ABAQUS 2018 and Fortran Compilers on Ubuntu 18.04LTS

Georgios TZORTZOPOULOS

14/01/2021

Installation steps of Abaqus 2018 on Ubuntu 18.04LTS

In this guide, I will explain you from scratch, the steps which are required in order to install Abaqus 2018 on a recently installed Ubuntu 18.04LTS on Windows 10 OS.

1. Install prerequisites

(a) First you have to update all the packages and libraries of Ubuntu 18.04LTS:

```
$ sudo apt-get update
$ sudo apt-get upgrade
```

(b) Then you should install some basic packages and libraries in order to successfully install Abaqus and to edit some files afterwards:

```
$ sudo apt-get install gedit csh tcsh ksh gcc g++ gfortran libstdc++5
build-essential make libjpeg62 libmotif-dev libglu1-mesa
```

(c) Finally, you should install Xming (or equivalent):

```
https://sourceforge.net/projects/xming/
```

2. Edit all Linux.sh files

Apart from Ubuntu, you should edit some installation files as well. Go to the downloaded folders of Abaqus 2018 and search for all `Linux.sh` files. Open them with an editor and edit them as follows:

(a) Find the 9th line:

```
DSY_OS_Release=‘lsb_release --short --id |sed ’s/ //g’‘
```

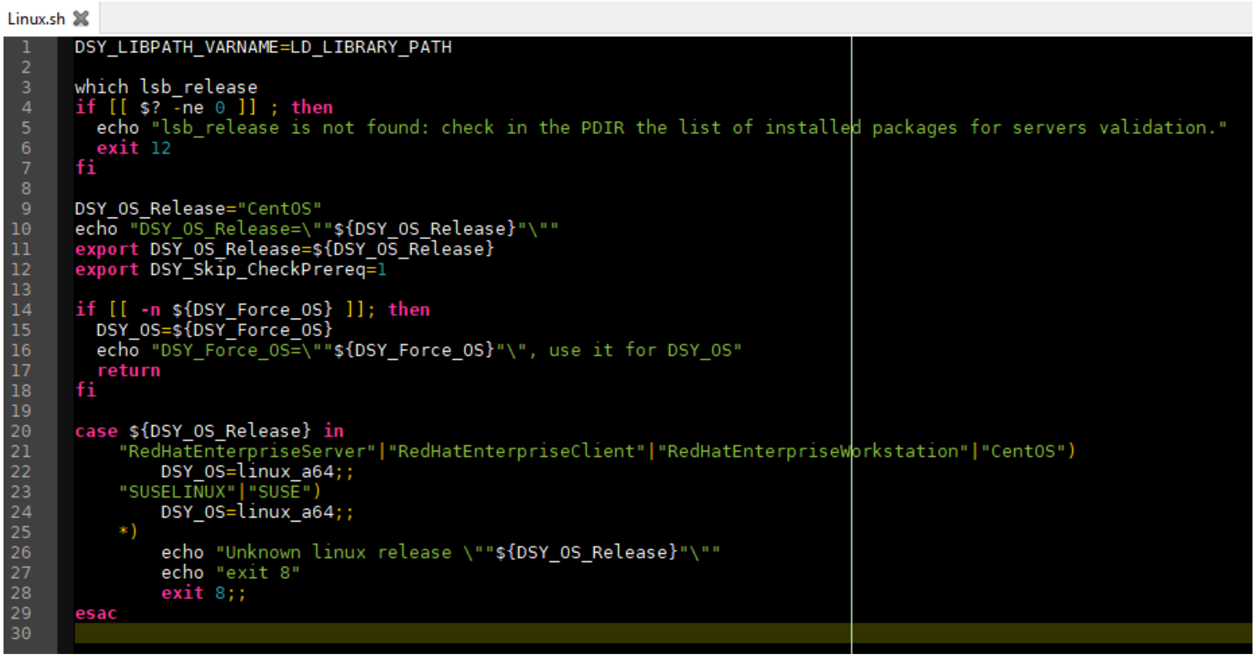
(b) And edit it as below:

```
DSY_OS_Release="CentOS"
```

(c) Also, enter a new line below the 11th line:

```
export DSY_Skip_CheckPrereq=1
```

So, the `Linux.sh` files should look like the one below:



```
 1  DSY_LIBPATH_VARNAME=LD_LIBRARY_PATH
 2
 3  which lsb_release
 4  if [[ $? -ne 0 ]] ; then
 5    echo "lsb_release is not found: check in the PDIR the list of installed packages for servers validation."
 6    exit 12
 7  fi
 8
 9  DSY_OS_Release="CentOS"
10  echo "DSY_OS_Release=\""${DSY_OS_Release}"\""
11  export DSY_OS_Release=${DSY_OS_Release}
12  export DSY_Skip_CheckPrereq=1
13
14  if [[ -n ${DSY_Force_OS} ]]; then
15    DSY_OS=${DSY_Force_OS}
16    echo "DSY_Force_OS=\""${DSY_Force_OS}"\", use it for DSY_OS"
17    return
18  fi
19
20  case ${DSY_OS_Release} in
21      "RedHatEnterpriseServer"|"RedHatEnterpriseClient"|"RedHatEnterpriseWorkstation"|"CentOS")
22          DSY_OS=linux_a64;;
23      "SUSELINUX"|"SUSE")
24          DSY_OS=linux_a64;;
25      *)
26          echo "Unknown linux release \""${DSY_OS_Release}"\""
27          echo "exit 8"
28          exit 8;;
29  esac
30
```

3. Run installation of Abaqus 2018

After all these steps, it is time to run the installation of Abaqus:

```
$ export DISPLAY=:0
```

```
$ cd /dir_of_downloaded_folders_Abaqus18/AM_SIM_Abaqus_Extend.AllOS/1/
```

```
$ sudo ./StartGUI.sh
```

Follow the instructions (better use the default options) and install the packages you are interested in. If you face problems with FLEXNET License, just continue the installation without activation and you will fix this problem later (see Step 5).

4. Make Abaqus available from any directory

After you have finished with the installation of Abaqus 2018, it is wise to make Abaqus available from any directory:

```
$ sudo ln /var/DassaultSystemes/SIMULIA/Commands/abq2018 /usr/bin/abaqus
```

Therefore, in order to run Abaqus CAE, just enter the following command in a terminal:

```
$ sudo abaqus cae
```

If you see a warning in the cmd saying that "Your system needs to be reconfigured to allow OpenGL rendering to a pixmap or Pbuffer;...", then you should use the following command in order not to face any transparency problems during the execution of the GUI:

```
$ sudo abaqus cae -mesa
```

5. FLEXNET License issues

If you have faced any problems during the installation with the FLEXNET License or when you run the previous command, then follow these instructions. When you run the previous command an error occurred concerning two files:

```
abaqus_v6.env
custom_v6.env
```

In order to fix this, type in the command line:

```
$ sudo gedit /usr/DassaultSystemes/SimulationServices/V6R2018x/linux_a64/
SMA/site/custom_v6.env
```

In the new window which was opened, edit the 6th and 7th lines:

```
license_server_type=FLEXNET
abaquslm_license_file="your_license"
```

where `"your_license"` is your license in the form: `"27000@maroon.abaqus.com"`.
When you are ready, save the file and close the window.

That's it! Let's launch Abaqus 2018.

6. Graphical issues while launching Abaqus 2018

After having installed Abaqus 2018 and activated your FLEXNET license, it is time to launch Abaqus 2018 from the terminal using the following command:

```
$ sudo abaqus cae -mesa
```

If you still face any transparency problems or there are any "residual" graphics from your background, the following command might help you:

```
$ sudo XLIB_SKIP_ARGB_VISUALS=1 LANG=en_US.UTF-8 abaqus cae -mesa
```

The link below could also be helpful if the installation was performed using Ubuntu 18.04 as host OS:

```
https://askubuntu.com/a/956407
```

Compiling and Linking steps of GFortran with Abaqus 2018

As you may have noticed, in the first step you installed gfortran in Ubuntu 18.04LTS. So, now, I will give you the instructions in order to compile and link User Subroutines written in Fortran with Abaqus 2018.

1. You need to open and edit the `lnx86_64.env` file because the default Fortran compiler in Abaqus 2018 is ifort and not gfortran:

```
$ sudo gedit /usr/DassaultSystemes/SimulationServices/V6R2018x/linux_a64/
SMA/site/lnx86_64.env
```

2. So, you should perform the following changes in this file. Let's start from the 12th line. It is like this:

```
fortCmd = "ifort"   # <-- Fortran compiler
```

and it should be like this:

```
fortCmd = "gfortran"   # <-- Fortran compiler
```

3. The 26th line is like this:

```
compile_fortran = [fortCmd,
                   '-V',
                   '-c', '-fPIC', '-auto', '-mP2OPT_hpo_vec_divbyzero=F',
                                                        '-extend_source',
                   '-fpp', '-WB', '-I%I', '-I'+abaHomeInc, '%P']
```

And it should be like this:

```
compile_fortran = [fortCmd,
                   '-c', '-fPIC', '-extend_source',
                   '-Warray-bounds', '-I%I']
```

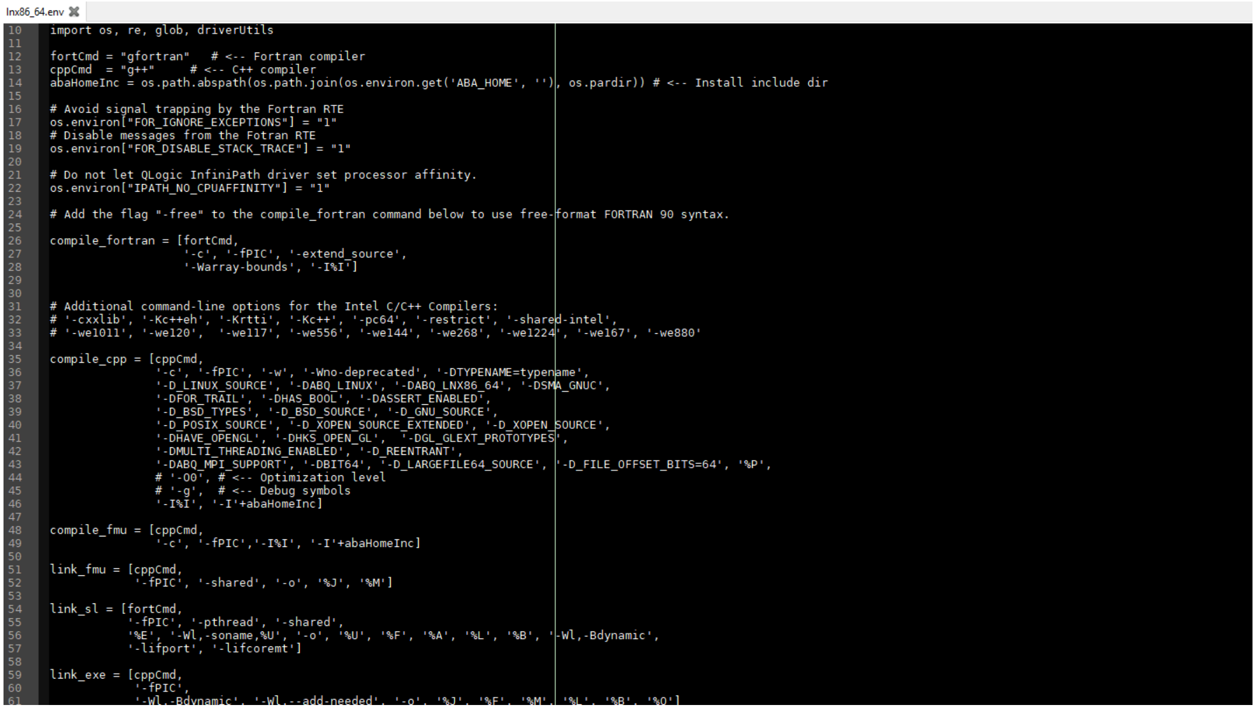
4. The 54th line is like this:

   ```
   link_sl = [fortCmd,
               '-V',
               '-cxxlib', '-fPIC', '-threads', '-shared','-Wl,--add-needed',
               '%E', '-Wl,-soname,%U', '-o', '%U', '%F', '%A', '%L', '%B',
                                                                '-parallel',
               '-Wl,-Bdynamic', '-shared-intel']
   ```

   And it should be like this:

   ```
   link_sl = [fortCmd,
               '-fPIC', '-pthread', '-shared',
               '%E', '-Wl,-soname,%U', '-o', '%U', '%F', '%A', '%L', '%B',
                                                           '-Wl,-Bdynamic',
               '-lifport', '-lifcoremt']
   ```

5. Your lnx86_64.env file should look like this, now:



```
10  import os, re, glob, driverUtils
11
12  fortCmd = "gfortran"   # <-- Fortran compiler
13  cppCmd  = "g++"     # <-- C++ compiler
14  abaHomeInc = os.path.abspath(os.path.join(os.environ.get('ABA_HOME', ''), os.pardir)) # <-- Install include dir
15
16  # Avoid signal trapping by the Fortran RTE
17  os.environ["FOR_IGNORE_EXCEPTIONS"] = "1"
18  # Disable messages from the Fotran RTE
19  os.environ["FOR_DISABLE_STACK_TRACE"] = "1"
20
21  # Do not let QLogic InfiniPath driver set processor affinity.
22  os.environ["IPATH_NO_CPUAFFINITY"] = "1"
23
24  # Add the flag "-free" to the compile_fortran command below to use free-format FORTRAN 90 syntax.
25
26  compile_fortran = [fortCmd,
27                     '-c', '-fPIC', '-extend_source',
28                     '-Warray-bounds', '-I%I']
29
30
31  # Additional command-line options for the Intel C/C++ Compilers:
32  # '-cxxlib', '-Kc++eh', '-Krtti', '-Kc++', '-pc64', '-restrict', '-shared-intel',
33  # '-we1011', '-we120',  '-we117', '-we556', '-we144', '-we268', '-we1224', '-we167', '-we880'
34
35  compile_cpp = [cppCmd,
36                 '-c', '-fPIC', '-w', '-Wno-deprecated', '-DTYPENAME=typename',
37                 '-D_LINUX_SOURCE', '-DABQ_LINUX', '-DABQ_LNX86_64', '-DSMA_GNUC',
38                 '-DFOR_TRAIL', '-DHAS_BOOL', '-DASSERT_ENABLED',
39                 '-D_BSD_TYPES', '-D_BSD_SOURCE', '-D_GNU_SOURCE',
40                 '-D_POSIX_SOURCE', '-D_XOPEN_SOURCE_EXTENDED', '-D_XOPEN_SOURCE',
41                 '-DHAVE_OPENGL', '-DHKS_OPEN_GL',  '-DGL_GLEXT_PROTOTYPES',
42                 '-DMULTI_THREADING_ENABLED', '-D_REENTRANT',
43                 '-DABQ_MPI_SUPPORT', '-DBIT64', '-D_LARGEFILE64_SOURCE', '-D_FILE_OFFSET_BITS=64', '%P',
44                 # '-O0', # <-- Optimization level
45                 # '-g',  # <-- Debug symbols
46                 '-I%I', '-I'+abaHomeInc]
47
48  compile_fmu = [cppCmd,
49                 '-c', '-fPIC','-I%I', '-I'+abaHomeInc]
50
51  link_fmu = [cppCmd,
52              '-fPIC', '-shared', '-o', '%J', '%M']
53
54  link_sl = [fortCmd,
55             '-fPIC', '-pthread', '-shared',
56             '%E', '-Wl,-soname,%U', '-o', '%U', '%F', '%A', '%L', '%B', '-Wl,-Bdynamic',
57             '-lifport', '-lifcoremt']
58
59  link_exe = [cppCmd,
60              '-fPIC',
61              '-Wl,-Bdynamic', '-Wl,--add-needed', '-o', '%J', '%F', '%M', '%L', '%B', '%O']
```

6. Now save the file and close the gedit window. Then, go to the folder where your Fortran Subroutine (.f file) is saved and enter the command:

   ```
   $ sudo abaqus make library=name_of_fortran_file.f
   ```

7. In order to use the shared object (.so file) that you created open the `lnx86_64.env` again and add to the end of the script the following line:

   ```
   usub_lib_dir = 'path_to_.so_file'
   ```

   Be careful, you should not point the .so file in the path above, but only the folder which is contained. In this way, it's not necessary to find the path for every single job. Just run the job without any changes in the job properties regarding the user subroutine file.