



Manual do Administrador CSGrid

Tecgraf/PUC-Rio
csgrid@tecgraf.puc-rio.br

Resumo

O objetivo deste documento é registrar em um único local todas as informações necessárias para instalar, executar e realizar as tarefas administrativas do CSGrid. Este documento está organizado da seguinte forma: na Seção 1 especificamos todos os requisitos exigidos do ambiente computacional sobre o qual o servidor executará. Em seguida, nas Seções 2 e 3, nós detalhamos os procedimentos de configuração, instalação e atualização do servidor do sistema. Na Seção 4 explicamos como iniciar os módulos do sistema. Na Seção 5 nós descrevemos as tarefas de administração. Por fim, na Seção 6, enumeramos alguns problemas que podem ocorrer durante a operação do sistema e detalhamos possíveis causas, formas de verificação e suas possíveis soluções.

Sumário

1	Requisitos de Hardware	4
1.1	Arquitetura	4
1.2	Premissas Básicas	5
1.3	Fatores Relevantes	5
1.4	Características dos Componentes	5
1.5	Escalabilidade	7
2	Configuração do Ambiente	8
2.1	Resumo de Requisitos do Servidor Central	8
2.1.1	Variáveis de Ambiente	9
2.1.2	Usuário Executor do Servidor	9
2.1.3	Diretório de Instalação	10
2.1.4	Java	10
2.1.5	Disponibilidade da porta 8080	10
2.2	Resumo de Requisitos dos SGAs	11
2.3	Resumo de Requisitos do Cliente	12
3	Instalação do Sistema	13
3.1	Descompactar o arquivo de instalação	13
3.2	Criar os diretórios src e config	13

3.3	Executar o instalador	14
3.4	Configuração dos Servidores SGAs	16
3.5	Conexão com o OpenBus	18
3.6	Atualização do Servidor	19
4	Execução do Servidor	20
4.1	Execução de Servidores SGAs adicionais	20
4.2	Inicialização Automática	21
4.3	Verificação do Estado do Servidor	23
5	Tarefas de Administração	25
5.1	Criar Usuários	25
5.2	Criar Grupos de Usuários	26
5.3	Criar Perfis de Usuários	27
5.4	Criar Plataformas de Execução de Algoritmos	28
5.5	Criar Permissões	29
5.5.1	Criar Permissão de Utilização de Servidores	30
5.5.2	Criar Permissão de Execução de Algoritmos	31
5.5.3	Criar Permissão de Gerenciamento de Algoritmos	33
5.5.4	Criar Permissão de Gerenciamento de Servidores de Execução de Algoritmos	34
6	Problemas e Soluções	36

1 Requisitos de Hardware

1.1 Arquitetura

A Figura 1 reproduz a visão geral dos componentes que compõem a arquitetura do CSGrid. O servidor do CSGrid (Servidor Central) que você vai instalar é acessível aos usuários através de um navegador. Para executar os programas (que chamamos algoritmos), precisamos de outro tipo de servidor chamado SGA (Servidor de Gerência de Algoritmos). O SGA é responsável pela monitoração das máquinas e pela execução dos algoritmos. O servidor é uma aplicação Java que roda em background (não possui GUI) e se comunica com os SGAs, através de CORBA. Cada usuário pode executar um cliente (Desktop Web) CSGrid através de navegadores. O cliente também é Java e se comunica apenas com o servidor, através de RMI.

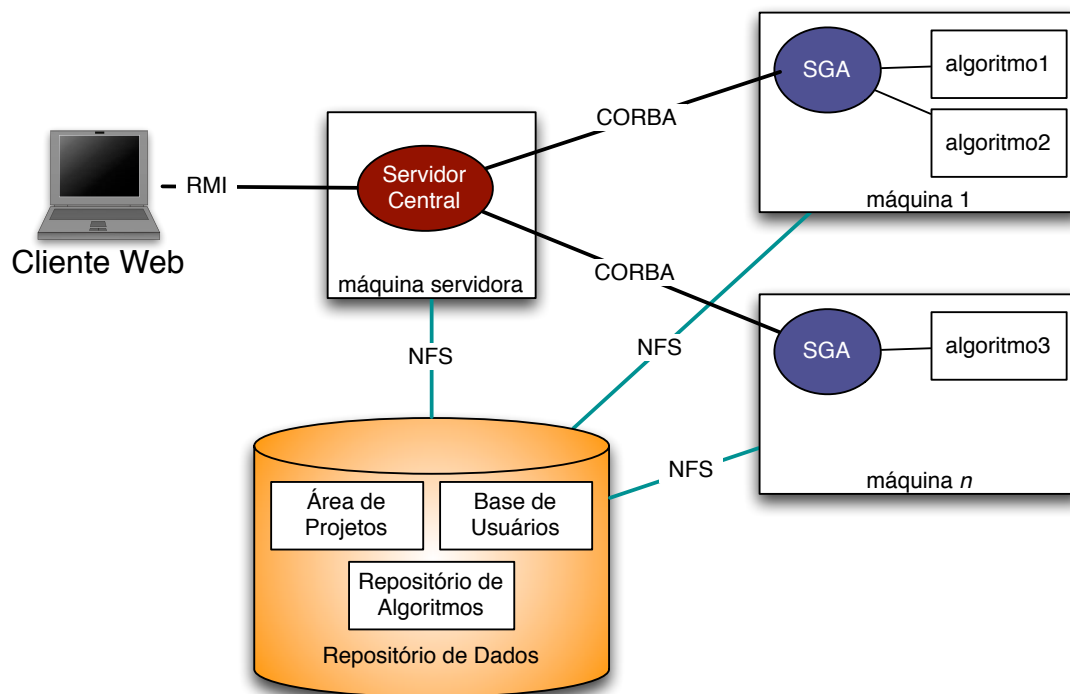


Figura 1: Arquitetura do CSGrid

1.2 Premissas Básicas

1. Todos os componentes (Servidor Central, SGA e Repositório de Dados) devem compartilhar uma mesma área de armazenamento de arquivos. No exemplo acima, este compartilhamento é feito via NFS (sistema de arquivos em rede do UNIX).
2. O Repositório de Dados (RD) tem que possuir espaço suficiente para armazenamento de:
 - (a) áreas de projeto dos usuários
 - (b) executáveis dos Algoritmos
 - (c) estruturas de controle (identificação dos usuários, permissões etc.)

1.3 Fatores Relevantes

No caso do CSGrid, os fatores relevantes para dimensionamento do parque de máquinas são:

- número de usuários (atual e projeção).
- número médio de acessos simultâneos (atual e projeção).
- ocupação média do espaço em disco.
- requisitos de disponibilidade (p.ex. disponibilidade 24x7 implica em redundância em alguns nós, configuração de RAID para armazenamento etc.).

1.4 Características dos Componentes

A Tabela 1 ilustra o perfil de cada componente com relação ao uso de memória, CPU, disco e rede. A definição de alto, médio e baixo depende bastante do contexto. De qualquer forma, a Tabela 2 pode ser usada como referência.

Observações:

- Um processador Intel com tecnologia hyper-threading é contabilizado como duas CPUs.
- Requisitos de disponibilidade e tolerância a falhas invariavelmente exigem múltiplos discos rígidos para armazenar os dados de forma redundante (RAID).

componente	memória	CPU	disco	rede
Servidor Central	alto	médio	baixo	alto
Repositório de Dados	médio	médio	alto	alto
Servidores de Execução (SGA)	alto	alto	baixo	alto

Tabela 1: Uso típico dos recursos de hardware

perfil	memória	CPU	disco	rede
alto	$\geq 4\text{GB}$	servidores (Intel, Sun, IBM etc.) com 4 ou mais CPUs	$\geq 120\text{GB}$ (SCSI)	1 Gigabit fibra ótica 100Mbps dedicada
médio	1GB	CPU Dual Core	80GB (IDE)	100Mbps
baixo	512MB	Pentium4 (1 CPU)	40GB (IDE)	100Mbps

Tabela 2: Referência para requisitos de hardware

A rigor, todos os componentes podem ser instalados em uma única máquina; porém, o uso típico consiste de:

- um nó para Servidor Central
- um nó para o Repositório de Dados
- um ou mais Servidores de Execução (SGAs)

Por ser desenvolvido em Java, o CSGrid é portátil e pode ser executado em sistemas operacionais diversos tais como Linux, Solaris, AIX e MS Windows. A Tabela 3 exemplifica um setup recomendado.

componente	sistema operacional
Servidor Central	UNIX (Linux / Solaris / AIX)
Repositório de Dados	UNIX (Linux / Solaris / AIX), compartilhamento de arquivos via NFS
Servidores de Execução	- MS Windows - UNIX (Linux / Solaris / AIX)

Tabela 3: Sistemas operacionais

Observações: a troca de arquivos entre nós MS Windows e UNIX deve ser feita via SAMBA.



1.5 Escalabilidade

Os Servidores de Execução (SGAs) podem e normalmente são hospedados em múltiplos servidores (inclusive envolvendo diferentes sistemas operacionais e configurações de hardware). O reconhecimento de novos servidores em ambos os componentes por parte do sistema é uma tarefa simples, exigindo essencialmente a atualização de arquivos de configuração. No caso do Repositório de Dados em particular, normalmente as mudanças são completamente transparentes ao sistema.

2 Configuração do Ambiente

Nessa seção descrevemos os procedimentos iniciais de configuração do ambiente e destacamos os requisitos que o ambiente computacional deve atender.

O CSGrid opera em um ambiente distribuído. Nesse ambiente existe o lado cliente, onde é executado o desktop web, e o lado servidor, que armazena o repositório de dados e onde os componentes Servidor Central e SGA são executados. Todos os componentes que estão do lado servidor devem compartilhar uma mesma área de armazenamento. Esse compartilhamento é feito atualmente via NFS para sistemas de arquivos em rede do UNIX. Para permitir o acesso por parte de componentes executando sobre MS Windows, o compartilhamento é feito usando o SAMBA [<http://us4.samba.org/samba/>] tornando, nesse caso, o uso do sistema operacional UNIX (Linux / Solaris / AIX) obrigatório no servidor. O SAMBA pode não ser necessário se o sistema for instalado em um *filer* capaz de exportar um mesmo sistema de arquivos de forma nativa para Linux e Windows.

Segue a lista de pré-requisitos para o servidor e para o cliente.

2.1 Resumo de Requisitos do Servidor Central

De forma resumida, os requisitos de software estão listados a seguir:

- Linux 2.6 ou Solaris 10 ou MS Windows
- JDK 1.6 (JDK 6u7 ou mais nova)
A instalação do JRE não é suficiente, é realmente necessário o SDK. Além disso, a última versão do Java deve ser usada sempre que possível, devido às correções e otimizações. Se o servidor for 64bits ou possuir mais do que 4GB de memória, é importante que a versão 64bits seja instalada.
- ksh (pacote "ksh" ou "pdksh")
- gtar (Linux da RedHat ou derivadas destas já possuem o executável gtar)
O arquivo de distribuição é binário, mas contém dados no formato tar.gz. Os caminhos contidos nesse arquivo excedem 100 caracteres, que é um limite estabelecido pelo programa **tar** do Solaris. Sendo assim, recomendamos que o programa **gtar** seja utilizado para abrir este arquivo. No caso de máquinas que possuam o programa **tar** compatível com GNU, basta definir um link de **tar** para **gtar**.

- Servidor Apache (pacote "httpd"), opcional

2.1.1 Variáveis de Ambiente

O administrador deve assegurar que o ambiente está configurado corretamente com o hostname e o ip do servidor.

Os seguintes comandos devem ser usados para consultar a configuração corrente do ambiente:

- `hostname` Deve mostrar o nome da máquina. Deve ser possível executar o comando `ping <nome>` a partir de outra máquina da mesma rede
- `hostname -i` Deve mostrar o ip da máquina. Deve ser possível executar o comando `ping <ip>` a partir de outra máquina da mesma rede

2.1.2 Usuário Executor do Servidor

A primeira ação a ser realizada para efetuar a instalação do Servidor Central é criar e configurar o usuário UNIX que executará o sistema. Sugerimos que seja criado um usuário **csgrid**. Esse usuário precisa existir no NIS para que todos os computadores utilizados pelo sistema o reconheçam e possam montar sua área home. Uma área de 40 Gb é suficiente para manter, com folga, os arquivos que compõem a instalação do sistema.

É preciso criar um grupo específico para esse usuário. Vamos considerar que seja criado um grupo **csgrid**. Esse deve ser o grupo primário do usuário **csgrid**, de forma que todos os arquivos criados por ele pertençam a esse grupo. A área home do usuário que executa o sistema deve ser configurada para permitir acessos de leitura e escrita para o próprio usuário e para seu grupo. Nenhum outro usuário deve ter acesso à essa área, nem mesmo acesso de leitura.

O usuário **csgrid** precisa ser reconhecido e ter seu home montado pelo computador que executa o servidor e pelos computadores que executam os **Servidores de Gerência de Algoritmos (SGA)**.

2.1.3 Diretório de Instalação

Na área *home* do usuário **csgrid**, crie um diretório para conter a instalação, por exemplo **cs-grid**. Dentro deste diretório, o procedimento de instalação expandirá os diretórios e arquivos necessários.

2.1.4 Java

O computador que executa o sistema precisa ter o Java SDK versão 1.6 instalado. Considerando que o SDK está instalado em `/usr/java`, os passos a seguir exemplificam um procedimento de atualização do PATH com versão do Java necessária.

- criar neste diretório um link simbólico chamado "jdk" para o diretório de instalação do java. Por exemplo: `jdk -> jdk1.6.0_05/`
- definir a variável de ambiente `JAVA_HOME` e colocá-la no o PATH. Normalmente, isso é feito acrescentando no arquivo `/.profile` do usuário criado, o seguinte trecho: (Atenção: a sintaxe e o nome do arquivo dependem do tipo de shell usado)

```
export JAVA_HOME = /usr/java/jdk
export PATH=${JAVA_HOME}/bin:${PATH}
```

2.1.5 Disponibilidade da porta 8080

Para que a aplicação cliente seja iniciada usando um servidor HTTP, a instalação disponibiliza o tomcat já configurado para receber requisições na porta 8080. Usando um browser, o cliente é iniciado usando o Java Web Start. A url para acessar a cliente CSGrid instalado no Tomcat é:

```
http://<server-name>:8080/csgrid/init
```

onde server-name é o nome do servidor ou o seu IP.

É importante assegurar que a porta 8080 não está sendo usada por nenhuma outra aplicação.

Para que o sistema possa ser acessado pelos usuários através de uma URL simples, ao invés da URL no Tomcat, é necessário que seja instalado um servidor Apache. Nesse caso, é também necessário acesso de escrita ao diretório do Apache, por exemplo `/var/www/html/csgrid`.

2.2 Resumo de Requisitos dos SGAs

Para a instalação dos SGAs, é preciso definir qual a plataforma de cada máquina. Para isso, a conta do usuário `csgrid` deve conter a definição da variável de ambiente `TEC_UNAME`, conforme o código da Figura 2.2¹. Esta variável depende da versão do sistema, da versão do compilador `gcc` instalado na máquina e de sua arquitetura.

O administrador deve indicar à equipe de desenvolvimento quais as plataformas onde o sistema deve executar SGAs, para que os binários necessários sejam gerados.

```
setenv TEC_SYSNAME `uname -s`
setenv TEC_SYSVERSION `uname -r|cut -f1 -d.`
setenv TEC_SYSMINOR `uname -r|cut -f2 -d.`
setenv TEC_UNAME "${TEC_SYSNAME}${TEC_SYSVERSION}${TEC_SYSMINOR}"

set GCCVER=`/usr/bin/gcc -dumpversion|cut -f1 -d.`
if ($GCCVER == '4') then
    setenv TEC_UNAME $TEC_UNAME'g4'
endif
unset GCCVER

setenv TEC_SYSARCH `uname -m`
if ($TEC_SYSARCH == 'x86_64') then
    setenv TEC_UNAME $TEC_UNAME'_64'
endif
echo $TEC_UNAME
```

Figura 2: Definição da variável de ambiente `TEC_UNAME`.

De forma resumida, os requisitos de software estão listados a seguir:

- `cygwin` (<http://www.cygwin.com>), para SGA na plataforma Windows. Após a instalação, o `cygwin` deve ser incluído no path do sistema.
- `ksh` (pacote "`ksh`" ou "`pkgsh`")

Para os SGAs em plataforma Windows terem acesso ao repositório de dados e de algoritmos de um servidor CSGrid instalado em Linux, o compartilhamento SAMBA configurado no servidor Linux deve ser mapeado como unidade de rede, usando as credenciais de acesso correspondentes (usuário **csgrid**). A identificação da unidade de rede pode ser escolhida arbitrariamente

¹Código desenvolvido para o shell `csh`.

(p.ex. T : \), mas o mapeamento deve ser configurado para que ela seja reconectada a cada logon. O SGA (Servidor de Gerência de Algoritmos) será executado nesta unidade de rede. O mapeamento pode ser feito via interface gráfica (Windows Explorer) ou via comando `net use`².

2.3 Resumo de Requisitos do Cliente

Qualquer sistema operacional suportado pelo Java Runtime Environment (JRE). Referência: <http://java.sun.com/javase/6/webnotes/install/system-configurations.html>.

Internet Explorer ou outro navegador (Netscape, Mozilla, Firefox ou Opera) compatível com plug-in do Java JRE 1.6 com a opção de instalar o plug-in do Java no navegador.

²http://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/net_use.mspx?mfr=true

3 Instalação do Sistema

Com o usuário do sistema criado e o ambiente configurado, passamos à instalação do sistema, que é feita a partir de um arquivo de distribuição. Este arquivo é sempre gerado pela equipe de desenvolvimento e é construído a partir da instalação final de teste do sistema, após todos os testes terem sido realizados.

3.1 Descompactar o arquivo de instalação

Copie para o diretório criado na seção 2.1.3, o arquivo de instalação fornecido pela equipe de desenvolvimento. Descompacte usando o seguinte comando:

```
tar -xf csgridInstall.tgz
```

supondo que **csgridInstall.tgz** é o arquivo de distribuição do CSGrid.

Ao descompactar, será criado um diretório **install** contendo:

- O executável, com extensão `.bin`, por exemplo, `CG-v1_00_00-2009_09_28.bin`
- O arquivo de propriedades de validação, `Validations.properties`
- O arquivo `sgad-cnf.lua` que serve de base para a configuração dos SGAs
- O arquivo `Readme` com instruções básicas para instalação

3.2 Criar os diretórios `src` e `config`

Dentro do diretório de instalação (ver 2.1.3), crie os diretórios **src** e **config**. Supondo que o diretório de instalação seja **csgrid**, você passará a ter os diretórios **csgrid/src** e **csgrid/config** que serão usados durante a instalação do sistema.

3.3 Executar o instalador

O instalador executará um script que fará a extração dos arquivos necessários e ajustará algumas propriedades do sistema. Atenção: todos os arquivos contidos no diretório **csgrid/src** serão removidos pelo instalador.

Usando o usuário **csgrid**, entre no diretório **src** criado no passo anterior e execute o comando:

```
../install/CG-v1_00_00-2009_09_28.bin -p ../install/Validations.properties
```

onde **CG-v1_00_00-2009_09_28.bin** é o nome do binário fornecido, de acordo com a versão do CSGrid.

O processo de instalação solicita algumas opções de configuração. Para usar o padrão sugerido, tecla <ENTER>. Para alterar a opção de configuração, preencha e tecla <ENTER>. Ao fim da instalação, o script indica quais arquivos de configuração foram alterados e podem ser conferidos e, se necessário, editados antes que o sistema seja executado. Um resumo da instalação é gravado no arquivo **src/init/deployment.log**, dentro do diretório de instalação (ver 2.1.3).

As seguintes opções de configuração são solicitadas:

- Inicia servidor do CSGrid? [true]
O script de inicialização do CSGrid iniciará o servidor central.
- Inicia SGA? [true]
O script de inicialização do CSGrid iniciará a execução do SGA na mesma máquina do servidor central.
- Inicia CSFS? [false]
Essa opção somente deve ser alterada para true se a máquina de algum dos SGAs não possuir acesso NFS ao repositório de dados e de algoritmos do servidor CSGrid. Nesse caso, o script de inicialização do CSGrid iniciará a execução do servidor CSFS para transferência de arquivos.
- Digite [1] para criar um novo arquivo de propriedades ou qualquer outra tecla para usar um arquivo existente:
Caso a instalação esteja sendo feita pela primeira vez, digite 1. Nesse caso, um arquivo de propriedades será criado e usado pelo script de inicialização do servidor CSGrid. Se já houver um arquivo de propriedades criado por uma instalação anterior, digite <ENTER>.

- Nome do arquivo de propriedades (Usar path relativo ao diretório de instalação): [../config/System.properties]
Digite <ENTER> caso o arquivo de propriedades (criado pelo processo de instalação ou proveniente de uma instalação anterior) possua o nome **System.properties** e esteja dentro do diretório **config**.
- Apelido do sistema: [CSGrid]
O nome com o qual o servidor CSGrid é reconhecido pela opções de administração e configuração.
- Nome do servidor para conexão com o cliente: [serverMachine]
Nome da máquina onde será executado o servidor do CSGrid.
- IP do servidor para conexão com o cliente: [10.0.8.169]
Endereço de IP usado para estabelecer conexão com o cliente CSGrid. No caso de máquinas com uma única placa de rede, digite <ENTER> para acatar a opção padrão que é o IP da própria máquina. No caso de máquinas que possuam mais de uma placa de rede, deve ser digitado aqui o IP externo com o qual o servidor CSGrid se comunicará com as máquinas que executarão o cliente web CSGrid.
- IP do servidor para conexão com o SGA: [10.0.8.169]
Endereço de IP usado para estabelecer conexão com as máquinas de execução de algoritmos (SGAs). No caso de máquinas com uma única placa de rede, digite <ENTER> para acatar a opção padrão que é o IP da própria máquina. No caso de máquinas que possuam mais de uma placa de rede, deve ser digitado aqui o IP com o qual a máquina se comunicará com os SGAs.
- IP do servidor para conexão com o barramento: [10.0.8.169]
Endereço de IP usado para estabelecer conexão com o servidor do barramento (OpenBus). No caso de máquinas com uma única placa de rede, digite <ENTER> para acatar a opção padrão que é o IP da própria máquina. No caso de máquinas que possuam mais de uma placa de rede, deve ser digitado aqui o IP com o qual a máquina se comunicará com o OpenBus.
- IP do servidor do barramento: [10.0.8.169]
Endereço IP da máquina onde o barramento OpenBus está executando. Digite <ENTER> no caso do barramento estar executando na mesma máquina onde o servidor CSGrid será iniciado.
- Porta do servidor do barramento: [3000]
Porta usada pelo barramento OpenBus. Digite o número da porta de acordo com o que foi configurado na instalação do barramento.

- Servidor de SMTP:
Digite o nome completo do servidor SMTP usado para envio de emails pelo servidor CS-Grid. Essa configuração é importante para envio de emails sobre erros ocorridos ou sobre término de execução de comandos submetidos. Sem um servidor de SMTP, o servidor CSGrid não enviará emails sobre erros ou término de execução de comandos submetidos.
- Digite [1] para instalar o CSGrid no diretório do apache ou qualquer outra tecla para prosseguir:
Caso o servidor apache esteja instalado na mesma máquina do servidor CSGrid e seja usado para disponibilizar o cliente web, digite 1. Caso o cliente web CSGrid seja disponibilizado pelo tomcat, que já faz parte da instalação, tecle <ENTER>.
- URL para acessar o csgrid [<http://csgrid.tecgraf.puc-rio.br:80/csgrid>]
Digite a URL que o cliente usará no browser para acesso ao desktop do CSGrid.
- Diretório do Apache: [/var/apache/htdocs/csgrid]
Digite o caminho para o diretório do apache onde o instalador disponibilizará os arquivos do cliente web do CSGrid (ver 2.1.5).

3.4 Configuração dos Servidores SGAs

Copie o arquivo `install/sgad-cnf.lua` para o diretório **config** do diretório de instalação (ver 2.1.3), por exemplo, **csgrid/config**. Esse arquivo deve ser editado para incluir os servidores que executarão os SGAs.

No arquivo **sgad-cnf.lua** existe uma seção comentada com um exemplo de configuração de um SGA simples. Copie esse trecho (deixando o exemplo comentado) para incluir as configurações específicas de cada SGA que será iniciado.

O exemplo a seguir mostra uma possível configuração para um SGA na plataforma Linux.

```
-----  
-- Configuração dos SGAs  
-----  
  
Node{  
    name = "bombas",  
    platform_id = "Linux26g4",  
    num_processors = 2,
```

```
clock_speed_mhz = 1000,  
memory_ram_info_mb = 2048,  
memory_swap_info_mb = 5376,  
disabled = NO,  
csfs_properties = { OAIAddr="kobke" },  
requirements = { "Excel" },  
}
```

Onde:

- **name** é um apelido que identifica o SGA para o servidor CSGrid. Como sugestão, usar o que é retornado pelo comando hostname (sem o domínio). É obrigatório.
- **platform_id** é o identificador de plataforma; ou seja, o valor de `TEC_UNAME` do seu ambiente. Esse identificador deve coincidir com uma das plataformas cadastradas pelo administrador do CSGrid (ver 5.4). É obrigatório.
- **num_processors** é o número de processadores na máquina. É obrigatório.
- **clock_speed_mhz** é o clock speed da sua máquina em Mhz. Essa configuração é opcional, ou seja, se não existir, o próprio SGA se encarrega de descobrir.
- **memory_ram_info_mb** e **memory_swap_info_mb** se referem à memória RAM e SWAP respectivamente. Essa configuração é opcional, ou seja, se não existir, o próprio SGA se encarrega de descobrir.
- **disabled** indica que o SGA não está habilitado para execução de algoritmos. Se estiver como NO ou não existir, o SGA está habilitado. Se estiver como YES, o SGA está desabilitado. É opcional.
- **csfs_properties** Usado apenas quando o SGA não possui acesso NFS ao repositório de dados e de algoritmos do servidor CSGrid. Indica as propriedades do servidor CSFS, responsável pela transferência de arquivos para este SGA. É opcional.
- **requirements** Lista de strings que representa requisitos que o SGA possui. O administrador pode definir, para cada algoritmo, os requisitos necessários para sua execução. Neste caso, serão escolhidos para a execução do algoritmo, apenas SGAs que possuam tais requisitos. É opcional.

Note que o `sgad-cnflua` é um arquivo escrito na linguagem de programação Lua³ e, portanto, pode ter qualquer comando dessa linguagem. Por exemplo, no caso de instalação em

³<http://www.lua.org>

um cluster com vários nós onde, em cada um desses nós, deseja-se instalar um SGA, pode-se usar um código como o ilustrado a seguir:

```
local default_config = {  
  num_processors = 4,  
  platform_id = "Linux26g4",  
}  
for i=1,12 do  
  local _node = cloneNode(default_config)  
  _node.name = "n"..i  
  Node(_node)  
end
```

onde o nomes dos nós são n1, n2, n3 até n12.

3.5 Conexão com o OpenBus

Para que o servidor CSGrid se conecte ao OpenBus, é preciso:

- Fornecer o certificado do CSGrid para que o administrador do OpenBus o cadastre no OpenBus. O certificado do CSGrid é o arquivo *CSGrid.crt* que fica no diretório *src/csgrid/security* do CSGrid. O nome do arquivo deve ser igual ao nome usado na propriedade *OpenBusService.entity.name*, com extensão *crt*.
- Solicitar o certificado do OpenBus ao administrador do OpenBus e copiá-lo para o diretório *src/csgrid/security* do CSGrid. O certificado do OpenBus é o arquivo *AccessControlService.crt* que fica no diretório *data/certificates* do OpenBus. O nome do arquivo deve ser igual ao nome usado na propriedade *OpenBusService.certificate.file*.
- Verificar se as seguintes propriedades (obrigatórias) estão configuradas corretamente no arquivo *config/System.properties*. Se alguma delas não estiver definida, ela deve ser incluída no arquivo de propriedades:
 - *OpenBusService.enabled = true* (*habilita a conexão com o OpenBus*)
 - *OpenBusService.host = servidor onde executa o OpenBus* (*isto é, o valor da propriedade *hostName* usada no OpenBus*)
 - *OpenBusService.port = porta onde executa o OpenBus* (*isto é, o valor da propriedade *hostPort* usada no OpenBus*)

- OpenBusService.certificate.file = security/AccessControlService.crt (*indica o nome do arquivo que contém o certificado do OpenBus*)
- OpenBusService.private.key.file = security/CSGrid.key (*indica o nome do arquivo que contém a chave privada do CSGrid*)
- OpenBusService.entity.name = CSGrid (*ou outro nome que identifique o sistema CS-Grid no OpenBus*)
- OpenDreamsService.export.to.openbus = true (*habilita a publicação do serviço OpenDreams no OpenBus*)
- OpenDreamsService.component.name = OpenDreams (*ou outro nome que identifique o componente OpenDreams publicado no OpenBus*)
- OpenDreamsService.component.version = 1.0.0 (*ou outra versão que identifique o componente OpenDreams publicado no OpenBus*)
- ProjectService.export.to.openbus = true (*habilita a publicação do serviço ProjectService no OpenBus*)
- ProjectService.component.name = ProjectService (*ou outro nome que identifique o componente do Serviço de Projetos publicado no OpenBus*)
- ProjectService.component.version = 1.0.0 (*ou outra versão que identifique o componente do Serviço de Projetos publicado no OpenBus*)

3.6 Atualização do Servidor

Para atualizar a versão do sistema, basta interromper a execução do servidor e seguir os passos contidos na Seção 3.3.

4 Execução do Servidor

O sistema CSGrid, quando em execução, é composto por diversos processos que executam em vários computadores e se originam de três programas:

1. O Servidor Central, servidor CSGrid propriamente dito.
2. O SGA, que permite o acesso aos computadores de execução de algoritmos.
3. O tomcat, que implementa um servidor HTTP/Servlets.

Todos esses programas são executados através de scripts que se encontram no diretório `init`. O script `csgrid` pode executar o servidor do CSGrid, o tomcat, e um SGA, de acordo com o que foi configurado na instalação. Ele atende ao padrão de serviços do UNIX, recebendo os parâmetros `start` para iniciar e `stop` para parar. Assim é possível colocá-los junto aos demais serviços do sistema. Para executar o servidor CSGrid e o tomcat, é necessário que o Java SDK versão 1.6.x esteja instalado. Note que a instalação do JRE não é suficiente, é realmente necessário o SDK.

Dependendo da opção selecionada na instalação (ver Seção 3.3), o `start` e o `stop` iniciam e param a execução do servidor central, do tomcat e de um servidor sga na mesma máquina do servidor central.

Para executar os comandos de iniciar, reiniciar e suspender a execução dos módulos do CS-Grid instalados, entre em **src/init** no diretório de instalação e execute os respectivos comandos:

- **`./csgrid start`** para iniciar a execução dos módulos instalados do CSGrid
- **`./csgrid stop`** para terminar a execução dos módulos instalados do CSGrid
- **`./csgrid restart`** para suspender e reiniciar a execução dos módulos instalados do CSGrid

4.1 Execução de Servidores SGAs adicionais

O programa SGA deve ser executado em cada computador de execução de algoritmos. Entre em **src/sgad/src** e execute o comando **`sga-daemon`**. Por default, o SGA assume que o apelido (ver 3.4) é o nome da máquina. Se o apelido não tiver sido configurado com o nome da máquina, o argumento **`sga`** deve indicar o apelido a ser usado. Outros argumentos que podem, opcionalmente, ser passados para o comando **`sga-daemon`** são:

- `--sga <name>` - o apelido do sga
- `--sga-addr <ip:port>` - ip e a porta do sga
- `--ssi <server>` - ip do servidor central do CSGrid
- `--ssi-port <port>` - porta do servidor central do CSGrid
- `--cnf-directory <directory>` - diretório onde está o arquivo de configuração de sgas (ver 3.4)
- `--log` - indica que o log do sga deve ser direcionado para um arquivo
- `--log-directory <directory>` - diretório onde o arquivo de log é gravado (depende de `--log`)
- `--help` - mostra o help do comando

No diretório **src/init** disponibilizamos outros scripts auxiliares para inicialização e suspensão dos SGAs.

Toda a saída gerada pelo servidor CSGrid é direcionada para arquivos no diretório logs, dentro do diretório de instalação (ver Seção 2.1.3). Por fim, a saída de cada SGA é direcionada para o arquivo `logs/sga/SGAxxx.log`, onde xxx é o nome do sga. Esses arquivos devem ser analisados para saber se os programas entram em execução sem erros.

4.2 Inicialização Automática

É conveniente que o ambiente esteja configurado para que o servidor seja iniciado e finalizado automaticamente de acordo com a inicialização e finalização do sistema operacional. Para fazer essa configuração de inicialização automática do servidor CSGrid, siga os passos a seguir:

- Copie o conteúdo da Figura 3 para o arquivo `/etc/init.d/csgrid` (substituindo `#CSGRID_HOME#` pelo diretório de instalação (ver 2.1.3) e `#USR#` pelo nome do usuário (ver 2.1.2).
- Executar o comando: `chkconfig --add csgrid`

```
#!/bin/bash
#
# chkconfig: 2345 99 15
# description: Script de inicializacao para o CSGrid
#

INIT_DIR=#CSGRID_HOME#/src/init

function action {
    su - #USR# -c "cd $INIT_DIR; csgrid $1 "
}

case $1 in
    start|stop)
        action $1
        ;;
    *)
        echo "ERRO: usar $0 (start|stop)"
        exit 1
        ;;
esac
```

Figura 3: Script de inicializacao para o CSGrid.

Para o caso de inicialização automática em máquinas que executam somente o SGA, o script de inicialização automática fica um pouco diferente, conforme descrito a seguir.

- Copie o conteúdo da Figura 4 para o arquivo `/etc/init.d/sga` (substituindo `#CSGRID_HOME#` pelo diretório de instalação (ver 2.1.3) e `#USR#` pelo nome do usuário (ver 2.1.2).
- Executar o comando: `chkconfig --add sga`

```
#!/bin/bash
#
# chkconfig: 2345 99 15
# description: Script de inicializacao para o SGA

#
INIT_DIR=/home/prj/cibase/csgrid/src/init
function action {
su - cibase -c "cd $INIT_DIR; run-sga-unix $1 "

}
case $1 in
start|stop)
action $1
;;
*)
echo "ERRO: usar $0 (start|stop)"
exit 1
;;
esac
```

Figura 4: Script de inicializacao para o SGA.

4.3 Verificação do Estado do Servidor

A forma mais eficaz de verificar se o servidor está executando e se sua configuração está correta, é tentar executar o cliente. Se o cliente conseguir acessar o servidor e responder corretamente a um conjunto de operações, significa que o servidor está fazendo o trabalho esperado.

Para executar o CSGrid Cliente, digite em um navegador o endereço do servidor escolhido (por exemplo, <http://csgrid.tecgraf.puc-rio.br>), ou o endereço que tenha sido definido na Seção 2.1.5. Deverá surgir uma tela de login. Entre com o usuário "admin", senha "1234". Se surgir a tela inicial do sistema, é garantido que o servidor foi encontrado e está disponível. Caso o cliente não seja iniciado ou ocorra alguma falha na sua comunicação com o servidor, os arquivos de log podem ser verificados. Os logs do servidor ficam no diretório de instalação em `logs/logs_AAAAMMDHMMSS`, onde AAAAMMDHMMSS é o ano, mês, dia, hora, minutos e segundos que correspondem a inicialização do servidor. O arquivo `csgrid.out` é a saída padrão do servidor. Dentro do sub-diretório `logs/logs_AAAAMMDHMMSS/develop` estão todos os logs do servidor, separados por serviços.



Você pode procurar por SEVERE. Qualquer problema, favor entrar em contato com a equipe de desenvolvimento pelo email

csguid@tecgraf.puc-rio.br.

É recomendável que, após a verificação do estado do servidor, o administrador altere a senha do usuário "admin".

5 Tarefas de Administração

Nessa seção nós descrevemos as tarefas administrativas que só podem ser realizadas pelo usuário administrador (que possui o login admin). Essas tarefas tem caráter global do sistema. Para realizar as tarefas de administração, o usuário admin deve abrir a janela de administração do CSGrid, que fica no menu **Configuração** da desktop principal do sistema, conforme ilustrado na Figura 5.

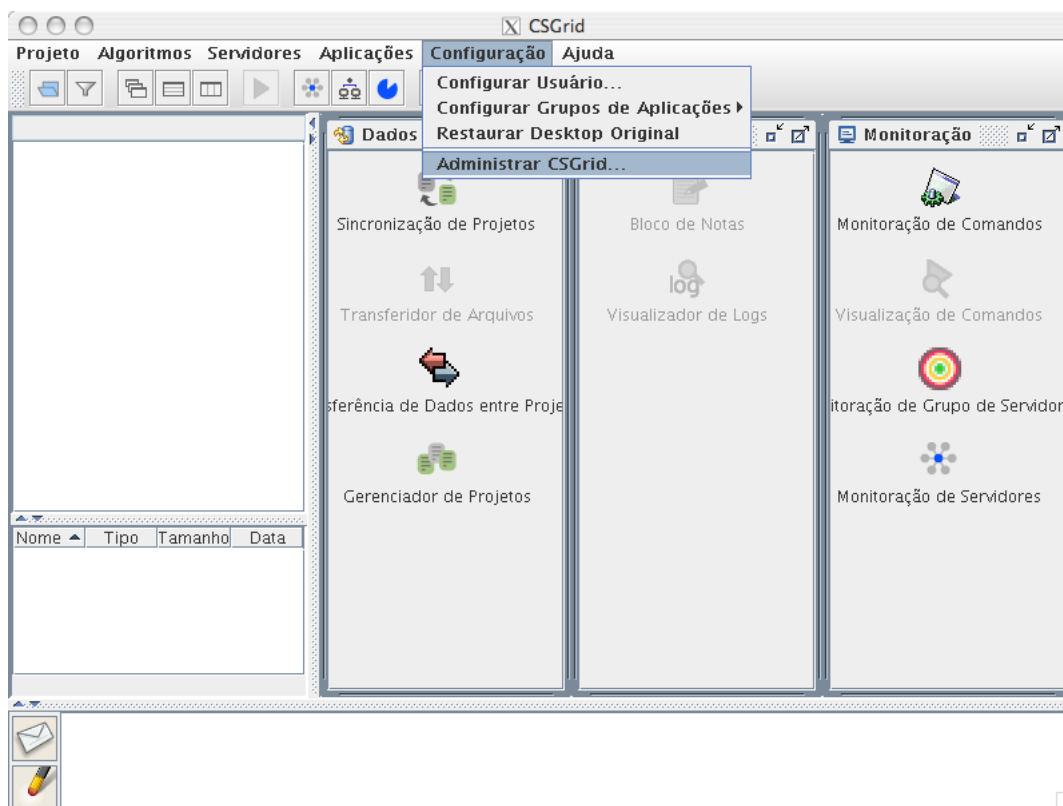


Figura 5: Menu de administração do CSGrid

5.1 Criar Usuários

O administrador pode adicionar usuários ao sistema. Cada usuário deve, obrigatoriamente, estar associado a um **Grupo de Usuários**. Ver na Seção 5.2 como criar um grupo de usuários. Para criar um usuário, o administrador deve fornecer um login (no campo **Usuário**), um nome (no campo **Nome**) e escolher o grupo ao qual esse usuário pertence. Opcionalmente, o administrador

pode também cadastrar um email (campo **Email**) para esse usuário. Após confirmar, o administrador deve cadastrar uma senha inicial que, posteriormente, o usuário pode alterar. A Figura 6 ilustra as janelas envolvidas na tarefa de criação de um usuário no CSGGrid.

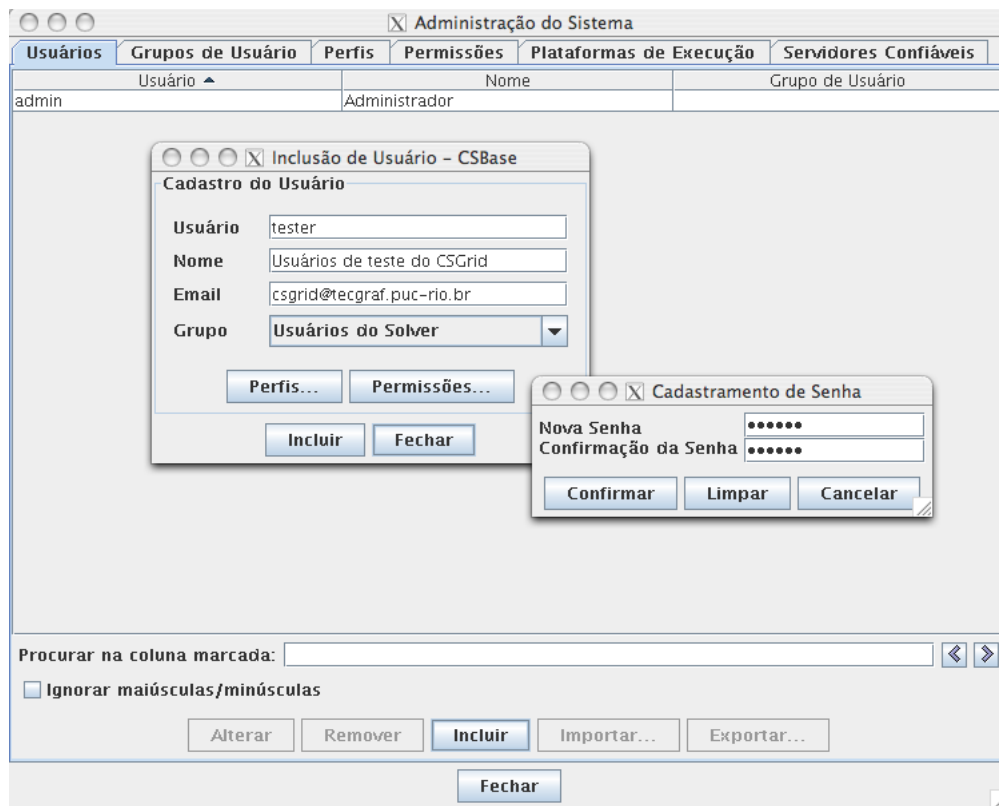


Figura 6: Cadastramento de usuários

O administrador pode atribuir permissões específicas aos usuários, que habilitam certas funcionalidades do CSGGrid. As permissões podem ser atribuídas diretamente a cada usuário ou a um perfil de usuário. No segundo caso, o perfil atribuído ao usuário é que determina as permissões que ele possui. Na Seção 5.3 descrevemos como criar um perfil e nas Seções 5.5.1, 5.5.2, 5.5.3 e 5.5.4 descrevemos alguns tipos comuns de permissões que podem ser dadas aos usuários.

5.2 Criar Grupos de Usuários

Um grupo de usuário é apenas uma representação de uma entidade que agrupa diversos usuários do CSGGrid. Pode ser uma unidade de negócio ou alguma outra organização lógica, por exemplo,

usuários de um outro sistema que usa o CSGrid para executar algoritmos específicos. Para criar um grupo de usuários, o administrador deve selecionar a aba **Grupos de Usuário** e acionar o botão de **Incluir**. Na janela de inclusão, o administrador deve fornecer um nome (campo **Grupo de Usuário**) e, opcionalmente, uma descrição (campo **Descrição**) para o grupo, conforme ilustrado na Figura 7.

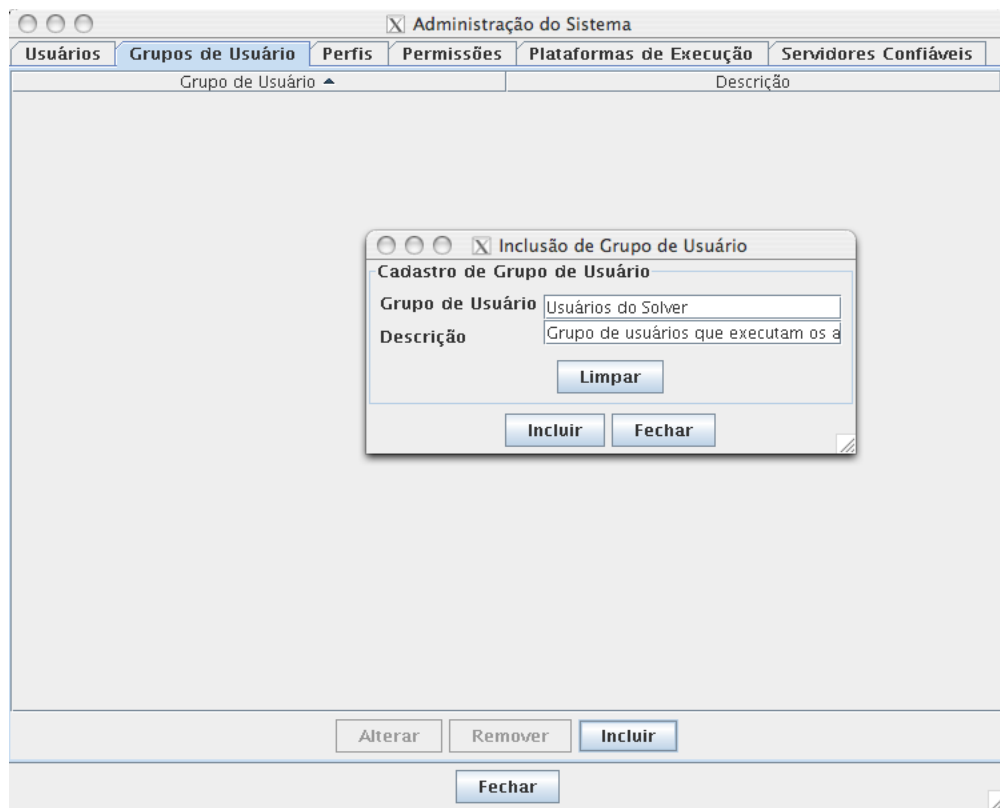


Figura 7: Cadastramento de grupo de usuários

5.3 Criar Perfis de Usuários

O perfil é apenas uma forma de definir um conjunto de permissões agrupadas que pode ser atribuído a um ou mais usuários. Para criar um perfil, o administrador deve selecionar a aba **Perfis** e acionar o botão de **Incluir**. Na janela de inclusão, o administrador deve fornecer um nome (campo **Nome**) e, opcionalmente, uma descrição (campo **Descrição**) para o perfil sendo cadastrado. A seguir, deve selecionar e adicionar quais as permissões fazem parte desse perfil. A janela da Figura 8 ilustra a inclusão de um perfil de usuários.

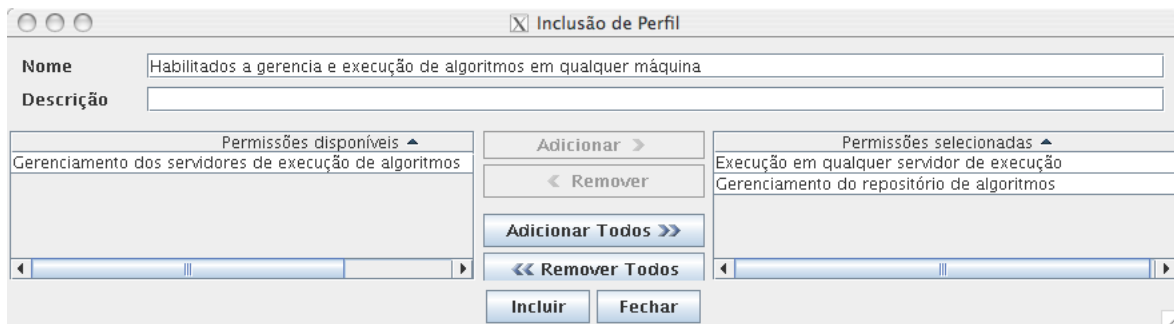


Figura 8: Cadastramento de perfis de usuários

Para que um usuário passe a ter as permissões de um determinado perfil, o administrador deve alterar os perfis desse usuário, na aba **Usuários**, selecionando o usuário e acionando o botão **Alterar**. A janela de alteração de um usuário (assim como a janela de inclusão), possui um botão **Perfis...** que abre a janela ilustrada na Figura 9.

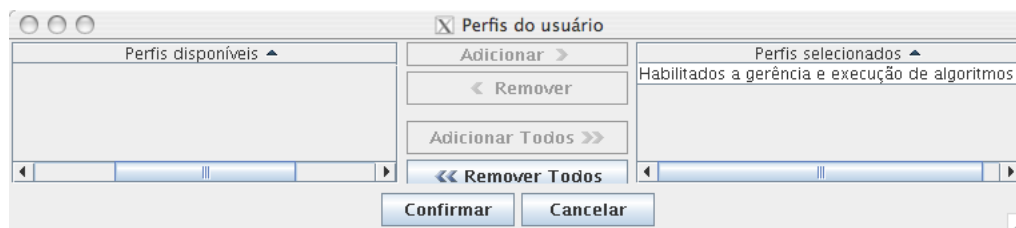


Figura 9: Associar um perfil a um usuário

Note que o administrador pode também associar as permissões diretamente aos usuários, sem que seja necessário criar perfis.

5.4 Criar Plataformas de Execução de Algoritmos

Os executáveis dos algoritmos são cadastrados no repositório por plataforma. Assim, a escolha da máquina de execução de cada algoritmo é feita de acordo com a plataforma para a qual existe um executável. O administrador deve cadastrar as plataformas disponíveis para que os usuários possam registrar os executáveis dos algoritmos nessas plataformas. Para cadastrar as plataformas, o administrador deve selecionar a aba **Plataformas de Execução** e acionar o botão de **Incluir**. Na janela de inclusão, o administrador deve fornecer um nome (campo **Plataforma**) e,

opcionalmente, uma descrição (campo **Descrição**) para a plataforma sendo cadastrada, conforme ilustrado na Figura 10.

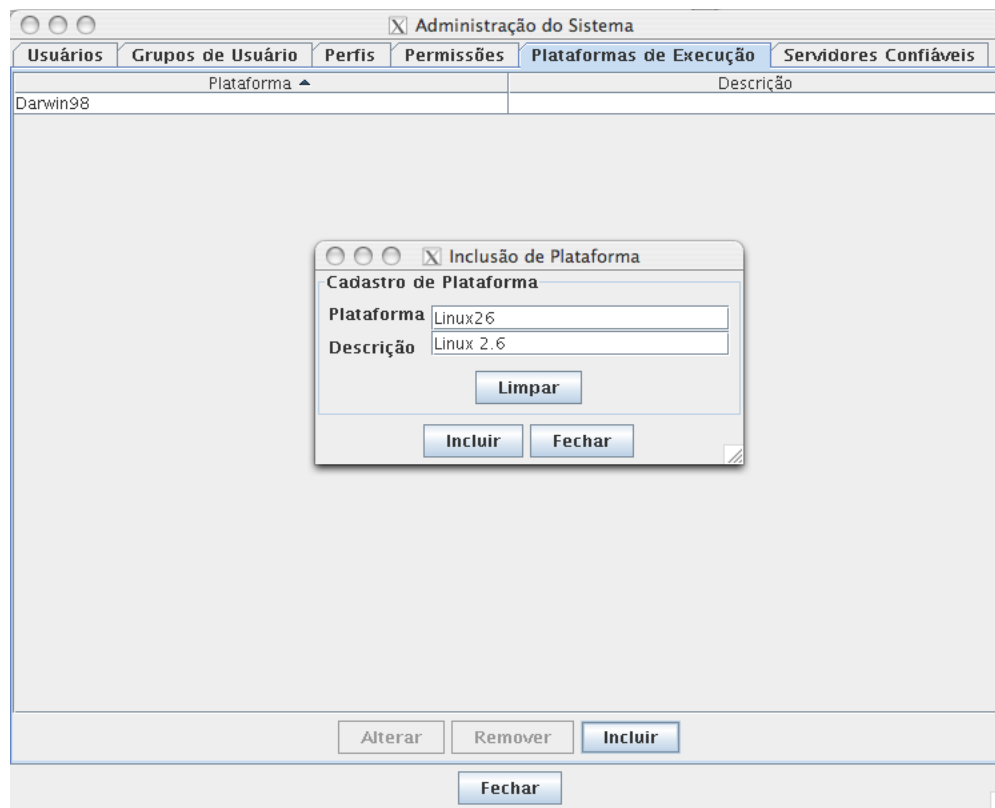


Figura 10: Cadastramento de plataformas de execução

5.5 Criar Permissões

O administrador pode criar um conjunto de permissões que servem para liberar determinadas funcionalidades para que um ou mais usuários possam usá-las. O CSGrid já possui um conjunto de classes de permissões que estão associadas a essas funcionalidades. O administrador, portanto, cria uma ou mais permissões de uma determinada classe, definindo atributos que customizam essas permissões de acordo com a necessidade do seu ambiente.

Por exemplo, o administrador pode criar permissões da classe **Utilização de servidores de execução de algoritmos** para permitir que usuários possam executar comandos nas máquinas de execução. Usando o atributo **local** dessa classe de permissão, o administrador pode criar permissões diferentes dessa classe para restringir em que máquinas os usuários estão liberados

para execução de comandos.

Para criar as permissões, o administrador deve selecionar a aba **Permissões** e acionar o botão de **Incluir**. Na janela de inclusão, o administrador fornece o nome (campo **Permissão** e, opcionalmente, uma descrição (campo **Descrição**) para a permissão sendo criada. O administrador deve obrigatoriamente escolher a classe da permissão e definir os atributos que essa classe exige. A seguir descrevemos algumas classes comuns de permissões e seus atributos.

5.5.1 Criar Permissão de Utilização de Servidores

O administrador pode eleger um ou mais usuários para que eles possam executar algoritmos em uma ou mais máquinas de execução (SGAs). A Figura 11 mostra a janela de inclusão de uma permissão da classe **Utilização de servidores de execução de algoritmos**, que serve para esse propósito.

O administrador deve incluir um atributo nessa permissão que limite os servidores de execução de algoritmos. O atributo deve ser da forma "nome", onde nome deve ser substituído pelo nome do servidor de execução (SGA) ao qual o usuário estará restrito para submeter comandos. Esse nome precisa ser idêntico ao apelido do servidor de execução (SGA) configurado durante a instalação (ver 3.4) ou pode ser uma expressão regular a ser casada com o apelido do SGA caso se deseje que o usuário possa executar algoritmos em mais de um servidor. Por exemplo, definindo o atributo como `.*`, a permissão libera o usuário para executar em qualquer SGA disponível.

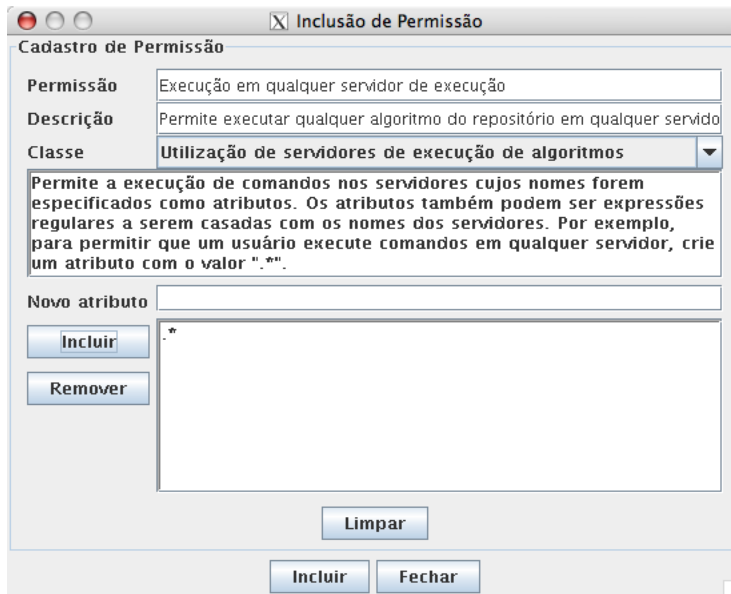


Figura 11: Incluir permissão de utilização de servidores

5.5.2 Criar Permissão de Execução de Algoritmos

O administrador pode eleger um ou mais usuários para que eles possam executar determinados algoritmos em um determinado sistema. A Figura 12 mostra a janela de inclusão de uma permissão da classe **Execução de algoritmos em um sistema específico**, para autorizar os usuários a executarem um ou mais algoritmos a partir de um determinado sistema. Esse sistema pode ser o mesmo que provê o repositório de algoritmos ou pode ser outro sistema acessado via Openbus.

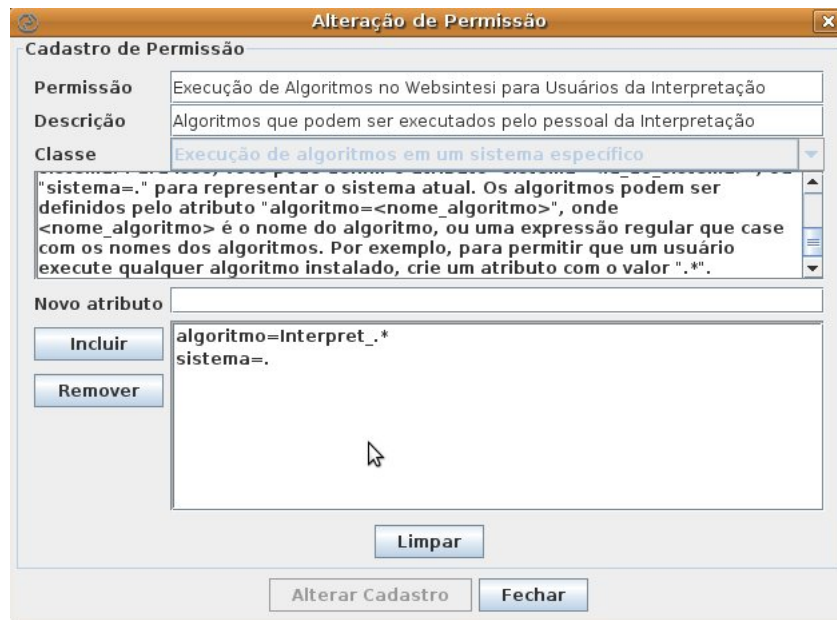


Figura 12: Incluir permissão de execução de algoritmos

O administrador deve incluir uma permissão da classe Execução de algoritmos em um sistema específico, que requer dois atributos obrigatórios:

- Atributo que limita a execução de algoritmos com origem em um determinado sistema.
- Atributo que limita a execução aos algoritmos desejados.

O primeiro atributo deve ser da forma "sistema=id_do_sistema", onde id_do_sistema deve ser substituído pelo identificador do sistema onde será originada a execução do algoritmo autorizado.

Se a permissão for especificar a execução de algoritmos originada no próprio sistema CSGrid, então esse atributo deve ser sistema=., onde "." representa o sistema atual.

Se a permissão for especificar a execução de algoritmos originada em outro sistema, então esse identificador precisa ser idêntico ao valor especificado na propriedade **OpenBusService.entity.name** do sistema que é publicado no OpenBus, e cujo valor será enviado pela credencial no barramento.

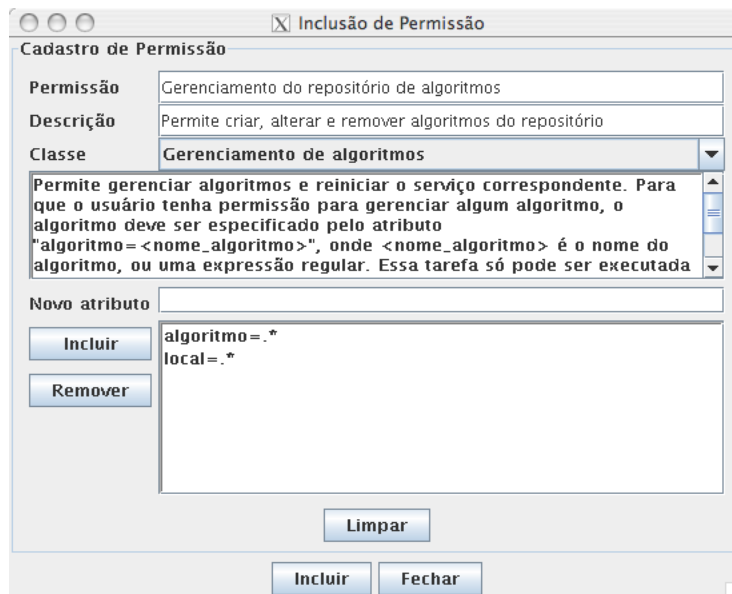
O segundo atributo deve ser da forma "algoritmo=nome", onde nome deve ser substituído pelo nome do algoritmo ao qual o usuário estará restrito. Esse nome também pode ser uma expressão regular a ser casada com os nomes dos algoritmos para permitir que o usuário execute mais de um algoritmo. Por exemplo, definindo o atributo como `algoritmo=.*`, a permissão libera o usuário para executar qualquer algoritmo do repositório, a partir do sistema desejado.

5.5.3 Criar Permissão de Gerenciamento de Algoritmos

O administrador pode eleger um os mais usuários para que eles possam gerenciar o repositório de algoritmos, criando, alterando ou removendo algoritmos do repositório. A Figura 13 mostra a janela de inclusão de uma permissão da classe **Gerenciamento de algoritmos**, que serve para esse propósito.

A permissão da classe **Gerenciamento de algoritmos** requer dois atributos:

- Atributo que limita a gerência de algoritmos ao servidor desejado.
Esse atributo deve ser da forma "local=nome", onde nome deve ser substituído pelo nome do servidor ao qual o usuário estará restrito. Esse nome precisa ser idêntico ao apelido do servidor fornecido durante a instalação 3.3 ou pode ser uma expressão regular a ser casada com o nome do servidor caso se deseje que o usuário possa gerenciar algoritmos em mais de um servidor. Por exemplo, definindo o atributo como `local=.*`, a permissão libera o usuário para gerenciar algoritmos em qualquer servidor CSGrid de uma mesma instalação.
- Atributo que limita a gerência ao algoritmo desejado.
Esse atributo deve ser da forma "algoritmo=nome", onde nome deve ser substituído pelo nome do algoritmo ao qual o usuário estará restrito. Esse nome também pode ser uma expressão regular a ser casada com os nomes dos algoritmos para permitir que o usuário gerencie mais de um algoritmo. Por exemplo, definindo o atributo como `algoritmo=.*`, a permissão libera o usuário para gerenciar qualquer algoritmo do repositório.



Inclusão de Permissão

Cadastro de Permissão

Permissão: Gerenciamento do repositório de algoritmos

Descrição: Permite criar, alterar e remover algoritmos do repositório

Classe: Gerenciamento de algoritmos

Permite gerenciar algoritmos e reiniciar o serviço correspondente. Para que o usuário tenha permissão para gerenciar algum algoritmo, o algoritmo deve ser especificado pelo atributo "algoritmo=<nome_algoritmo>", onde <nome_algoritmo> é o nome do algoritmo, ou uma expressão regular. Essa tarefa só pode ser executada

Novo atributo

Incluir

Remover

algoritmo=.*
local=.*

Limpar

Incluir Fechar

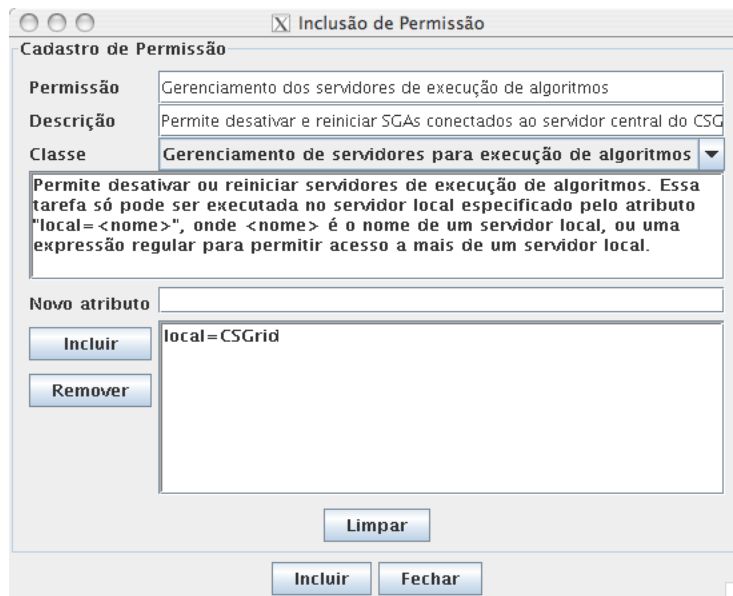
Figura 13: Incluir permissão de gerenciamento de algoritmos

5.5.4 Criar Permissão de Gerenciamento de Servidores de Execução de Algoritmos

O administrador pode eleger um os mais usuários para que eles possam gerenciar os servidores de execução de algoritmos, desativando e reiniciando esses servidores pela própria interface do CSGrid. A Figura 14 mostra a janela de inclusão de uma permissão da classe **Gerenciamento de servidores para execução de algoritmos**, que serve para esse propósito.

A permissão da classe **Gerenciamento de servidores para execução de algoritmos** requer o seguinte atributo:

- Atributo que limita a gerência de computadores de execução de algoritmos ao servidor desejado.
Esse atributo deve ser da forma "local=nome", onde nome deve ser substituído pelo nome do servidor ao qual o usuário estará restrito. Esse nome precisa ser idêntico ao apelido do servidor fornecido durante a instalação 3.3 ou pode ser uma expressão regular a ser casada com os nomes dos servidores caso se deseje que o usuário possa gerenciar computadores de execução de algoritmos em mais de um servidor. Por exemplo, definindo o atributo como local=.*, a permissão libera o usuário para gerenciar algoritmos em qualquer servidor CSGrid de uma mesma instalação.



Inclusão de Permissão

Cadastro de Permissão

Permissão: Gerenciamento dos servidores de execução de algoritmos

Descrição: Permite desativar e reiniciar SGAs conectados ao servidor central do CSG

Classe: Gerenciamento de servidores para execução de algoritmos

Permite desativar ou reiniciar servidores de execução de algoritmos. Essa tarefa só pode ser executada no servidor local especificado pelo atributo "local= <nome>", onde <nome> é o nome de um servidor local, ou uma expressão regular para permitir acesso a mais de um servidor local.

Novo atributo: local= CSGGrid

Incluir

Remover

Limpar

Incluir Fechar

Figura 14: Incluir permissão de gerenciamento de servidores de execução de algoritmos

6 Problemas e Soluções

Independentemente do problema que venha a ocorrer, lembre-se que toda a saída gerada pelo servidor CSGrid é direcionada para o arquivo `<instalação>/init/csgrid.out` e a saída de cada SGA é direcionada para o arquivo `<instalação>/sgad/logs/SGAxxx.log`, onde xxx é o nome do sga. Esses arquivos podem fornecer informações importantes sobre eventuais problemas.

Sintoma: SGA não inicia ou inicia mas não se conecta ao servidor.

Causas possíveis:

- Configuração dos servidores e da porta do serviço de SGA do CSGrid não casam com os parâmetros passados para o script sga-daemon.

Sintoma: Erro ao executar um algoritmo em um computador de execução de algoritmos.

Causas possíveis:

- Falta de espaço em disco na área do CSGrid.
- O binário do algoritmo para a plataforma em questão está com problemas. Para testar se esse é o caso, execute um SGA manualmente no computador de execução de algoritmos em questão e veja o comando que está sendo enviado para execução. Interrompa o SGA e execute o comando diretamente em um terminal: será possível ver erros como, por exemplo, falta de alguma biblioteca dinâmica ou caminhos inválidos.

Sintoma: Erros ao criar arquivos no projeto.

Causas possíveis:

- Falta de espaço em disco.

Sintoma: Erro ao executar o cliente usando o MS Explorer.

Causas possíveis:

- O browser está protegido, por questões de segurança.

1. Entrar em 'Tools / Internet Options / Security / Trusted Sites' e adicionar a URL onde se encontra o servidor do CSGrid, por exemplo:
`http://csgrid.tecgraf.puc-rio.br` (desmarcar a opção de usar apenas https).
2. Entrar em 'Tools / Internet Options / Security / Internet / Custom Level' e colocar: 'Download / File downloads' em 'Enabled'; 'Miscellaneous / Launching Applications and Unsafe Files' em 'Prompt (recommended)'; 'Miscellaneous / Installation of Desktop Items' em 'Prompt (recommended)'.