



IBM XL Fortran for AIX

Compile with a List of Options Defined in a File

By: Ying Zhang

Level: Introductory

March 2012

Contents

IBM XL Fortran for AIX.....	1
About this Tutorial	3
Objectives	3
Prerequisites	3
System Requirements.....	3
Glossary	4
Start the Terminal Emulator to AIX System.....	5
Get Started with IBM XL Fortran Compiler	5
Conclusion	12
Trademarks.....	12
Resources	12

Before you start

About this Tutorial

This demo demonstrates the use of command line option `-qoptfile=filename` of IBM XL Fortran compiler which allows the user to specify a file containing a list of command line options to be used during compilation. The option provides a friendly interface for users to handle long lists of command line options. The principles demonstrated here also apply to C and C++ programs using the IBM XL C/C++ compilers.

In this demo, there are three tests, using the same set of options for compilation. Test one uses the options specified through command line. Test two uses the options specified through option file defined by `-qoptfile=filename`. Test three tests a combination of both command line and option file. As the set of options for compilation is the same, compiler is expected to behave the same regardless on how the options are passed in.

Objectives

- Use IBM XL Fortran Compiler for AIX to compile Fortran source code with options defined in option file through the use of `"-qoptfile=filename"` option
- Total time: 30 minutes

Prerequisites

- Basic Unix skills
- Basic command line compilation experience
- Basic command line option knowledge for IBM XL Fortran compiler

System Requirements

<http://www.ibm.com/software/awdtools/fortran/xlfortran/aix/sysreq>

Glossary

IBM XL C/C++ Compiler: IBM® XL C and C++ compilers offer advanced compiler and optimization technologies and are built on a common code base for easier porting of your applications between platforms. They comply with the latest C/C++ international standards and industry specifications and support a large array of common language features.

IBM XL Fortran Compiler: The IBM® XL Fortran compiler offers advanced compiler and optimization technologies and is built on a common code base for easier porting of your applications between platforms. It complies with the latest Fortran international standards and industry specifications and supports a large array of common language features.

Getting Started

Start the Terminal Emulator to AIX System

Figure 1 Get Started

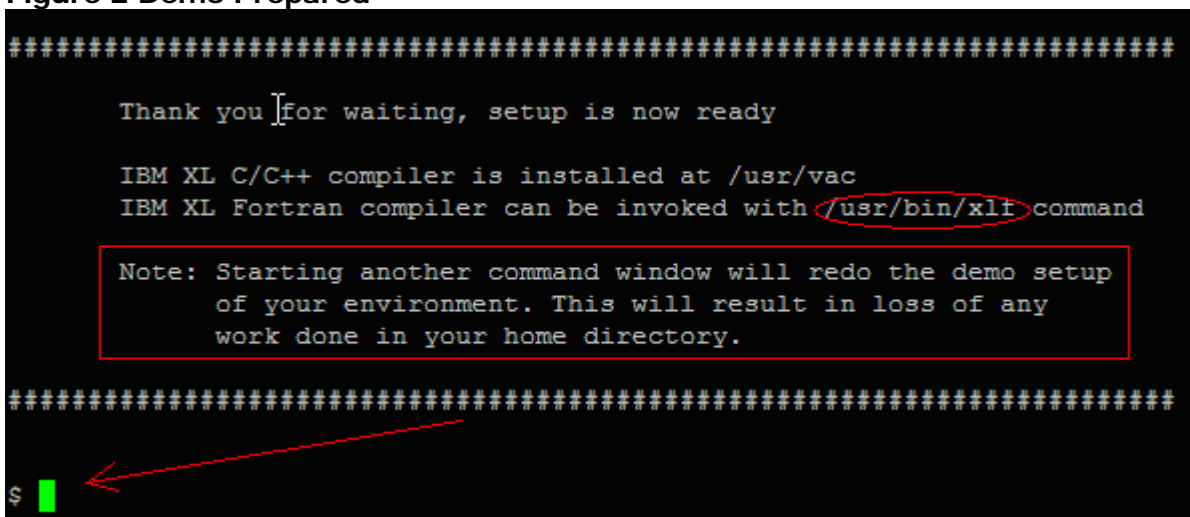


Double click the "Launch AIX" icon on the desktop (See Figure 1) to start the character terminal to AIX system.

Get Started with IBM XL Fortran Compiler

A successful login will result with the user presented with a menu of the demos hosted on the server. Type 18 and press Enter to select the "Using option file to pass options" demo.

Figure 2 Demo Prepared



On the terminal window you will see important information and the directory path to the compiler invocation command (See Figure 2 Demo Prepared *oval red*).

Note: Starting another command window will start the demonstration setup of your environment. This will result in loss of any work done in your home directory. This will impact any progress you have made on demo steps going forward.

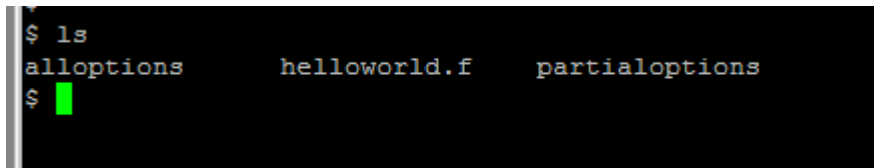
This demo does not require more than one terminal window. However, if you prefer more than one terminal window then you may open them before going forward.

The terminal window is now ready for commands. Your home directory contains one Fortran program (helloworld.f) and two option files (alloptions, and partialoptions). Type the ls command to see the directory content (See Figure 3 Directory Contents).

Command:

```
ls
```

Figure 3 Directory Contents



```
$ ls
alloptions      helloworld.f   partialoptions
$
```

The source code of the Fortran program is a simple hello world program. For the two option files, option file alloptions contains full set of options used in this demo, option file partialoptions contains partial options.

Steps:

1. Use command line options to compile helloworld.f program. See Figure 4

Here is the set of options we use for compilation:

```
-qlist -qfree=f90 -qhot -c
```

This test will pass all the options via command line, type following command:

Command:

```
xlf -qlist -qfree=f90 -qhot -c helloworld.f
```

Figure 4 Compile helloworld.f using command line options:

```
$ xlf -qlist -qfree=f90 -qhot -c helloworld.f
** helloworld === End of Compilation 1 ===
1501-510 Compilation successful for file helloworld.f.
$
```

The option `-c` compiles the source file, generates an object file called `helloworld.o`. With the use of option `-qlist`, a listing file `helloworld.lst` is generated by the compiler. Type the `ls` command to see the two generated files (highlighted in yellow in Figure 5)

Command:

```
ls
```

Figure 5 helloworld.o and helloworld.lst files are generated

```
$ ls
alloptions      helloworld.f    helloworld.lst  helloworld.o    partialoptions
$
```

We can `grep` the listing file to check if the other two options `-qfree=f90` and `-qhot` are used in compilation. Type the following two `grep` commands to check the options.

Command:

```
grep FREE helloworld.lst
```

Figure 6 checking `-qfree=f90` option

```
$ grep FREE helloworld.lst
      FREE (F90)                HALT (S)                IEEE (NEAR)
$
```

Command:

```
grep HOT helloworld.lst
```

Figure 7 checking –qhot option

```
$ grep HOT helloworld.lst
      HOT (VECTOR, FASTMATH, LEVEL (1) )
$
```

Figure 6 and Figure 7 confirm –qfree=f90 and –qhot are used in compilation.

The above test demonstrates the use of command line options in compile step. Before moving to the next test, remove the two generated files:

Command:

```
rm helloworld.lst helloworld.o
```

2. Use option file to pass options to compile helloworld.f program.

In this test, the set of options for compilation are all defined in option file called alloptions. The test uses option –qoptfile to pass them to compilation.

Type cat command to see the content of alloptions file. See Figure 8 alloptions content. Lines starting with # are comment lines.

Command:

```
cat alloptions
```

Figure 8 alloptions content

```
$ cat alloptions
# generating listing file
-qlist

# free source form
-qfree=f90

-qhot -c
$
```

Now use option –qoptfile to compile the helloworld.f program. Options are passed to compilation via option file alloptions.

Command:

```
xlf -qoptfile=alloptions helloworld.f
```

Figure 9 Compile helloworld.f using –qoptfile option:


```
$ xlf -goptfile=alloptions helloworld.f
** helloworld   == End of Compilation 1 ==
1501-510  Compilation successful for file helloworld.f.
$
```

Helloworld.o and helloworld.lst will be generated due to the use of `-c` and `-qlist` options. Type the `ls` command to see the two generated files in the directory.

Command:
ls

Figure 10 helloworld.o and helloworld.lst files are generated

```
$ ls
alloptions      helloworld.f  helloworld.lst  helloworld.o  partialoptions
$
```

Let's grep the helloworld.lst file to verify the other two options: `-qfree=f90` and `-qhot`

Command:
grep FREE helloworld.lst

Figure 11 checking `-qfree=f90` option

```
$ grep FREE helloworld.lst
                FREE (F90)                HALT (S)                IEEE (NEAR)
$
```

Command:
grep HOT helloworld.lst

Figure 12 checking `-qhot` option

```
$ grep HOT helloworld.lst
                HOT (VECTOR, FASTMATH, LEVEL (1) )
$
```

Figure 11 and Figure 12 confirm the two options `-qfree=f90` and `-qhot` are used in compilation.

The above test demonstrates the use of option file.

Before moving to the next test, remove the two generated files:

Command:

```
rm helloworld.lst helloworld.o
```

3. Use options in option file and command line compiling helloworld.f program.

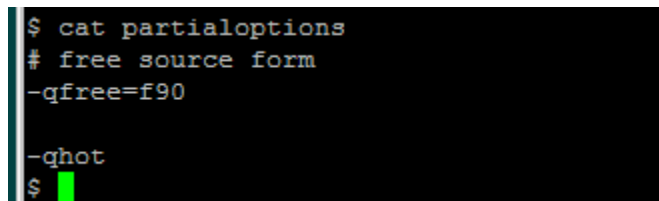
In this test, the set of options for compilation are split into 2 sets. The first set is passed in the command line and the second set is passed via option file.

Type cat command to see the content of partialoptions file. As illustrated in Figure 13 partialoptions content, only `-qfree=f90` and `-qhot` are defined in the file.

Command:

```
cat partialoptions
```

Figure 13 partialoptions content



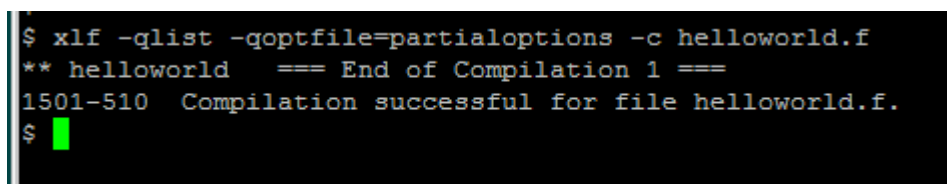
```
$ cat partialoptions
# free source form
-qfree=f90
-qhot
$
```

Now compile the helloworld.f program with options some from option file and some from command line.

Command:

```
xlf -qlist -qoptfile=partialoptions -c helloworld.f
```

Figure 14 Compile helloworld.f using `-qoptfile` option and command line options:



```
$ xlf -qlist -qoptfile=partialoptions -c helloworld.f
** helloworld === End of Compilation 1 ===
1501-510 Compilation successful for file helloworld.f.
$
```

Helloworld.o and helloworld.lst will be generated due to the use of `-c` and `-qlist` via command line. Type the ls command to see the two generated files in the directory.

Command:

```
ls
```

Figure 15 helloworld.o and helloworld.lst files are generated

```
$ ls
alloptions      helloworld.f  helloworld.lst helloworld.o  partialoptions
$
```

Now grep the helloworld.lst file to verify the other two options: `-qfree=f90` and `-qhot`

Command:

```
grep FREE helloworld.lst
```

Figure 16 checking `-qfree=f90` option

```
$ grep FREE helloworld.lst
      FREE (F90)                HALT (S)                IEEE (NEAR)
$
```

Command:

```
grep HOT helloworld.lst
```

Figure 17 checking `-qhot` option

```
$ grep HOT helloworld.lst
      HOT (VECTOR, FASTMATH, LEVEL (1) )
$
```

Figure 16 and Figure 17 confirm the two options `-qfree=f90` and `-qhot` are used in compilation.

The above test demonstrates the use of options via option file and command line.

What you have learned

In this exercise you learned how to:

- Use the IBM XL Fortran for AIX compiler to compile programs.
- Use command line options.
- Use option `-qoptfile=filename` to pass a list of options through option file.

Conclusion

This concludes the tutorial on using the IBM XL Fortran compiler. This tutorial has shown how to compile programs with the IBM Fortran for AIX compiler and use the `-qoptfile=filename` option to pass a list of options through an option file.

Trademarks

IBM and the IBM logo are trademarks of International Business Machines Corporation in the United States, other countries or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product and service names may be trademarks or service marks of others.

Resources

Learn

Overview of the IBM XL C/C++ and XL Fortran Compiler Family

<http://www.ibm.com/support/docview.wss?uid=swg27005175>

Getting Started with XL Fortran for AIX, V14.1

<http://www.ibm.com/support/docview.wss?uid=swg27024216>

Compiler Reference - XL Fortran for AIX, V14.1

<http://www.ibm.com/support/docview.wss?uid=swg27024215>

Language Reference - XL Fortran for AIX, V14.1

<http://www.ibm.com/support/docview.wss?uid=swg27024218>

Optimization:

Optimization and Programming Guide - XL Fortran for AIX, V14.1

<http://www.ibm.com/support/docview.wss?uid=swg27024219>

Code Optimization with the IBM XL Compilers

<http://www.ibm.com/support/docview.wss?uid=swg27005174>