Plano De Estudos para a Disciplina MAC 5701 - Tópicos em Ciência da Computação

Aluno - Andrei Goldchleger
Orientador - Prof. Fabio Kon

12 de setembro de 2002

1 Introdução

Este documento descreve as atividades de pesquisa que realizarei como parte da disciplina MAC 5701 - Tópicos em Ciência da Computação. O trabalho será realizado na área de sistemas distribuídos e posteriormente será integrado à minha dissertação de mestrado, que versará sobre a arquitetura de um sistema de computação em grade a ser desenvolvido no IME-USP, *InteGrade*.

1.1 Motivação

A computação atingiu um patamar surpreendente em termos da qualidade de recursos. Processadores que hoje equipam computadores pessoais possuem capacidade de processamento encontrada em supercomputadores há menos de uma década; as redes de computadores estão cada vez mais rápidas e disseminadas e o software evolui juntamente tornando-se cada vez mais dinâmico, configurável e modularizável.

Mesmo com os avanços em hardware, certos problemas ainda persistem. Alguns tipos específicos de usuários necessitam de grande poder computacional e não o possuem, ao passo que, em um instante qualquer, milhões de computadores pessoais estão ociosos ao redor do mundo. A Internet trouxe uma maneira inédita de conexão entre sistemas, mas certos recursos ainda não podem ser compartilhados e acabam sub-utilizados. Paradoxos que pareciam não possuir solução.

Visando resolver essas e outras questões, surge a idéia de Grades de Computação (*Computational Grids*). O objetivo é prover maneiras de interconectar computadores e recursos computacionais diversos para estender

a fronteira do que é possível realizar utilizando-se sistemas de computação. O nome deriva da rede de energia elétrica (*Power Grid*), e simboliza a esperança de que, quando bem estabelecidas, as grades permitirão acesso a recursos computacionais de maneira tão simples quanto utilizar a rede elétrica. Da mesma forma que podemos entrar em qualquer sala do mundo e ligar um aparelho na tomada, esperamos desenvolver uma infra-estrutura que nos permitirá obter um enorme poder computacional através de uma simples ligação à Grade.

2 InteGrade

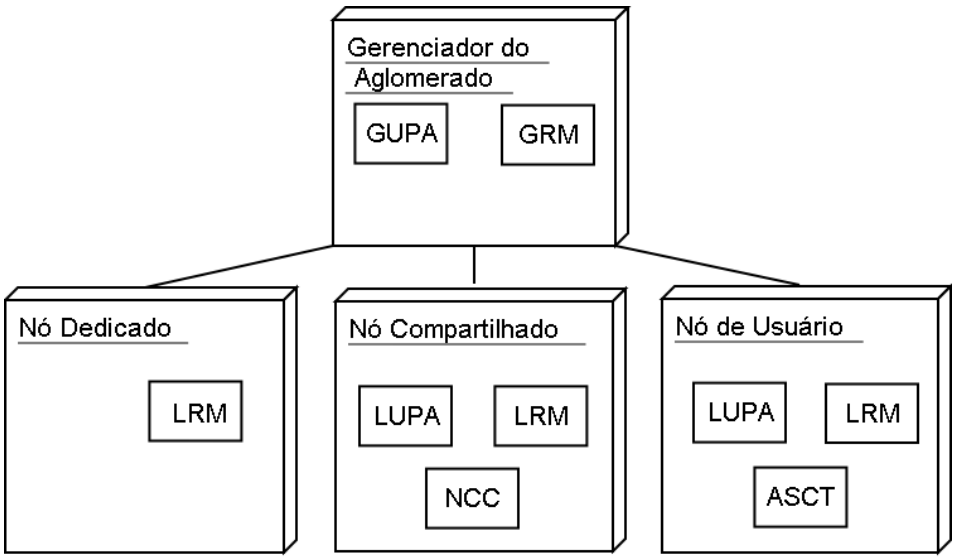
InteGrade é um projeto de computação em grade em início de desenvolvimento no IME-USP. Dentre os outros projetos similares em desenvolvimento, acreditamos que InteGrade possui ao menos os diferenciais listados abaixo.

- **Reutilização da base instalada**: é um princípio chave do projeto InteGrade. Tal característica também pode ser observada em poucos dos projetos existentes, porém o diferencial é que InteGrade considerará essa característica no desenvolvimento de sua arquitetura.
- **Utilização de tecnologias modernas de objetos distribuídos**: característica única do InteGrade, que utilizará CORBA na sua construção. Com isso, obtemos duas vantagens: (1) poderemos re-utilizar os inúmeros serviços já disponíveis na arquitetura CORBA e (2) será mais fácil a integração de outros serviços e aplicações à grade dado que esta utilizará uma tecnologia padrão de interconexão orientada a objetos.
- **Baixa sobrecarga**: o middleware do InteGrade vai priorizar o proprietário do recurso. Assim, é necessário que o sistema não consuma muitos recursos de maneira a não se tornar um problema para o usuário. Em particular, pretendemos utilizar ORBs CORBA que consumam poucos recursos.

2.1 Proposta Inicial de Arquitetura

A arquitetura e os requisitos de InteGrade ainda não estão completamente definidos, mas algumas características necessárias já estão evidentes. A seguir apresentamos a nossa proposta atual de arquitetura para um aglomerado do InteGrade. Futuramente, diversos aglomerados serão interligados,

dando origem a uma poderosa Grade de computação com potencial para abrigar milhões de máquinas, utilizando uma arquitetura hierárquica.



2.1.1 Descrição de cada nó

- **Nó Dedicado**: representa um recurso computacional dedicado, isto é, que não é utilizado como estação de trabalho.
- **Nó Compartilhado**: nó da Grade não dedicado, isto é, que é utilizado diretamente por um usuário, e que provê recursos computacionais para uso da Grade.
- **Nó de Usuário**: nó da Grade capaz de submeter aplicações para execução.
- **Gerenciador do Aglomerado**: nó da Grade responsável pelo monitoramento dos demais nós, escalonamento de aplicações, análise de uso dos nós do aglomerado e coleta de informações.

2.1.2 Descrição dos Módulos

- **LRM - *Local Resource Manager***: responsável por gerenciar recursos de uma máquina.

- **GRM - *Global Resource Manager***: responsável pelo gerenciamento de um aglomerado do InteGrade.
- **LUPA - *Local Usage Pattern Analyzer***: módulo que é executado em máquinas utilizadas como estações de trabalho. Seu objetivo é compilar um conjunto de informações que representem com certo grau de confiança o padrão de uso da máquina, com o objetivo de possibilitar melhorias de desempenho no sistema como um todo.
- **GUPA - *Global Usage Pattern Analyzer***: módulo presente apenas no nó gerenciador de um aglomerado, objetiva utilizar as informações fornecidas pelos diversos **LUPA** de maneira a consolidar padrões que reflitam as condições típicas de uso do aglomerado.
- **NCC - *Node Control Center***: este módulo permite que o proprietário de uma estação de trabalho configure as condições de utilização de seus recursos pela Grade.
- **ASCT - *Application Submission and Control Tool***: módulo que permitirá que usuários da Grade submetam aplicações para execução, monitorem o estado das execuções e recebam resultados.

3 Plano de Trabalho

Meu objetivo será implementar parte dos módulos responsáveis pelo gerenciamento de recursos em um aglomerado InteGrade, a saber: LRM e GRM. A funcionalidade a ser implementada é o protocolo de atualização dinâmica das quantidades dos recursos disponíveis em cada nó de um aglomerado InteGrade. Essa tarefa será derivada de um trabalho já existente em nosso grupo de pesquisa, porém implicará em re-escrever o código fonte com adaptações para o InteGrade, uma vez que o código atual atende às necessidades de outro sistema. Outra motivação para tal tarefa é a necessidade de mantermos boa parte de nossa implementação compacta para não causar sobrecarga perceptível nas máquinas clientes. Assim, necessitamos desvincular a funcionalidade desejada da base de código atual, que está voltada às necessidades de outro sistema.