



Intall Abaqus 2017 on a Linux System

Yadong Jiang¹

¹College of Engineering and Informatics, National University of Ireland
Galway

Description

This document aims to guide users to install and run Abaqus 2017 on a Linux machine. Table 1 summarizes the environments of Linux machine used in this guidance. Despite Ubuntu, this instruction also applies for installation of Abaqus on other Linux distributions (e.g. Archlinux, Fedora, etc.). The Linux desktop environment is not required. All installation steps are completed in terminal. Thus, this document is also suitable for users who want to install Abaqus on a Linux server remotely via SSH connection. Furthermore, if you want to install other version of Abaqus, the methods given by this guide can be indicative. It should be highlighted that the administration right is required to complete the installation.

Table 1: Environments of Installation

Linux Distribution:	Ubuntu 14.04.4 LTS (64 bits)
Linux desktop environment:	N/A
Administration Right:	Yes
Abaqus version:	Abaqus 2017 (64 bits)

Prerequisites

Installation Files

The Abaqus installation files should be transferred to the Linux machine. In this document, the root directory of the installation files is referred as *Ins_File_Dir*. It should be noted that the *Ins_File_Dir* should not contain any “space”.

To avoid any permission issues, the permission of *Ins_File_Dir* should be modified so that the root user can read/write/execute the installation files. To achieve this simply, run the following command in terminal.



```
sudo chmod -R 777 Ins_File_Dir
```

Install Required Packages

The Linux Abaqus installation program is based on “ksh” while the FLEXNET requires the “lsb-core”. The two packages should be installed before starting the installation of Abaqus.

```
sudo apt-get install ksh lsb-core
```

It should be noted that “apt-get” is the package handling utility of Ubuntu. If you are working with other Linux distribution, the command of installing packages may vary.

Disguise Your Linux Distribution

The Abaqus installation program will check your Linux release name. If your Linux distribution is not “Red Hat Enterprise Linux Server”, “Red Hat Enterprise Linux Client”, “Red Hat Enterprise Linux Workstation”, “Cent OS” or “Suse Linux”, the program will pop out error and exit, as shown in Figure 1.

```
Current operating system: "Linux"
DSY_OS_Release="Ubuntu"
Unknown linux release "Ubuntu"
exit 8
```

Figure 1: Unknown linux release

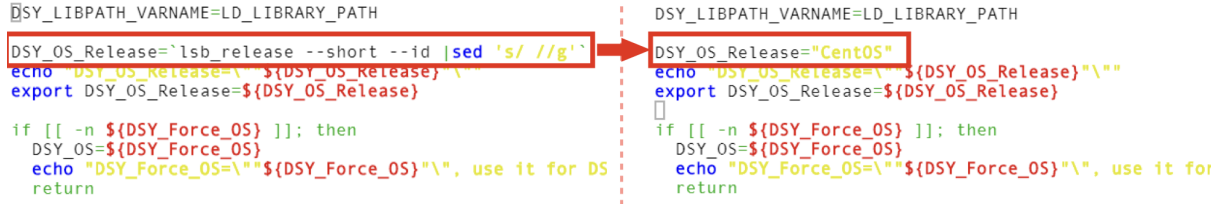
To avoid this error, you have to disguise your Linux distribution as one of the aforementioned Linux release names. To achieve this, you have to modified the “Linux.sh” file, a script which is called by the Abaqus installation program to detect your Linux release name. In this script, you have to change the line:

```
“DSY_OS_Release =‘lsb_release -short -id —sed 's/ //g'”
```

to

```
“DSY_OS_Release=“CentOS””
```

In terminal, you can edit text script file with software “nano” or “vim”. The modification results are highlighted in Figure 2. By modifying the “Linux.sh” script, the installation program will detect your Linux release name as “Cent OS”.



```
DSY_LIBPATH_VARNAME=LD_LIBRARY_PATH
DSY_OS_Release=`lsb_release --short --id | sed 's/ //g'`
echo "DSY_OS_Release=\`${DSY_OS_Release}\`"
export DSY_OS_Release=${DSY_OS_Release}

if [[ -n ${DSY_Force_OS} ]]; then
DSY_OS=${DSY_Force_OS}
echo "DSY_Force_OS=\`${DSY_Force_OS}\`, use it for DSY"
return
fi

DSY_LIBPATH_VARNAME=LD_LIBRARY_PATH
DSY_OS_Release="CentOS"
echo "DSY_OS_Release=\`${DSY_OS_Release}\`"
export DSY_OS_Release=${DSY_OS_Release}

if [[ -n ${DSY_Force_OS} ]]; then
DSY_OS=${DSY_Force_OS}
echo "DSY_Force_OS=\`${DSY_Force_OS}\`, use it for DSY"
return
fi
```

Figure 2: Change the “Linux.sh” file

There are eight “Linux.sh” scripts in the *Ins_File_Dir*, you need to modify all the eight files. The directories which contain the scripts are listed in Table 2.

Table 2: Directories which contain the “Linux.sh” files

<i>Ins_File_Dir</i> /1/inst/common/init
<i>Ins_File_Dir</i> /1/SIMULIA_Documentation/Allos/1/inst/common/init
<i>Ins_File_Dir</i> /2/SIMULIA_FLEXnet_LicenseServer/Linux64/1/inst/common/init
<i>Ins_File_Dir</i> /2/SIMULIA_AbaqusServices/Linux64/1/inst/common/init
<i>Ins_File_Dir</i> /2/SIMULIA_AbaqusServices_CAA_API/Linux64/1/inst/common/init
<i>Ins_File_Dir</i> /2/SIMULIA_Abaqus_CAE/Linux64/1/inst/common/init
<i>Ins_File_Dir</i> /2/SIMULIA_Tosca/Linux64/1/inst/common/init
<i>Ins_File_Dir</i> /3/SIMULIA_Isight/Linux64/1/inst/common/init

Installation

First of all, login as root user before the installation. In Ubuntu, you can do this by:

```
sudo -s
```

To start the installation program, the following command should be executed:

```
Ins_File_Dir/1/StartTUI.sh
```



This command will start a installation program in terminal. Alternatively, if you have Linux with Desktop Environment installed, you can call the following command to start the installation program with a GUI.

```
Ins_File_Dir/1/StartGUI.sh
```

It should be highlighted that the installation program cannot be started from the *Ins_File_Dir* or its subdirectories.

When the installation program starts, you can select and install the SIMULIA products. To install Abaqus, the products “FLEXnet License Server”, “Abaqus Simulation Services”, “Abaqus Simulation Services CAA API” and “Abaqus/CAE” are necessary.

It should be noted that the “FLEXnet License Server” should be installed before installing “Abaqus/CAE”. When install the “FLEXnet License Server”, you need to tick the option “Files only : do not start the license server program.”. When “FLEXnet License Server” is installed, you need to start the FLEXnet Server before starting installing “Abaqus/CAE”. To achieve this, you need to open a new terminal window (or a new SSH connection) and start server as root user with the following command (if you install the “FLEXnet License Server” into the default directory).

```
/usr/SIMULIA/License/2017/linux_a64/code/bin/lmgrd
```

During the installation of “Abaqus/CAE”, the “SIMULIA FLEXnet” should be ticked as the license server. The license server of NUI Galway is “27004@cert1.srv.nuigalway.ie”.

After the installation, the Abaqus licence type should be changed from academic to research. To do this, one of the Abaqus environment files, “custom_v6.env”, should be modified. If the Abaqus is installed into its default directory, the file can be edit with the command:

```
nano /usr/DassaultSystemes/SimulationServices/V6R2017x/linux_a64/SMA/  
site/custom_v6.env
```

The line “academic=TEACHING” in the “custom_v6.env” file should be commented or deleted, as shown in Figure 3.

After finishing the installation, exit the root mode with command “exit”.

```
# Installation of Abaqus CAE 2017
# Thu Jun 28 15:12:11 2018
plugin_central_dir="/var/DassaultSystemes/SIMULIA/CAE/plugins/2017"
doc_root="file:///usr/DassaultSystemes/SIMULIA2017doc/English"
license_server_type=FLEXNET
abaquslm_license_file="27004@cert1.srv.nuigalway.ie"
#academic=TEACHING
```

Figure 3: Change the license setting in the “custom_v6.env” file

Start Abaqus

If Abaqus is installed into its default directory, it can be called with command:

```
/var/DassaultSystemes/SIMULIA/Commands/abaqus
```

To avoid typing the full path to start Abaqus, the path “/var/DassaultSystemes/SIMULIA/Commands” needs to be appended to the “PATH” variable of Linux. The system “PATH” variable is defined in the file “/etc/environment”. Open the file with “nano” or “vim” and append the Abaqus command path into the “PATH” variable, as shown in Figure 4.

```
PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
:/usr/games:/usr/local/games:/var/DassaultSystemes/SIMULIA/Command
s"
```

Figure 4: Append path to “PATH” variable

After modifying the “/etc/environment” file, a reboot or re-login is required to take the modification into effects. Now the Abaqus can be simply called by:

```
abaqus
```

The Abaqus CAE can be started by:

```
abaqus cae
```

It should be noted that other libraries (e.g. libstdc++.so.6, libjpeg62, etc.) may be required by Abaqus CAE. Just follow the error message given by Abaqus CAE and install the missing libraries with “apt-get”.



Link C/C++ and Fortran compiler

Change the default Fortran compiler

On Linux, the default C/C++ compiler and Fortran compiler which Abaqus will search for are “gcc” and “ifort”, respectively. As the usage of free version “ifort” is very limited, the “gfortran” compiler is considered as an alternative during linking. To do this, these two compilers should be installed:

```
sudo apt-get install gcc gfortran
```

Then, one of the Abaqus environment files, “lnx86_64.env”, should be modified. If you installed Abaqus into the default directory, the file is located in “/usr/DassaultSystemes/SimulationServices/V6R2017x/linux_a64/SMA/site”. It is always recommended to create a backup file before editing the environment file.

To switch the default Fortran compiler, change the line:

```
fortCmd = “ifort”
```

to:

```
fortCmd = “gfortran”
```

Correspondingly, the compiling arguments should be modified from:

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

to:

```
compile_fortran = [fortCmd,  
                  '-c', '-fPIC',  
                  '-I%i']
```

Then, the linking arguments should also be changed from :

```
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']
```

to:

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

Verify the linking

To verify if the linking is successful, you can try to compile and link a sample Fortran file.

```
abaqus make library=testUMAT.f
```

If the compilation and linking are successful, you will find two output files, one “*.o” file and one “*.so” file, created, as shown in Figure 5.

libstandardU.so testUMAT.f testUMAT-std.o

Figure 5: Output files of the compilation and linking