Pela complexidade do funcionamento de *grids*, identificar e reportar um problema não é uma tarefa tão trivial como em outras situações.

Esse guia lista alguns procedimentos para facilitar essa tarefa. Todas as dúvidas ou notificações de problemas devem ser encaminhadas para o email support@ncc.unesp.br.

Problemas com uma aplicação específica

Quando a suspeita for de um problema com uma aplicação ou caso específico:

- **Recomendamos a utilização do *go* (submissão simplificada) pois ela permite obter os arquivos de saída mesmo quando a simulação falha.**

- Verifique o arquivo <nome_do_caso>.go.out. Esse arquivo informa o código de retorno do programa e se os arquivos de entrada foram copiados corretamente.

- Verifique os arquivos <nome_do_caso>.stderr e <nome_do_caso>.stdout. Eles ajudam a entender qual foi o erro da aplicação. O seu conteúdo varia de aplicação para aplicação.

Como reportar o problema

- Envie um email para support@ncc.unesp.br, informando o nome de usuário e o diretório onde estão os arquivos da simulação.

Problemas com a infraestrutura do GridUNESP

Caso suspeite que o problema é com a infraestrutura do GridUNESP, execute os seguintes passos:

Teste de geração de proxy

Comando:

```
myproxy-logon
```

Resultado (aproximado):

```
Enter MyProxy pass phrase:
A credential has been received for user winckler in /tmp/x509up_u10001.
```

Teste de autenticação

Comando:

```
globus-job-run ce.grid.unesp.br/jobmanager-fork /usr/bin/id
```

O resultado pode demorar alguns segundos para aparecer. Resultado (aproximado):

```
uid=10001(winckler) gid=10005(ncc) groups=10005(ncc),9997(supercomputing),10000(users)
```

Teste de submissão para fila

Comando:

```
globus-job-run ce.grid.unesp.br/jobmanager-pbs -x "(queue=short)" /bin/hostname
```

O resultado pode demorar até alguns minutos para aparecer, dependendo da utilização do GridUNESP. Resultado (aproximado):

```
node031.local
```

Teste das aplicações

Esse teste dispara todos os testes das aplicações.

Comando:

```
mkdir go_tests
cd go_tests
/opt/GridUNESP/go/go-tests.sh
```

Esse comando enviará diversas simulações para o grid. Você poderá listá-las com o condor_q.

Existe um comando especial para listar o código de retorno das simulações. Todas elas devem eventualmente retornar **0**.

Comando para verificar o resultado:

```
/opt/GridUNESP/go/go-check.sh
```

Algumas simulações demoram apenas segundos, outras até uma hora para produzir o resultado. Exemplo de resultado intermediário:

```
beast.go: return  0
crystal-mpi.go: return  0
crystal-properties.go: return  0
crystal-serial.go: return  0
dacapo-parallel.go: no return code (yet?)
dacapo-serial.go: return  0
dirac-openmp.go: no return code (yet?)
dirac-parallel.go: no return code (yet?)
dirac-serial.go: return  0
emcee-demo.go: return  0
emcee-mpi.go: no return code (yet?)
emcee-smp.go: no return code (yet?)
emcee-tests.go: return  0
espresso-mpi.go: no return code (yet?)
espresso-serial.go: return  0
gamess-parallel.go: no return code (yet?)
gamess-serial.go: return  0
gamess-smp.go: return  0
gamit-2000.go: no return code (yet?)
garli-mpi.go: no return code (yet?)
garli-serial.go: no return code (yet?)
gaussian-parallel-short.go: no return code (yet?)
gaussian-serial-short.go: return  0
gaussian-smp-short.go: return  0
go-hpl.go: no return code (yet?)
gpaw-parallel.go: no return code (yet?)
gpaw-serial.go: return  0
gromacs-parallel.go: no return code (yet?)
gromacs-serial.go: no return code (yet?)
gromacs-smp.go: return  0
ima2-serial.go: return  0
im-serial.go: return  0
```

lammps-parallel.go: no return code (yet?)
lammps-serial.go: return 0
mopac.go: return 0
mrbayes-parallel.go: no return code (yet?)
mrbayes-serial.go: return 0
mugsy_serial.go: return 0
namd-parallel.go: no return code (yet?)
namd-smp.go: return 0
networkx_test.go: return 0
openfoam-airfoil.go: return 0
openfoam-cavity.go: return 0
orca-serial-multstep.go: no return code (yet?)
orca-serial-short.go: return 0
poy-parallel.go: no return code (yet?)
poy-serial.go: return 0
python-version.go: return 0
raxml-mpi.go: no return code (yet?)
raxml-serial.go: return 0
raxml-smp.go: return 0