

Capítulo 13: Sistemas de E/S



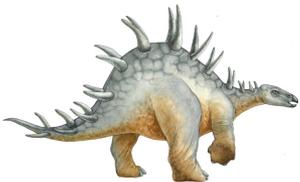
Capítulo 13: Sistemas de E/S

- ❑ Hardware de E/S
- ❑ Interface de E/S da aplicação
- ❑ Subsistema de E/S do kernel
- ❑ Transformando requisições de E/S em operações de hardware
- ❑ Fluxos
- ❑ Desempenho



Objetivos

- ❑ Explorar a estrutura do subsistema de E/S de um sistema operacional
- ❑ Discutir os princípios do hardware de E/S e sua complexidade
- ❑ Oferecer detalhes dos aspectos de desempenho do hardware e software de E/S

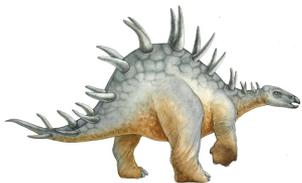
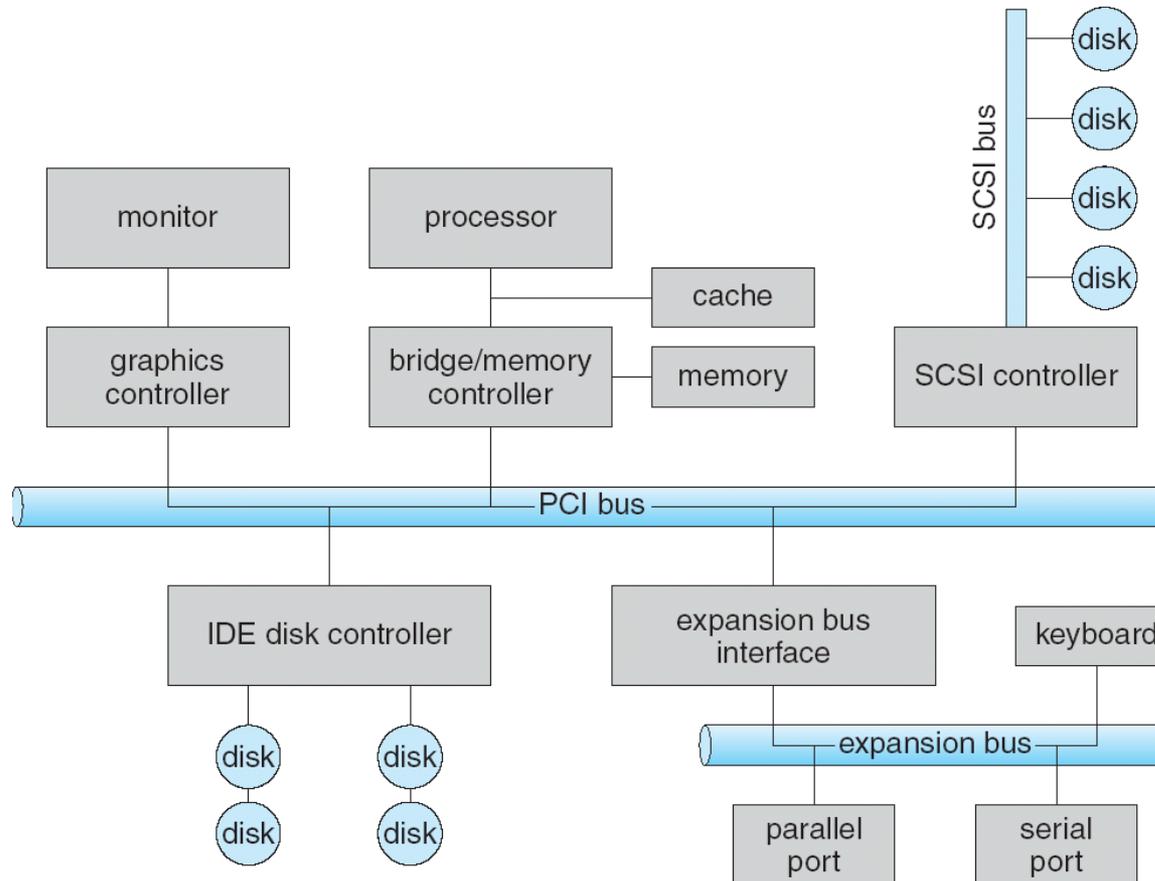


Hardware de E/S

- ❑ Variedade incrível de dispositivos de E/S
- ❑ Conceitos comuns
 - **Porta**
 - **Barramento (daisy chain ou acesso direto compartilhado)**
 - **Controlador (adaptador de host)**
- ❑ Dispositivo de controle das instruções de E/S
- ❑ Dispositivos têm endereços, usados por
 - Instruções de E/S diretas
 - E/S mapeada na memória

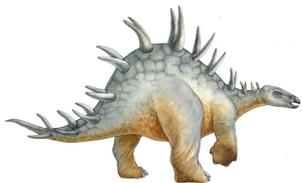


Estrutura típica de barramento de PC



Locais de porta de E/S de dispositivo nos PCs

I/O address range (hexadecimal)	device
000–00F	DMA controller
020–021	interrupt controller
040–043	timer
200–20F	game controller
2F8–2FF	serial port (secondary)
320–32F	hard-disk controller
378–37F	parallel port
3D0–3DF	graphics controller
3F0–3F7	diskette-drive controller
3F8–3FF	serial port (primary)



Protocolo de interação entre host e controladores

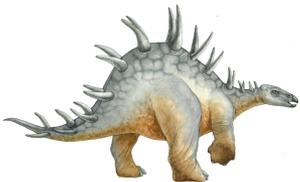
- Estado do dispositivo:
 - pronto
 - ocupado
 - erro

- Ciclo de **busy-wait** para esperar E/S do dispositivo (se o dispositivo demorar demais para atender requisições, pode haver perda de dados no buffer)



Interrupções

- ❑ **Linha de requisição de interrupção** da CPU disparada por dispositivo de E/S
- ❑ **Manipulador de interrupção** recebe interrupções
- ❑ **Mascarável** para ignorar ou adiar algumas interrupções
- ❑ Vetor de interrupção para despachar interrupção ao tratador correto
 - Baseado em prioridade
 - Algumas **não-mascaráveis**
- ❑ Mecanismo de interrupção também usado para exceções



Ciclo de E/S controlado por interrupção

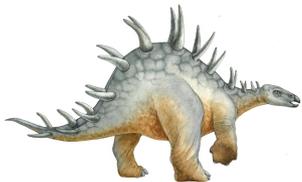
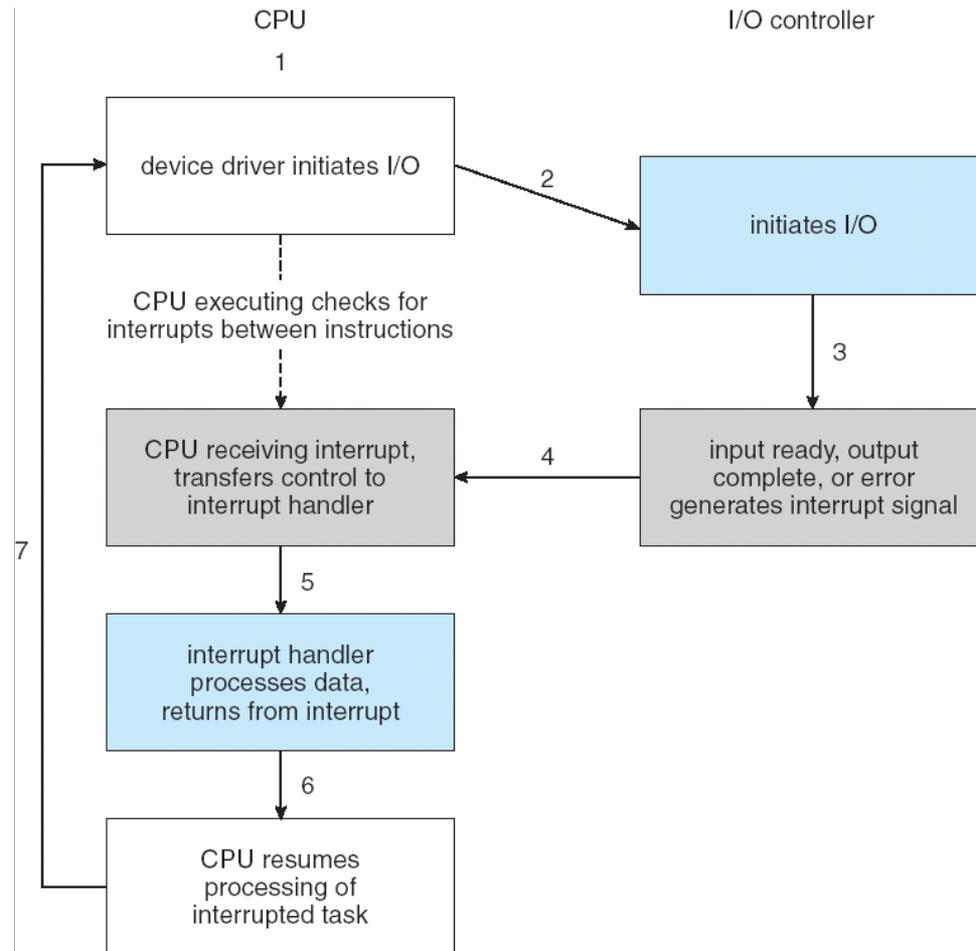


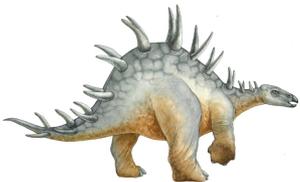
Tabela de vetor de evento do Intel Pentium

vector number	description
0	divide error
1	debug exception
2	null interrupt
3	breakpoint
4	INTO-detected overflow
5	bound range exception
6	invalid opcode
7	device not available
8	double fault
9	coprocessor segment overrun (reserved)
10	invalid task state segment
11	segment not present
12	stack fault
13	general protection
14	page fault
15	(Intel reserved, do not use)
16	floating-point error
17	alignment check
18	machine check
19–31	(Intel reserved, do not use)
32–255	maskable interrupts

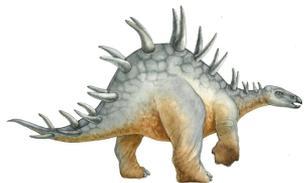
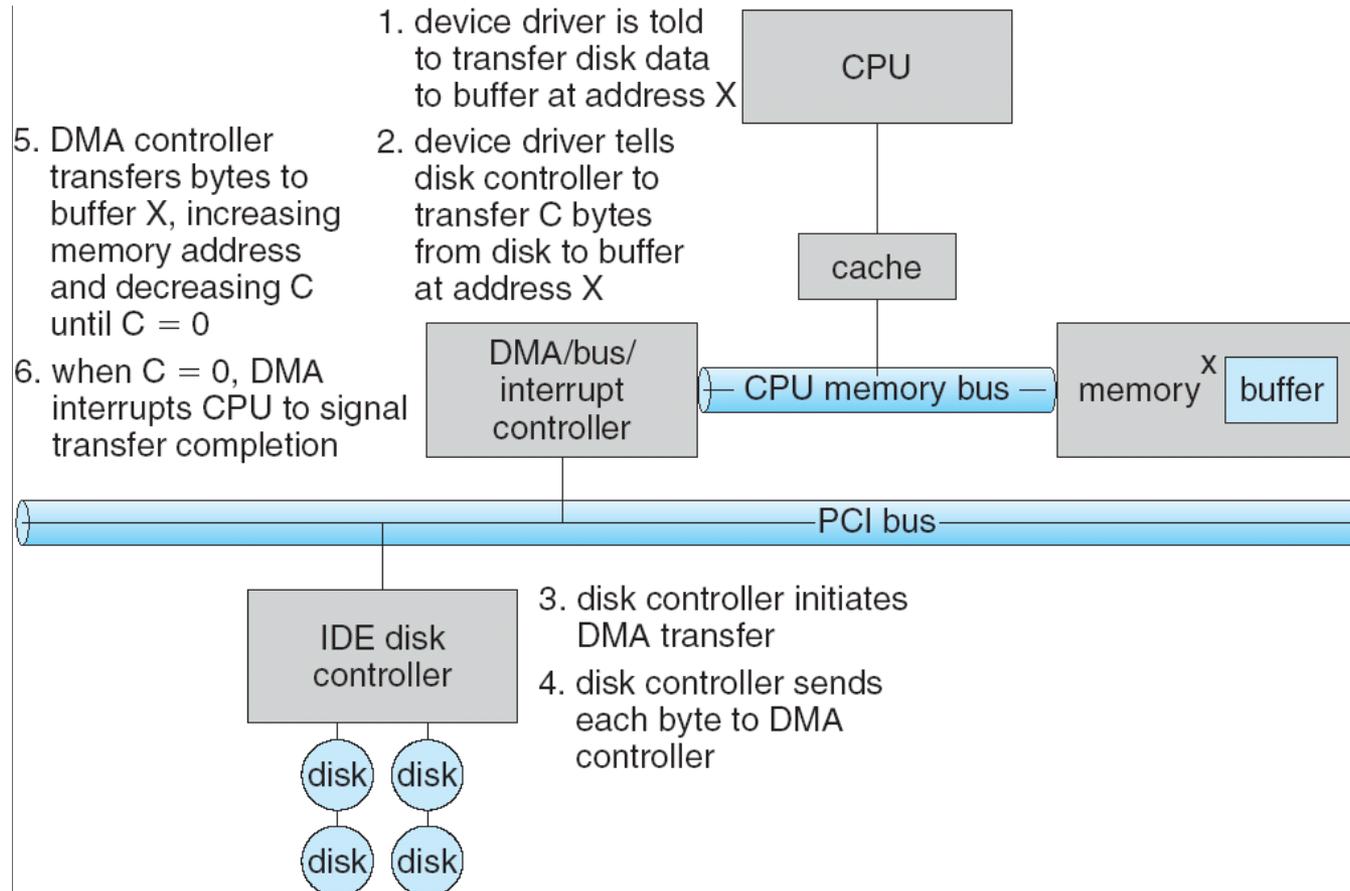


Acesso direto à memória

- ❑ Usado para evitar **E/S programada** para grande movimento de dados
- ❑ Requer controlador de **DMA (Direct Memory Access)**
- ❑ Contorna CPU para transferir dados diretamente entre dispositivo de E/S e memória



6 etapas para realizar transferência por DMA

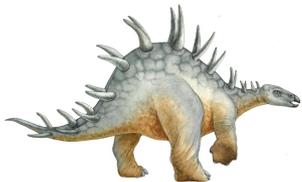
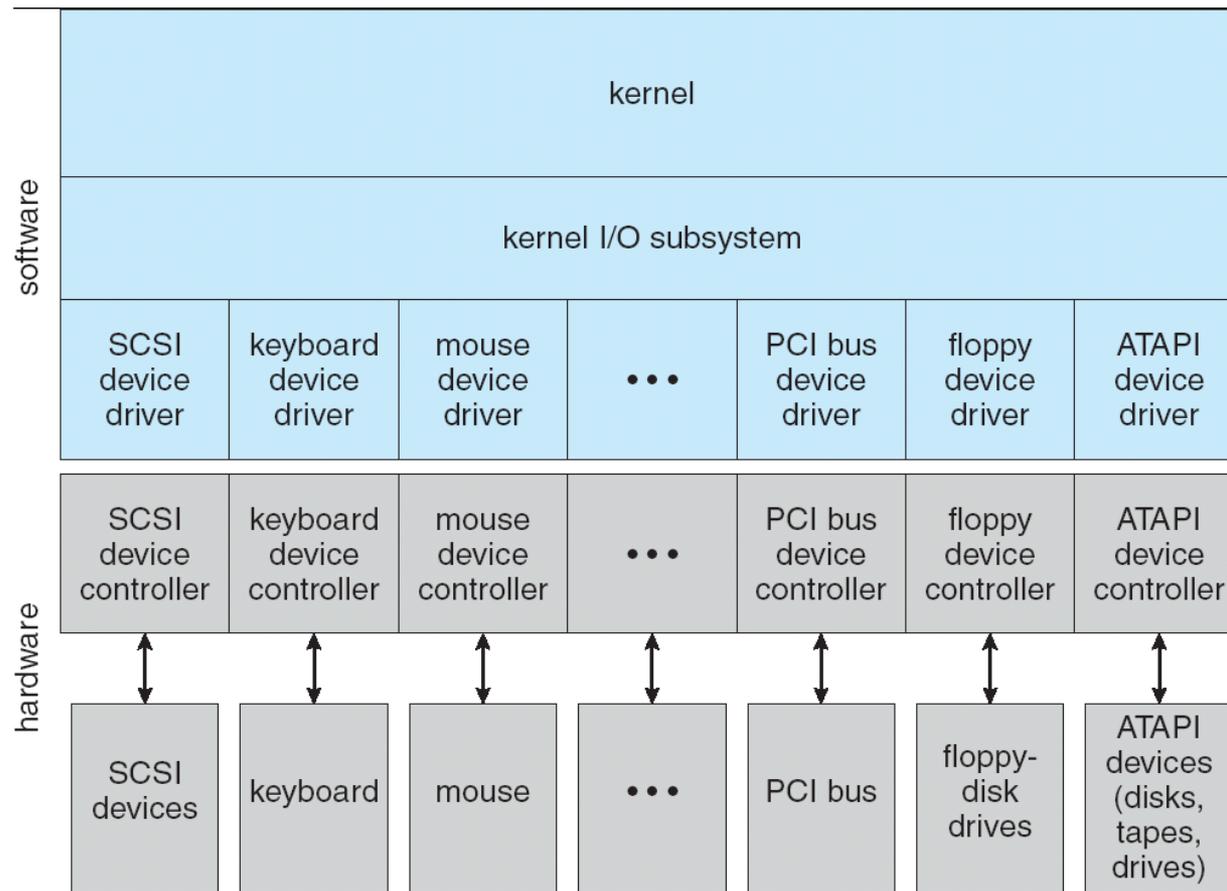


Interface de E/S

- Sistema de E/S chama comportamentos de dispositivo em classes genéricas
- Camada de driver de dispositivo esconde diferenças entre controladores de E/S do kernel
- Dispositivos variam em muitas dimensões
 - **Fluxo de caracteres ou bloco**
 - **Acesso seqüencial ou aleatório**
 - **Compartilhável ou dedicado**
 - **Velocidade de operação**
 - **Leitura/escrita, somente de leitura ou somente de escrita**



Uma estrutura de E/S do kernel



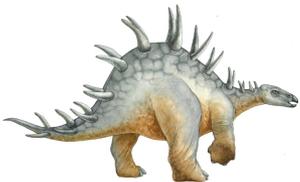
Características de dispositivos de E/S

aspect	variation	example
data-transfer mode	character block	terminal disk
access method	sequential random	modem CD-ROM
transfer schedule	synchronous asynchronous	tape keyboard
sharing	dedicated sharable	tape keyboard
device speed	latency seek time transfer rate delay between operations	
I/O direction	read only write only read–write	CD-ROM graphics controller disk



Dispositivos de bloco e caractere

- ❑ Dispositivos de bloco incluem unidades de disco
 - Comandos incluem read, write, seek
 - E/S bruta ou acesso ao sistema de arquivos
 - Possível acesso a arquivo mapeado na memória
- ❑ Dispositivos de caractere incluem teclado, mouse, porta serial
 - Comandos incluem `get`, `put`



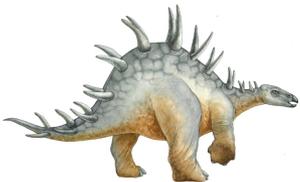
Dispositivos de rede

- ❑ Variação suficiente de bloco e caractere para ter interface própria
- ❑ Unix e Windows NT/9x/2000 incluem interface de socket
- ❑ Técnicas variam bastante (pipes, FIFOs, fluxos, filas, mailboxes)



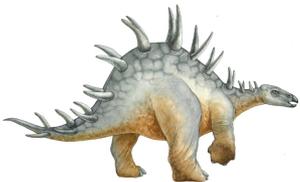
Relógios e temporizadores

- ❑ Oferecem hora atual, tempo transcorrido, temporizador
- ❑ **Temporizador de intervalo programável** usado para temporizações, interrupções periódicas

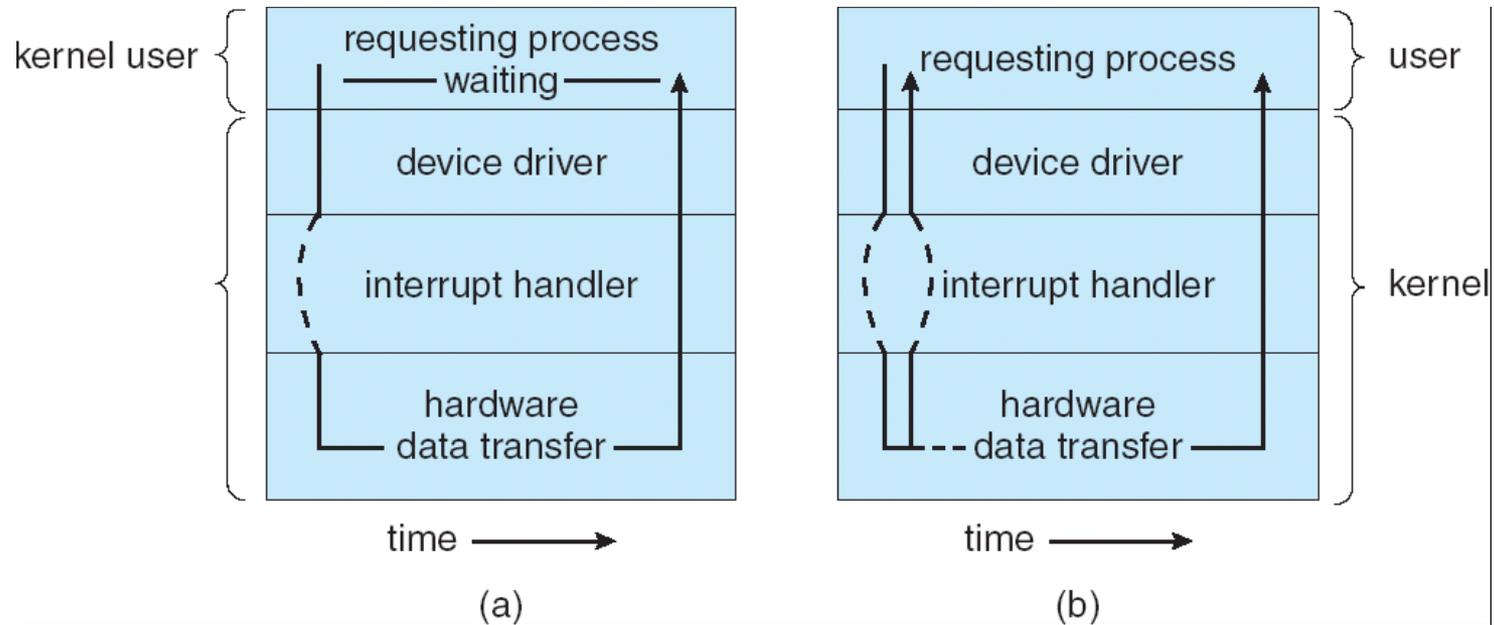


E/S bloqueante e não bloqueante

- ❑ **Bloqueante** – processo suspenso até E/S terminar
 - Fácil de usar e entender
 - Insuficiente para algumas necessidades
- ❑ **Não bloqueante** – Chamada de E/S retorna o máximo possível
 - Interface de usuário, cópia de dados (E/S em buffer)
 - Implementado via multithreading
 - Retorna rapidamente com contagem de objetos lidos ou gravados
- ❑ **Assíncrona** - processo executa enquanto E/S executa
 - Difícil de usar
 - Subsistema de E/S sinaliza processo quando E/S termina

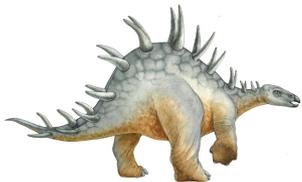


Dois métodos de E/S



Síncrona

Assíncrona

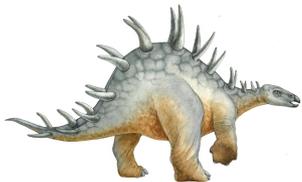
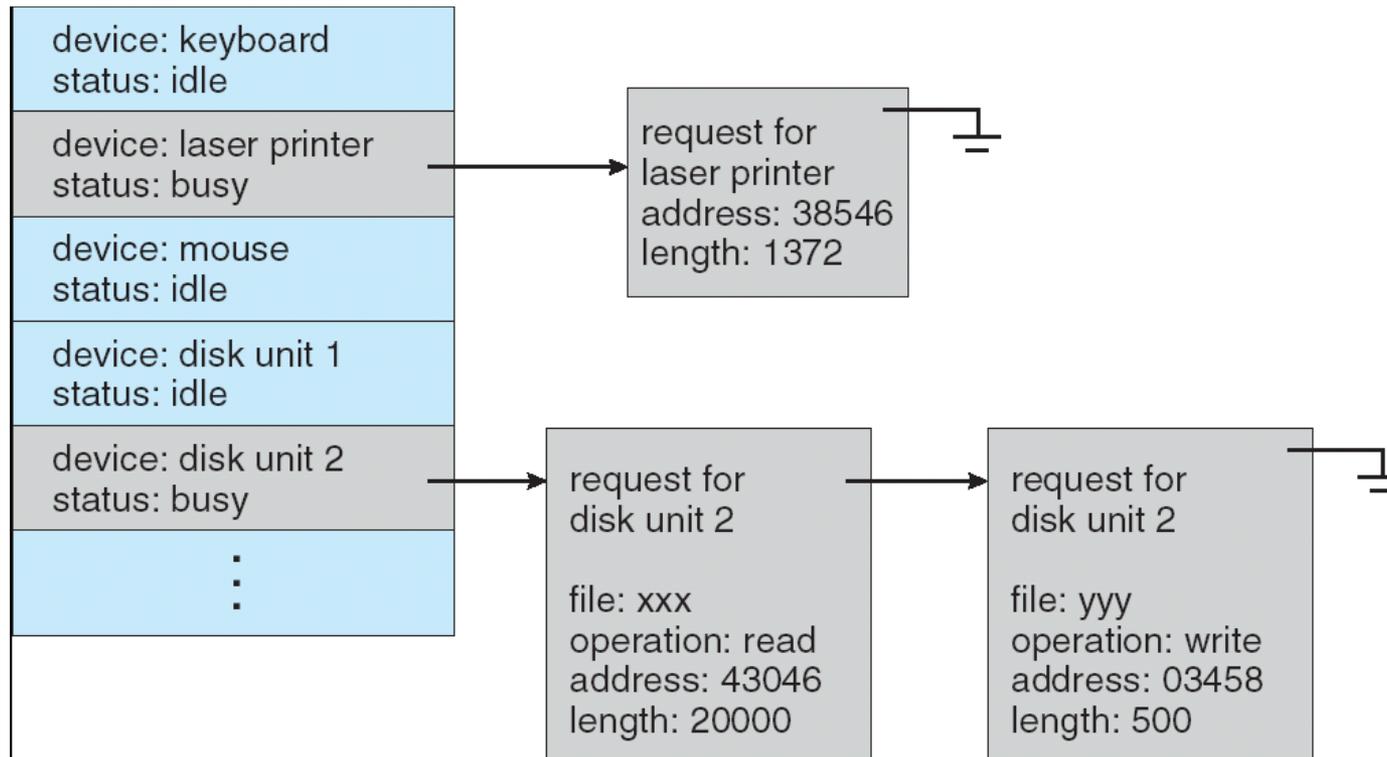


Subsistema de E/S do kernel

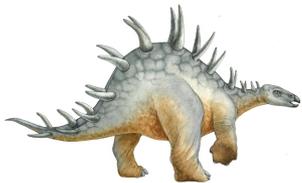
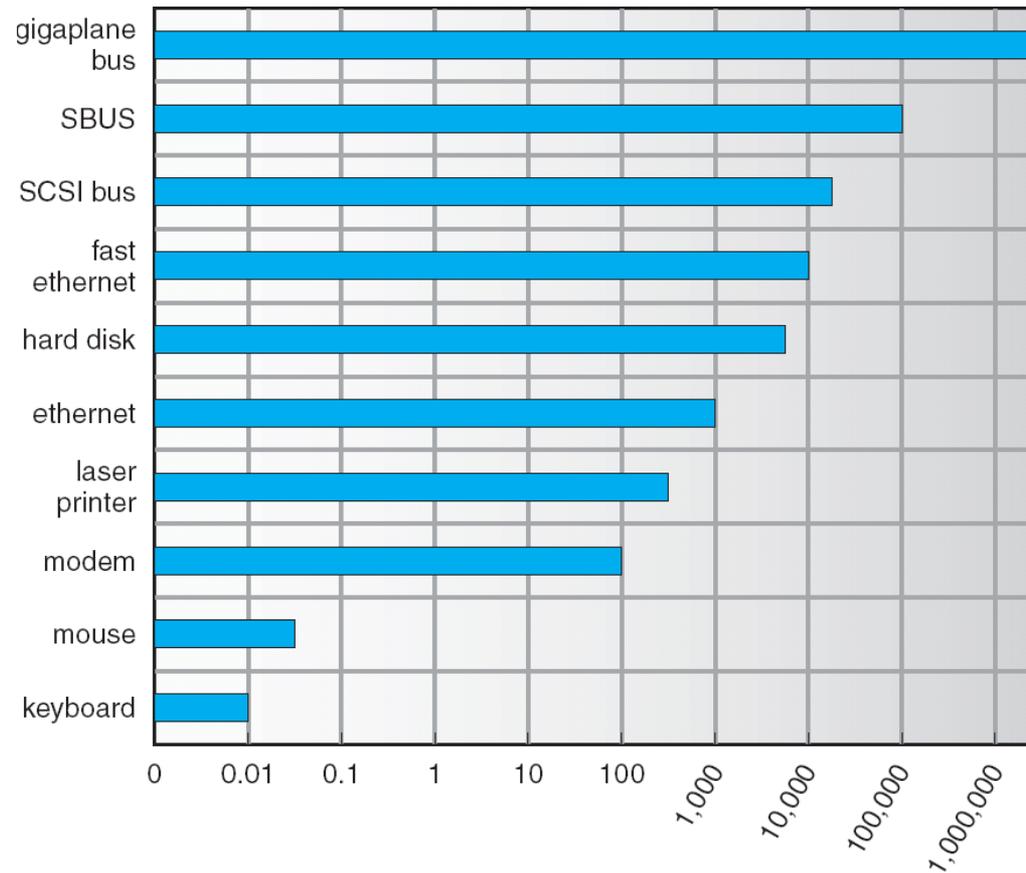
- Escalonamento
 - Alguma ordenação de requisição de E/S via fila por dispositivo
 - Alguns SOs tentam imparcialidade
- Buffering – armazena dados na memória enquanto transfere entre dispositivos
 - Para enfrentar divergência de velocidade de dispositivo
 - Para enfrentar divergência de tamanho de transferência do dispositivo



Tabela de status de dispositivo

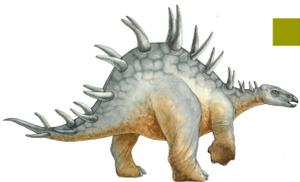


Taxas de transferência de dispositivo do Sun Enterprise 6000



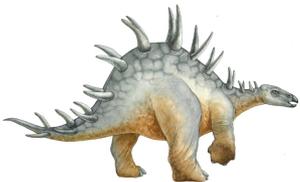
Subsistema de E/S do kernel

- **Caching** – memória rápida mantendo cópia dos dados
 - Sempre apenas uma cópia
 - Fundamental para o desempenho
- **Spooling** – mantém saída para um dispositivo
 - Se dispositivo pode atender apenas uma requisição por vez
 - Por exemplo, impressão
- **Reserva de dispositivo** – oferece acesso exclusivo a um dispositivo
 - Chamadas do sistema para alocação e desalocação
 - Observa deadlock



Tratamento de erros

- ❑ O SO pode se recuperar de falhas na leitura de disco, dispositivo não disponível, falhas de gravação
- ❑ A maioria retorna um número ou código de erro quando a requisição de E/S falha
- ❑ Logs de erro do sistema mantêm relatórios de problemas



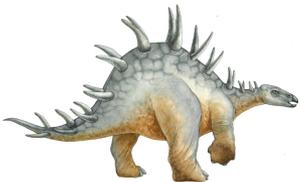
Proteção da E/S

- O processo do usuário pode acidental ou propositalmente tentar atrapalhar a operação do sistema via instruções de E/S ilegais
 - Todas as instruções de E/S definidas como privilegiadas
 - E/S precisa ser realizada via chamadas do sistema
 - Locais de memória mapeada e de porta de E/S também precisam ser protegidos

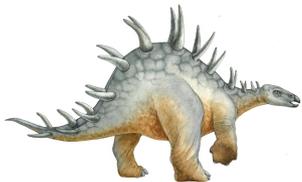
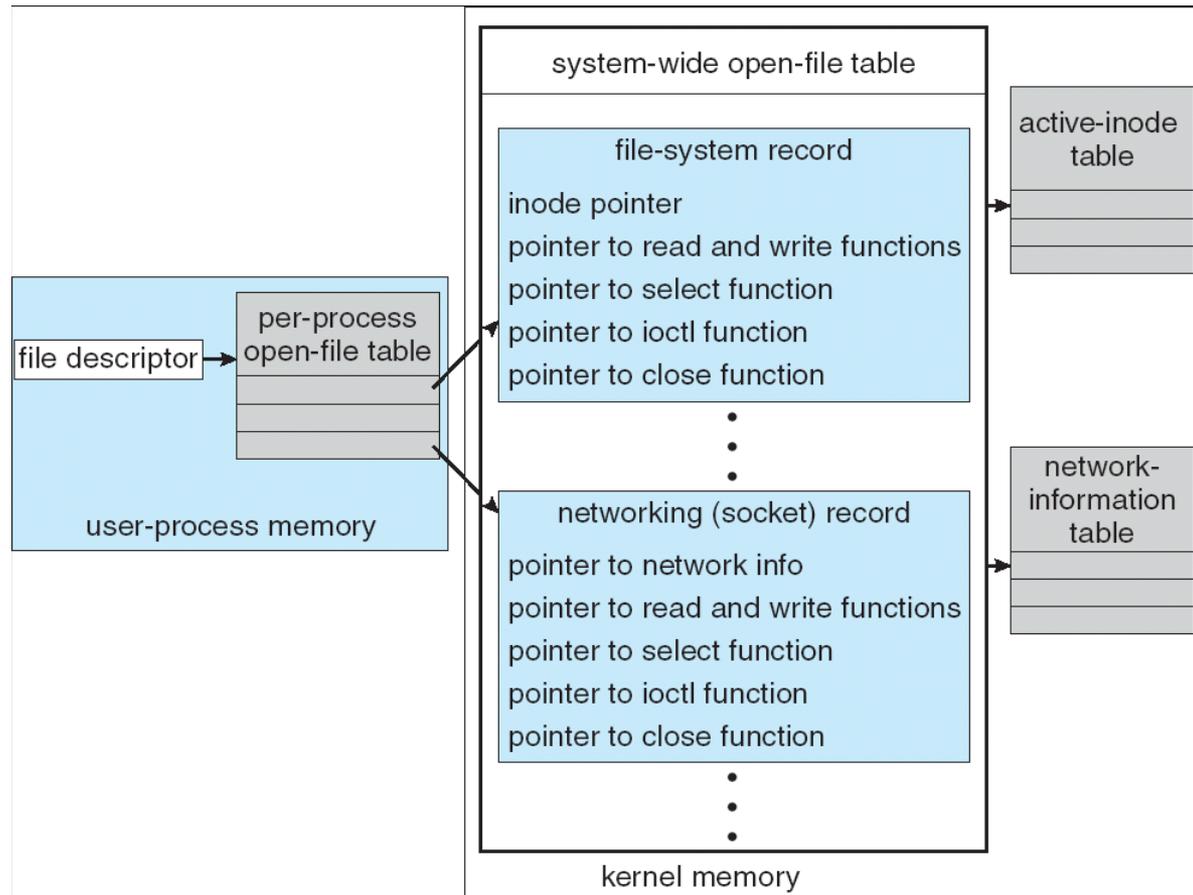


Estruturas de dados do kernel

- O kernel mantém informações de estado para componentes de E/S, incluindo tabelas de arquivos abertos, conexões de rede, estado de dispositivo de caractere



Estrutura de E/S do kernel no UNIX



Requisições de E/S para operações de hardware

- Considere a leitura de um arquivo do disco para um processo:
 - Determinar dispositivo mantendo arquivo
 - Traduzir nome para representação de dispositivo
 - Ler dados fisicamente do disco para o buffer
 - Tornar dados disponíveis ao processo solicitante
 - Retornar controle ao processo

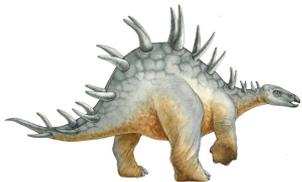
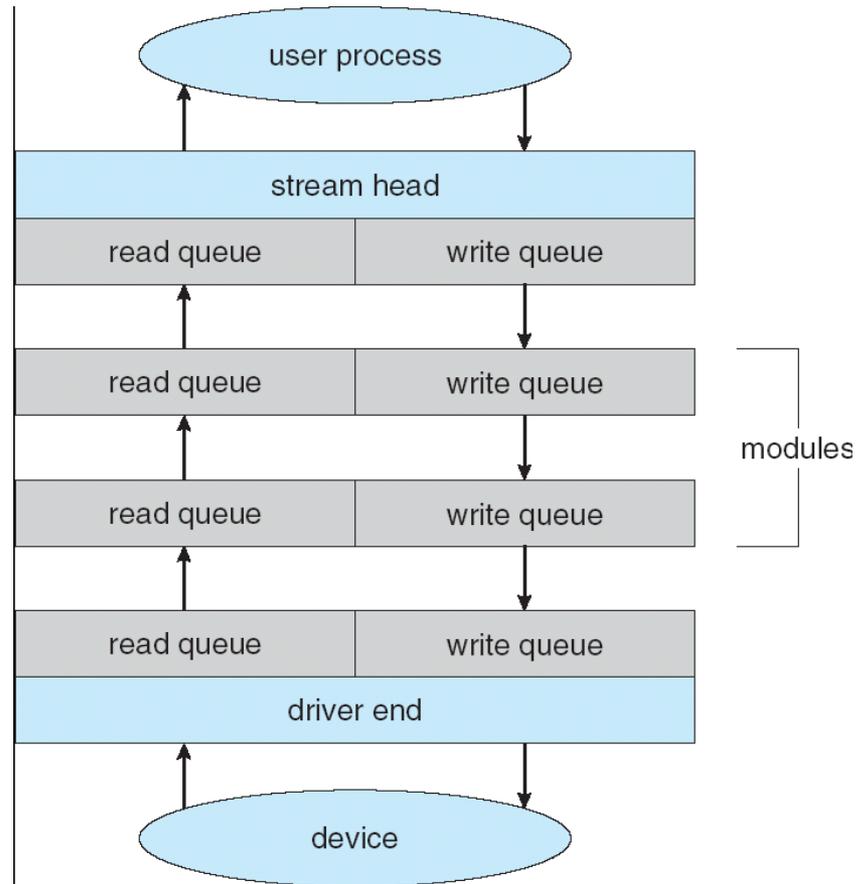


STREAMS

- ❑ **STREAM** – um canal de comunicação full-duplex entre um processo em nível do usuário e um dispositivo no Unix System V em diante
- ❑ Um STREAM consiste em:
 - Interfaces de cabeça STREAM com o dispositivo
 - zero ou mais módulos STREAM entre elas.
- ❑ Cada módulo contém uma **fila de leitura** e uma **fila de gravação**
- ❑ Passagem de mensagens é usada para comunicar entre as filas



A estrutura STREAMS



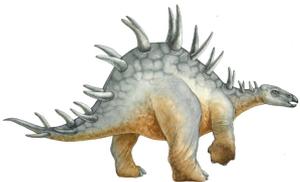
Desempenho

- E/S como fator principal no desempenho do sistema:
 - Demanda CPU para executar o driver de dispositivo, código de E/S do kernel
 - Trocas de contexto devido às interrupções
 - Cópia de dados
 - Tráfego de rede especialmente estressante

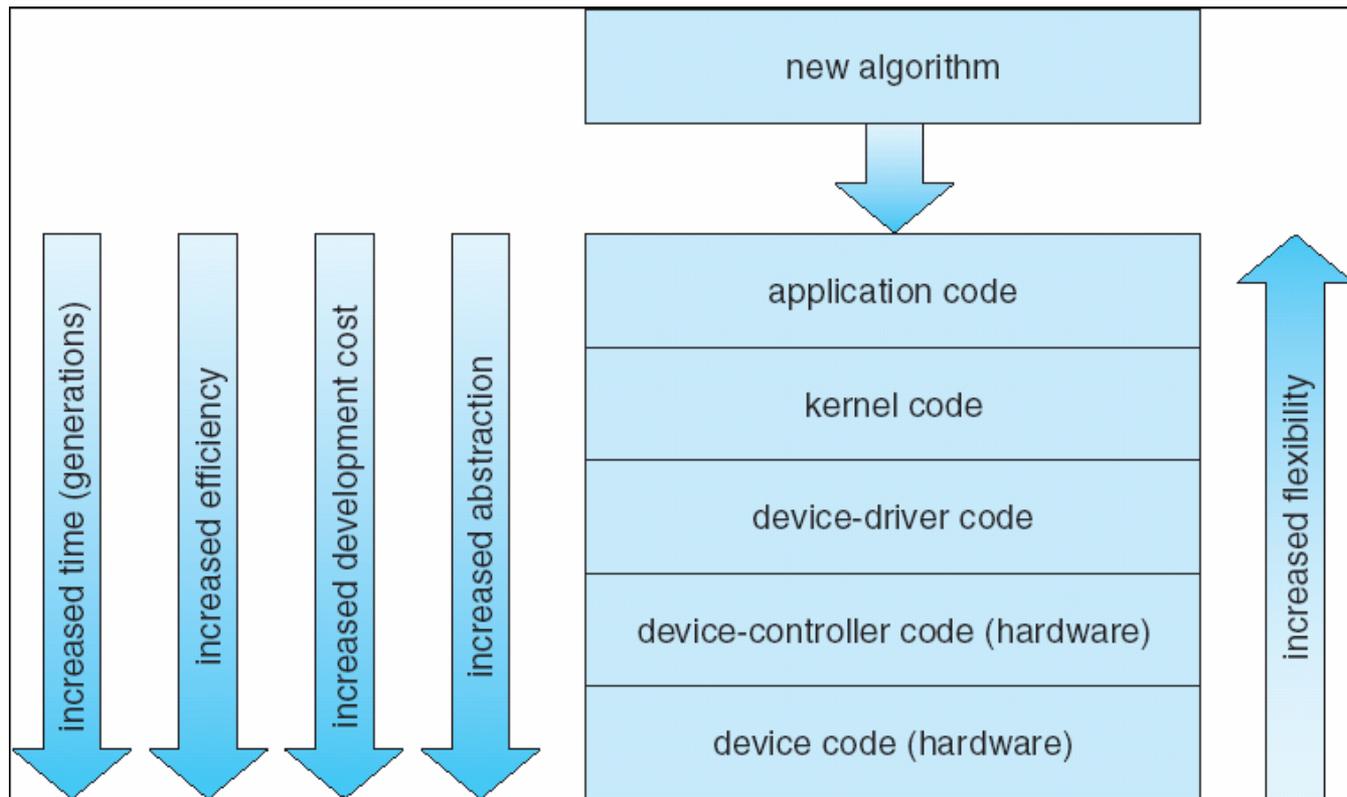


Melhorando o desempenho

- ❑ Reduzir o número de trocas de contexto
- ❑ Reduzir a cópia de dados
- ❑ Reduzir interrupções usando grandes transferências, controladores inteligentes, polling
- ❑ Usar DMA
- ❑ Balancear CPU, memória, barramento e desempenho de E/S para um throughput mais alto



Progressão da funcionalidade do dispositivo



Final do Capítulo 13

