



---

National Resource Centre  
for EHR Standards  
India

---

# **A guide for using C-DAC's SDK for DICOM PS3.0-2015 in .Net Application**

---

**Created by:**  
National Resource Centre for EHR Standards,  
Centre for Development of Advanced Computing (C-DAC), Pune, India

**Published: June 2019  
Version v1.0**

## REVISION HISTORY

Date	Version	Description
27/06/2019	1.0	Document describes the way to convert existing jars of C-DAC's SDK for DICOM PS3.0-2015 to .net dlls using IKVM.NET which can be referred in .Net application

## TABLE OF CONTENTS

<b>Table of Contents</b>	<b>3</b>
<b>Introduction</b>	<b>4</b>
<b>Purpose</b>	<b>4</b>
<b>Scope</b>	<b>4</b>
<b>Prerequisite</b>	<b>4</b>
<b>Steps to convert jars to dll</b>	<b>4</b>
<b>Steps to use dll in .Net Application</b>	<b>5</b>
<b>Note</b>	<b>5</b>
<b>Reference</b>	<b>6</b>

## INTRODUCTION

There may arise instances where Java and .Net code need to communicate with each other e.g., there exists only Java libraries for some algorithm, protocol, or feature which may be required in a .Net application. One solution could be to redevelop the whole algorithm, protocol, or feature in .Net and consume it for application. The other solution could be to work upon calling Java based library in .Net and focus on application development.

There are some initiatives on interoperability between Java and .Net. This guide particularly focuses on converting Java library to .Net for use in .Net based applications.

IKVM.NET is an open source implementation of Java for Microsoft .Net Framework that provides tools to enable Java and .Net interoperability. IKVM.NET provides a tool `ikvmc` that converts Java bytecode to .NET dlls and exe.

## PURPOSE

This document provides a step by step approach to build java library to produce a .dll using IKVM.Net. The Java library referred in this guide is C-DAC's SDK v3.5. developed by Centre for Development of Advanced Computing (C-DAC) Pune. The DICOM SDK is a Free and Open Source Software (FOSS) Library used for enabling Parser Serializing of DICOM image and supporting DICOM services and Service Object Pair.

## SCOPE

The scope of the document is to provide steps on how to create a dll using IKVM.NET's tool, `ikvmc` and use that dll in the .Net Application. This document is only for reference purpose. The approach for implementation may vary from application to application based on the requirements, resources, and scope. User may choose a different platform/mechanism than IKVM.NET to achieve Java and .Net interoperability.

## PREREQUISITE

.Net Framework 2.0 or above

## STEPS TO CONVERT JARS TO DLL

- Download IKVM.NET binary `ikvmbin-8.1.5717.0.zip`
- Extract the zip to some folder having permission
- Copy SDK's jars (including third party jars) in extracted bin folder of IKVM.NET
- Open Command prompt
- Navigate to IKVM.NET bin folder
- Execute the following command to make Parser Serializer jar to dll:

```
ikvmc -target:library -out:ParserSerializer.dll commons-codec-1.3.jar  
DICOMBase30_04.jar en_US.jar IOD30_04.jar servlet-api.jar
```

Where, *ParserSerializer.dll* is output dll after converting DICOMBase30\_04.jar en\_US.jar IOD30\_04.jar along with dependency jars. In above example, IKVM options used:

- `-target:<target-type>`: “-target:library” : generates a .dll
- `-out:<outputfile>`: “-out: *ParserSerializer.dll*”: Specifies the name of the output file.

Here, commons-codec-1.3.jar, en\_US.jar and servlet-api.jar\_04.jar are the dependency jars required for Parsing and Serializing

- There are multiple [options] that can be used with the ikvmc tool. For more details please go through IKVM user guide link provided in reference.
- As per user’s requirement, other SOP’s jars can be converted to dll with same steps.
- The dll will be created in the bin folder of IKVM.NET by the name provided to the -out option in the command.

For e.g. To convert whole SDK jar to dll, all sdk jars are required with dependency jars, then the command will be:

```
ikvmc -target:library -out:DICOMSDK.dll derby.jar derbyclient.jar commons-codec-1.3.jar javax.ws.rs.jar org.osgi.core-4.2.0.jar servlet-api.jar DICOMBase30_04.jar en_US.jar IOD30_04.jar mail.jar DIMSECServices30_04.jar DIMSEMServices30_04.jar basicworklistmgt30_04.jar CIRWBDSERVICE30_04.jar colorpaletteservice30_04.jar DICOMNet30_04.jar DIMSENServices30_04.jar displaysystemmanagement30_04.jar hangingprotocol30_04.jar iflr30_04.jar implanttemplate30_04.jar mediastorageprofilemgt30_04.jar mediastorageprofiles30_04.jar miscservices30_04.jar printmgt30_04.jar rtmachineverification30_04.jar securityprofiles30_04.jar SPSSService30_04.jar SRService30_04.jar storagecommitment30_04.jar substanceadministrationservice30_04.jar UPSService30_04.jar
```

- The dll will be created in the bin folder of IKVM.NET by the name provided to the -out option in the command.

## STEPS TO USE DLL IN .NET APPLICATION

- To use created dll, add created dll along with IKVM.NET dlls based on type of application to project reference.

For e.g. For a .Net Console Application below IKVM.NET dlls are required in project reference:

- IKVM.OpenJDK.Core.dll
- IKVM.Runtime.dll
- IKVM.OpenJDK.XML.Parse.dll
- IKVM.OpenJDK.Security.dll
- IKVM.OpenJDK.Util.dll
- IKVM.OpenJDK.XML.API.dll

## NOTE

- The guide is constructed assuming the implementer has access and knowledge of DICOM PS3.0-2015 version specification.

- The dll created from SDK for DICOM PS3.0-2015 libraries using IKVM.NET is tested only for following purposes,
  - Parsing of existing DICOM file and serializing it
  - DIMSE C-Services on DICOM files.

In evidence of any material error, change, correction, concerns or assistance regarding this document, you are requested to contact at: [nrc-help@cdac.in](mailto:nrc-help@cdac.in)

## REFERENCE

- Refer below link to know brief about DICOM Standard PS3.0 2015  
<https://www.nrces.in/standards/dicom/>
- Refer below link for more information about ikvmc command  
<http://www.ikvm.net/userguide/ikvmc.html>
- Refer below link to download IKVM.NET binary  
<http://weblog.ikvm.net/2015/08/26/IKVMNET81ReleaseCandidate0.aspx>.