



# Sistemas Operacionais

Prof. Marcos Ribeiro Quinet de Andrade  
Instituto de Ciência e Tecnologia - ICT  
Universidade Federal Fluminense - UFF

# Bibliografia

## **Bibliografia Básica:**

- TANENBAUM, A. S. **Sistemas Operacionais Modernos**. São Paulo : Ed. Pearson, 2004 (2ª edição), 2009 (3ª edição).
- SILBERSCHATZ, A., GAGNE, G., GALVIN, P. B. **Fundamentos de Sistemas Operacionais**. Rio de Janeiro: Ed. LCT, 2015 (9ª edição)

## **Bibliografia Complementar:**

- DEITEL, H. M., DEITEL, P.J., CHOFINES, D.R. **Sistemas Operacionais**. São Paulo : Pearson Prenticce-Hall, 2005 (3ª edição).
- TANENBAUM, A. S., WOODHULL. **Sistemas Operacionais: Projeto e Implementação**. 2a. ed.. Porto Alegre : Bookman, 2000.

## **Material das aulas:**

<http://www.professores.uff.br/mquinet>

# Introdução

- **Sistema computacional:** conjunto de recursos computacionais, parte *hardware* e parte *software*
- Essencialmente, um sistema computacional consiste em:
  - Recursos computacionais físicos (*hardware*);
  - Programas do sistema;
  - Programas de aplicação.

# Por que ?

- Sistemas computacionais modernos são compostos por diversos elementos:

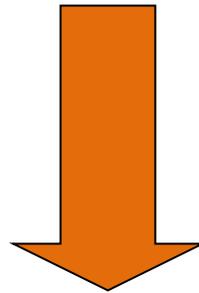
- Processadores;
- Memória;
- Controladoras;
- Monitor;
- Teclado;
- Mouse;
- Impressoras;
- Etc.



**Alta  
Complexidade**

# Por que ?

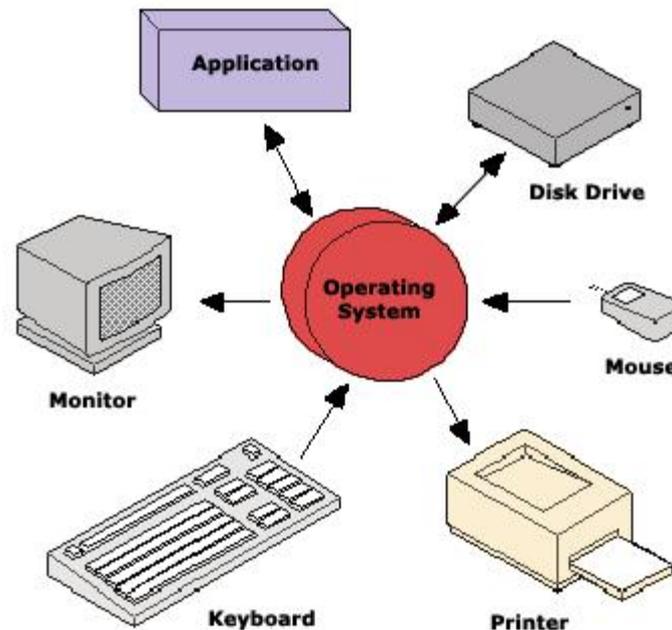
- Com tantos dispositivos, surge a necessidade de gerenciamento e manipulação desses diversos dispositivos
  - Tarefa difícil



**SISTEMAS OPERACIONAIS**

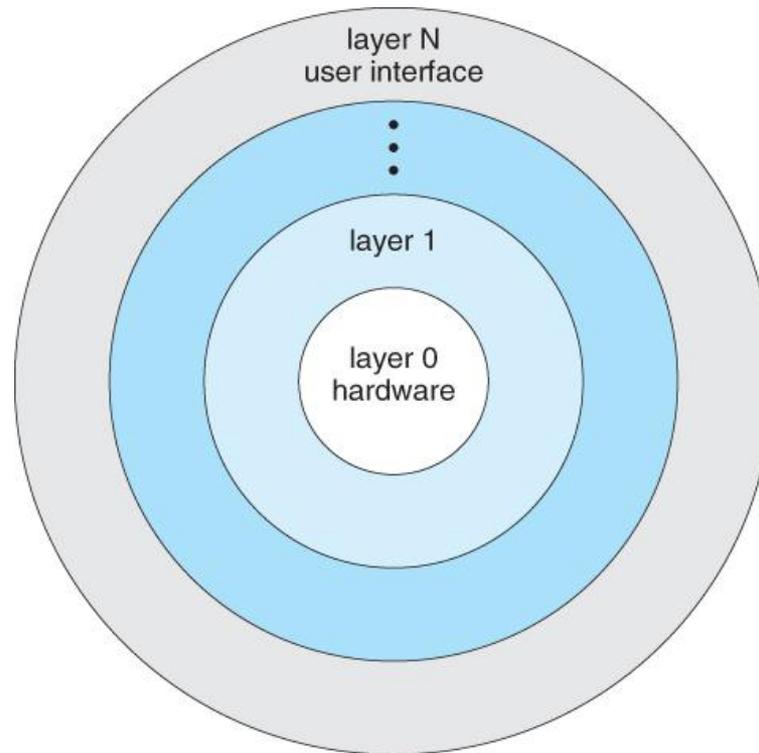
# O que é um Sistema Operacional?

- **Sistema Operacional:** *software* que controla os recursos do sistema computacional e oferece ao usuário uma interface para interagir com cada um destes recursos

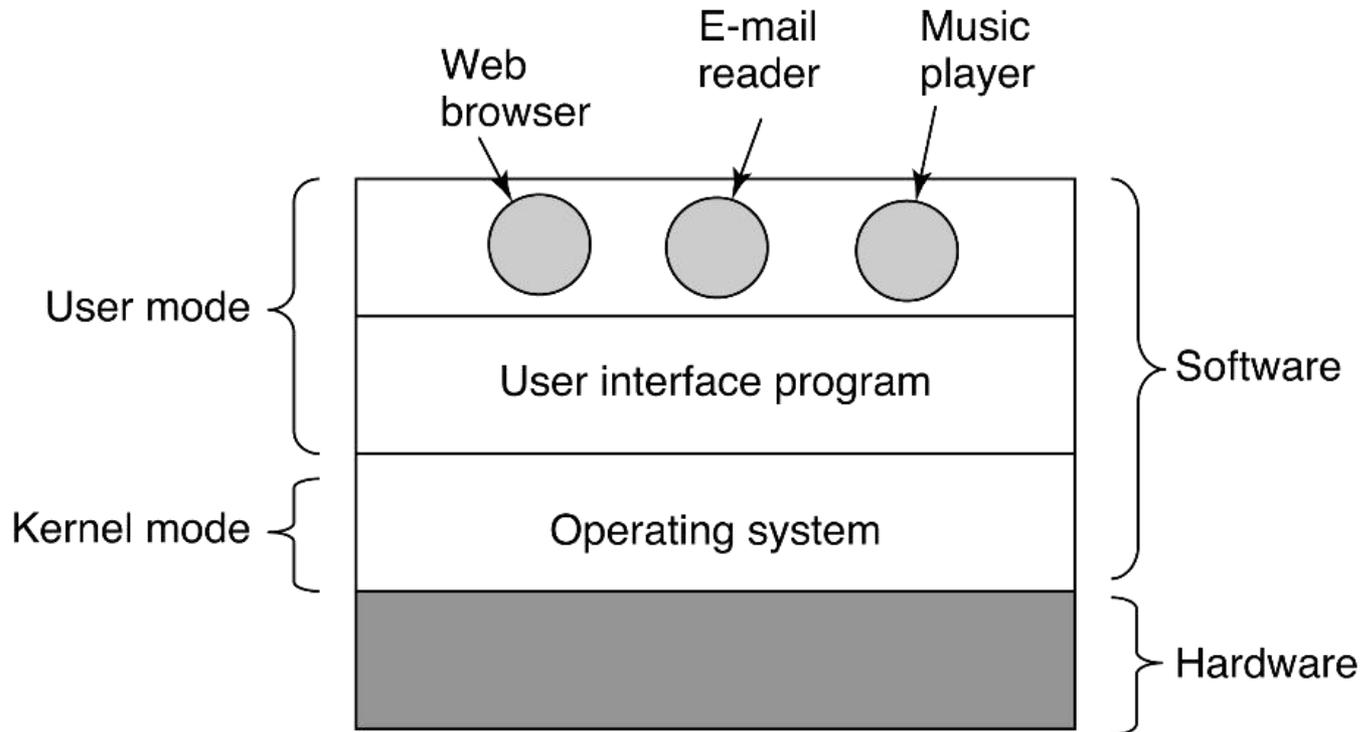


# O que é um Sistema Operacional?

Um sistema operacional típico é formado por múltiplas camadas, cada uma projetada para atender uma tarefa específica

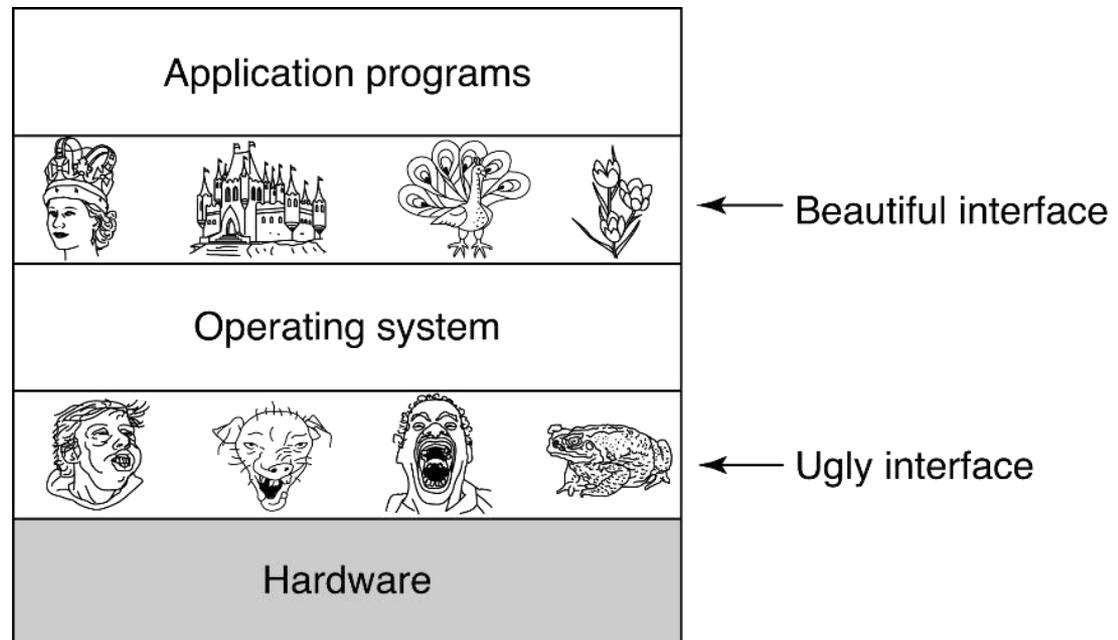


# Estrutura em Níveis de um SO



# Definição de um S.O. de acordo com a função desejada

- É uma máquina estendida (abordagem *top-down*, “abstração do todo para as partes”)
  - Oculta os detalhes complicados que têm que ser executados
  - Apresenta ao usuário uma máquina virtual, mais fácil de usar



# Definição de um S.O. de acordo com a função desejada

- É um gerenciador de recursos (abordagem *bottom-up* “abstração das partes para o todo”)
  - Precisa fornecer a alocação ordenada e controlada de recursos (UCP, memória, E/S) entre processos que competem por eles;
  - Gerencia todas as partes de um sistema complexo;
  - Cada programa tem um tempo com o recurso;
  - Cada programa tem um espaço no recurso;
  - Faz a **multiplexação** (compartilhamento) dos recursos.

# Diferença de tratamento de uma operação de acordo com a abordagem

- Ex.: como é realizada a entrada/saída de um disco flexível – tarefa: Leitura e Escrita
  - Usuário: alto nível – abstração simples
    - Visualização do arquivo a ser lido e escrito;
    - Arquivo é lido e escrito;
    - Arquivo é fechado.
  - SO: baixo nível de detalhes
    - Número de parâmetros;
    - Endereço de bloco a ser lido;
    - Número de setores por trilha;
    - Modo de gravação;
    - Em um chip controlador baseado no NEC PD765, são 16 comandos para a carga de 1 a 9 bytes no registrador de dispositivos (ver livro)

# Sistema Operacional como Gerenciador de Recursos

- Gerenciar todos os dispositivos e recursos disponíveis no computador
  - Ex.: se dois processos querem acessar um mesmo recurso, por exemplo, uma impressora, o SO é responsável por estabelecer uma ordem para que ambos os processos possam realizar sua tarefa.
- Coordena a alocação controlada e ordenada dos recursos;

# Objetivos de um Sistema Operacional

- Apresentar ao usuário do sistema uma forma amena de utilizar o computador. Criar uma máquina virtual, de fácil compreensão para o usuário, com características diferentes da máquina física;
- Utilizar o *hardware* disponível da forma mais otimizada possível, para aumentar o desempenho do sistema;
- Garantir o maior nível de segurança e funcionabilidade possível, sem incomodar os usuários com detalhes técnicos do funcionamento.

# Histórico de Evolução (SO)

- Meados do século XIX: Charles Babbage (1792-1871), por volta de 1833, projetou o primeiro computador. No entanto, a pouca tecnologia da época não permitiu que o projeto tivesse sucesso.
- Máquina analítica:
  - Apenas '*hardware*', e complexa de ser manipulada;
  - Charles Babbage conheceu Ada Lovelace, que ficou muito interessada no seu trabalho; traduziu um artigo sobre a máquina analítica e acrescentou uma grande quantidade de notas, explicando seu funcionamento, e mostrando como utilizá-la para calcular uma sequência de Bernoulli
  - Sua metodologia foi tão precisa e descritiva, que hoje é considerada a primeira programadora da história

# Histórico de Evolução

## Primeira Geração

- Primeira Geração (1940-1955): Válvulas
  - Computadores experimentais, enormes, que ocupavam salas imensas e dependiam de potentes sistemas de refrigeração para seu funcionamento;
  - Dezenas de milhares de válvulas (mais de 20.000) e outros componentes eletrônicos;
  - Não existiam ainda os conceitos de sistema operacional e linguagem de programação de alto nível; programação feita diretamente no hardware.

# Histórico de Evolução

## Primeira Geração

- Mesmo grupo de pessoas projetava, construía, programava, operava e fazia a manutenção de cada máquina;
- O acesso às máquinas era feito por meio de reserva de tempo: cada usuário fazia sua programação diretamente nos painéis das máquinas (programação "*hardwired*");
- Praticamente todos os problemas submetidos eram cálculos numéricos diretos (tabelas de seno, logaritmos, etc);

# Histórico de Evolução

## Primeira Geração

- Final dos anos 40: primeiro computador eletrônico, o ENIAC (***E**lectronic **N**umerical **I**ntegrator **A**nd **C**omputer*);
- 1950: surgem os cartões perfurados
  - Os programas eram codificados nos cartões e sua leitura era feita por máquinas específicas, que necessitavam de operadores;
- John Von Neumann propõe uma programação não “*hardwired*”. Nasce o Assembler/Assembly;

# Histórico de Evolução

## Segunda Geração

- Segunda Geração (1955-1965) - Transistores e Sistemas em *Batch*
  - O desenvolvimento dos transistores tornou o computador mais confiável possibilitando sua comercialização - *Mainframes*;
  - Separação entre projetistas, fabricante, programadores e técnicos de manutenção;
  - No entanto, devido aos altos custos, poucos tinham acesso a essa tecnologia - somente grandes empresas, órgãos governamentais ou universidades;

# Afinal, qual foi o primeiro S.O.?

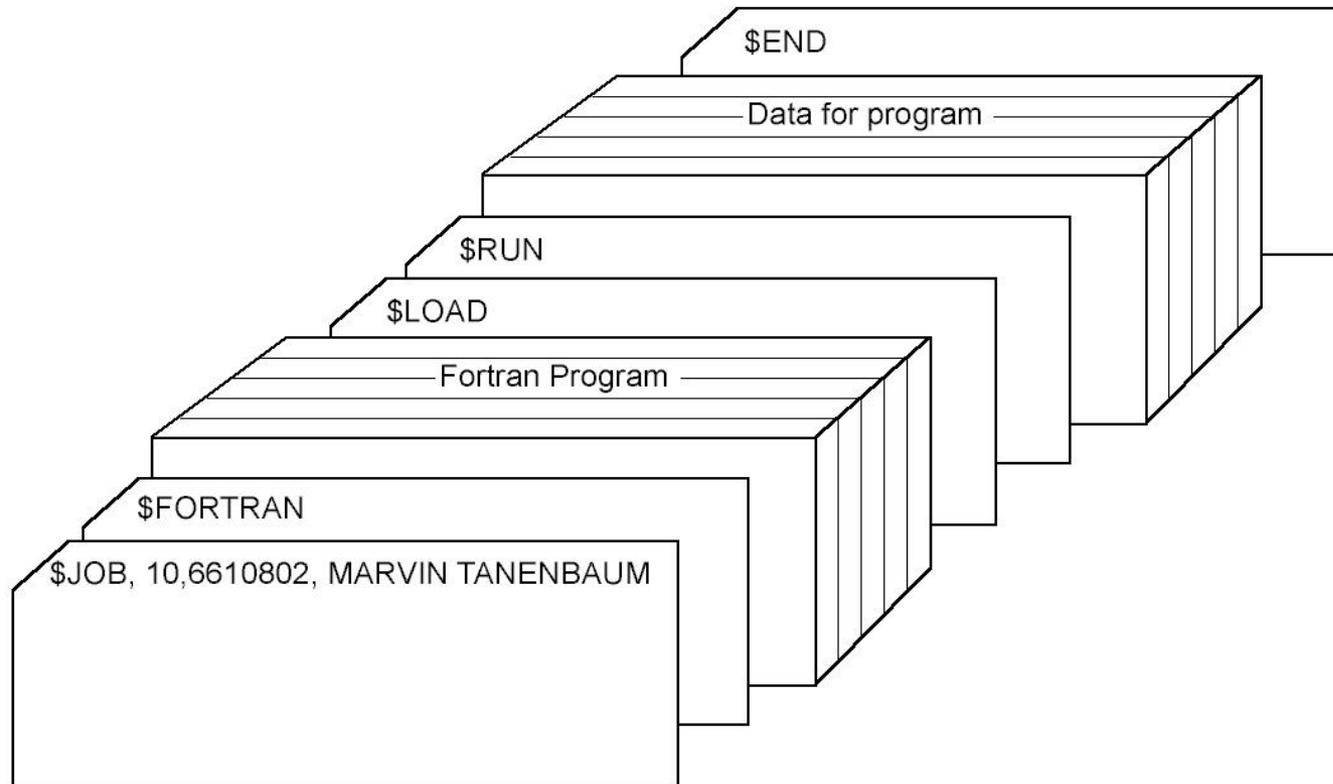
- O GM-NAA I/O foi o primeiro sistema operacional **comercial**, criado em 1956 por Robert L. Patrick para mainframes IBM 704.
- Porém, existem registros de quatro sistemas operacionais criados antes do GM-NAA I/O, sendo o LEO I (Lyons Eletronics Office) o mais antigo, criado em 1951. A diferença é que estes sistemas foram somente de uso interno dos centro de pesquisas e empresas que os desenvolveram
- Vale lembrar que todos os S.O. anteriores ao IBM OS/360 eram desenvolvidos para um hardware em particular, a cada nova versão de mainframe lançado, era necessário reescrever o sistema de acordo com as novas características do hardware utilizado.

# Histórico de Evolução

## Segunda Geração

- Surge a idéia de linguagem de programação de alto nível - Fortran (desenvolvida pela IBM - 1954-1957);
- Cartões perfurados ainda são utilizados:
  - Operação: cada programa (*tarefa*) ou conjunto de programas escrito e perfurado por um programador era entregue ao operador da máquina para que o mesmo fosse processado - alto custo
  - Sistemas em *Batch* (lote)
    - Consistia em coletar um conjunto de tarefas e fazer a gravação desse conjunto para uma fita magnética

# Histórico de Evolução Segunda Geração

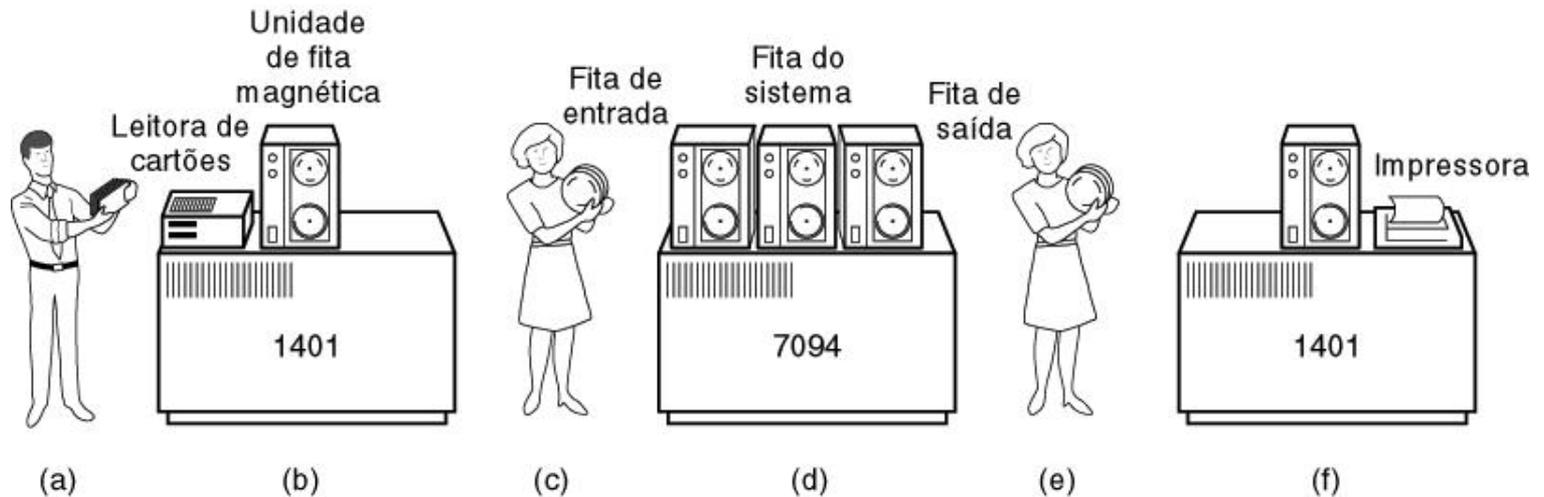


Estrutura de uma tarefa em Fortran

# Histórico de Evolução Segunda Geração

Antigo sistema em batch (lote):

- traz os cartões para o 1401;
- lê os cartões para a fita;
- coloca a fita no 7094 que executa o processamento;
- coloca a fita no 1401 que imprime a saída.



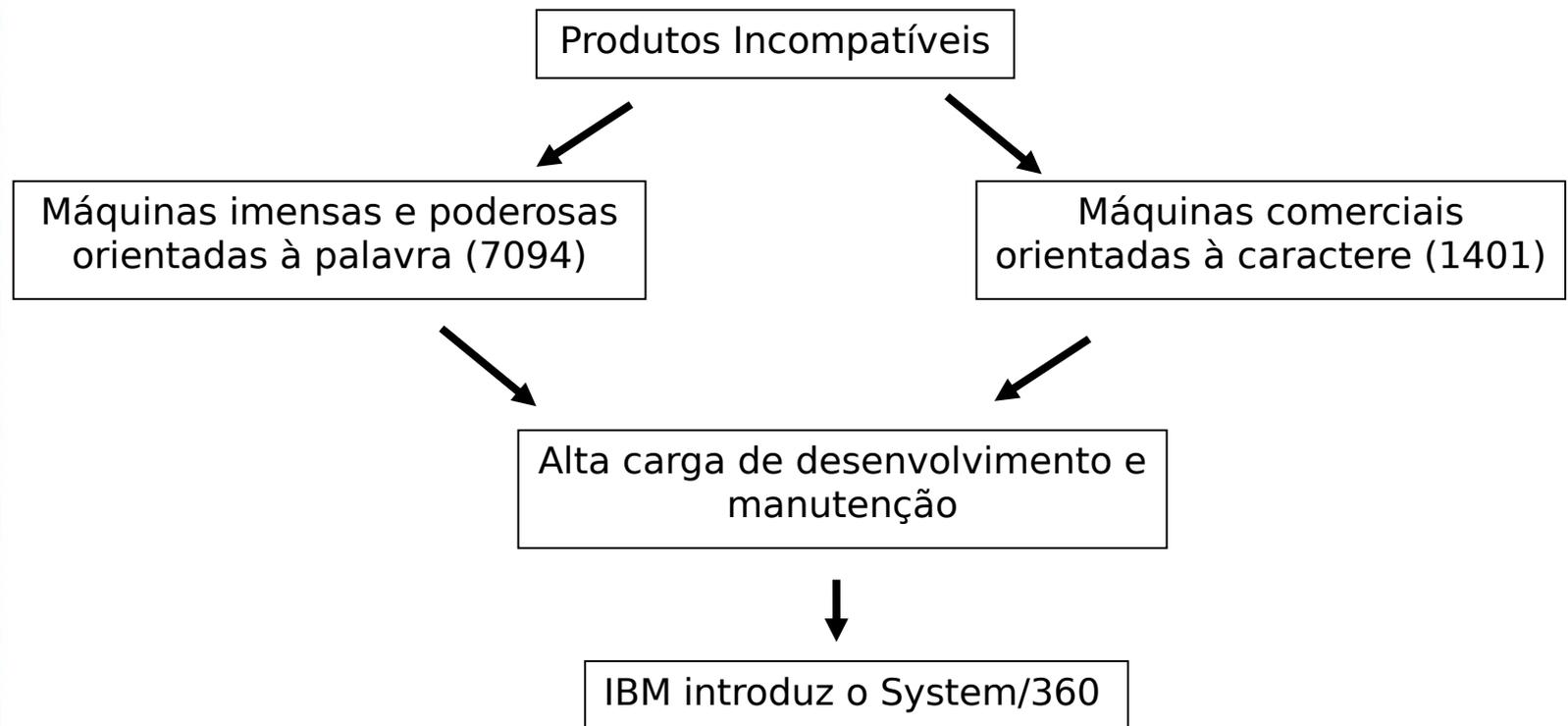
FMS (*Fortran Monitor System*)

Processamento: IBSYS – SO IBM para o 7094

# Histórico de Evolução

## Terceira Geração

- Terceira Geração (1965-1980) - Circuitos integrados e Multiprogramação



# Histórico de Evolução

## Terceira Geração

- System/360
  - Série de máquinas com software compatível;
  - Essas máquinas diferiam apenas no preço e desempenho, variando da 1401 até a 7094;
  - Foi a primeira a usar circuito integrado em pequena escala, ao invés de transistores;
  - O sistema operacional era o OS/360
    - Sua maior vantagem era também sua maior fraqueza: SO enorme e muito complexo, pois precisava realizar as funções de todas as máquinas → ineficiente, cheio de erros (milhões de linhas de código assembly escritas por milhares de programadores = milhares de erros)

# Histórico de Evolução

## Terceira Geração

- Os sistemas computacionais dessa época eram monoprogramados, ou seja, só executavam a tarefa seguinte se a anterior tivesse sido concluída
- Aplicações ***CPU-bound*** não tinham problema com relação ao tempo que se precisava esperar para realizar E/S
- Aplicações ***IO-bound*** gastavam de 80 a 90% do tempo realizando E/S
  - Enquanto isso, a UCP ficava parada
  - Solução: Multiprogramação

# Histórico de Evolução

## Terceira Geração

- **Multiprogramação:**
  - Dividir a memória em diversas partes e alocar a cada uma dessas partes uma *tarefa*.
  - Manter na memória simultaneamente uma quantidade de tarefas suficientes para ocupar 100% do tempo do processador, diminuindo a ociosidade.
  - Importante: o *hardware* é que protegia cada uma das tarefas contra acesso indevidos de outras tarefas. Futuramente, esta tarefa torna-se parte do sistema operacional.

# Histórico de Evolução

## Terceira Geração

- **Spooling** (*Simultaneous Peripheral Operation On Line*):
  - Passou a permitir que operações como a leitura de cartões de tarefas fosse feita direta do disco;
  - Assim que uma tarefa terminava, o sistema operacional já alocava uma nova *tarefa* a uma partição livre da memória direto do disco;
  - Eliminação de máquinas como as 1401 e a necessidade de se ficar transportando fitas de um lugar para outro.

# Histórico de Evolução

## Terceira Geração

- Mesmo com o surgimento de novas tecnologias, o tempo de processamento ainda era algo crítico. Para corrigir um erro de programação, por exemplo, o programador poderia levar horas pois cada *tarefa* era tratada dentro de um lote



*Timesharing*

# Histórico de Evolução

## Terceira Geração

- ***Timesharing***: cada usuário tinha um terminal *on-line* à disposição; a utilização da UCP de um módulo servidor é compartilhada entre os usuários
  - Primeiro sistema *Timesharing*: CTSS (*Compatible Time Sharing System*) – 7094 modificado.
  - Ex.: se 20 usuários estão ativos e 17 estão ausentes, o processador é alocado a cada um dos 3 *jobs* que estão sendo executados;

# Histórico de Evolução

## Terceira Geração

- Surge o MULTICS (*Multiplexed Information and Computing Service*), predecessor do UNIX;
  - Fruto de uma idéia do MIT, Bell Labs e General Electric, de desenvolver um computador que suportasse centenas de usuários simultâneos em *timesharing*
  - Codificado em PL/I, o que atrapalhou seu desenvolvimento (compilador fraco)
  - Apesar do fracasso comercial, teve enorme influência em SO's futuros
- Família de minicomputadores PDP da DEC;
  - Diferente da família System/360, eram incompatíveis;
  - Unix original rodava no PDP-7 (Ken Thompson - cientista da Bell Labs)
  - O PDP-1 custava 120 mil (5% do valor de um 7094)
    - Tinha 4K palavras de 18 bits

# Histórico de Evolução

## Quarta Geração

- Quarta Geração (1980-1990) – Computadores Pessoais
  - Com a tecnologia de circuitos integrados de larga escala (LSI) surgem *chips* com milhares de transistores encapsulados em um centímetro quadrado de silício
    - Intel – 8080 (1974)
    - IBM – PC (início dos anos 80)
    - Apple – Apple e Macintosh

# Histórico de Evolução

## Quarta Geração

- Intel 8080 - CP/M da Digital Research, de Gary Kildall)
  - CP/M (*Control Program for MicroComputer*) - sistema operacional baseado em disco;
- IBM PC - DOS
  - Inicialmente, a IBM tentou utilizar o CP/M, mas Kildall não quis nenhum acordo;
  - IBM procurou Bill Gates pedindo um sistema operacional para rodar e ser vendido juntamente com o IBM PC;
  - Bill Gates comprou da Seattle Computer Products o DOS (*Disk Operating System*), desenvolvido por Tim Paterson;

# Histórico de Evolução

## Quarta Geração

- Evolução do DOS para o MS-DOS (*MicroSoft DOS*)
  - Tanto o CP/M quanto o MS-DOS eram baseados em comandos;
- Macintosh Apple - Sistemas baseados em janelas (*GUI - Graphical User Interface*)
- Microsoft - Plataforma Windows
- A história deste período da computação está muito bem retratada no filme “Piratas da Informática” (*“Pirates of Silicon Valley”*) e no documentário em três partes “O Triunfo dos Nerds”

# Histórico de Evolução

## Quinta Geração (1990-hoje)

- O protocolo de comunicações TCP/IP tornou-se largamente utilizado (Depto. de Defesa dos EUA) e as LANs (*Local Area Networks*) tornaram-se mais práticas e econômicas com o surgimento do padrão **Ethernet**, desenvolvido pela Xerox;
- Desenvolvimento e popularização do modelo cliente/servidor;
- Proliferação das redes de computadores
  - *Internet*

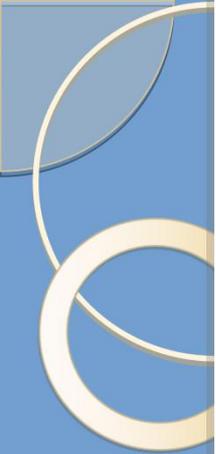
# Histórico de Evolução

## Quinta Geração (1990-hoje)

- Sistemas Operacionais Distribuídos:
  - Apresenta-se como um sistema operacional centralizado, mas que, na realidade, tem suas funções executadas por um conjunto de máquinas independentes;
- Sistemas Operacionais em Rede;
  - Usuários conhecem a localização dos recursos que estão utilizando e não têm a visão de um sistema centralizado
- Sistema Operacionais para dispositivos móveis;
  - Execução de tarefas com economia de energia (baterias limitadas), aplicações voltadas principalmente para *web*

# História dos Sistemas Operacionais - Resumo

- Primeira geração – anos 50
  - Válvulas, painéis de programação; processamento em lotes
- Segunda geração – anos 60
  - Multiprogramação, multiprocessamento, *timesharing*, tempo real
- Terceira geração – meados 60 a meados 70
  - Sistemas de propósito geral; desenvolvimento em linguagens de alto nível
- Quarta geração – meados 70 a meados 80
  - Cliente/servidor, processamento distribuído, interface gráfica
- Quinta geração – meados 80 aos dias atuais
  - Sistemas operacionais de redes, computação distribuída, software livre, multiprocessamento



# O Zoológico de Sistemas Operacionais

- Sistemas operacionais para computadores de grande porte;
- Sistemas operacionais para servidores;
- Sistemas operacionais para multiprocessadores;
- Sistemas operacionais para computadores pessoais;
- Sistemas operacionais de tempo real;
- Sistemas operacionais embarcados;
- Sistemas operacionais para cartões inteligentes;
- Sistemas operacionais para dispositivos móveis.

# Qual S.O. devo usar?

- Para saber qual o sistema operacional mostra-se mais adequado para uso em seu computador, vários critérios devem ser considerados:
  - Preço;
  - Confiabilidade;
  - Desempenho;
  - Compatibilidade com o *hardware*;
  - Facilidade de interação;
  - Aplicativos disponíveis;
  - Filosofia de uso.

# Na dúvida...

