TRADUÇÃO DO LIVRO DESIGN E IMPLEMENTAÇÃO DO SISTEMA OPERACIONAL MTX

PREFÁCIO

O propósito desse livro é prover uma plataforma suitável para ensinar e aprender a teoria e prática de desenvolvimento de sistemas operacionais. Abrange os conceitos e princípios de sistemas operacionais, e demonstra como aplicar os mesmos no designer e implementação de um sistema operacional completo gerado e produzido passo-a-passo. Esse livro usa o desenvolvimento do sistema operacional MTX para transmitir a real intenção desse livro, o qual tem prentenção de apresentar os princípios de design e técnicas de implementação de sistemas operacionais em geral.

Devido seu conteúdo técnico, esse livro não é indicado para cursos que ensinam somente, em nível introdutório, os conceitos básicos e princípios de sistemas operacionais sem a pratica da programação. É indicado para tecnicamente para cursos de sistemas operacionais que têm foco, em ambos, na prática e teoria. O estilo envolucional do livro, acoplado de código fonte detalhado e completo sistemas simples de trabalho, torna-lo especial suíte para estudo próprio. Esse livro contém vários novos e originais materiais especialmente no designe e uso de algoritimos paralelos em SMP – Symetric Multiprocessing, processamento simétrico e paralelo de processos (programas, aplicativos etc). Também é um guia para estudo avançado de sistemas operacionais.

Empreender este projeto de livro demonstrou ser um grande desafio. Enquanto este livro estava sendo preparado para publicação, eu tive bênçãos de encorajamentos e ajudas de varias pessoas. Eu gostaria de dedicar essa oportunidade para agradecer a todos. Eu também agradeço a Publicadora Internacional Springer AG por permitir me de dispor o código fonte de esse livre gratuito e publicamente. Espero, que o livro ajude os desenvolvimentos do sistema MTX e que possa-se aprimorar esse livro no futuro.

O código fonte incluído no livro pode ser baixado diretamente do website do autor em <http://www.eecs.wsu.edu/~cs460/mtxhome>.

CAPITULO 1

INTRODUÇÃO

* 1. SOBRE ESSE LIVRO

Esse livro é sobre o design e implementação de sistemas operacionais. Um sistema operacional (SO) é um série de programas e arquivos de suporte, os quais rodam em um sistema de computador para que esse sistema possa ser conveniente e de fácil uso. Sem um sistema operacional, um sistema de computador é essencialmente sem utilidade. O estudo de sistemas operacionais envolvem uma ampla variedades de matérias subjetivas em ciência da computação e engenharia da computação. Isso inclui arquitetura de computador, programação de computador e, o mais importante, interface entre o software e hardware do computador. Prover uma completa serie de conhecimento sobre sistemas computacionais. Tal conhecimento não é somente necessário para profissionais da computação mas também beneficia todos os tipos de esforços na careira relacionada a computação. Esse livro abrange os conceitos básicos e princípios de sistemas operacionais. Mostra-se como aplicar esses conceitos e princípios para fazer design e implementação do sistema operacional MTX em detalhes. MTX é um pequeno sistema baseado em Unix, com intenção principalmente ao ensinamento e aprendizagem da teoria e pratica de sistemas operacionais. Designado para PC´s Intel x86, e roda em PCs reais e maquinas virtuais que emulam o hardware do PC. Escrito em grande parte na linguagem de alto nível C, com pouco de 2% em código assembly. O núcleo do MTX inclui gerenciamento de processo e memoria, I/O drivers de dispositivos e um compatível arquivo de sistema Linux EXT2. Permite que usuários façam login via console do PC e via portas seriais, e suporta execuções de comandos simples com redirecionamento I/O, bem como mutiplos comandos conectados por pipes. Ao invés de apresentar um sistema completo em única etapa com tentativa de dissecar o sistema, esse livro mostra o desenvolvimento de um sistema operacional em etapas que evoluem-se progressivamente, iniciando a partir de um simples programa com múltipla tarefa, ate um completo sistema operacional. Em cada etapa, mostra-se como aplicar os conceitos e princípios de sistemas operacionais para que seja realizado o design e implementação de vários componentes de sistemas. Cada etapa consiste em uma pequena do sistema, o qual permite ao leitor que teste e observer as operações de um sistema operacional a partir de seu interior. Em cada estapa sucessiva, adiciona-se novas características ao sistema e expande-se a capacidade do sistema enquanto se caminha ao resultade de um sistema operacional completo. O design e implementação do MTX abrange o aspecto quase que absoluto do sistema de arquiteturas baseadas no Intel x86, partindo do modo real de 16-bit ate o modo protegido de 32-bit e eventualmente ate o multiprocessamento simétrico (SMP). Desconsiderando as diferenças de arquitetura, demonstra-se que os mesmos princípios de design e técnicas de implementações ode ser aplicado em todos os casos.

* 1. OBJETIVO E PUBLICO ALVO

O objetivo desse livro é prover uma plataforma suitavel para ensinamento e aprendizagem da teoria e pratica de sistemas operacionais. Use-se o desenvolvimento do sistema MTX visando demonstrar os princípios de design e técnicas de implementações de sistemas operacionais em geral. Devido ao conteúdo técnico, esse livro não é indicado para cursos de SO que ensinam somente os conceitos básicos e princípios de SO sem qualquer pratica de programação. É indicado aos estudantes de ciência da computação e profissionais da computação, que desejam estudar os detalhe internos de sistema operacionais. Esse livro abrange ambos aspectos práticos e teóricos de sistemas operacionais. Descreve o design e implementação de um sistema operacional real em detalhe. É um guia como os manuais tecnicamente orientados a cursos de sistemas operacionais que tem fim de servir de balança entre a teoria e a pratica. O estilo evolucional do livro, acoplado com código fonte detalhado e sistemas simples de trabalho completos, torna esse livro essencialmente um guia para próprio estudo de entusiastas da computação. O livro contem vários novos e originais materiais, especialmente no design e uso algoritimos paralelos em SMP. É também um guia para estudo avançado de sistemas operacionais.

* 1. CARACTERISTICAS ÚNICA DESSE LIVRO

Esse livro tem varias características únicas que diferenciam-lo de outros livros.

1. O livro é auto-organizado. Capitulo 2 abrange todas informações fundamentais e de segundo-plano que são necessárias ao estudo de sistemas operacionais. Isso inclui a CPU e operações I/O , endereços virtuais e reais, etapas de desenvolvimento de programas, montadores de programas em C com código assembly, imagem de execução, convenções de chamadas de sistema e tempo-real no uso da pilho. Quando se fala driveres de dispositivos explana-se as operações de cada dispositivo antes de mostrar o driver atual quanto ao seu design e implementação.Quando se fala dos sistemas de arquivo descreve-se operações de arquivo em detalhe antes de mostrar a implementações de um sistema de arquivo completo. Quando se trabalho com PCs em modo protegido de 32-bit também explana-se as operações do modo protegido em detalhes.
2. Capitulo 3 prover uma completa abragencia acerca da inicialização de sistemas operacionais, o qual não está presente em outros livros de SO. Complementado ao principio de inicialização, mostra-se como escrever carregadores de sistemas operacionais para carregar-se sistemas operacionais reais, tal como Linux, a partir de uma variedade de dispositivos carregáveis. Todos os carregadores usados no MTX são desenvolvidos no capitulo. Eles são comparáveis ao GRUB (GNU GRUB 2010) e isolinux (Syslinux 2013) em performance.
3. Esse livro mostra o design e implementação de um sistema operacional completo com etapas evolutivas e sucessivas.No capitulo 4, usa-se um programa simples para introduzir o conceito de processo e demonstrar o principio da mudança de contexto – salvar o estado do processo atual e preparar para que a cpu passe a dedicar ao processo conseguinte ao atual. Usa-se o modelo de ciclo de vida de processo para ilustrar o principio de multiplatarefa. O mesmo principio é usado no decorrer do livro para criar e gerenciar imagens de processo, independente do hardware gerenciador de memoria do sistema. Essa abordagem única ajuda ao leitor melhor entender gerenciamento de processo em um sistema operacional.
4. Esse livro inicia o desenvolvimento do MTX em modo real de 16-bit. Então extende o modo real do MTX ate o modo protegido de 32-bit e eventualmente ate SMP – Multiprocessamento Simetrico. Despido das diferences de arquitetura, demonstra que o mesmo principio de design e técnica de implementação pode ser aplicada a todos os casos.
5. Em vez de abranger somente conceitos e princípios de threads, esse livro mostra como extender-se o modelo de processo para que se possa implementar suporte de threads e demonstrar aplicações de threads por programas concorrentes.
6. MTX usa uma mistura de ferramentas para sicronização de processo, os quais são escolhidos para melhor ajudar as necessidades, e justificar o uso deles. No modo real de 16-bit, usa-se sleep/wakeup para gerenciamento de processo mas usa-se semaphores nos drivers de dispositivos e para melhor eficiência do sistema de arquivo. Também permite uma transição suave do ambiente uniprocessador para SMP. Por complementação, mostra-se como aplicar alguns dos clássicos problemas de sicronização de processo para o design do SO. Especificamente, usa-se o problema produtor de consumo como modelo para pipes e aplica ao problema leitor-escritor para gerenciamento de I/O buffer em SMP para improvisar concorrência.
7. Capitulo 8 abrange processamento de interrupção e agendamento de processo. Abrange interrupção de hardware do PC e processamento de interrupções em detalhes, e apresenta um geral framework o design de drivers de dispositivos como interrupção-driver. Explana por que manipuladores de interrupção não devem dormir ou ficar bloqueados. Discute os objetivos, politicas e algoritimos do agendamento de processo, e demonstra as diferenças dos efeitos dos algoritimos dos agendamentos de processo através de sistemas simples.
8. Capitulo 9 apresenta um tratamento unificado de interrupções e sinal de processamento, que ajuda clarear os papeis intencionais e uso adequado de sinais. Complementamente, usa exemplos para ilustrar os conceitos e técnicas de coletores de sinal.
9. Capitulo 12 abrange gerenciamento de I/O buffer. Discute o algoritimo de gerenciamento de buffer Unix (Bach, 1990) e aponta suas deficiências. Então mostra como fazer design de novos algoritimos para melhorar a performance do cache de buffer. Varios materiais apresentados nesse capitulo são novos e originais.
10. Explana a relação da hierarquia de operações de arquivo em um SO, a partir do espaço de usuário ate baixo espaço de núcleo (kernel) ate nível de driver de dispositivo. Então mostra a implementação de um sistema de arquivo Linux compatível em detalhes, sistema de arquivo EXT2.
11. Na ordem mostra a diferença de capacidades de segmentação e paginação, apresenta três versões diferentes do MTX no modelo protegido; mtx32.1 usa segmentação, mtx32.2 usa paginação estática e pmtx usa paginação dinâmica.
12. Capitulo 15 no SMP é uma variedade única. Mostra como configurar sistemas SMP-compativel e explana a sequencia inicial de sistemas SMP-compativel. Extende os mecanismos de sicronização de processo convencional, tal como sleep/wakeup e semaphores, ate ambiente de SMP suitavel. Apresenta uma metodologia geral para design do kernel SMP, e mostra como aplicar os princípios para adaptar o kernel pmtx para SMP. Então foca no design e usa algoritimos paralelos para SMP. Especificamente, usa algoritimos paralelo em gerenciamento de processo, gerenciamento de recurso, pipes e I/O buffers para melhorar ambos a concorrência e eficiência. Esse nova abordagem do SMP é novo e original. Compara kernel monolítico e microkernel baseado no SO design e demonstra um sistema hibrido, que incorpora a força de ambas abordagens. No decorrer do livro, usa-se sistemas simples para demonstrar os princípios de design e técnicas de implementações.
    1. CONTEÚDO DO LIVRO

Esse livro está organizado como asseguir, capitulo 2 abrange as fundamentações e segundo-plano que são necessários para estudar sistemas operacionais. Esses incluem operações da CPU, endereços real e virtual, operações de I/O, etapas de desenvolvimento de programa, motadores C com código assembly, imagem de execução de programa, funções de chamadas e pilha em uso em tempo-real. Tambem abrange alguma informação básica do sistema de arquivo EXT2.

Capitulo 3 abrange carregamento do sistema operacional. Discute o carregamento do SO em geral e descreve o processo de carregamento a partir de dispositivos real e virtual em detalhes. Esses incluem carregamento a partir do floppy disco, partições de hard disco, CD/DVD-ROM e USB drivers. Em vez de escrever um carregador customizado para carregar somente o MTX, mostra como desenvolver programas de carregadores para carregar outros sistemas operacionais, tal como Linux, a partir de uma variedade de dispositivos carregáveis.

Capitulo 4 inicia o desenvolvimento do kernel do MTX. Primeiro, usa um programa simples para introduzir o conceito de processo e demonstrar a técnica da mudança de contexto de processo, Então implementa um ambiente multitarefa para dar suporte a múltiplos processos. Extende o sistema multitarefa para suportar a criação do processo dinâmico, gerenciamento de processo e terminação de processo. Mostra como parar e continuar um processo, e extende stop/continue para operações sleep/wakeup para sicronização e explana o uso deles em um núcleo de SO. Então implementa a operações de wait para permitir processos a aguardar para terminação de processo filho. Complementarmente, também mostra como ajustar prioridades de processo para agendamento de processo baseado em prioridade. Esses levão a um simples kernel de SO para gerenciamento de processo. Em cada etapa, demonstra-se os princípios de design e técnica de implementação através de um completo simples sistema, que permite ao leitor testar e observar as operações internas de um núcleo de SO.