

5

Estudo de Caso: CSBase

O CSBase [15] é um *framework* que permite o desenvolvimento de ambientes customizados de computação em grade. Ele foi criado para permitir o gerenciamento de recursos e a execução de algoritmos em um ambiente computacional distribuído e heterogêneo. Entre suas principais funcionalidades temos o gerenciamento de usuários, a monitoração de plataformas de execução de algoritmos, a execução remota de algoritmos e o suporte ao trabalho cooperativo entre seus usuários.

No CSBase, as aplicações que executam no cliente utilizam os serviços disponíveis no servidor principal para ter acesso e gerenciar os recursos computacionais distribuídos. Um desses serviços é o de execução remota de algoritmos que, nesse contexto, são programas executáveis implementados em qualquer linguagem e que admitem parâmetros de entrada, geram uma saída e não possuem nenhum tipo de interação com o usuário durante sua execução.

Uma característica importante na arquitetura do CSBase é a interoperabilidade entre o servidor principal do sistema e os servidores executando nas diversas máquinas de execução de algoritmos. Independente da linguagem de implementação dos algoritmos e da máquina onde esses algoritmos serão executados, o CSBase oferece uma interface padrão para execução remota e monitoração das máquinas do ambiente.

Originalmente, o CSBase dependia de um sistema de arquivos compartilhado para permitir o acesso aos dados. Ao executar um algoritmo, os binários do algoritmo e os arquivos de dados eram acessados via NFS e processados. Como discutido anteriormente, esse procedimento limitava o uso do CSBase, fazendo com que apenas máquinas onde o NFS estava configurado pudessem ser utilizadas para a computação.

Atualmente, o GridFS está sendo usado no CSBase para permitir outros cenários de uso. Em vez de ter o NFS como único meio de compartilhamento

de dados, o sistema é capaz de usar o GridFS para verificar se uma máquina de execução possui acesso a todos os arquivos necessários para executar um algoritmo e, caso algum arquivo não esteja disponível, um procedimento automático transfere os arquivos ausentes. Dessa forma, o ambiente para a execução do algoritmo é construído sob demanda e, no momento que a execução é concluída, um outro procedimento sincroniza os dados gerados para a área de projetos do CSBase. Assim, o GridFS permite um uso mais amplo do CSBase, fazendo com que máquinas sem acesso NFS possam ser utilizadas.

5.1

Arquitetura

A figura 5.1 apresenta uma visão geral da arquitetura do CSBase. As caixas preenchidas com uma tonalidade acinzentada indicam o protocolo de comunicação utilizado pelos diversos módulos do sistema. Por exemplo, o Desktop Web se comunica com o Servidor Principal por meio de uma interface RMI, enquanto CORBA é utilizado na comunicação entre o Servidor Principal, o Repositório de Dados e os Nós de Execução.

O GridFS foi utilizado na implementação do CSFS, o servidor de arquivos presente no CSBase. Como podemos observar na figura, o CSFS é utilizado juntamente ao NFS para permitir que as máquinas de execução e o Servidor Principal tenham acesso aos arquivos armazenados no repositório de dados, que engloba o repositório de algoritmos, a base de usuários e a área de projetos, que mantém os arquivos relacionados aos projetos gerenciados pelo sistema. A seguir entramos em mais detalhes sobre os principais componentes dessa arquitetura.

5.1.1

Desktop Web

O *Desktop Web* é a interface com o usuário, onde este acessa os recursos computacionais distribuídos como se estivesse trabalhando localmente. Utilizando um navegador, os usuários podem criar suas áreas de trabalho, monitorar as máquinas da grade, instalar algoritmos, executar aplicações instaladas no *desktop*, comandar a execução remota de algoritmos e acompanhar os processos nas máquinas do ambiente.

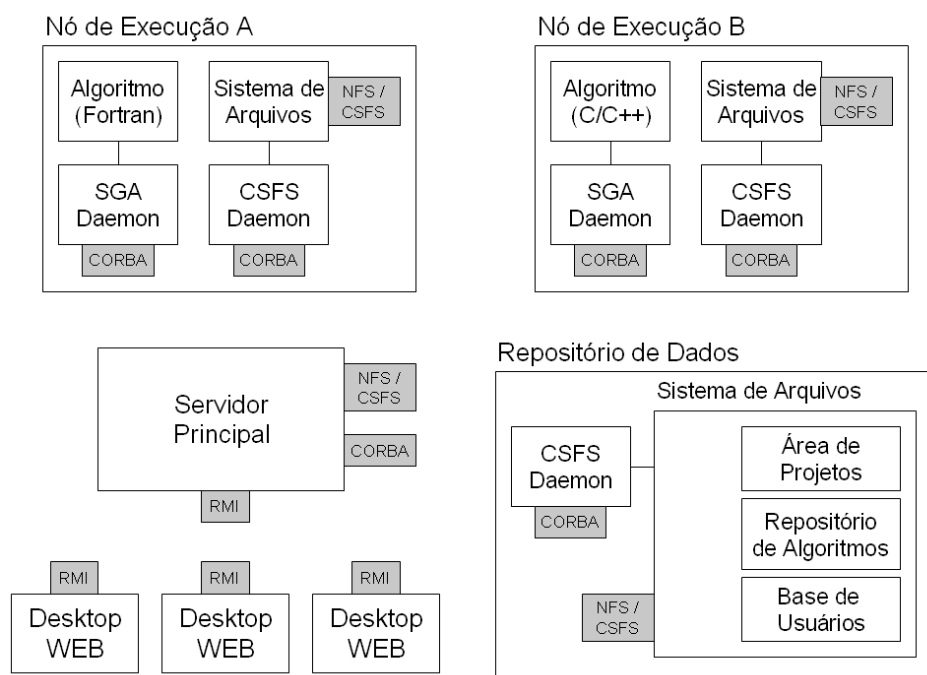


Figura 5.1: Arquitetura do CSBase

O *desktop web* possui um componente que monitora a comunicação com o servidor principal. Quando essa comunicação é perdida, o *desktop web* informa ao usuário e entra num estado de monitoração até que a comunicação se restabeleça. Dessa forma, o *desktop* não precisa ser reiniciado devido, por exemplo, a uma falha no servidor principal. Além disso, o *desktop* implementa um mecanismo de observadores remotos que mantém os dados dos projetos consistentes entre os diversos *desktops* que acessam os projetos simultaneamente.

5.1.2

Repositório de Dados

O repositório de dados é uma área central de armazenamento, onde estão armazenados os projetos dos usuários, a base de usuários do sistema e o repositório de algoritmos. A base de usuários é composta por arquivos com os dados dos usuários e suas permissões. A área de projetos possui um diretório específico para armazenar os arquivos relacionados a um determinado projeto. O repositório de algoritmos é uma outra estrutura hierárquica em que cada algoritmo é representado por um diretório, contendo suas diversas versões e configurações.

Para que um algoritmo possa ser executado, a máquina de execução

precisa ter acesso ao binário do algoritmo e aos dados do projeto. Esse acesso pode ser dado via NFS, onde as máquinas acessam o arquivo remotamente, ou pela utilização do CSFS Daemon, presente nos nós de execução e no repositório de dados. No caso do CSFS Daemon, em vez de termos um acesso remoto aos arquivos, uma cópia dos arquivos necessários é realizada entre o repositório de dados e a máquina de execução.

5.1.3

Servidor Principal

O Servidor Principal, ou o Servidor do Sistema Integrador (SSI), faz o papel de intermediário entre as requisições dos *desktops web*, o repositório de dados e as máquinas para execução dos algoritmos. Pelo uso de diversos protocolos de comunicação, tais como RMI, CORBA, e o NFS/CSFS, ele faz a orquestração de todas as atividades no CSBase, permitindo que os usuários tenham acesso aos recursos disponíveis. O SSI está desenvolvido em Java e é composto por um conjunto de serviços que implementam as diversas funcionalidades disponíveis em uma instância do CSBase.

Uma grade de computadores pode ser dividida em regiões, um SSI central fica responsável pelas tarefas de administração e pelo repositório principal das informações administrativas (usuários e permissões). Sob todos os demais aspectos, os servidores locais são equivalentes funcionalmente ao SSI central. Cada SSI local possui uma réplica da base de usuários, atualizada automaticamente pelo sistema. A replicação da base de usuários nas diferentes regiões permite a autenticação de usuários que se conectam a um SSI local mesmo quando a conexão com o servidor central não está disponível.

5.1.4

Servidor de Gerência de Algoritmos

O Servidor de Gerência de Algoritmos (SGA) é um *daemon* instalado nas máquinas de execução de algoritmos. Ele permite que o SSI comande a execução de algoritmos, assim como monitore o estado das máquinas e dos processos disparados a partir do SGA. Toda interação entre o SSI e o SGA, implementado em Lua [28] e C++, é realizada através de uma interface CORBA.

Ao ser ativado, o SGA se registra junto ao SSI de sua região e fica disponível aos clientes que se conectam a tal SSI. Quando um cliente solicita a execução de um algoritmo através do SSI, este passa a monitorar o processo até a sua conclusão (sucesso, falha ou interrupção). Os SGAs e os processos dos algoritmos disparados nas máquinas de execução permanecem ativos mesmo quando a conexão com o SSI é interrompida. Quando a conexão é reativada, o SGA se registra novamente no SSI. Periodicamente, o SSI solicita aos SGAs informações sobre o estado de sua máquina hospedeira. Essas informações podem ser exibidas ao usuário para que ele escolha a máquina onde o algoritmo deve ser executado ou podem ser utilizadas pelo SSI na seleção automática da máquina para a execução de um determinado algoritmo.

5.1.5

Servidor de Arquivos

O GridFS foi usado como servidor de arquivos no CSBase. Os serviços no SSI podem acessar um sistema de arquivos remoto utilizando a interface CORBA exportada pelo GridFS. O processo do servidor é disparado automaticamente pelo SGA e, dessa forma, para cada máquina onde um SGA está disponível, o servidor do GridFS também está disponível e pode ser utilizado para preparar o ambiente de execução para os algoritmos.

Além da funcionalidade de transferência dos arquivos, o GridFS permite que um escalonador de processos no CSBase utilize a informação sobre as taxas de transferência entre os diversos nós como parâmetro para a escolha automática do local para a execução de um algoritmo.

5.2

Serviços do CSBASE

O SSI exporta uma série de serviços remotamente, viabilizando o controle e o gerenciamento do ambiente da grade. Os serviços exportam uma interface RMI que permite que o *Desktop Web* acesse suas funcionalidades. O serviço de administração, por exemplo, é responsável por permitir o gerenciamento da base de usuários do sistema. O serviço de projetos permite que as aplicações desenvolvidas para o Desktop Web acessem os arquivos da área de projetos, permitindo a visualização dos arquivos. Além desses serviços, temos o serviço

de envio de e-mails, o serviço HTTP, entre outros. Nas próximas subseções, vamos descrever os principais serviços existentes no CSBase que possuem relação com o GridFS.

SGAService

O SGAService é responsável por fazer a ponte entre os clientes e os SGAs executando nos nós. Ele mantém a lista de SGAs cadastrados junto ao SSI, bem como é responsável por fornecer as características de cada um dos nós de execução e o seu estado corrente, provendo informações sobre uso de CPU, memória, e comandos em execução. O serviço também é capaz de receber as solicitações via RMI para a execução e término de um comando e realiza as chamadas CORBA aos nós para efetivar a solicitação do cliente.

ProjectService

O serviço de projetos é responsável pela criação e manutenção dos projetos dos usuários. O serviço disponibiliza os arquivos para o cliente utilizando um protocolo baseado em soquetes. As funções de mais alto nível estão disponíveis via RMI para o cliente. O serviço de projetos persiste as informações dos projetos e arquivos diretamente no disco local, ou em uma área mapeada via um cliente NFS. Para isso ele faz uso das classes básicas de IO de Java.

CSFSService

O CSFSService foi criado para auxiliar o SGAService na execução dos comandos remotos. Ele é responsável por verificar a disponibilidade dos dados e binários na máquina de execução. Caso os binários ou dados não sejam encontrados, o CSFSService utiliza os servidores do GridFS para copiar os arquivos do repositório de dados para a máquina de execução. Quando o ambiente está pronto para a execução, o SGAService é informado e pode disparar o comando na máquina. Após o término do comando, o SGAService solicita ao CSFSService que sincronize o estado da máquina de execução e o repositório de dados, ou seja, que copie os arquivos que foram gerados ou modificados após a execução do algoritmo.

5.3

Considerações sobre o uso do GridFS no CSBase

O servidor de arquivos presente na arquitetura do CSBase é exatamente o GridFS. Ele foi integrado ao SGA de forma que a execução do servidor fosse realizada de forma automática ao se iniciar a execução de um SGA. Mantendo o servidor ativo em cada nó de execução, seus sistemas de arquivos locais ficam amplamente disponíveis para as demais máquinas que compõem a rede controlada pelo CSBase. Dessa forma, é possível a troca de arquivos entre o repositório de dados e os nós de execução de uma forma simplificada e sem a dependência de nenhum outro mecanismo.

A implementação do CSFSService foi realizada em conjunto com a disponibilização dos servidores de arquivos nos nós. O CSFSService mantém uma referência para o servidor CSFS executando no repositório de dados, e pode criar uma referência dinamicamente para o nó de execução utilizando o seu nome para a construção do endereço `corbaloc` para o servidor executando no nó. No momento em que o SGAService deseja executar um comando em algum nó, ele solicita ao CSFSService que prepare o ambiente de execução. Assim, o CSFSService copia os arquivos necessários a partir do repositório de dados e cria um *sandbox* contendo os dados necessários para a execução do algoritmo. Um *snapshot* da árvore copiada é realizado antes da execução do algoritmo, de forma que após a execução o CSFSService seja capaz de identificar quais arquivos foram criados ou modificados pela execução do algoritmo. Nesse momento, o CSFSService é responsável por atualizar o repositório de dados central em função do efeito da execução do algoritmo na máquina de execução.

O CSFSService faz um uso extensivo da API de programação do GridFS, especialmente para realizar a cópia de arquivos e a recuperação da árvore de arquivos sob um determinado caminho. Além da estrutura hierárquica, funções para a recuperação da data de modificação do arquivo remoto são essenciais para que a avaliação do efeito de um algoritmo sobre a árvore de dados seja realizada. Dessa forma, o GridFS foi implantado no CSBase e permite a transferência de dados e a execução de algoritmos nas diversas máquinas gerenciadas pelo *framework*, inclusive nas máquinas sem acesso NFS ao repositório de dados.