

INSTRUÇÕES PARA EXECUTAR O MODELO DECOMP VERSÃO LINUX NA PLATAFORMA WINDOWS

O modelo DECOMP é disponibilizado apenas em sua versão compatível com o sistema operacional Linux. No entanto é possível executá-lo no sistema operacional Windows usando dispositivos de virtualização. Neste tutorial são propostas duas formas diferentes para realizar uma rodada de DECOMP em Windows.

O processo de virtualização permite emular um sistema operacional (hóspede) em outro (hopedeiro), por meio de um programa virtualizador, é necessário observar que os programas podem ter licenças pagas que devem ser levadas em consideração antes das suas instalações. Aqui vamos abordar duas formas de realizar esta virtualização: através de uma máquina virtual ou através de containers. As duas formas possuem vantagens e desvantagens, assim deixamos a cargo do usuário a escolha da tecnologia de sua preferência.

Este tutorial é dividido em duas partes:

1. Máquinas virtuais:
 - 1.1. Criação de uma máquina virtual.
 - 1.2. Como executar o DECOMP em uma máquina virtual Linux a partir da plataforma Windows (opcionalmente em paralelo).
2. Containers:
 - 2.1. Instalação de um ambiente de suporte de containers
 - 2.2. Execução do DECOMP em um container Linux a partir da plataforma Windows

1 MAQUINAS VIRTUAIS

1.1 CRIAÇÃO DE UMA MÁQUINA VIRTUAL

Uma vez criada a máquina virtual, ela pode ser desligada e posteriormente ligada novamente, comportando-se, dessa forma, como se fosse um micro comum. Assim, os procedimentos de criação de uma máquina virtual, descritos a seguir, são executados uma única vez.

i) Instalar o programa virtualizador na máquina Windows

Qualquer programa virtualizador com suporte à plataforma Windows e com suporte para máquinas virtuais Linux pode ser utilizado. Exemplos são: VirtualBox (<https://www.virtualbox.org/>) e VMware (<http://www.vmware.com/br>).

Faça download da versão do instalador para o Windows utilizado no computador e instale o programa.

ii) Criar uma máquina virtual Linux

Qualquer distribuição Linux pode ser utilizada, porém é importante ressaltar que a máquina hospedeira, a instalação Windows e a instalação Linux devem ser 64 bits para executar o DECOMP. Escolha uma distribuição Linux e faça download do instalador no formato “.iso”. São exemplos: CentOS (<http://www.centos.org/download/>) e Ubuntu (<http://www.ubuntu.com/download/desktop>).

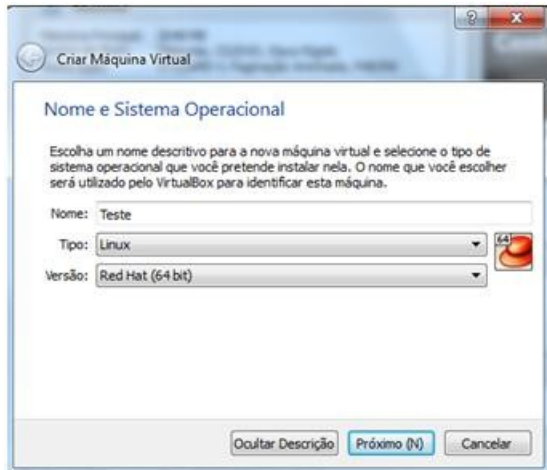
Comunicação entre hospedeiro e máquina virtual através de pastas compartilhadas: Para que arquivos e executáveis DECOMP possam ser compartilhados entre o sistema operacional hospedeiro (Windows) e sistema operacional hóspede (Linux), deve-se criar uma interface de compartilhamento. Os programas virtualizadores têm essa opção em sua configuração. Faça de forma a conseguir acessar os arquivos necessários para executar o DECOMP no sistema Linux.

Comunicação via ip host-only:

A troca de arquivos entre hospedeiro e hóspede pode ser feita através de uma comunicação IP entre elas, isso permite que transferências de arquivos e execução de comandos sejam feitas nos dois sentidos.

Exemplo: Passo a passo para criação de máquina virtual com sistema CentOS-Linux utilizando VirtualBox

Após instalado, executar o programa VirtualBox e clicar na opção do menu “Máquina -> novo” para criação de uma nova máquina virtual.

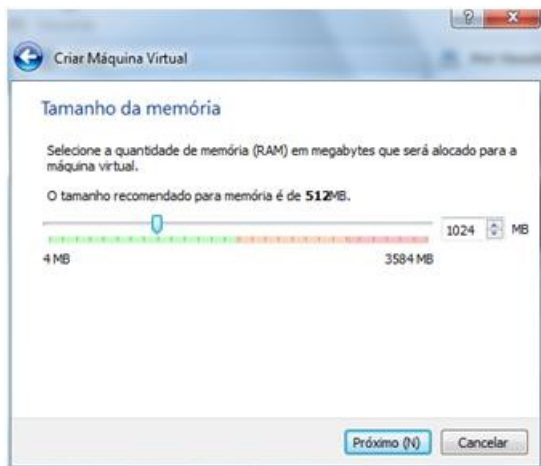


Dados da máquina

O usuário deve informar:

- Um nome para a máquina.
- O tipo de máquina: plataforma Linux.
- E a distribuição Linux, que poderá ser escolhida pelo usuário, neste exemplo RedHat.

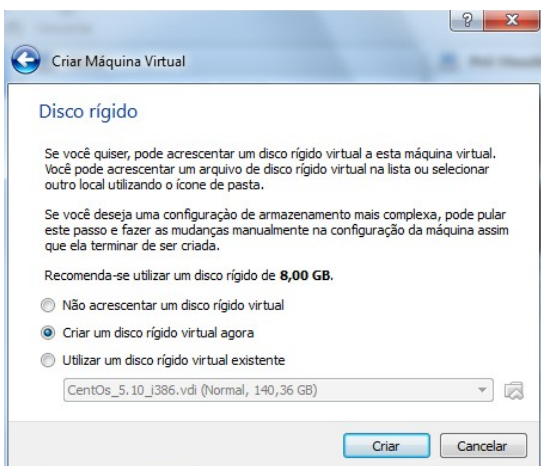
Na sequência, é preciso configurar a memória RAM disponibilizada para a máquina virtual.



Memória

O usuário deve escolher um valor de memória dedicada à máquina virtual. Esse valor dependerá da quantidade de memória disponível na máquina e afeta a eficiência de ambas as plataformas, hospedeiro e hóspede.

O próximo passo é a criação do disco rígido dedicado à máquina.



Disco Rígido

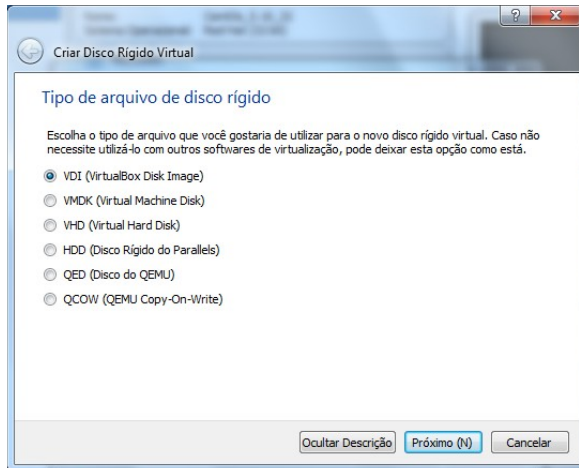
É recomendável que se tenha espaço em disco dedicado à máquina virtual para instalação de programas e armazenamento de arquivos. A opção “Criar um disco rígido virtual agora” possibilita a alocação de um espaço em disco vazio para a máquina virtual.

Se o usuário possuir uma imagem de um disco rígido virtual, também pode utilizá-la pela opção “Utilizar um disco rígido virtual existente”. Dessa forma,

dados e instalações podem ser replicados de uma máquina para outra.

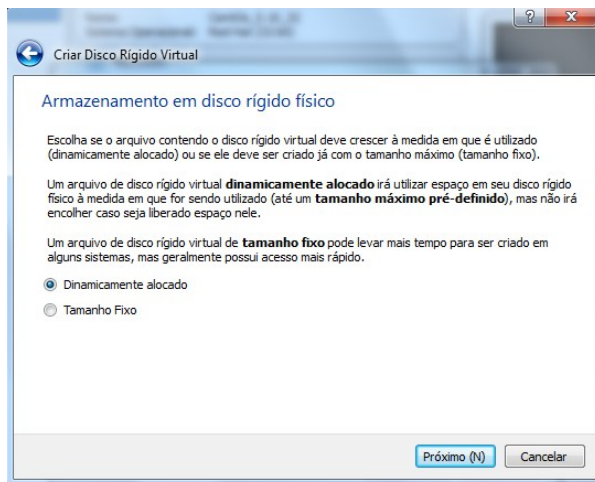
Neste exemplo criaremos um disco rígido virtual vazio. Assim ao criar o disco deve-se escolher:

- tipo de arquivo utilizado,
- tipo de alocação e
- tamanho do disco



Disco Rígido - Tipo de arquivo

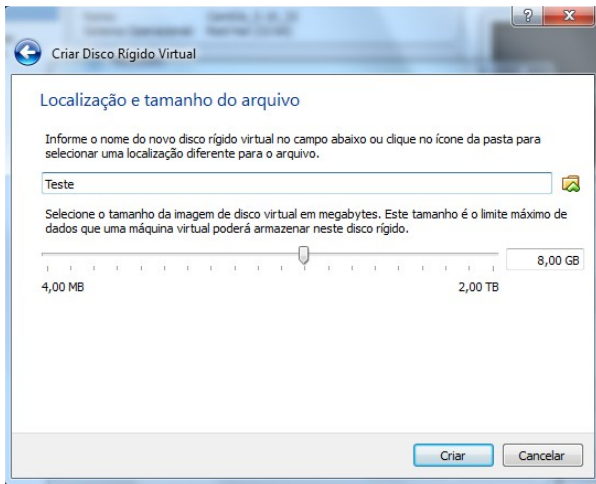
A escolha do tipo de arquivo do disco rígido é pessoal. Caso não haja necessidade de reutilizá-lo em outro virtualizador, pode-se criar um arquivo próprio do VirtualBox : “.vdi”



Disco Rígido - Tipo de alocação

A escolha do tipo de alocação depende da utilização:

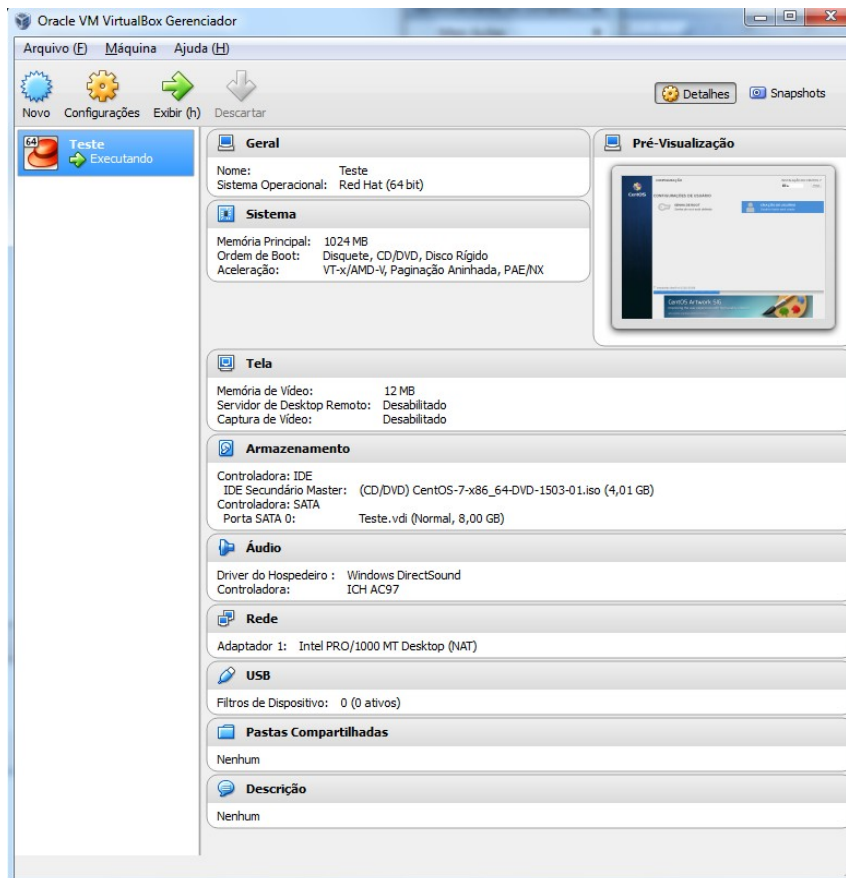
- A alocação dinâmica faz com que o disco virtual cresça conforme a demanda da máquina virtual, possui um processo de criação mais rápido, porém acesso mais lento, uma vez que os dados pode estar espalhados no disco.
- A alocação fixa reserva previamente o espaço pré-definido, possui criação mais lenta, porém acesso mais rápido.



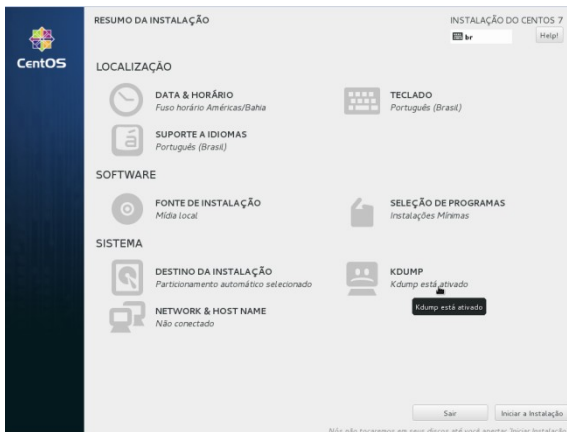
Disco Rígido - Tamanho

A escolha do tamanho do disco virtual vem com a necessidade. Aqui, alocaremos 8GB.

Uma vez criada a máquina virtual, suas características podem ser observadas na tela principal do virtualizador. A qualquer momento as opções podem ser alteradas através do *menu* “Máquina→Configurações”



O Próximo passo é a inicialização da máquina virtual. Basta clicar no botão “Iniciar”.

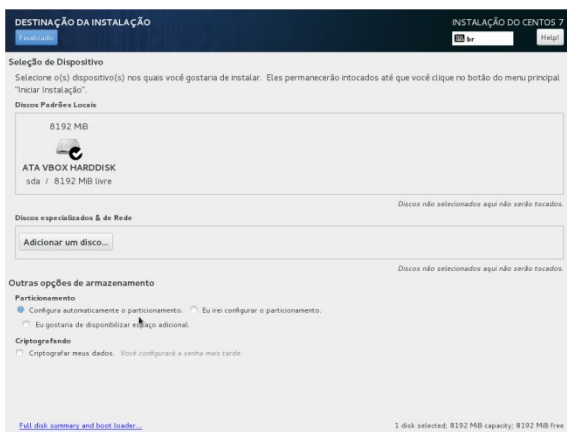


Resumo da instalação

O resumo da instalação define algumas propriedades importantes do sistema operacional que está sendo criado.

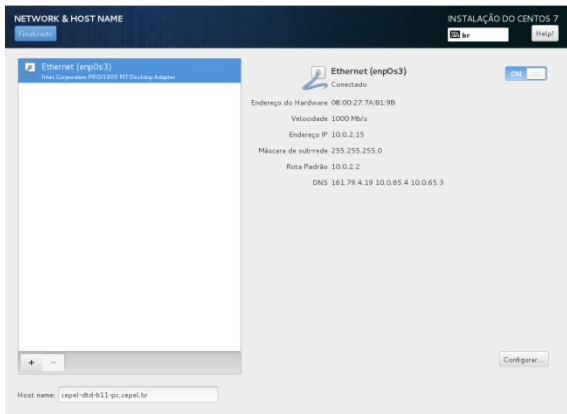
Alguns pontos importantes são:

- Destino da instalação
- Seleção de programas
- Network & Host name



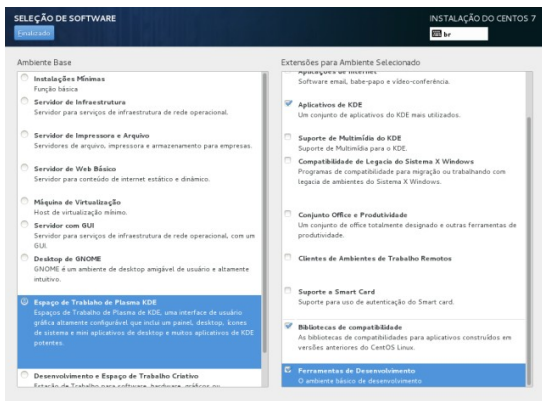
Destino da Instalação

É possível que apareça uma mensagem de atenção. O usuário deve entrar na tela e manter a configuração padrão como na figura ao lado.



Network & Host name

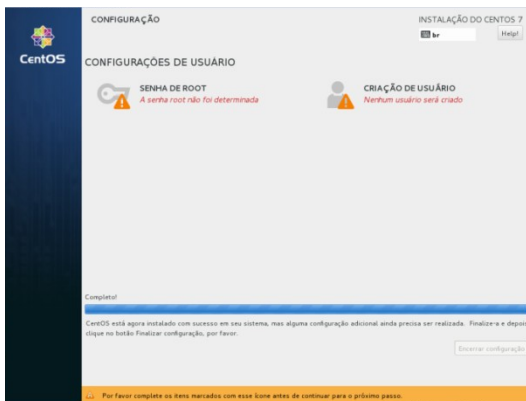
Para que a internet e conexões de rede seja configuradas na máquina virtual, a conexão ethernet deve ser indicada "on"



Seleção de software

Sugerimos a instalação de um ambiente de interface gráfica com alguns pacotes básicos conforme na figura ao lado.

Terminado o resumo da instalação:



Configurações

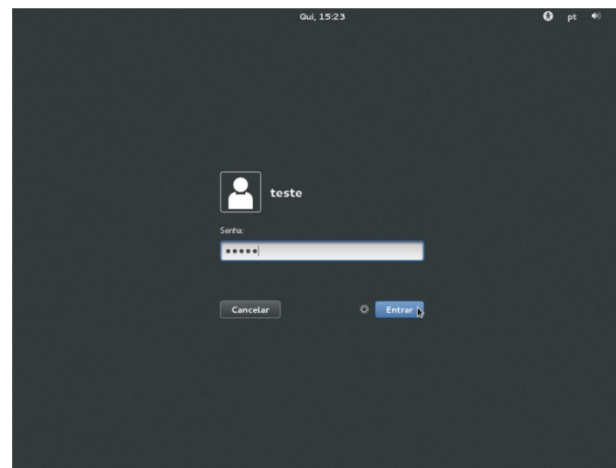
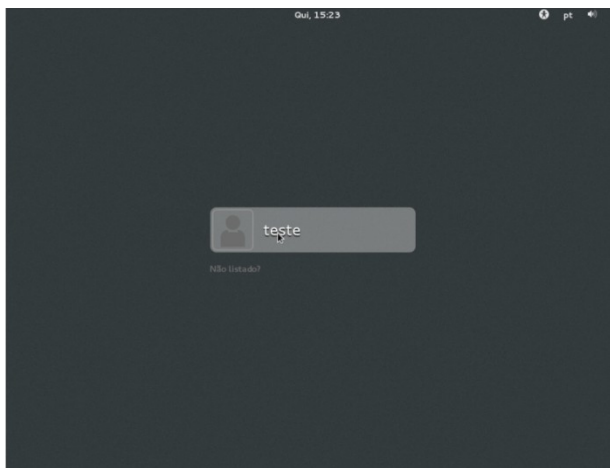
O usuário deve definir uma senha de root e um usuário para o novo sistema. E esperar que as configurações de instalação terminem.

Reinicialize ao terminar.

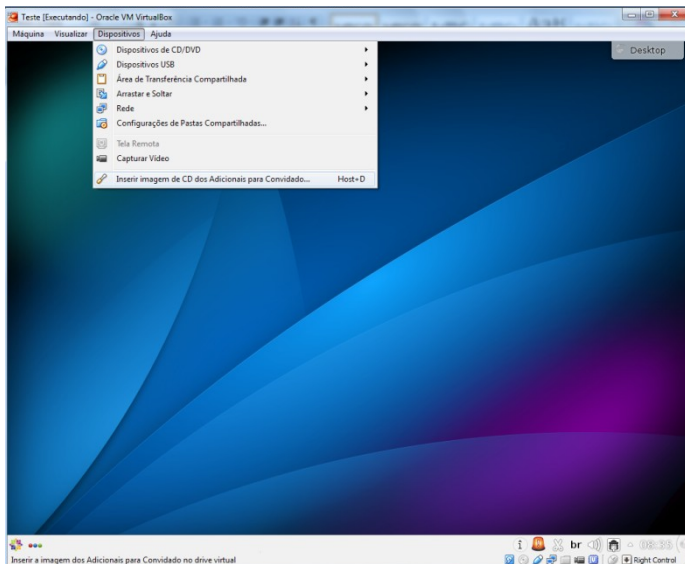
Senha root

Criar usuário

Agora a máquina virtual foi criada, faça o login com usuário e senha:



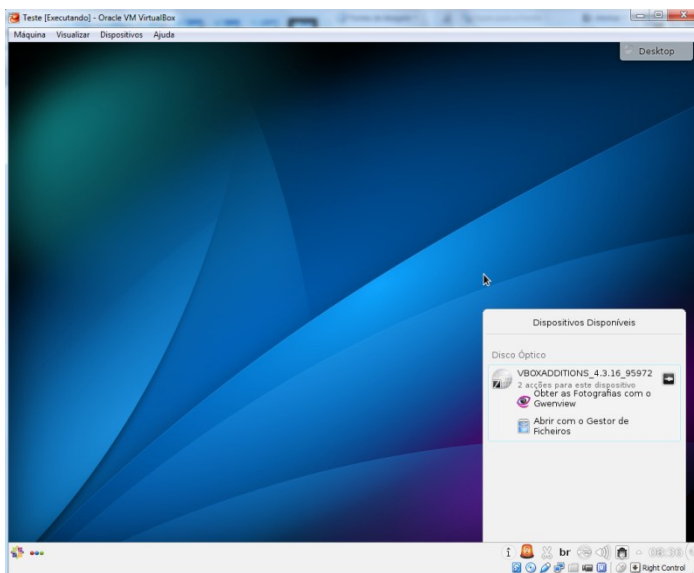
Próximo passo é instalar os adicionais da máquina virtual:



Visualizando os adicionais

No menu "dispositivos", selecione a opção "Inserir imagem do CD dos Adicionais para Convidado"

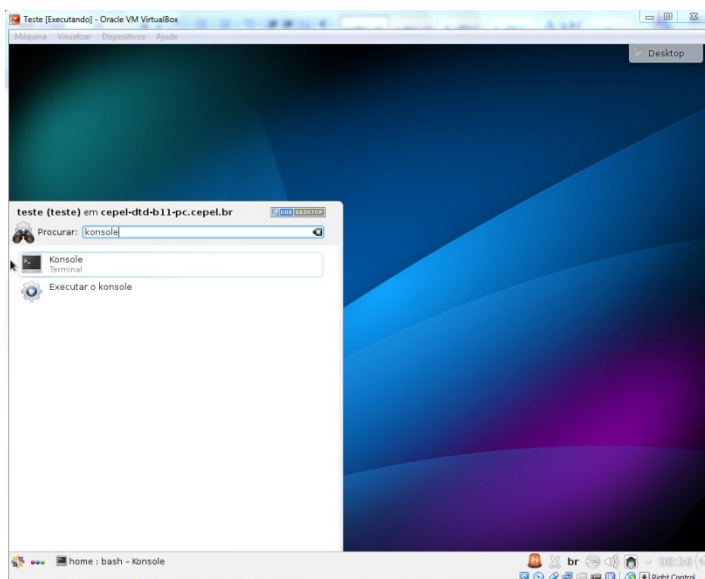
Assim será inserido o CD com as informações dos adicionais



Visualizando os adicionais

Monte o CD, clicando no botão ao lado do nome do disco.

Pose-se certificar-se a devida abertura do CD visualizando os arquivos no gestor de ficheiros. Em seguida abra um terminal Konsole pelo menu iniciar



Visualizando os adicionais

Em seguida abra um terminal Konsole pelo menu iniciar.

Digite konsole no espaço de busca ou vá em "Aplicações→Sistema→Terminal (konsole)"

```
home : bash - Konsole
Ficheiro Editar Ver Favoritos Configuração Ajuda
[teste@cepel-dtd-b11-pc home]$ su
Senha:
ABRT detectou 1 problema(s). Para mais informações execute: abrt-cli list
[root@cepel-dtd-b11-pc home]# cd /run/media/teste/VBOXADDITIONS_4.3.16_95972/
[root@cepel-dtd-b11-pc VBOXADDITIONS_4.3.16_95972]# ./VBoxLinuxAdditions.run
Verifying archive integrity... All good.
Uncompressing VirtualBox 4.3.16 Guest Additions for Linux.....
VirtualBox Guest Additions installer
Copying additional installer modules ...
Installing additional modules ...
Removing existing VirtualBox non-DKMS kernel modules      [ OK ]
Building the VirtualBox Guest Additions kernel modules     [ OK ]
Building the main Guest Additions module                  [ OK ]
Building the shared folder support module                  [ OK ]
Building the OpenGL support module                        [ OK ]
Doing non-kernel setup of the Guest Additions              [ OK ]
Starting the VirtualBox Guest Additions                    [ OK ]
Installing the window system drivers                      [ OK ]
Installing X.Org Server 1.15 modules                      [ OK ]
Setting up the window system to use the Guest Additions   [ OK ]
You may need to restart the hal service and the window system (or just restart the guest system) to enable the Guest Additions.

Installing graphics libraries and desktop services componen[ OK ]
[root@cepel-dtd-b11-pc VBOXADDITIONS_4.3.16_95972]#
```

Instalando os Adicionais

No terminal digite os comandos:

```
$ su
```

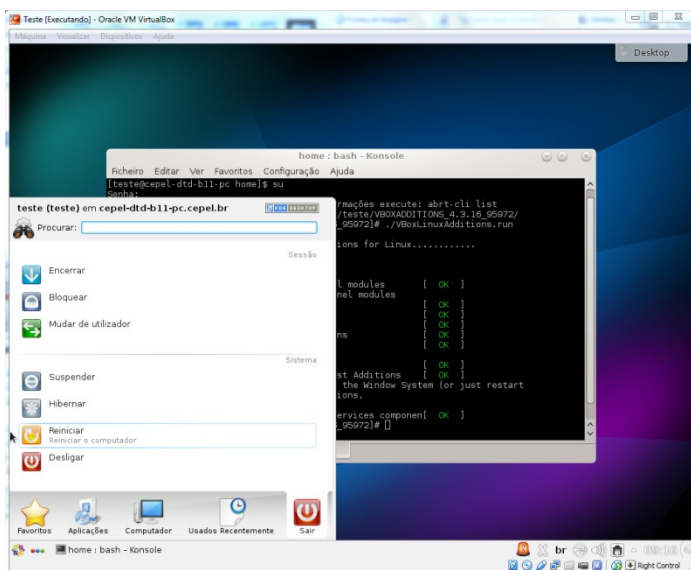
Para logar como root em seguida a senha previamente cadastrada

Em seguida:

```
# cd /run/media/teste/VBOXADDITIONS_4.3.16_95972
```

Para entrar na pasta do cd (deve-se observar qual a versão do vboxadditions utilizada para entrar na pasta corretamente). Execute com:

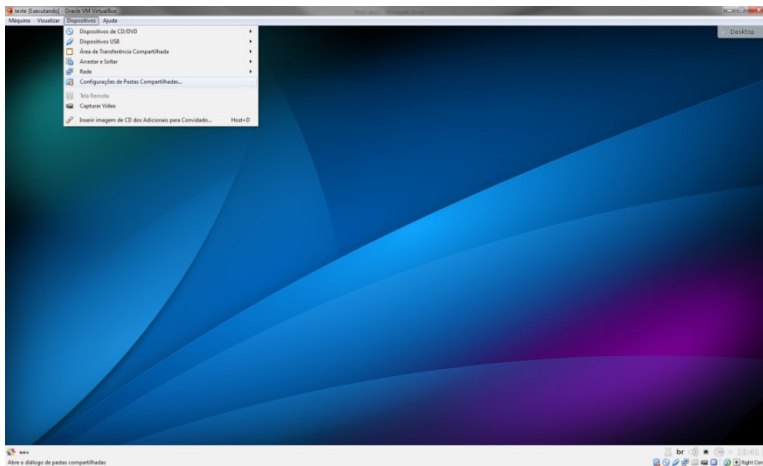
```
# ./VBoxLinuxAdditions.run
```



Verifique se os pacotes foram devidamente instalados.

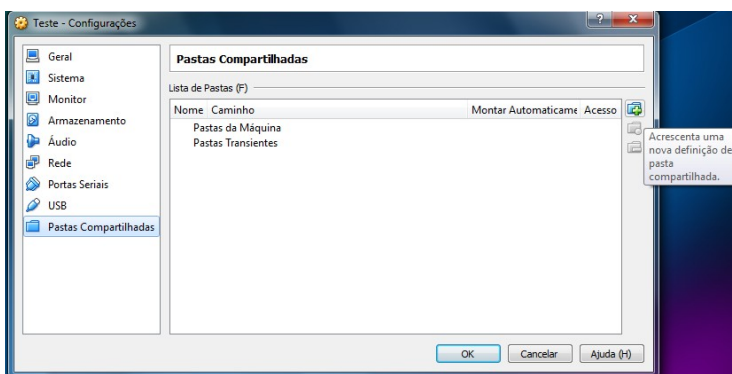
Reinicie a máquina.

O próximo passo consiste em habilitar a comunicação entre o computador Windows e a máquina virtual Linux, esse processo pode ser feito através do compartilhamento de pastas:



Compartilhar uma pasta

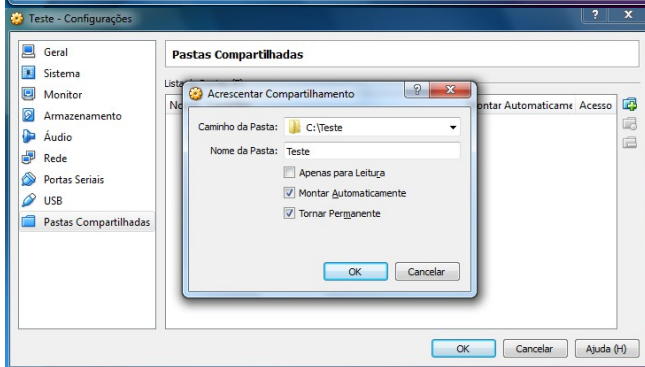
Acesse o menu: “Dispositivo→Configuração de Pasta Compartilhada”



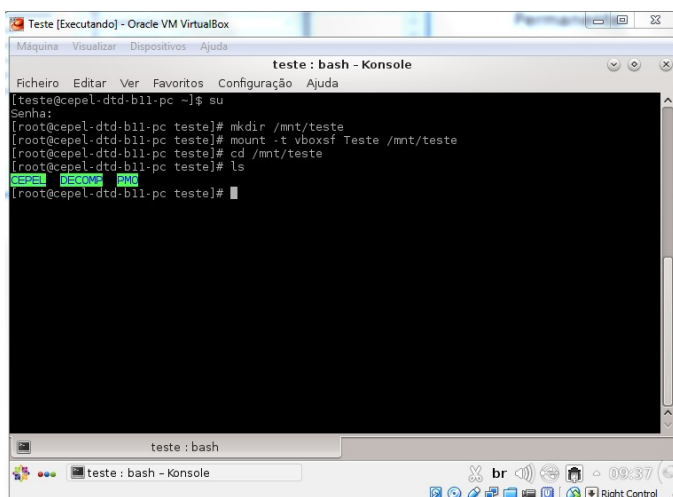
Compartilhar uma pasta

Acrescente uma pasta no botão de acrescentar pasta.

Em seguida, selecione a pasta dentro dos dados encontrados no Windows, essa pasta ficará visível para ambos os sistemas operacionais.



Marque as opções de Montar automaticamente e Tornar Permanente.



Montar a Pasta Compartilhada

A pasta que foi compartilhada deve ser montada no Linux. Abra um konsole - terminal.

`$ su`

Para logar como root e a senha

Criação do diretório onde será montado:

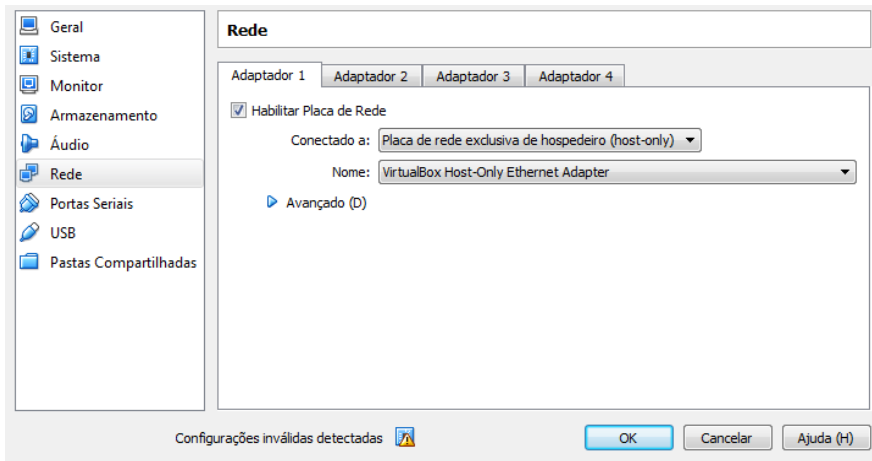
`# mkdir /mnt/teste`

Montagem feita na pasta

```
/mnt/teste:  
# mount -t vboxsf Teste  
/mnt/teste
```

Pode-se verificar que a pasta local teste pode ser acessada. Esta pasta, já previamente existente no Windows, contém três outros onde estão instalador e casos do DECOMP que serão utilizados na sequência deste tutorial.

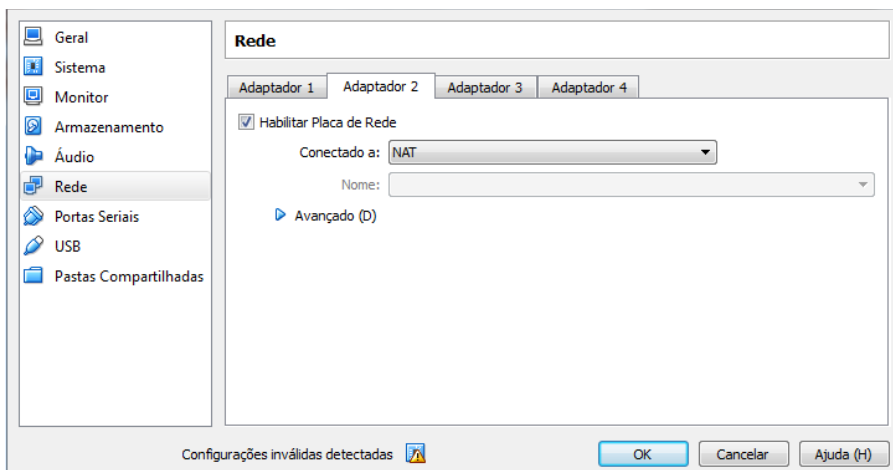
Alternativamente ao compartilhamento de pastas, pode ser realizada uma comunicação via IP.



Comunicação via IP

Com a máquina desligada abra o menu “Maquinas->Configuração”, item rede. Na opção “conectado a” selecione o item “placa de rede exclusiva de hospedeiro (host-only)”

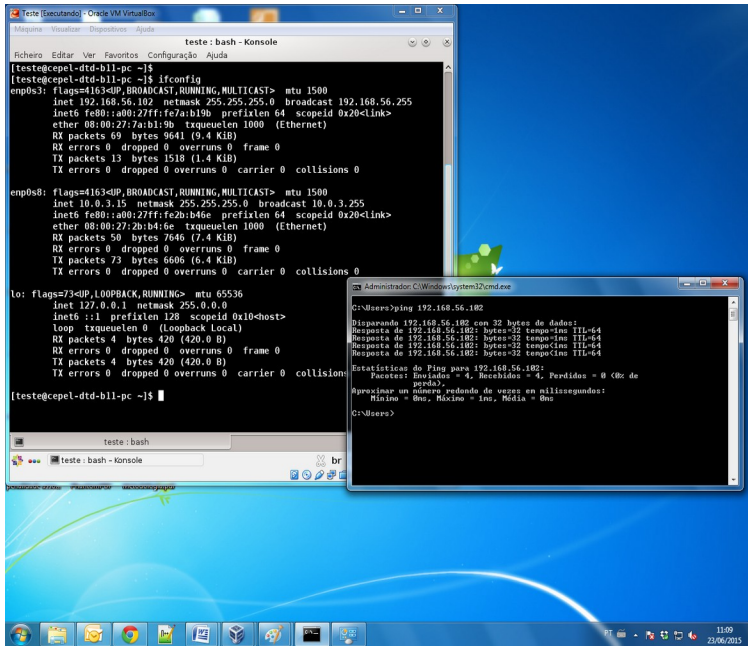
Para manter a conexão da máquina virtual à internet configure o adaptador 2 da seguinte forma:



Comunicação via IP

Na aba Adaptador 2 habilite a opção NAT

Para verificar se a conexão está funcionando ligue a máquina virtual e abra um console:



Comunicação via IP

Digite o comando:

`#ifconfig`

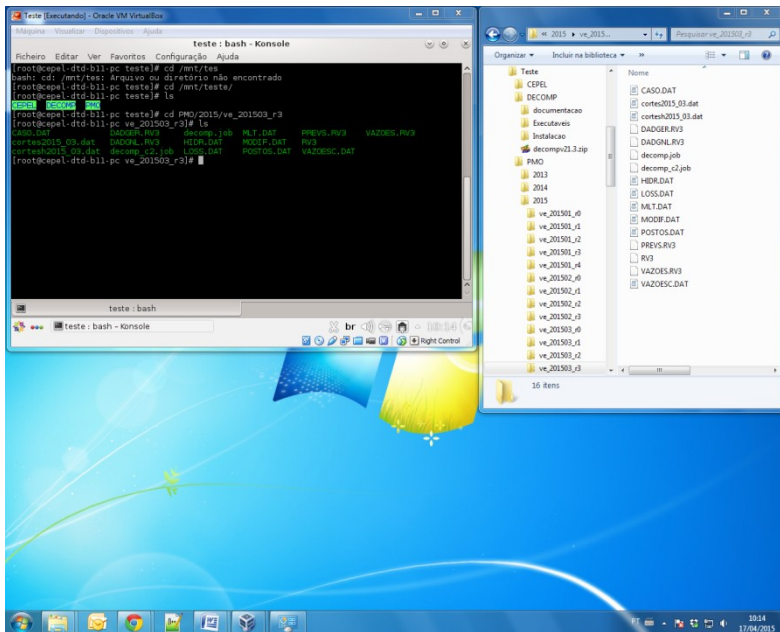
Observe o IP da máquina virtual. No Windows abra um terminal de comando cmd e digite o comando ping no IP, exemplo:

`#ping 192.168.56.102`

Verifique a comunicação das duas máquinas.

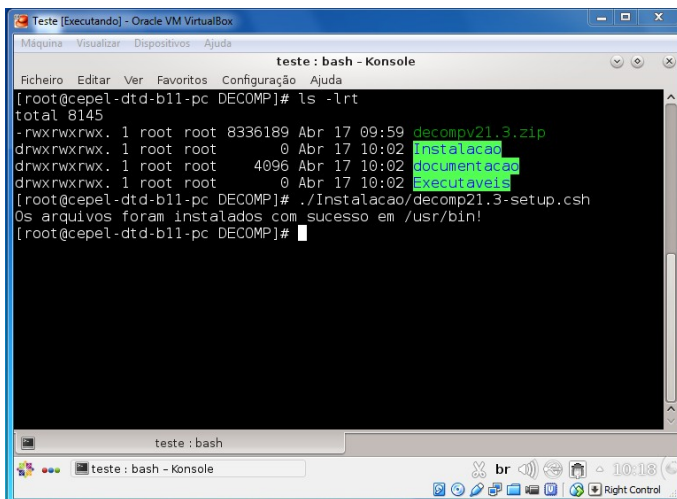
1.2 Como executar o DECOMP em uma máquina virtual Linux a partir da plataforma Windows

Esta segunda parte do tutorial mostra como usar o Programa DECOMP - Linux, na plataforma Windows por meio de uma máquina virtual. O processo é simples e rápido uma vez que a máquina virtual já está configurada. A máquina virtual passa a ser uma janela no sistema Windows assim como a janela de comandos cmd.



A pasta compartilhada entre os sistemas contém casos de PMO e o instalador do programa DECOMP Versão Linux.

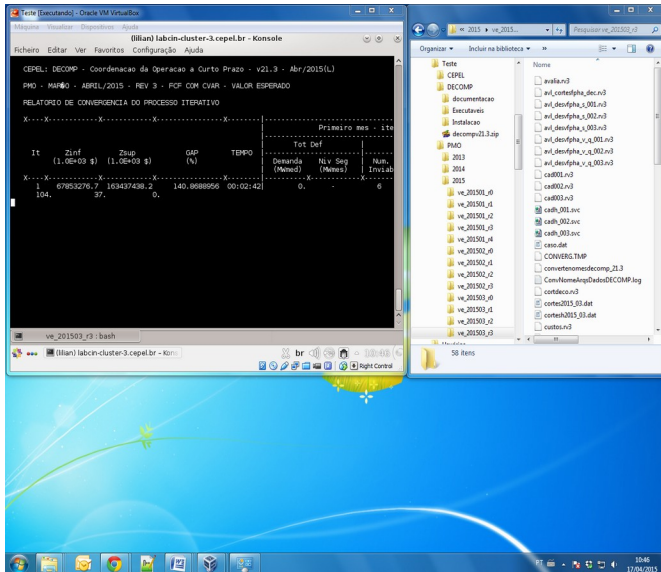
Essa pasta e suas sub pastas estão acessíveis para a máquina virtual, após a execução dos passos anteriores.



Instalação do DECOMP

Na pasta DECOMP tem-se os arquivos para instalação da versão Linux, basta executar o arquivo de setup via terminal:

```
./Instalacao/decomp21.3-setup.csh
```

Resultados

À medida que o caso roda na máquina virtual, os arquivos gerados com resultados já estão disponíveis na pasta compartilhada tanto no Windows quanto na máquina virtual!

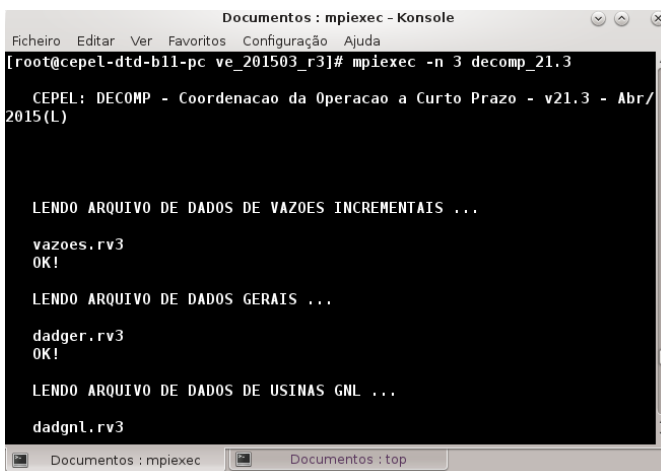
i) Execução do DECOMP em paralelo em uma máquina virtual Linux

Até então vimos como executar o DECOMP serialmente assim como é feito no Windows. A plataforma Linux permite que o DECOMP seja executado em paralelo melhorando o desempenho. Para executar o DECOMP em paralelo é preciso instalar a biblioteca MPICH ou MPICH2.

Faça download da versão desejada em: <http://www.mpich.org/>

Descompacte o arquivo e siga as instruções de instalação fornecidas pela biblioteca (normalmente arquivo README ou INSTALL dentro da pasta compactada).

Após instalado é possível executar o DECOMP em paralelo através do MPIEXEC



Executando o DECOMP em paralelo

Na pasta aonde está o caso a ser rodado execute o DECOMP com o seguinte comando:

```
# mpiexec -n <N>
decomp_21.3
```

Onde <N> é o número de processos usados.


```

top - 14:08:29 up 21 min, 4 users, load average: 7.04, 4.51, 2.74
Tasks: 187 total, 5 running, 182 sleeping, 0 stopped, 0 zombie
%Cpu(s): 34.2 us, 43.8 sy, 0.0 ni, 20.5 id, 0.4 wa, 0.0 hi, 1.0 si, 0.0 st
KiB Mem : 1884472 total, 999036 free, 692000 used, 193436 buff/cache
KiB Swap: 839676 total, 485304 free, 354372 used, 1028324 avail Mem

  PID USER      PR  NI   VIRT   RES   SHR  S  %CPU  %MEM     TIME+ COMMAND
 4106 root      20   0 1542164 175704 2268  R  92.9  9.3   0:06.95 decomp 21.3
 4104 root      20   0 1542540 179096 2412  R  92.6  9.5   0:06.21 decomp 21.3
 4105 root      20   0 1542164 186848 2268  R  91.7  9.9   0:06.70 decomp 21.3
 3208 teste    20   0 1646196  7708  1216  S   7.4  0.4   0:55.11 knotify4
 3735 teste    20   0 638324  14188  7160  S   5.3  0.8   0:39.24 konsole
 1298 root     20   0 228352  11336  3728  S   2.4  0.6   0:59.72 Xorg
 3240 teste    20   0 1489712  3504    0  S   2.1  0.2   0:20.04 mysqld
 4094 teste    20   0 130024   880   448  R   2.1  0.0   0:09.01 top
 3211 teste    20   0 1174884 33524 10452  S   1.8  1.8   0:53.24 plasma-des+
 3317 teste    9  -11 560280   724   336  S   0.9  0.0   0:12.92 pulseaudio
 3176 teste    20   0 636480  5740 3564  S   0.6  0.3   0:20.18 kwin
 3229 root     20   0 177436   680   436  S   0.6  0.0   0:02.00 cupsd
 3290 teste    20   0 446192   616   432  S   0.6  0.0   0:05.57 akonadi_ag+
  14 root     20   0    0      0    0  S   0.3  0.0   0:06.75 rcuotf/0
  48 root     99  19    0      0    0  S   0.3  0.0   0:00.30 khugepaged
 316 root     0 -20    0      0    0  S   0.3  0.0   0:10.70 kworker/0:+
 653 root     20   0 19152  368  288  S   0.3  0.0   0:03.43 irqbalance
 755 root     20   0 324352  140   52  S   0.3  0.0   0:06.38 VBoxService

```

Executando o DECOMP em paralelo

Ao executar mpiexec com 3 processos podemos observar através do comando top os 3 processos criados e sendo executados em diferentes processadores.

ii) Execução do DECOMP através de um sistema de filas (opcional)

Em clusters onde se tem muitos nós (conjuntos de processamentos) é necessária a utilização de um sistemas de filas para gerenciamento da execução de processos e uso dos nós. Em caso de máquinas virtuais é possível criar um sistema de gerenciamento de filas, com a particularidade de conter somente um nó, o servidor. O sistema de filas não é compulsório para rodar o DECOMP em paralelo, porém ele provê alguns recursos computacionais interessantes.

O uso do sistema de filas necessita, além da instalação, configuração e manuseio do mesmo, exigindo-se um conhecimento prévio mínimo em Linux e no sistema torque-PBS.

Para instalação do torque-PBS faça o download na página:

<http://www.adaptivecomputing.com/products/open-source/torque/>

Antes de iniciar, instale os pré-requisitos necessários através dos comandos:

```

$ su
# yum update
# yum install libxml2-devel openssl-devel gcc gcc-c++ boost-devel

```

Descompacte o arquivo e entre na pasta criada:

```

# tar -xzf torque-__versao__.tar.gz
# cd torque-__versao__

```

Instale o torque:

```

# ./configure
# make
# make install

```

Uma vez instalado o torque tem que ser configurado. Aqui criaremos um servidos com um nó apenas:

Copie os arquivos para o servidos e confira os resultados:

```
# cp contrib/init.d/trqauthd /etc/init.d/  
# cp contrib/init.d/pbs_mom /etc/init.d/pbs_mom  
# cp contrib/init.d/pbs_server /etc/init.d/pbs_server  
# cp contrib/init.d/pbs_sched /etc/init.d/pbs_sched  
# chkconfig --add trqauthd  
# chkconfig --add pbs_mom  
# chkconfig --add pbs_server  
# chkconfig --add pbs_sched  
# echo '/usr/local/lib' > /etc/ld.so.conf.d/torque.conf  
# ldconfig  
# service trqauthd start
```

Criando um servidor pbs:

```
# ./torque.setup root
```

No arquivo /var/spool/torque/server_priv/nodes coloque o seguinte conteúdo:

```
MYSERVER np=N
```

Onde MYSERVER é o nome do servidor (pode ser encontrado em /var/spool/toque/server_name) e N é o número de processadores do nó (de acordo com a máquina virtual utilizada).

No arquivo ./var/spool/torque/mom_priv/config coloque o seguinte conteúdo:

```
$pbsserver MYSERVER
```

```
$logevent 255
```

Para reinicializar o servidor digite:

```
# qterm -t quick  
# pbs_server
```

Verifique os nós do sistema com:

```
# pbsnodes -a
```

Inicie o scheduler

```
# pbs_sched
```

Por ultimo crie uma fila:

```
# qmgr -c "create queue decomp_fila queue_type=execution"  
# qmgr -c "set queue test enabled=true"  
# qmgr -c "set queue test started=true"  
# qmgr -c "set server scheduling=True"
```

PBS está configure e pronto para executar Jobs PBS. Use:

qsup : para submeter um job

qstat: para verificar o status da fila

qdel : para derrubar um job na fila

2 CONTAINERS

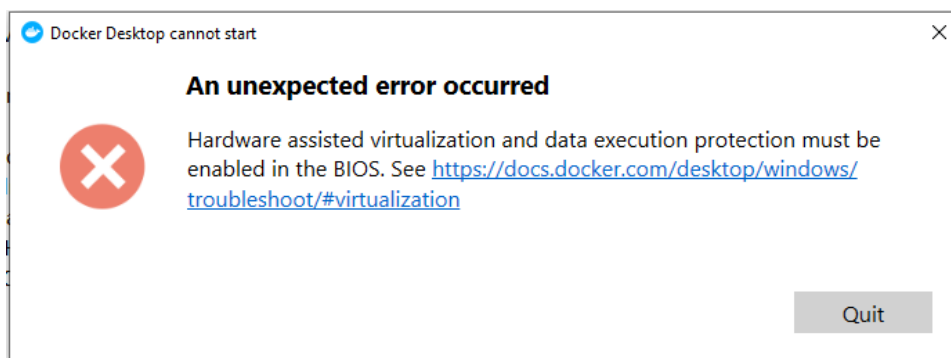
2.1 INSTALAÇÃO DE UM AMBIENTE DE SUPORTE DE CONTAINERS

Um ambiente de suporte de containers deve ser escolhido pelo usuário. Neste tutorial vamos exemplificar a rodada do modelo DECOMP utilizando o docker e docker-compose (<https://docs.docker.com>).

Inicialmente é preciso instalar a o docker descktop para Windows, as instruções para instalação podem ser encontradas no site do programa: <https://docs.docker.com/desktop/windows/install/> . Este tutorial está descrito para instalação no sistema Windows 10 Professional, para outras versões pode ser necessário executar passos adicionais.

Durante a instalação e/ou execução podem ocorrer alguns erros, aqui estão pontuadas algumas soluções para erros comuns:

Caso após a instalação do docker e a reinicialização do sistema apareça a seguinte mensagem de erro



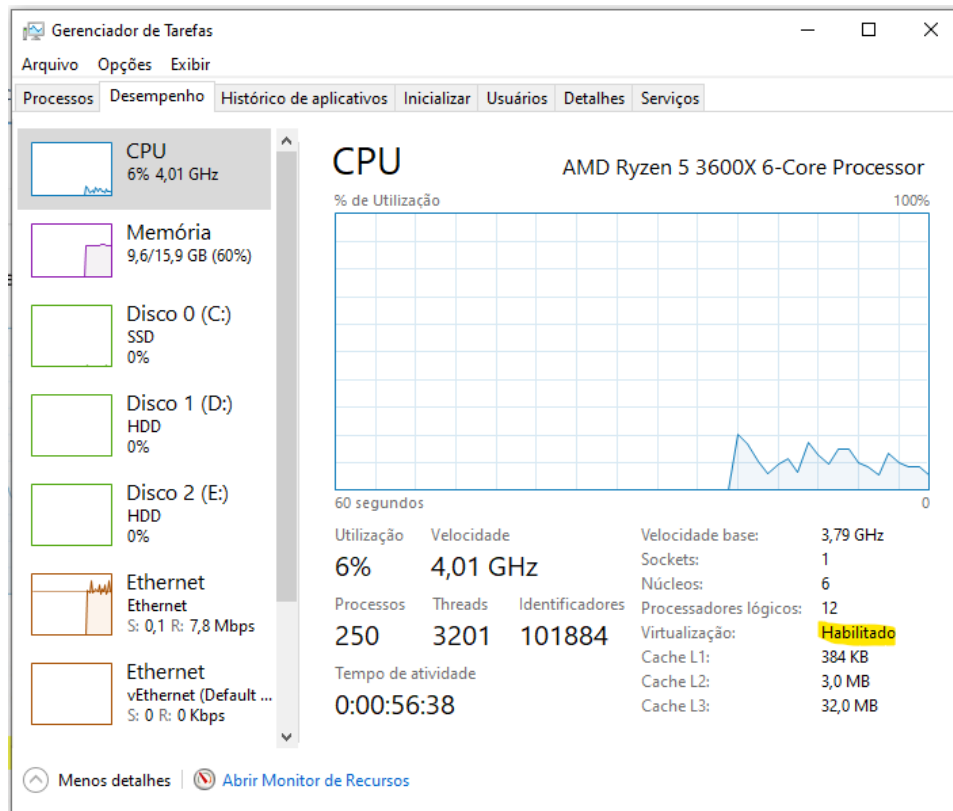
Os seguintes passos devem ser executados:

- Abrir o “Ativar ou desativar recursos do Windows”
- Ligar a opção “Plataforma de máquina virtual”
- Ligar a opção “Subsistema do Windows para Linux”
- Ligar a opção “Hyper V”
- Reiniciar o sistema e testar

Caso continue com o problema continue realizar os seguintes passos

- Abrir o prompt de comando como administrador e executar o comando “`bcdedit /set hypervisorlaunchtype auto`”
- Reiniciar o sistema e testar

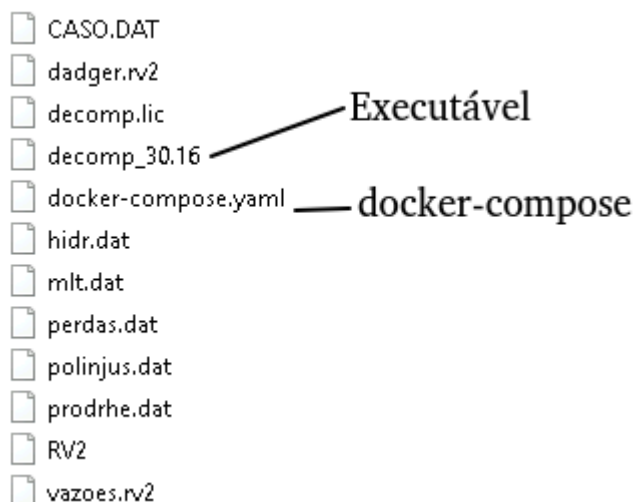
- Caso não resolva, seguir o tutorial descrito no site: [Passo a Passo](#)
- Checar se a visualização está ativada no Gerenciador de Tarefas



- Caso não esteja é necessário ativar a visualização na BIOS.

2.2 EXECUÇÃO DO DECOMP EM UM CONTAINER LUNUX A PARTIR DA PLATAFORMA WINDOWS

Uma vez instalado, é preciso preparar um arquivo docker-compose.yaml, para facilitar este arquivo deve estar na mesma localidade do caso executado e do executável como no exemplo:



O conteúdo deste arquivo deve ser:

```
version: "3.3"

services:
  decomp:
    image: ubuntu:latest
    container_name: decomp
    entrypoint: ["/bin/bash","-c", "/caso/decomp_30.16"]
    working_dir: /caso/
    volumes:
      - "./:/caso:Z"
```

Onde:

“image”: é a imagem do sistema onde será feita a execução, o sistema ubuntu é compatível com o decomp.

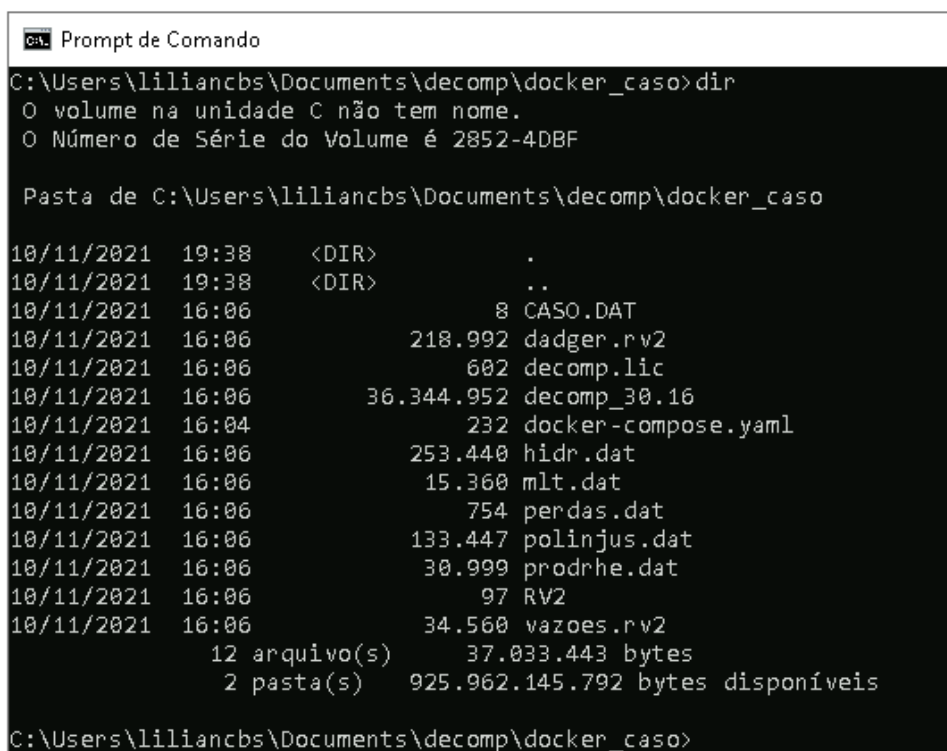
“container_name”: é o nome escolhido para o container.

“entrypoint”: é o comando a ser executado do container (no caso estamos executando o modelo DECOMP versão 30.16 que está localizado na pasta /caso/).

“working_dir”: é o diretório de trabalho.

“volumes”: é o mapeamento do diretório Windows no Linux, no caso em questão vamos mapear o diretório corrente “./”, pois é onde será executado o DECOMP, na pasta caso, que é o diretório de trabalho.

Após criar o arquivo, basta abrir um terminal (cmd) do Windows e navegar até a pasta onde está o caso, o executável e o arquivo docker-compose.yaml:



```
cmd Prompt de Comando
C:\Users\liliancbs\Documents\decomp\docker_caso>dir
O volume na unidade C não tem nome.
O Número de Série do Volume é 2852-4DBF

Pasta de C:\Users\liliancbs\Documents\decomp\docker_caso

10/11/2021  19:38    <DIR>          .
10/11/2021  19:38    <DIR>          ..
10/11/2021  16:06             8 CASO.DAT
10/11/2021  16:06          218.992 dadger.rv2
10/11/2021  16:06             602 decomp.lic
10/11/2021  16:06        36.344.952 decomp_30.16
10/11/2021  16:04             232 docker-compose.yaml
10/11/2021  16:06          253.440 hidr.dat
10/11/2021  16:06          15.360 mlt.dat
10/11/2021  16:06             754 perdas.dat
10/11/2021  16:06          133.447 polinjus.dat
10/11/2021  16:06           30.999 prodrhe.dat
10/11/2021  16:06             97 RV2
10/11/2021  16:06          34.560 vazoes.rv2
                12 arquivo(s)        37.033.443 bytes
                2 pasta(s)    925.962.145.792 bytes disponíveis

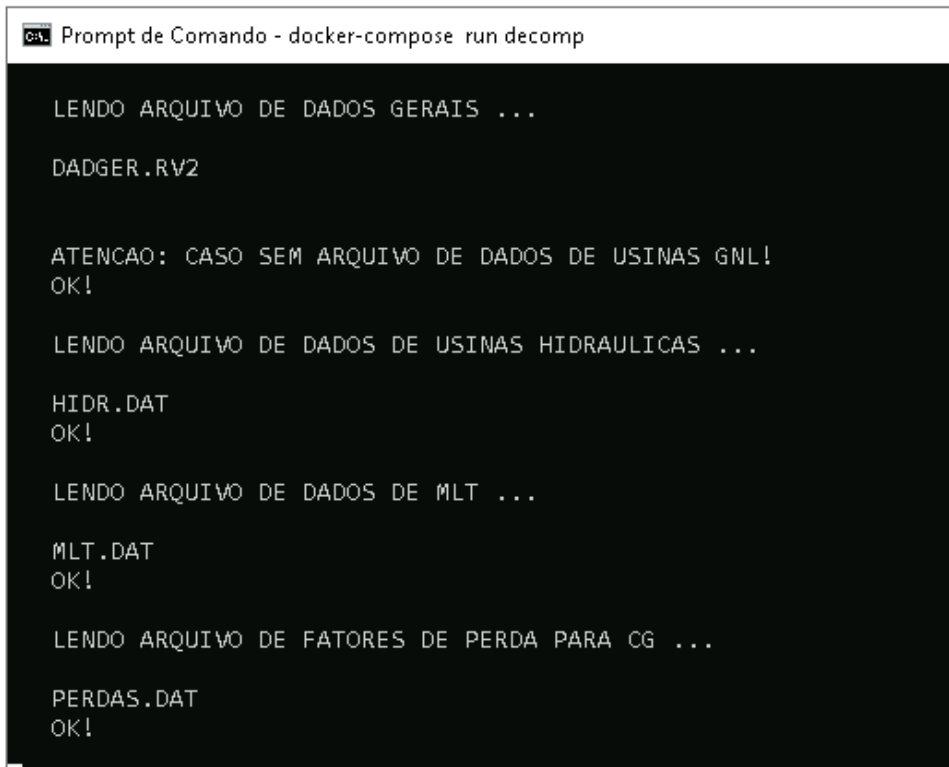
C:\Users\liliancbs\Documents\decomp\docker_caso>
```

E executar o comando `docker-compose run <nome do container>`, conforme abaixo:

```
C:\Users\liliancbs\Documents\decomp\docker_caso>docker-compose run decomp_
```

O container será criado e o DECOMP será executado na sequência:

```
C:\Users\liliancbs\Documents\decomp\docker_caso>docker-compose run decomp
Creating network "docker_caso_default" with the default driver
Creating docker_caso_decomp_run ... done
```



```
Prompt de Comando - docker-compose run decomp

LENDO ARQUIVO DE DADOS GERAIS ...

DADGER.RV2

ATENCAO: CASO SEM ARQUIVO DE DADOS DE USINAS GNL!
OK!

LENDO ARQUIVO DE DADOS DE USINAS HIDRAULICAS ...

HIDR.DAT
OK!

LENDO ARQUIVO DE DADOS DE MLT ...

MLT.DAT
OK!

LENDO ARQUIVO DE FATORES DE PERDA PARA CG ...

PERDAS.DAT
OK!
```

É possível manter o arquivo `docker-compose.yaml` fora da pasta do caso, assim como o executável, basta configurar o `docker-compose.yaml` de outras formas.