faculdade de tecnologia | departamento de engenharia elétrica

UnB

Faculdade de Tecnologia, Universidade de Brasília, Campus Universitário Darcy Ribeiro, Gleba A, Avenida L3 Norte, CEP 70.910-900, Brasília-DF
Caixa Postal 4386, fone +55 61 3307 2300, fone/fax +55 61 3273 8893, ftd@unb.br, www.ft.unb.br

Trabalho 1 – Sistemas Operacionais de Redes

Instruções

O trabalho poderá ser feito em grupos de até 4 integrantes e será desenvolvido em sala de aula, no ambiente de execução/programação indicado pelo professor. Cada grupo deverá organizar o trabalho da seguinte forma:

1. Código fonte C/C++ (baseado no código fonte disponibilizado para a turma no campusvirtual): O ambiente de compilação e execução deverá ser UNIX. Além disso, é preciso que seja criado um Makefile para compilação automática do projeto. Trabalhos que apresentem qualquer erro de compilação ou execução serão automaticamente descartados da avaliação e receberão nota ZERO. Trabalhos copiados (ou plagiados) também receberão nota ZERO.

UM INTEGRANTE DO GRUPO SERÁ SORTEADO PARA APRESENTAR O TRABALHO AO PROFESSOR E DEVERÁ SER CAPAZ DE RESPONDER A QUALQUER PERGUNTA EM NOME DO GRUPO, OU SEJA, RESPONSABILIZANDO-SE PELA NOTA DA EQUIPE.

Motivação & Objetivos

Os sistemas operacionais têm como principal meta abstrair a complexidade do hardware e gerenciar de forma eficiente os recursos limitados da máquina que o hospeda, tais como memória, discos e processadores. Este último, por exemplo, executa um Processo (ou Thread) por vez (no caso de máquinas "single-core"), dentre um conjunto de Processos e Threads que podem ser escalonados pelo SO. Com o advento de arquiteturas "multi-core", cada núcleo pode executar um Processo ou Thread em paralelo com os demais núcleos, contanto que as tarefas sejam independentes.

A divisão de tarefas entre os processos e threads pode ser feita de duas formas: **estática** e **dinâmica**. Na primeira, o programador, no ato da codificação, já decide quais partes devem ser executadas por cada processo e thread. Na segunda, as tarefas são dinamicamente associadas a cada processo e thread. Para tanto, os processos e threads trabalhadores podem "retirar" as tarefas de uma "sacola de tarefas", computá-las e devolver o resultado à "sacola", para que um processo junte e organize os resultados finais. Este modelo dinâmico é em geral melhor que o estático, em especial por acabar balanceando a carga de processamento entre todos as unidades de computação disponíveis.

O gerenciamento de processos e threads é fundamental para o uso eficiente dos processadores por diferentes programas. Sendo assim, é crucial que os alunos entendam e pratiquem os conceitos de criação, execução e administração de processos e threads.

O Algoritmo de Traçado de Raios Monte-Carlo

O algoritmo de traçado de raios pertence à classe dos algoritmos de iluminação global, muito usados em filmes de animação 3-D. É um algoritmo com alto custo computacional, pois produzir imagens com qualidade e realismo quase-fotográfico, podendo levar semanas ou meses para produzir uma sequência de

Faculdade de Tecnologia, Universidade de Brasília, Campus Universitário Darcy Ribeiro, Gleba A, Avenida L3 Norte, CEP 70.910-900, Brasília-DF
Caixa Postal 4386, fone +55 61 3307 2300, fone/fax +55 61 3273 8893, ftd@unb.br, www.ft.unb.br

animação curta. Para cada pixel da imagem original, o algoritmo simula o trajeto de todos os raios de luz presentes na cena 3-D a fim de compor a cor de cada pixel, como mostra a Figura 1 abaixo.

Figura 1: uso de traçado de raios Monte-Carlo para renderização de um modelo de estátua 3-D.

O Trabalho

Este trabalho consiste em produzir um sistema de renderização usando "Bag of Tasks" (sacola de tarefas), onde processos clientes poderão inserir tarefas (bloco da imagem a ser processado) enquanto servidores remotos computam as tarefas concorrentemente (possivelmente em paralelo). Em cada servidor, threads deverão ser criadas para utilizar todas as CPUs disponíveis. A Figura 2 apresenta uma **proposta** de arquitetura da solução. O código do algoritmo de traçado de raios será disponibilizado no campusvirtual e deverá ser usado, isto é, alterado, para criação da solução. **Atenção! Nenhum outro algoritmo será aceito. Não é permitido o uso de bibliotecas e funções que fujam do escopo da disciplina, ou seja, use somente o que foi apresentado em sala de aula. Use as máquinas virtuais do laboratório (ou outras apontadas pelo professor) para construir o ambiente apresentado na Figura 2. Não é permitido levar código fonte para casa, ou trazer código fonte de casa.**

Faculdade de Tecnologia, Universidade de Brasília, Campus Universitário Darcy Ribeiro, Gleba A, Avenida L3 Norte, CEP 70.910-900, Brasília-DF
Caixa Postal 4386, fone +55 61 3307 2300, fone/fax +55 61 3273 8893, ftd@unb.br, www.ft.unb.br

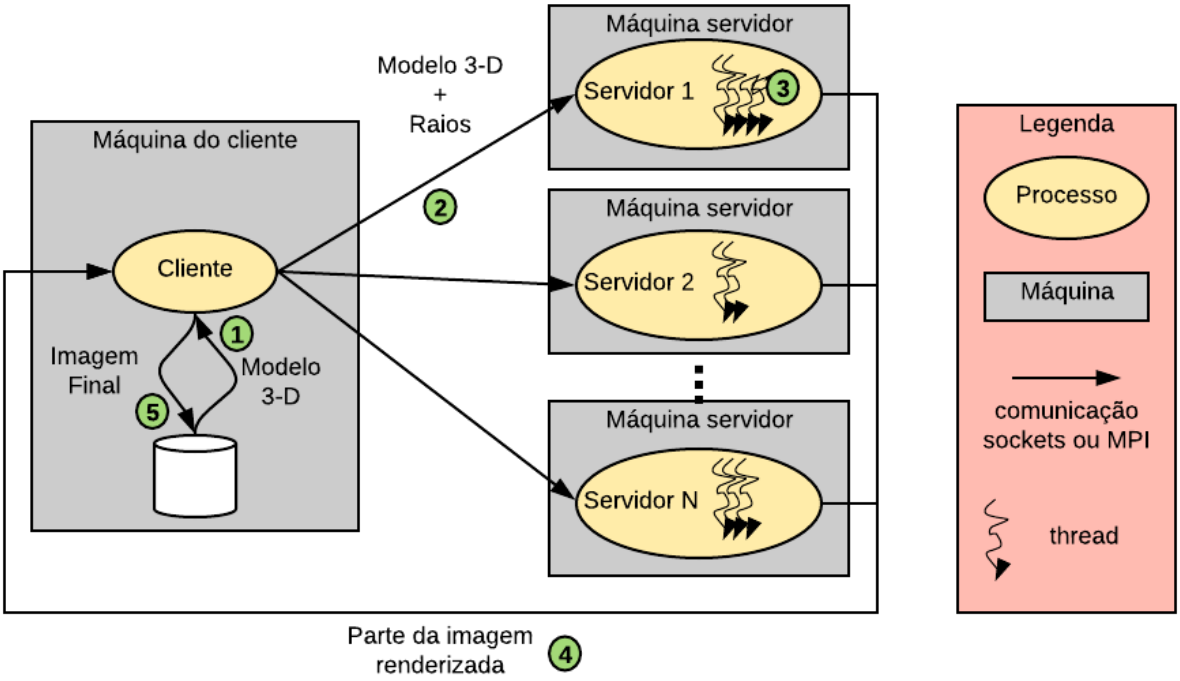

Figura 2: proposta de arquitetura de rendering-farm. Os números indicados em círculos verdes apresentam uma sugestão de fluxo de execução.

Pontuação (Nota)

O trabalho vale 10 e será pontuado em 3 partes:

1. 50% para o grupo que apresentar uma solução concorrente (com processos ou threads) para execução do algoritmo usando apenas uma máquina (sem comunicação com outras máquinas).
2. 75% para o grupo que apresentar uma solução concorrente **estática** em conformidade (ou semelhante) com a arquitetura proposta na Figura 2.
3. 100% para o grupo que apresentar uma solução concorrente **dinâmica ("bag of tasks")** em conformidade (ou semelhante) com a arquitetura proposta na Figura 3, isto é, usando um modelo de "sacola de tarefas".

Atenção: a implementação dos itens 1,2 e 3 não garante nota 10. A pontuação também leva em consideração a qualidade da solução (legibilidade, organização, etc.), a qualidade do relatório e a apresentação oral.

Faculdade de Tecnologia, Universidade de Brasília, Campus Universitário Darcy Ribeiro, Gleba A, Avenida L3 Norte, CEP 70.910-900, Brasília-DF
Caixa Postal 4386, fone +55 61 3307 2300, fone/fax +55 61 3273 8893, ftd@unb.br, www.ft.unb.br

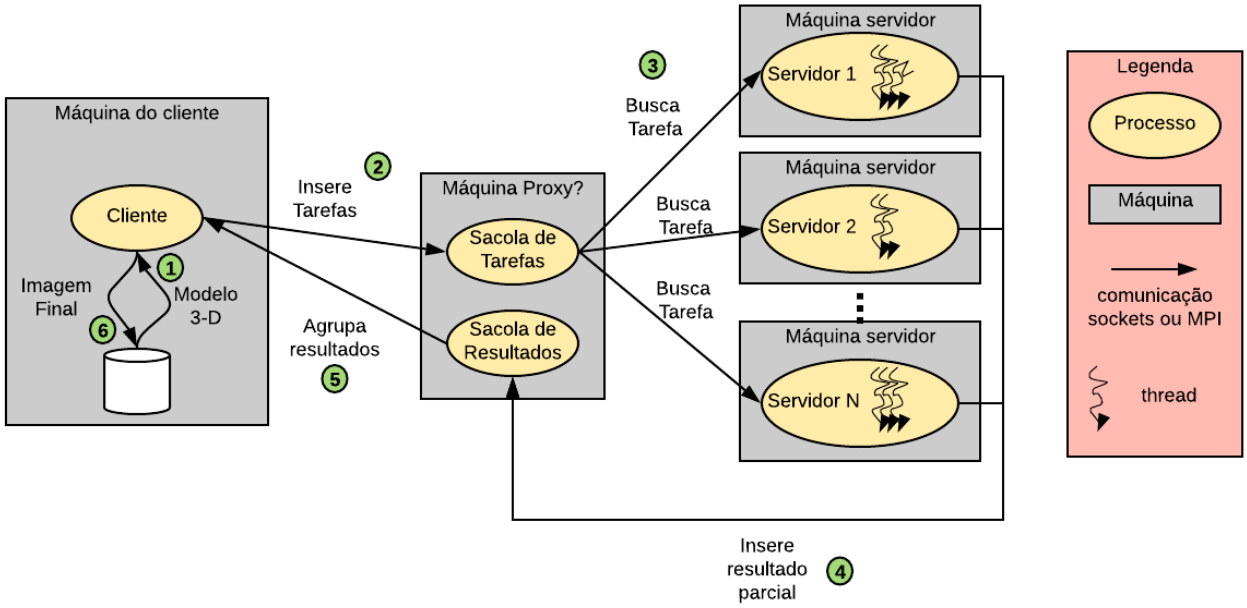

Figura 3: proposta de arquitetura "rendering-farm" com escalonamento dinâmico de tarefas por "sacola de tarefas" (Bag of Tasks).