REMOTE api call
Hands on

Introduction:

1. It is assumed that the „AutoDock" gUSE workflow application has been developed by a third party and it is available. This workflow is runnable on a local submitter.
2. Your task is to submit the workflow from your client machine on the server http://192.168.143.129:8080/remote/RemoteServlet

Major Steps:

I. **Preparation:**

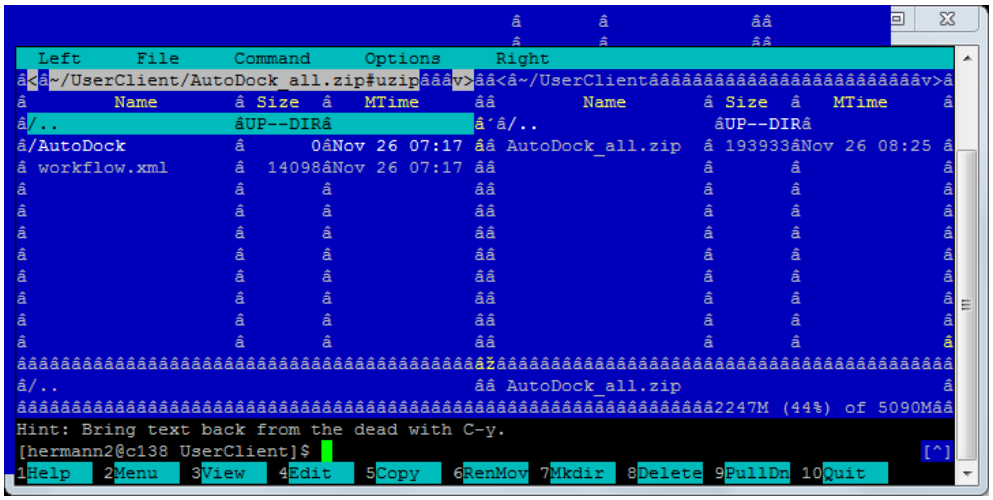
Creating the workflow descriptor files „**workflow.xml**" „**portmapping.txt**" „**inputs.zip**" from the original workflow definition.

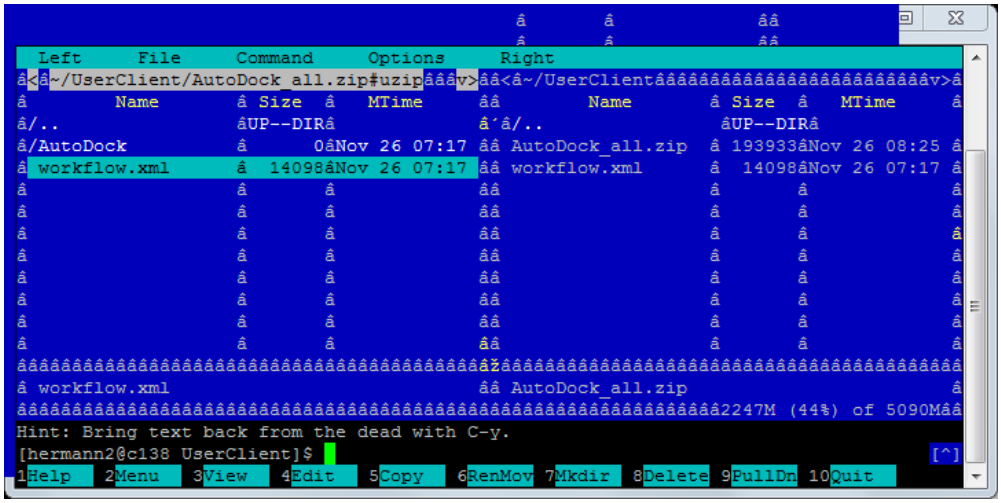
1. Download the following file to your local machine. It contains the compressed gUSE code of the AutoDock workflow.

   For example:

   ```
   mkdir UserClient
   cd UserClient
   wget
   https://www.lpds.sztaki.hu/services/sw/download.php?download=aaee
   a21f486f345eebe61db01db81e70
   ```

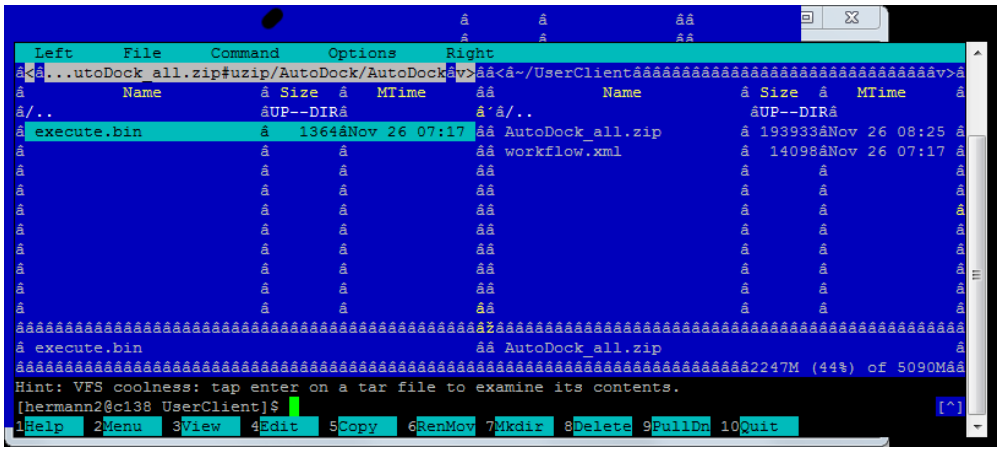
2. Separate the file „**workflow.xml**" file within the „AutoDock_all.zip" file.
   (Use the midnight commander (mc) for copy it, you may use the same name as destination):

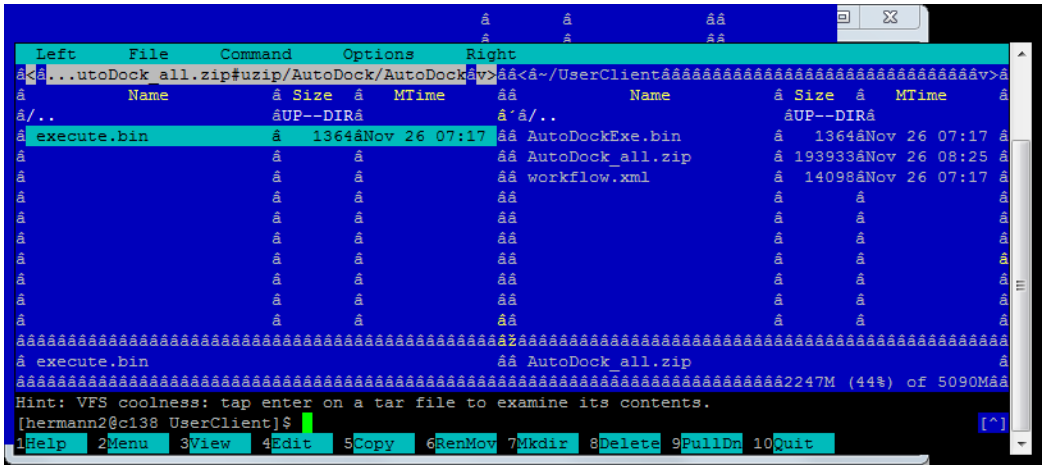

Left File Command Options Right
~/UserClient/AutoDock_all.zip#uzip
~/UserClient
Name Size MTime
/..
UP--DIR
/AutoDock 0 Nov 26 07:17
workflow.xml 14098 Nov 26 07:17
AutoDock_all.zip 193933 Nov 26 08:25
Hint: Bring text back from the dead with C-y.
[hermann2@c138 UserClient]$
1Help 2Menu 3View 4Edit 5Copy 6RenMov 7Mkdir 8Delete 9PullDn 10Quit

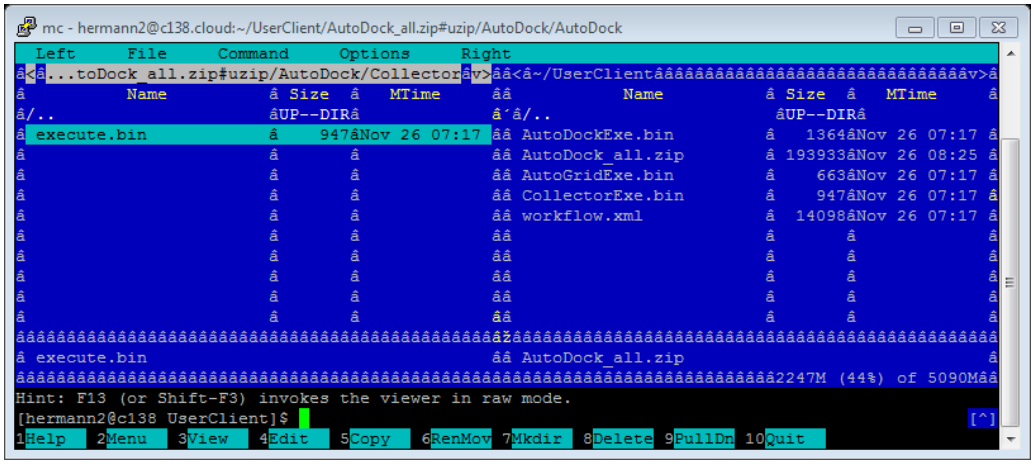

Left File Command Options Right
~/UserClient/AutoDock_all.zip#uzip
~/UserClient
Name Size MTime
/..
UP--DIR
/AutoDock 0 Nov 26 07:17
workflow.xml 14098 Nov 26 07:17
AutoDock_all.zip 193933 Nov 26 08:25
workflow.xml 14098 Nov 26 07:17
Hint: Bring text back from the dead with C-y.
[hermann2@c138 UserClient]$
1Help 2Menu 3View 4Edit 5Copy 6RenMov 7Mkdir 8Delete 9PullDn 10Quit

3. Find and copy the **executable files**.
   Hint: go to sub directory „AutoDock” where the included jobs („AutoDock”,”AutoGrid”,”Collector”) are listed. Go inside the jobs and copy the files named „execute.bin” with separate names.

   Example:



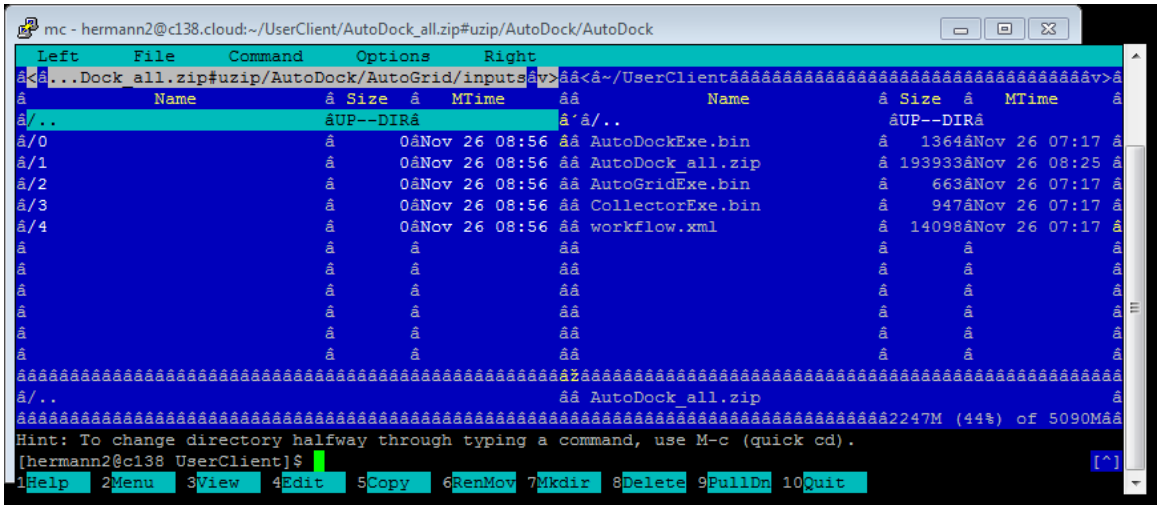


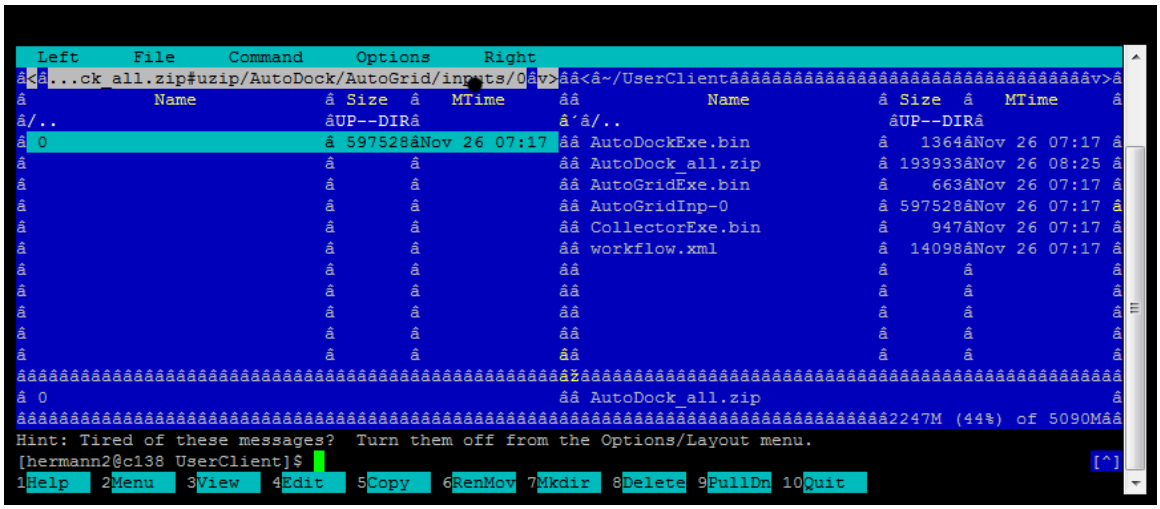


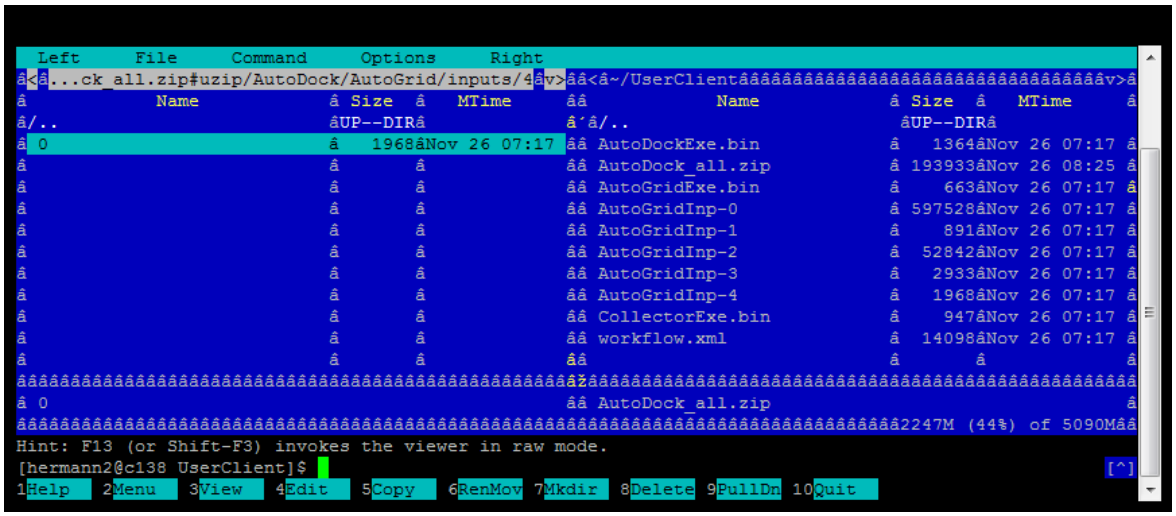
4. Find and copy the **input files**:

   Hint: They all have the common name „0" and are located within the job „AutoGrid" and listed within separate ports coded as sub directories. You have to rename them.

   Example:







5. Create the association file „**portmapping.txt**”:
   (remember the right hand side should have the following form:
   <workflow_name><job_name><input_port_number> for free input files and
   <workflow_name><job_name> for job executables

```
AutoGridInp-0=AutoDock/AutoGrid/0
AutoGridInp-1=AutoDock/AutoGrid/1
AutoGridInp-2=AutoDock/AutoGrid/2
AutoGridInp-3=AutoDock/AutoGrid/3
AutoGridInp-4=AutoDock/AutoGrid/4
AutoDockExe.bin=AutoDock/AutoDock
AutoGridExe.bin=AutoDock/AutoGrid
CollectorExe.bin=AutoDock/Collector
```

6. Create the file „**inputs.zip**” from the listed inputs:

```
zip inputs.zip AutoGridInp-0 AutoGridInp-1 AutoGridInp-2
AutoGridInp-3 AutoGridInp-4 AutoGridExe.bin AutoDockExe.bin
CollectorExe.bin
```

II. Submit the workflow

1. Create the submitter command „**LocalSubmit.sh**”:
   (Hint: you can download „LocalSubmit.sh” from here)

```
#!/bin/bash
#submit without certs (local submitter)
curl -k -F m=submit \
            -F pass=password \
            -F wfdesc=@workflow.xml \
            -F inputzip=@inputs.zip \
            -F portmapping=@portmapping.txt
http://192.168.143.129:8080/remote/RemoteServlet
```

2. Create an observation command „**Status.sh**” :
   (Hint: you can download „Status.sh” from here)

```
#!/bin/bash
#get WF status
```

```
echo $1
curl -k -F m=detailsinfo \
        -F ID=$1 \
        -F pass=password \
        http://192.168.143.129:8080/remote/RemoteServlet
```

3. Create an output reading command „**GetOutput.sh**”:
   (Hint: you can download GetOutput.sh from here)

```
#!/bin/bash
#get output and cleanup
echo $1
curl -k -F m=download \
        -F ID=$1 \
        -F pass=password \
        -o output.zip \
        http://192.168.143.129:8080/remote/RemoteServlet
```

4. Submit the workflow:
   Example:

   ./LocalSubmit.sh
   1353924639496876

   Notice that it returns a unique identifier which must be used as argument in the subsequent **Status** and **GetOutput** commands.

5. Observe the progress of the workflow:
   Example:

```
./Status.sh 1353924639496876
1353924639496876
running;AutoGrid:init=0:running=0:finished=1:error=0;AutoDock:ini
t=0:running=4:finished=1:error=0;Collector:init=1:running=0:finis
hed=0:error=0;

./Status.sh 1353924639496876
1353924639496876
finished;AutoGrid:init=0:running=0:finished=1:error=0;AutoDock:in
it=0:running=0:finished=5:error=0;Collector:init=0:running=0:fini
shed=1:error=0;
```

6. Fetch the output:

```
./GetOutput.sh 1353924639496876
1353924639496876
  % Total    % Received % Xferd  Average Speed   Time    Time
Time  Current
                                 Dload  Upload   Total   Spent
Left  Speed
100 3322k    0 3322k  100   352  6279k    665 --:--:-- --:--:-- -
-:--:-- 6527k
```

7. The output has been created in the **output.zip** file: