos se tornarem indisponíveis.

Distribuído

- Os recursos de cada máquina estão disponíveis globalmente, de forma transparente aos usuários.
- Para o usuário e suas aplicações é como se não existisse uma rede de computadores, e sim um único sistema centralizado.
 - Usuário desconhece o computador em uso.
- Ainda não são uma realidade de mercado.
- Exemplo: Amoeba.

Interatividade

Ao longo do período estudado, observamos uma evolução no gerenciamento do sistema operacional e sua estrutura técnica.

Selecione o período que pode ser marcado pelos seguintes fatos:

- I. Utilização dos transistores.
 - II. Redução do tamanho das máquinas.
 - III. Processamento na ordem de milissegundos.
-
- a) 1ª geração de computadores.
 - b) 2ª geração de computadores.
 - c) 3ª geração de computadores.
 - d) 4ª geração de computadores.
 - e) 5ª geração de computadores.

Resposta

Ao longo do período estudado, observamos uma evolução no gerenciamento do sistema operacional e sua estrutura técnica.

Selecione o período que pode ser marcado pelos seguintes fatos:

- I. Utilização dos transistores.
 - II. Redução do tamanho das máquinas.
 - III. Processamento na ordem de milissegundos.
-
- a) 1ª geração de computadores.
 - b) 2ª geração de computadores.
 - c) 3ª geração de computadores.
 - d) 4ª geração de computadores.
 - e) 5ª geração de computadores.

Gerência de processador

- Conhecida como gerência de processos ou de atividades.
- Visa distribuir a capacidade de processamento de forma justa.
 - “Justa é diferente de igual”.
- Sincronização de atividades
 - Comunicação entre processos.

Gerência de memória

- Fornecer a cada aplicação um espaço próprio de memória
 - Independente e isolado das demais aplicações.
- Uso do disco como memória complementar
 - Aplicação desconhece o tipo da memória em uso.

Gerência de recursos

- Para cumprir sua função, deve atuar em várias frentes.
- São vários os recursos a serem gerenciados.
 - Processador.
 - Memória.
 - Dispositivos de E/S.
 - Arquivos.

Gerência de dispositivos

- Gerência de Entrada/Saída (E/S) – *Input/Output* (I/O).
- Vários dispositivos diferentes.
 - *Pen drive*, disquetes, discos IDE e SCSI.
- Permitir, por meio de *drivers*, o uso de forma comum
 - Vetor de blocos de dados.

Gerência de arquivos

- Construída sobre a gerência de dispositivos
 - Criando abstrações de arquivos e diretórios.
- Outros dispositivos podem ser utilizados como arquivos
 - Gravar arquivos numa saída TCP (Unix).

Gerência de proteção

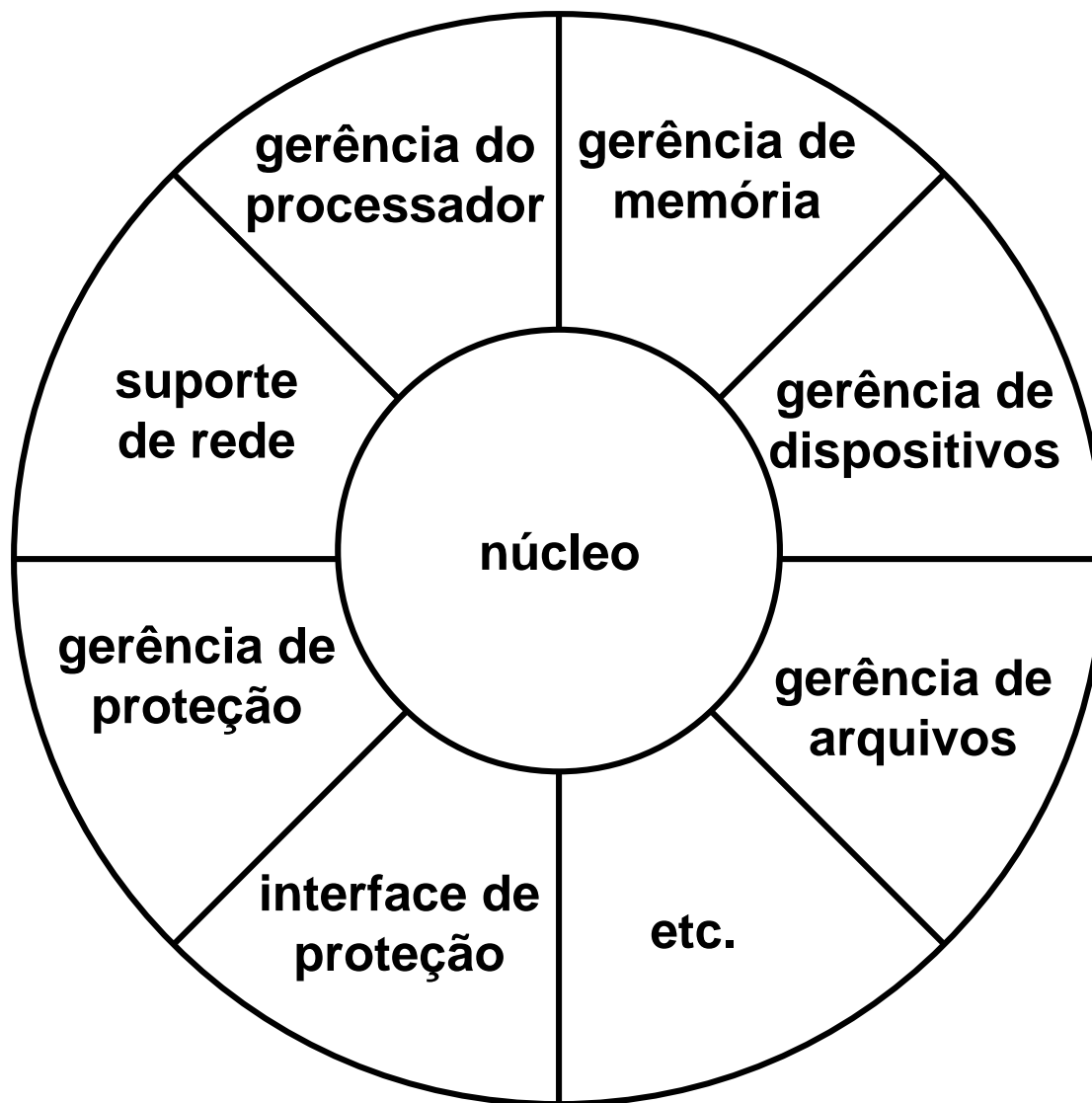
- Políticas de acesso para sistemas
 - Em rede.
 - Multiusuários.
- Definição de usuários, grupos de usuários, registro de recursos por usuários.

Outros tipos de gerência

- Gerência de energia.
- Gerência de rede.
- Gerência de recursos multimídia.
- Gerência da interface.
- Gerência de armazenamento secundário.

Gerência de recursos

- Os módulos de gerência são interdependentes.

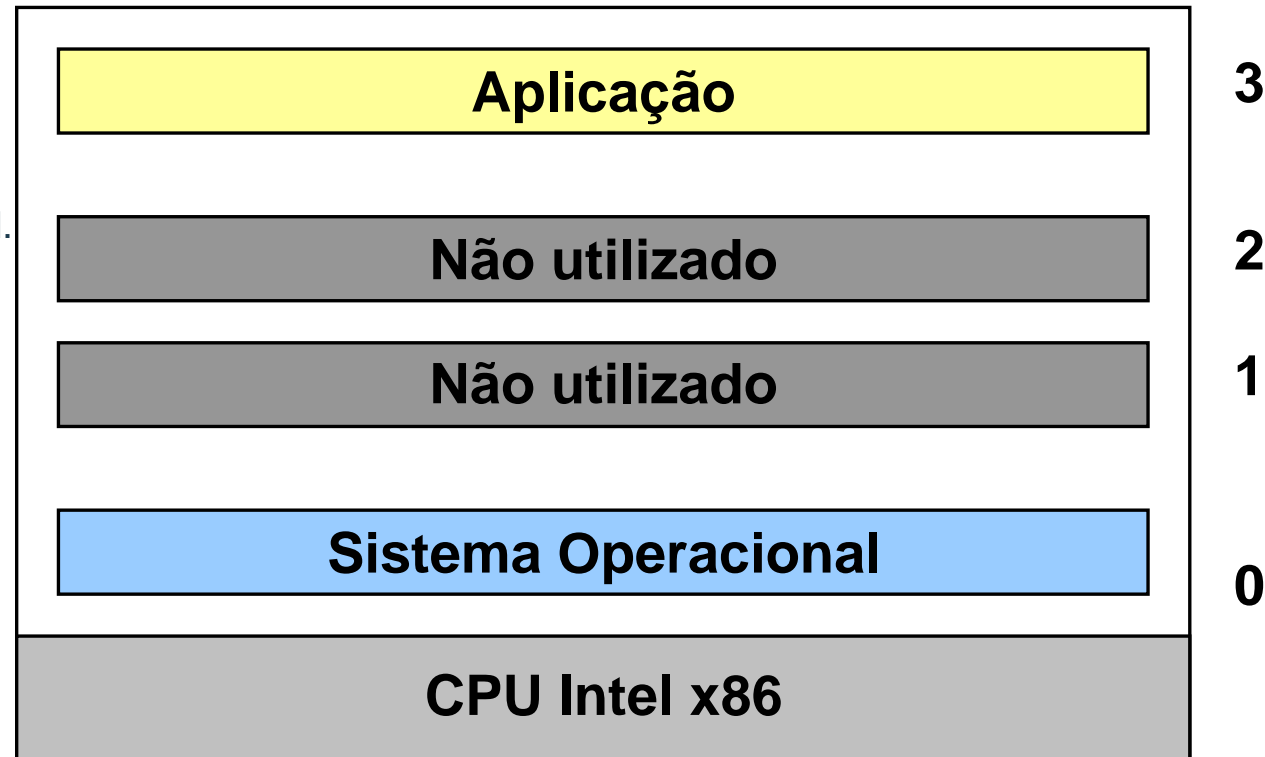


Fonte: Adaptado de: acervo pessoal.

Proteção ao núcleo

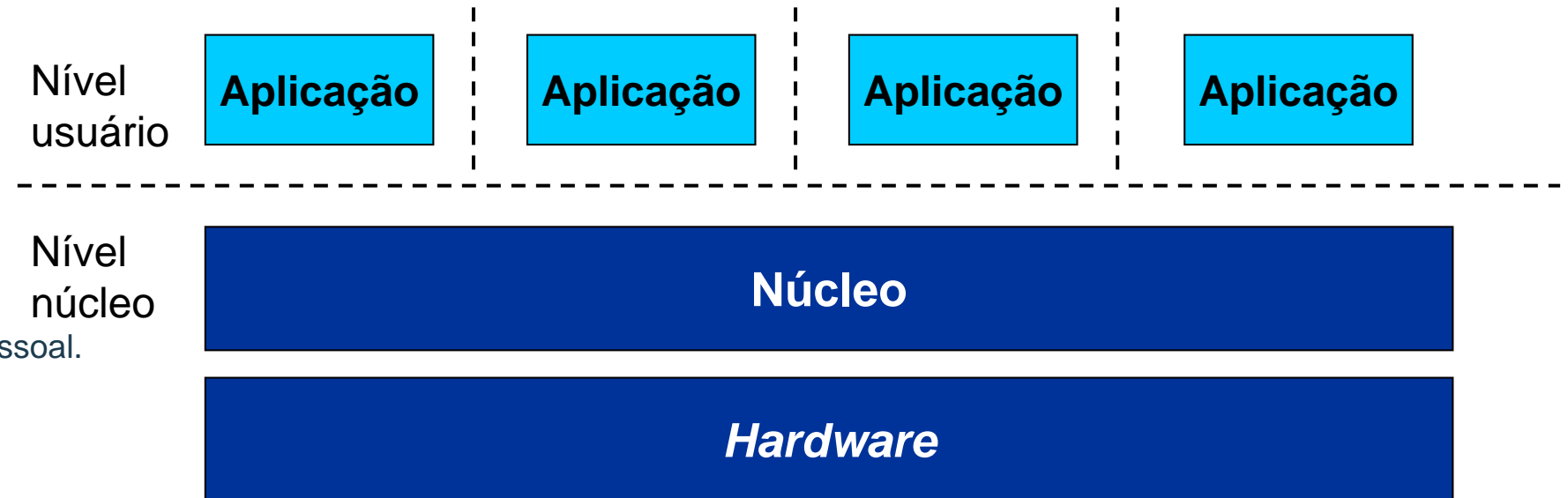
- Certas instruções não podem estar disponíveis para as aplicações, pois isso poderia ocasionar um sério problema de integridade no sistema.
- Por exemplo, um acesso ao disco rígido para gravação de um arquivo.
- Então, existem 2 tipos de instruções: instruções privilegiadas e instruções não privilegiadas.
- O processador implementa 2 modos de acesso: modo usuário e modo *kernel* (ou supervisor).

Fonte: Adaptado de: acervo pessoal.



Separação entre núcleo e aplicações

- O sistema operacional realiza um isolamento entre si e entre as aplicações e o núcleo do sistema operacional.

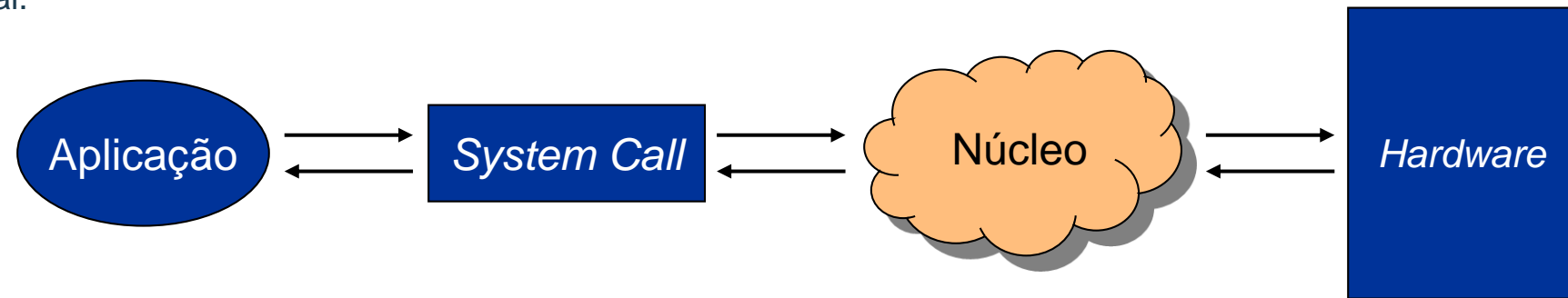


Fonte: Adaptado de: acervo pessoal.

Chamadas de sistema

- São portas de entrada para se ter acesso ao núcleo do sistema operacional.
- Exemplo: quando o usuário deseja algum serviço, realiza uma chamada a uma de suas rotinas pelo *system calls* (chamadas ao sistema).
- Para cada serviço existe um *system call* associado e cada sistema operacional tem o seu próprio conjunto de chamadas.

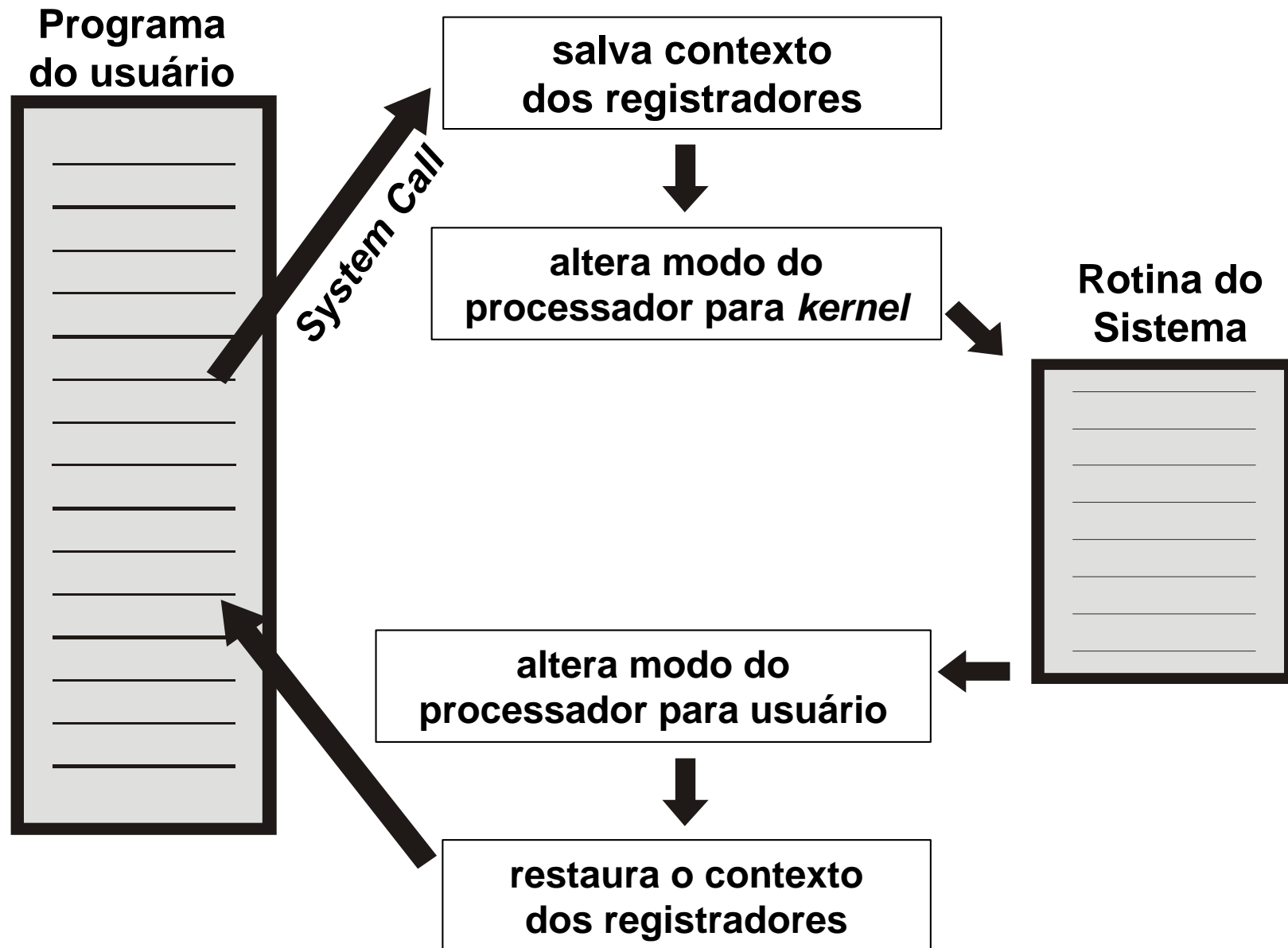
Fonte: Adaptado de: acervo pessoal.



***Syscalls* – grupos de função**

- Gerência de processos
 - Criação e eliminação de processos, alteração das características do processo e sincronização e comunicação entre processos.
- Gerência de memória
 - Alocação e desalocação de memória.
- Gerência de entrada/saída
 - Operações de entrada e saída e manipulação de arquivos e diretórios.

Chamada a uma rotina de acesso



Fonte: Adaptado de: acervo pessoal.

Application Programming Interface

- Conjunto de chamadas de sistema oferecidas por um núcleo é chamada de API
 - API Win32 (Windows)
 - POSIX (Unix/Linux)

Arquitetura de SO

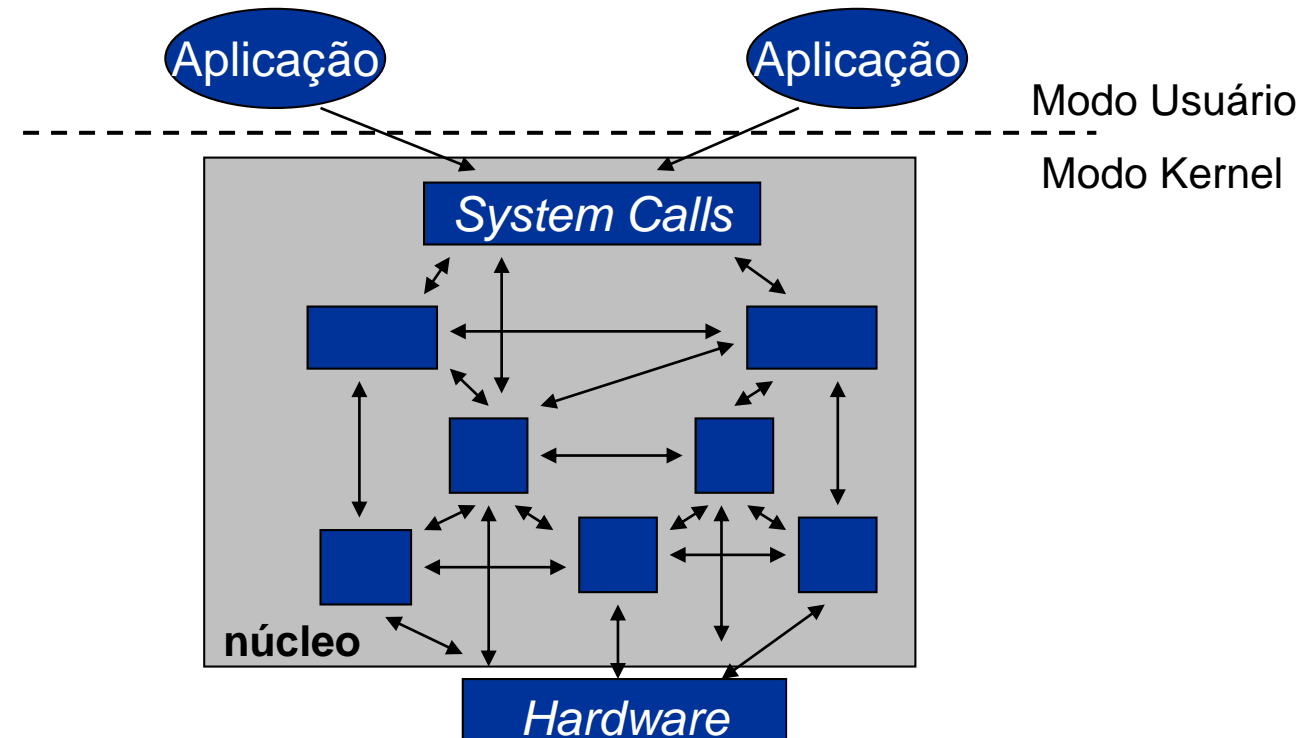
- Sistemas monolíticos
- Sistemas em camadas
- Sistemas micronúcleo
- Máquinas virtuais

Sistema monolítico

- A organização mais comum é aquela que estrutura o sistema como um conjunto de rotinas que podem interagir livremente umas com as outras.
- Pode ser comparada com uma aplicação formada por vários procedimentos que são compilados separadamente e depois *linkados*, formando um grande e único programa executável.
 - Grande desempenho.
 - Uma falha pode paralisar todo o núcleo.

Sistema monolítico

- A organização mais comum é aquela que estrutura o sistema como um conjunto de rotinas que podem interagir livremente umas com as outras.
- Pode ser comparada com uma aplicação formada por vários procedimentos que são compilados separadamente e depois *linkados*, formando um grande e único programa executável.
- Grande desempenho.
- Uma falha pode paralisar todo o núcleo.



Sistema em camadas

- Divide o sistema operacional em sistemas sobrepostos. Cada módulo oferece um conjunto de funções que pode ser usado por outros módulos.
 - No sistema MULTICS VMS, as camadas inferiores são as mais privilegiadas.
- A vantagem da estruturação em camadas é isolar o sistema operacional, facilitando sua alteração e depuração, além de criar uma hierarquia de níveis de modos, protegendo as camadas mais internas.

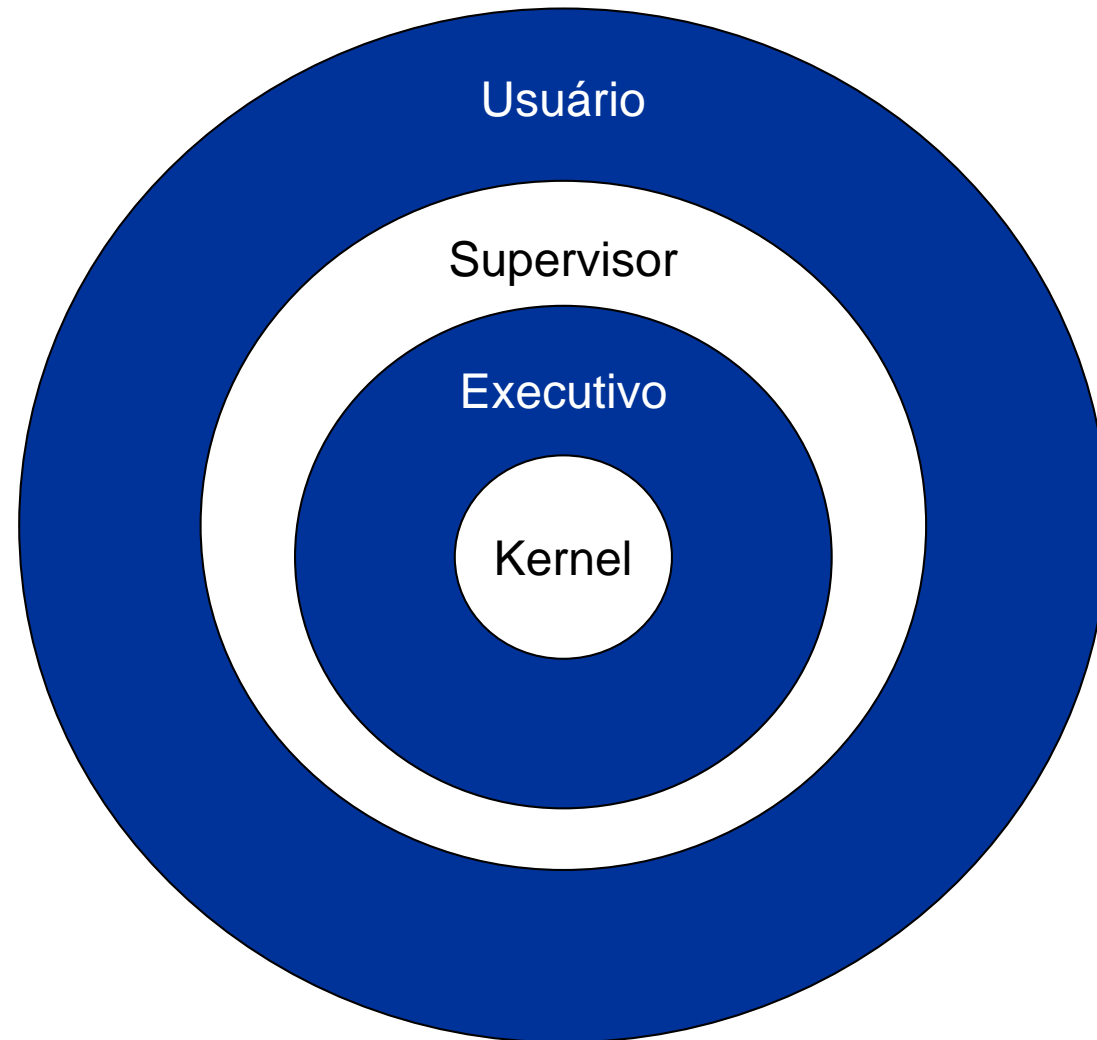
Sistema em camadas

- O empilhamento de várias camadas de *software* faz com que cada pedido de uma aplicação demore mais tempo para chegar até o dispositivo periférico ou recurso a ser acessado, prejudicando o desempenho do sistema.
- Não é óbvio dividir as funcionalidades de um núcleo de sistema operacional em camadas horizontais de abstração crescente, pois essas funcionalidades são interdependentes, embora tratem muitas vezes de recursos distintos.

Sistema em camadas

▪ Sistema Multics

5	Operador
4	Programas de Usuário
3	Entrada/Saída
2	Comunicação
1	Gerência de Memória
0	Multiprogramação



Sistema VMS

Sistemas micronúcleo (*microkernel*)

- Uma tendência dos sistemas operacionais é tornar o núcleo menor e o mais simples possível, e para implementar esta ideia o sistema é dividido em processos.
- Desta forma, sempre que uma aplicação deseja algum serviço ela solicita ao processo responsável; assim, a aplicação que solicita um serviço é chamada de cliente e o processo que responde à solicitação é chamado de servidor.

Sistema micronúcleo

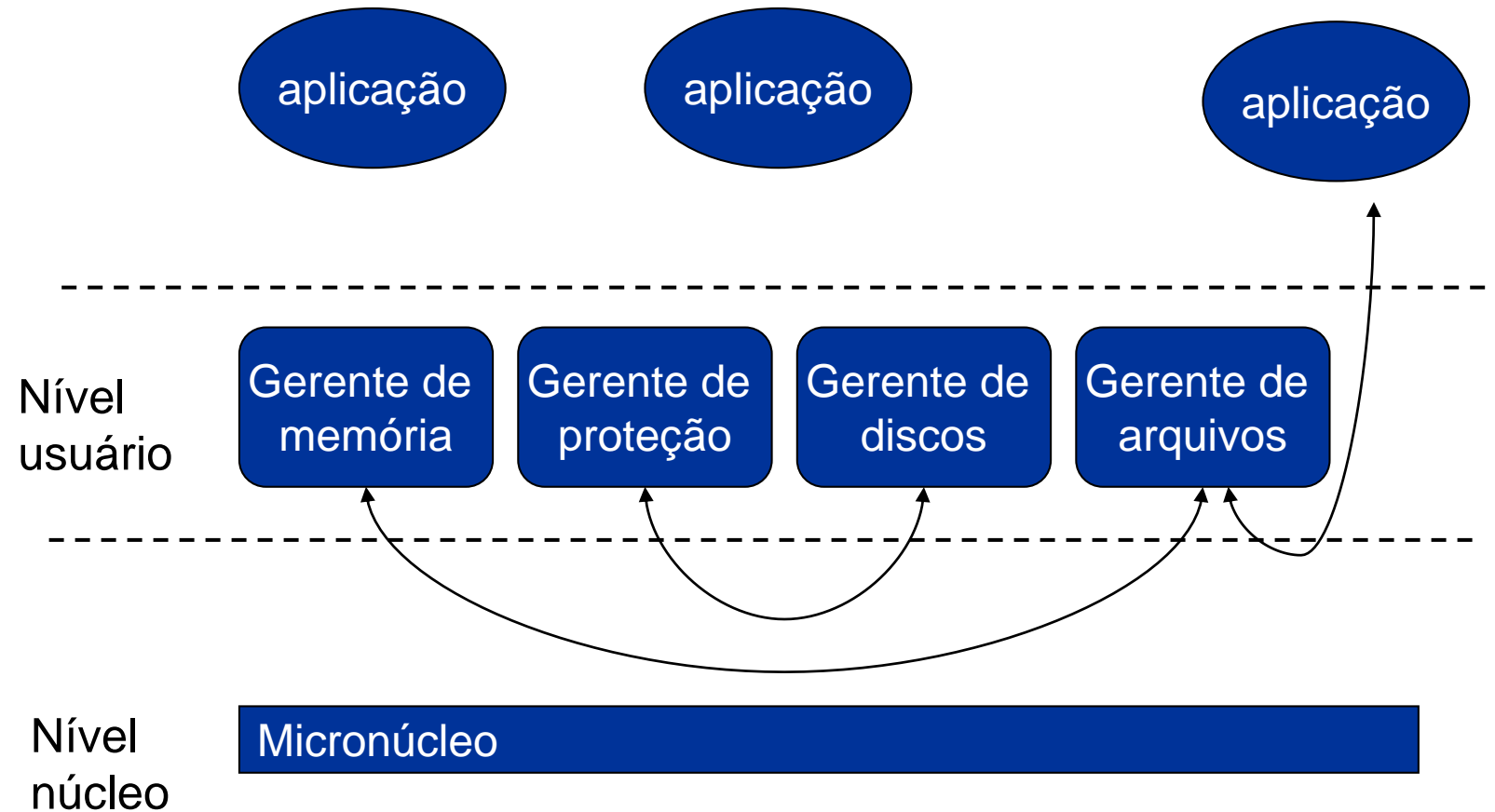
- A utilização deste modelo permite que os servidores executem em modo usuário.
- Apenas o núcleo do sistema, responsável pela comunicação entre clientes e servidores, executa no modo kernel.
- O sistema operacional passa a ser de mais fácil manutenção.
- Não importa se o serviço está sendo processado em um único processador, com múltiplos processadores (fortemente acoplado) ou em sistema distribuído (fracamente acoplado).

Sistema micronúcleo

- Em ambiente distribuído permite que um cliente solicite um serviço e a resposta seja processada remotamente.
- Sua implementação é difícil e mais usualmente é implantada uma combinação do modelo de camadas com o cliente-servidor.
- O núcleo do sistema passa a incorporar o escalonamento e gerência de memória, além das funções de *device drivers*.

Sistema micronúcleo

- Estrutura do sistema micronúcleo



Interatividade

O modo clássico de encarar um sistema operacional é como um gerenciador de recursos. No caso de um programa com defeito (ou malicioso), escrever no espaço de memória do outro programa, o segundo programa travará, na melhor das hipóteses, ou produzirá resultados incorretos, na pior das hipóteses. Ou ainda, se o programa ofensivo modificar a memória do sistema operacional poderá afetar o comportamento de todo o sistema. A atividade em que um sistema operacional garante essa proteção é:

- a) Programa antivírus.
- b) Gerenciamento do processador.
- c) Gerenciamento de memória.
- d) Gerenciamento de arquivos.
- e) Gerenciamento de entrada/saída.

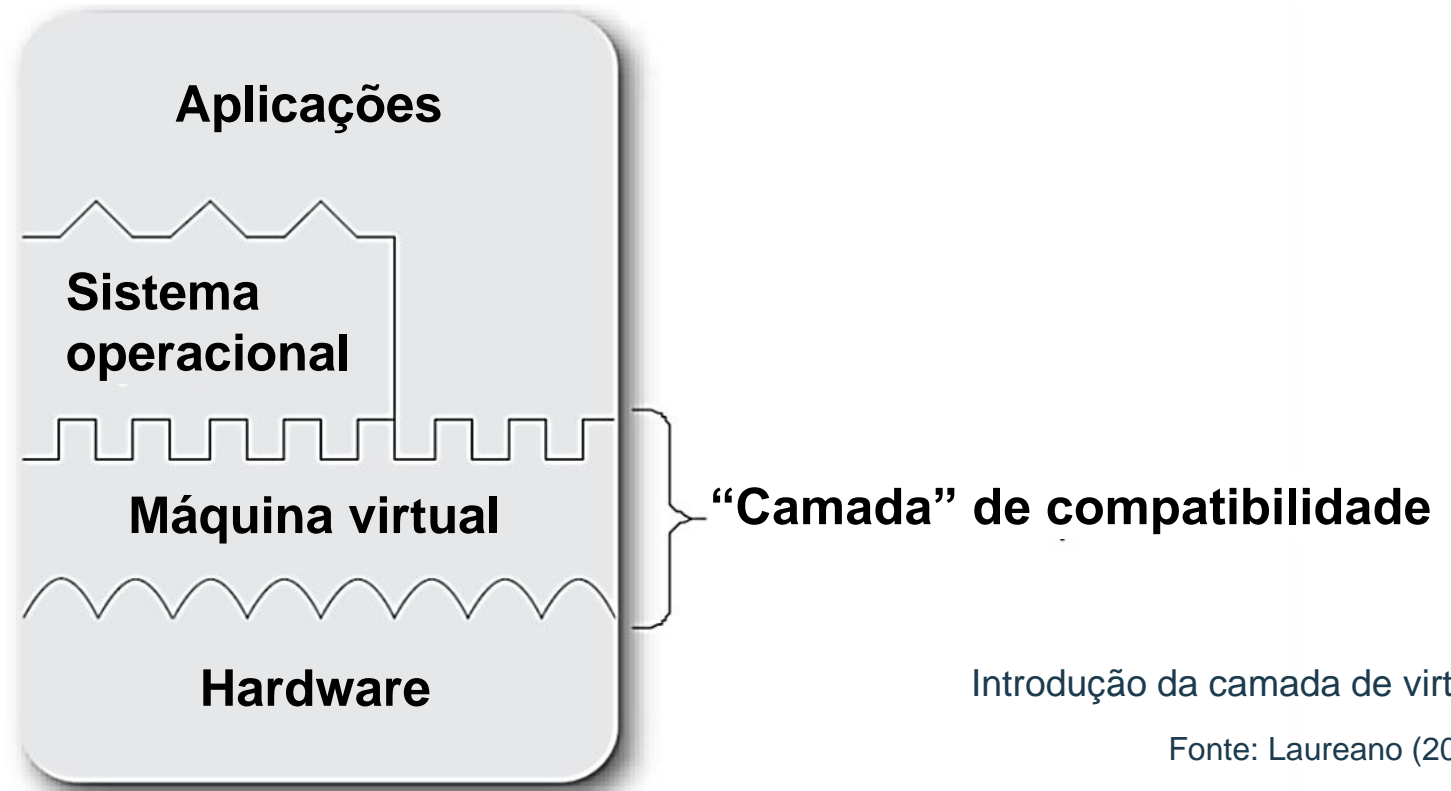
Resposta

O modo clássico de encarar um sistema operacional é como um gerenciador de recursos. No caso de um programa com defeito (ou malicioso), escrever no espaço de memória do outro programa, o segundo programa travará, na melhor das hipóteses, ou produzirá resultados incorretos, na pior das hipóteses. Ou ainda, se o programa ofensivo modificar a memória do sistema operacional poderá afetar o comportamento de todo o sistema. A atividade em que um sistema operacional garante essa proteção é:

- a) Programa antivírus.
- b) Gerenciamento do processador.
- c) Gerenciamento de memória.
- d) Gerenciamento de arquivos.
- e) Gerenciamento de entrada/saída.

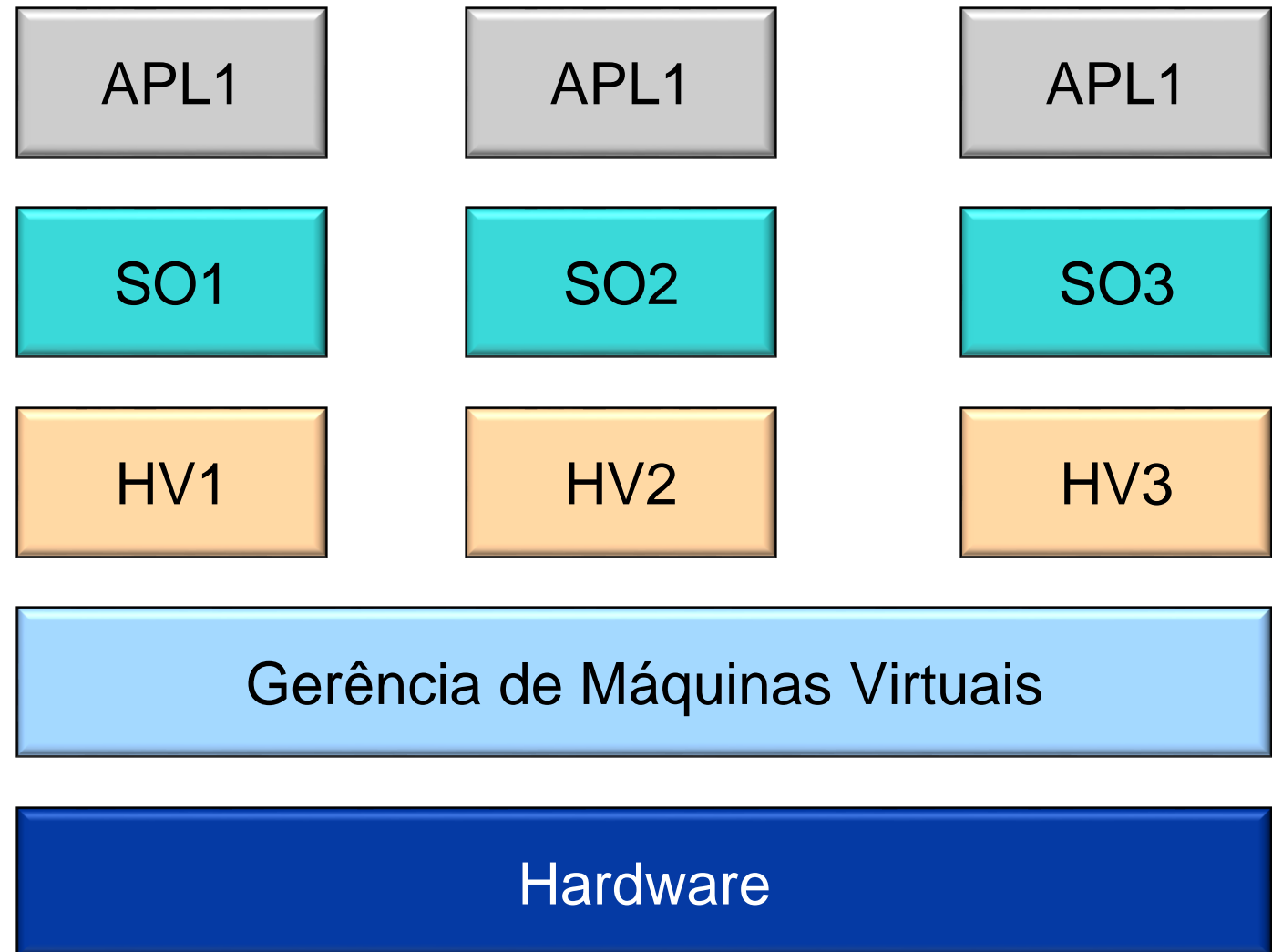
Máquina virtual

- Um sistema operacional só funciona sobre o *hardware* para o qual foi construído.
- É possível contornar os problemas de compatibilidade através de técnicas de virtualização.
- Constrói-se uma camada de *software* que ofereça aos demais componentes serviços com outra interface, essa camada.
- O Sistema computacional visto através dessa camada é chamado Máquina Virtual.



Máquina virtual

- Estrutura em camadas de máquina virtual



Processos

- Na década de 60, surgiu o conceito de SO multiprogramáveis e *Time Sharing*, em que se atendem diversas tarefas dos usuários e se mantêm informações a respeito de vários programas que estão sendo executados concorrentemente.
- Neste sistema, o processador executa a tarefa durante um *Time Slice* e, no instante seguinte, pode processar outra tarefa.
- A cada troca de tarefa, é necessário salvar todas as informações da tarefa que foi interrompida para que, quando ela retornar a ser executada, não lhe falte nenhuma informação para continuar seu processamento.

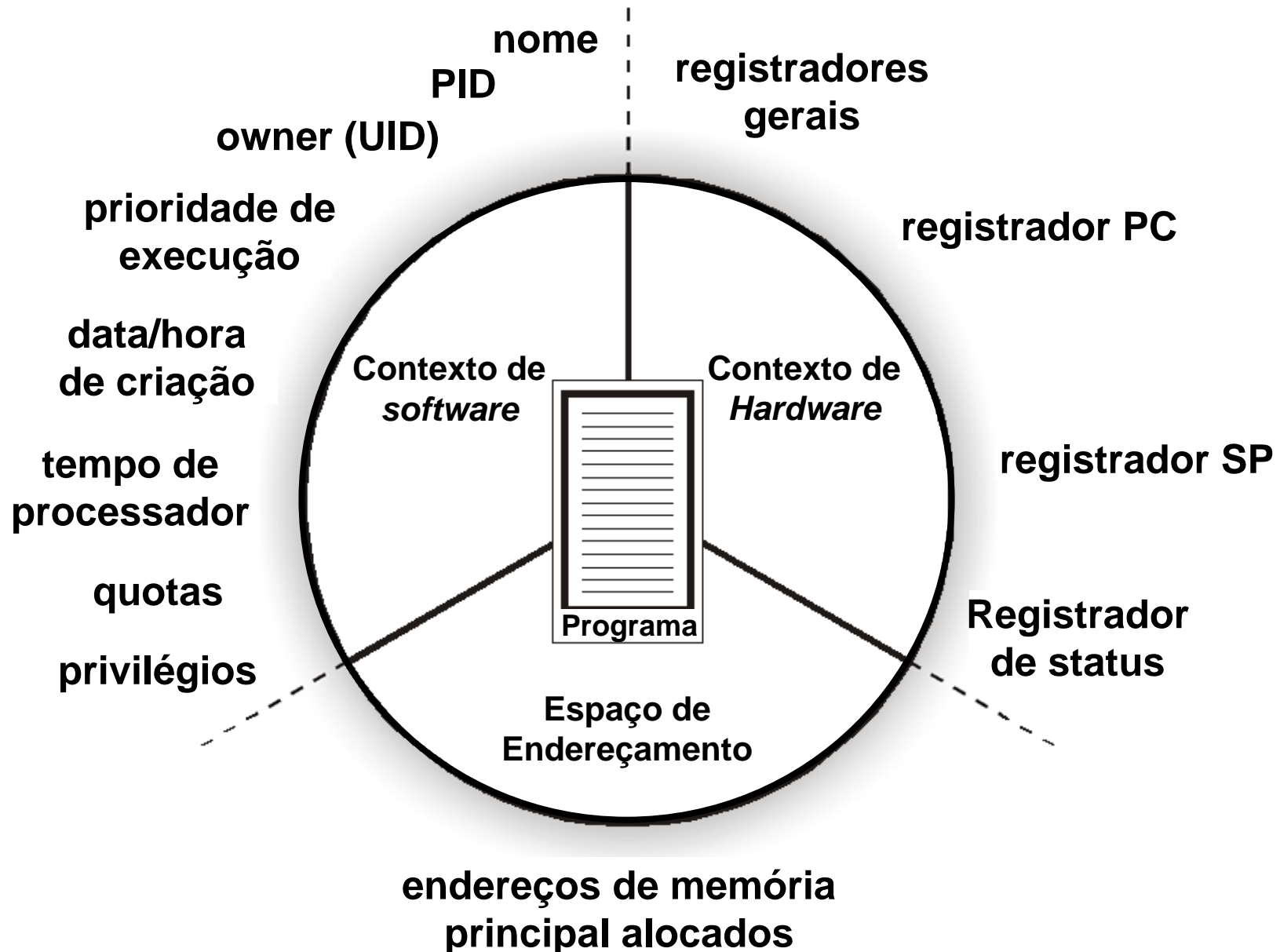
Processos

- O Processo mantém todas essas informações para execução de um programa, como por exemplo, conteúdo de registradores e espaço de memória.
- Processo também é o ambiente em que se executa um programa.
- Nenhum programa é executado diretamente na Memória principal e sim dentro de um processo, pois se não o programa faria uso indiscriminado de qualquer área da Memória Principal, efetuando operações de E/S indevidas comprometendo a integridade e a consistência dos dados.

Estrutura de um processo

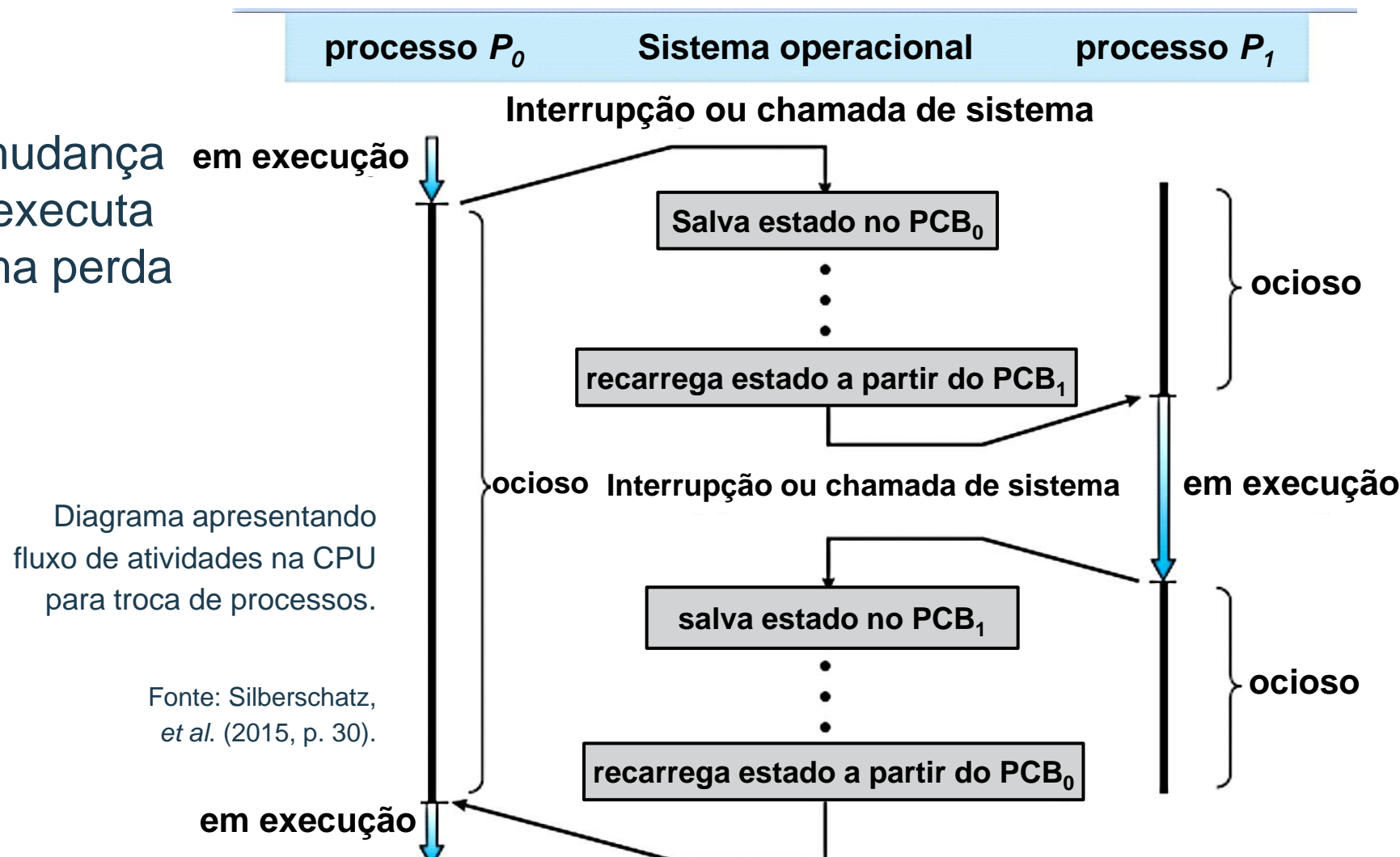
Composto por:

- Programa
- Contexto de *hardware*
- Contexto de *software*
- Espaço de endereçamento



Mudança de contexto

- A Mudança de Contexto realiza a alocação da CPU a outro processo e requer a execução do armazenamento do estado do processo atual e a restauração do estado de um processo diferente.
- Durante o tempo gasto na mudança de contexto, o sistema não executa trabalho útil e representa uma perda no processamento.



Criação de processos

- Processos são criados e destruídos constantemente nos sistemas.
- Essas operações disponibilizam aplicações por meio de chamadas de sistema que diferem entre sistemas operacionais.

Existem quatro eventos que fazem com que processos sejam criados no sistema operacional:

- na inicialização do sistema;
- um processo em execução procedendo a uma chamada de sistema de criação de um processo;
- requisição do usuário para criar um novo processo; e
- gerenciar a tarefa em lote (*batch job*) que está sendo iniciada.

Término de processos

Após o término, o processo é finalizado com base nas quatro condições típicas:

- normal, por erro, erro fatal e cancelado por terceiros – sendo as duas primeiras voluntárias e as duas últimas involuntárias.
- Processos terminados de forma involuntária não são comuns num sistema em perfeito funcionamento.
- O primeiro caso, que é a condição normal de se encerrar um processo, é verificado pela chamada *exit* no *Unix* ou *ExitProcess* no *Windows*. Nesses casos, o processo termina após finalizar as tarefas que estavam previstas, mesmo que seja um usuário finalizando um programa, fechando a janela no ambiente GUI ou pela opção relativa no ambiente *Shell*.

Término de processos

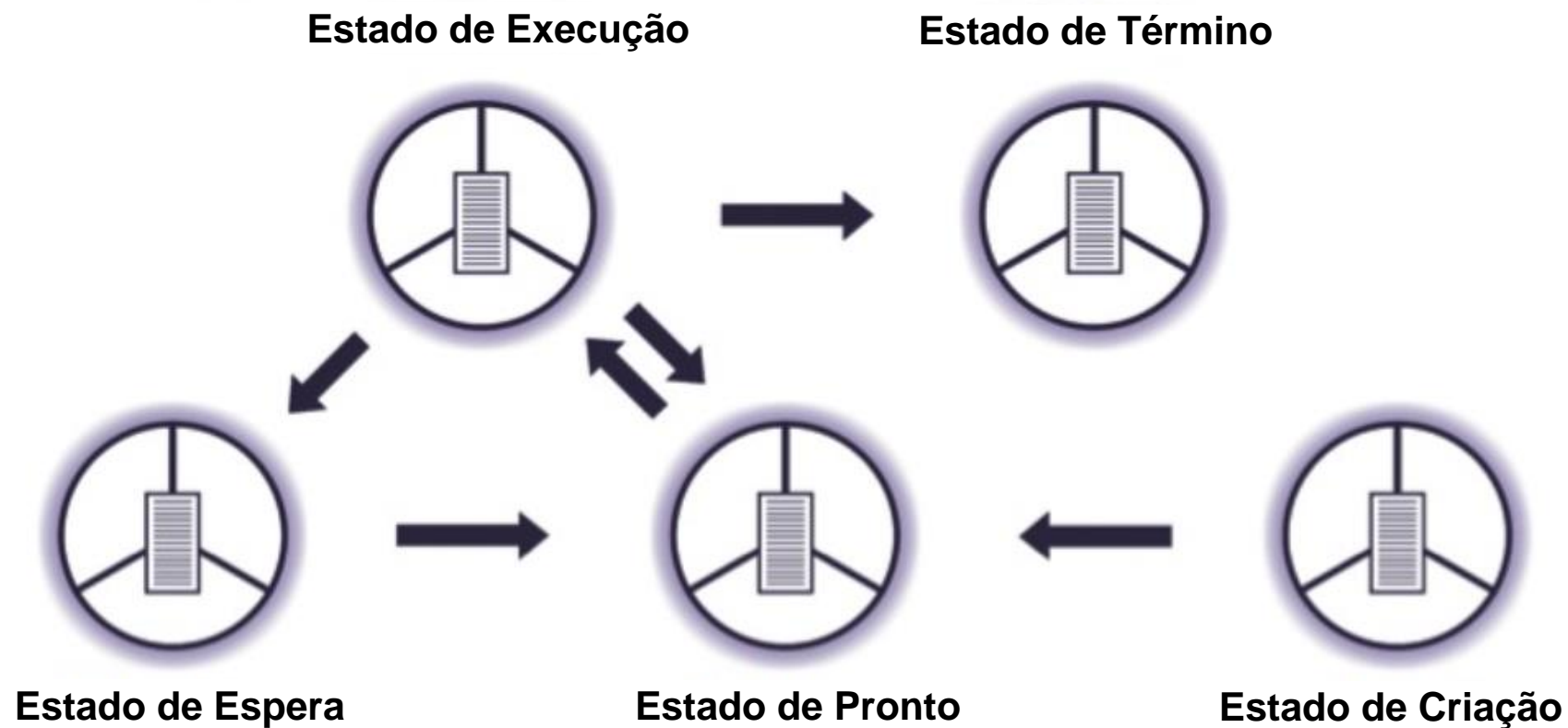
- Num ambiente *Unix*, a chamada de sistema *exit* serve para informar ao núcleo do sistema operacional que o processo em questão não é mais necessário e pode ser eliminado, liberando todos os recursos a ele empregados.
- Processos podem solicitar ao núcleo o encerramento de outros processos, mas essa operação só é aplicável a processos do mesmo usuário ou se o processo solicitante pertencer ao administrador do sistema.
- Os processos que interagem com outros não podem ser concluídos quando algum parâmetro errado é fornecido.
 - Vamos considerar o caso de um usuário tentando colocar o nome duplicado entre dois arquivos no sistema, então uma caixa de diálogo emerge e pergunta ao usuário se ele quer tentar novamente; desta forma, teremos por consequência a segunda condição que é a saída por erro.

Término de processos

- Erro fatal é um erro causado pelo processo e, normalmente, por um erro de programa. Como exemplo podemos ter a execução de uma instrução ilegal, a referência à memória inexistente ou a divisão por zero; em todos esses casos, teremos como resultado um erro fatal.
- O cancelamento por outro processo ocorre quando um processo x executa uma chamada de sistema determinando que o sistema operacional cancele outro(s) processo(s) n.
- Tanto no *Unix/Linux*, a chamada é o *kill* e, no ambiente Windows, a função *Win32* correspondente é a *TerminateProcess*.

Mudanças de estado de processos

- Um processo sofre alteração do seu estado em função de eventos voluntários, originados por ele mesmo, ou eventos involuntários, originados pelo sistema operacional.



Estados de um processo

Estado de execução: Processo que está sendo executado pela CPU.

Estado de pronto (*ready*):

- Processo aguardando para ser executado;
 - Geralmente organizado em listas encadeadas;
 - Escalonamento da fila (lista) a critério do SO.
-
- Estado de espera (*wait* ou *blocked*).
 - Processo que aguarda algum evento externo ou liberação de recurso (ex.: operação de E/S, relógio).

Interatividade

Em relação à Arquitetura de Sistemas Operacionais, assinale a alternativa correta.

- a) No sistema monolítico, garante-se a robustez do sistema, pois todos os recursos estão em uma única camada.
- b) Em um sistema monolítico, caso um serviço falhe, somente ele será afetado, devido ao confinamento de memória entre os serviços.
- c) A divisão do SO em camadas é um problema difícil, pois muitas dessas funcionalidades da gerência são interdependentes.
 - d) A implementação de camadas contribui para melhoria do desempenho do sistema operacional.
 - e) A abordagem micronúcleo não oferece maior modularidade no desenvolvimento, pois cada serviço desenvolvido depende dos demais.

Resposta

Em relação à Arquitetura de Sistemas Operacionais, assinale a alternativa correta.

- a) No sistema monolítico, garante-se a robustez do sistema, pois todos os recursos estão em uma única camada.
- b) Em um sistema monolítico, caso um serviço falhe, somente ele será afetado, devido ao confinamento de memória entre os serviços.
- c) A divisão do SO em camadas é um problema difícil, pois muitas dessas funcionalidades da gerência são interdependentes.
- d) A implementação de camadas contribui para melhoria do desempenho do sistema operacional.
- e) A abordagem micronúcleo não oferece maior modularidade no desenvolvimento, pois cada serviço desenvolvido depende dos demais.

Referências

- DENARDIN, G. W.; BARIQUELLO, C. H. *Sistemas operacionais de tempo real e sua aplicação em sistemas embarcados*. São Paulo: Editora Blucher, 2019.
- FURUKAWA, F.; NUNES, R. *Fundamentos de sistemas operacionais*. São Paulo: Sol, 2011.
- MACHADO, F. B.; MAIA, L. P. *Arquitetura de sistemas operacionais*. 5. ed. Rio de Janeiro: LTC, 2013.
- MAZIERO, C. A. *Sistemas operacionais: conceitos e mecanismos [recurso eletrônico]*. Curitiba: DINF - UFPR, 2019.
 - SILBERSCHATZ, A.; GALVIN, P. B.; GAGNE, G. *Fundamentos de sistemas operacionais*. 9. ed. Rio de Janeiro: LTC, 2015.
 - TANENBAUM, S. A.; WOODHULL, S. *Sistemas operacionais: projetos e implementação - o livro do Minix*. 3. ed. Porto Alegre: Bookman, 2008.

ATÉ A PRÓXIMA!