VMS Software

Xerces-C++ for VSI OpenVMS x86-64
August 2023

`VSI-X86VMS-XERCES_C-V0400-0-1.PCSI`

1.   Introduction

Thank you for your interest in this port of Xerces-C++ to VSI OpenVMS x86-64. The current release of Xerces C++ for OpenVMS is based on the 4.0.0 distribution.

Apache Xerces-C++ is a collection of software libraries written in a portable subset of C++ that can be used by C and C++ programs to provide applications with the ability to read, write, and manipulate XML data. A comprehensive set of API's are provided for parsing, generating, manipulating, and validating XML documents using the DOM, SAX, and SAX2-type parsers.

This OpenVMS port of Xerces-C++ includes all functionality provided by the Open Source 4.0.0 release available on GitHub (https://github.com/apache/xerces-c).

Additional information about the Xerces C++ library and its various components can be found at https://xerces.apache.org/xerces-c/.

2.   Acknowledgements

VMS Software Inc. would like to acknowledge the work of the Apache Software Foundation members for the creation and ongoing maintenance of the Xerces-C++ software.

3.   Requirements

The kit you are receiving required the following products and product versions to be installed in order for the software to operate correctly. In general it will also be possible to install and use the software on higher versions of the operating system.

- VSI OpenVMS 9.2-1 for x86-64 or higher

- It is recommended that the software is installed on an ODS-5-enabled file system

In addition to the specific software requirements listed above, you will also require a C or C++ compiler in order to develop applications using the Xerces-C++ API's.

It is assumed that the reader has a good knowledge of OpenVMS and software development in the OpenVMS environment.

4.   Recommended reading

Before using Xerces-C++, it is recommended that developers review the documentation available at https://xerces.apache.org/xerces-c/ and examine the various example programs that are provided with the Xerces-C++ for VSI OpenVMS kit.

5. Installing the kit

The kit is provided as an OpenVMS PCSI kit (`VSI-X86VMS-XERCES_C-V0400-0-1.PCSI`) that can be installed by a suitably privileged user using the following command:

`$ PRODUCT INSTALL XERCES_C`

The installation will then proceed as follows (output may differ slightly from that shown):

```
Performing product kit validation of signed kits ...

The following product has been selected:
    VSI X86VMS XERCES_C V4.0-0              Layered Product

Do you want to continue? [YES]

Configuration phase starting ...

You will be asked to choose options, if any, for each selected
product and for any products that may be installed to satisfy
software dependency requirements.

Configuring VSI X86VMS XERCES_C V4.0-0

    VMS Software Inc.

* This product does not have any configuration options.

Execution phase starting ...

The following product will be installed to destination:
    VSI X86VMS XERCES_C V4.0-0              DISK$X86SYS:[VMS$COMMON.]

Portion done: 0%...10%...20%...60%...90%...100%

The following product has been installed:
    VSI X86VMS XERCES_C V4.0-0              Layered Product

VSI X86VMS XERCES_C V4.0-0

    Post-installation tasks are required.


    To start XERCES-C at system boot time, add the following lines to
    SYS$MANAGER:SYSTARTUP_VMS.COM:

        $ file := SYS$STARTUP:XERCES-C$STARTUP.COM
        $ if f$search("''file'") .nes. "" then @'file'

    To stop XERCES-C at system shutdown, add the following lines to
    SYS$MANAGER:SYSHUTDWN.COM:

        $ file := SYS$STARTUP:XERCES-C$SHUTDOWN.COM
        $ if f$search("''file'") .nes. "" then @'file'
```

5.1.   Post-installation steps

After the installation has successfully completed, include the commands displayed at the end of the installation procedure into `SYSTARTUP_VMS.COM` to ensure that the logical name `XERCES-C$ROOT` (required in order for developers to be able to use the software) is defined system-wide at start-up.

5.2. *Privileges and quotas*

There are no special quota or privilege requirements in order for application developers to use Xerces-C++.

6. Sample applications

The directory `XERCES-C$ROOT:[SAMPLES]` contains a number of example programs that can be used to learn about the Xerces-C++ API's or as a source of inspiration for the development of new applications. These examples can be compiled and linked using the provided build procedure (`SAMPLES.COM`).

Note that two versions of the Xerces-C++ object library are provided for linking with application code, namely `LIBXERCES32.OLB` and `LIBXERCES64.OLB`, where the former should be linked with applications compiled to use 32-bit pointers, and the latter should be linked with applications that use 64-bit pointers. The examples build procedure (`SAMPLES.COM`) compiles the example code to use 64-bit pointers and therefore links the examples with `LIBXERCES64.OLB`.

From a development perspective, it should also be noted that symbols in the Xerces-C++ object libraries are mixed-case, and application developers must therefore use the C and C++ compiler option `/NAMES=(AS_IS,SHORTENED)`, or include in their code appropriate `#pragma` directives (C only) to ensure that symbols are correctly resolved when linking. Developers will also need to include in their application code one or more of the header files found in `XERCES-C$ROOT:[INCLUDE]`.

7. What's missing?

The supplied kit for OpenVMS includes all functionality supported by version 4.0.0 of the Xerces-C++ library.