



digital
vision
group

TRIP Connectivity Framework

Installation Guide

Version 1.7
UNIX and Windows

End User License Agreement

All rights to this software, its documentation and logotypes of the TRIP product family and software (altogether “Software”) supplied by DVG Operations GmbH (DVG) are exclusively owned by DVG.

The transfer of this Software, solutions or parts thereof requires the prior written agreement of DVG. Furthermore, the customer has the right to use licensed Software and / or process solutions supplied by DVG to the extent specified in his contract with DVG.

The free-to-use non-commercial version doesn't require a prior written agreement with DVG but such customers, organizations and/or third parties agree by using the software and / or solution of DVG to be strongly obliged to keep all rights to this software, documentation and logotypes of the TRIP product family absolutely unfringed and protected.



Table of Contents

INTRODUCTION	4
ABOUT THIS DOCUMENT.....	4
INSTALLATION TYPES	4
<i>Server-Side Installation</i>	4
<i>Client-Side Installation</i>	4
COMPATIBILITY WITH OTHER TRIP SDK PRODUCTS.....	4
JAVA DEPENDENCY.....	4
OTHER DEPENDENCIES	5
DIRECTORY STRUCTURE	6
UNIX/LINUX	6
WINDOWS.....	6
SERVER-SIDE INSTALLATION	7
OVERVIEW.....	7
UNINSTALL TRIPVIEW.....	7
<i>TRIPview uninstallation on UNIX/Linux</i>	7
<i>TRIPview uninstallation on Windows</i>	7
STOP IMPORT CONNECTOR SERVICES ON WINDOWS	7
UNPACK THE FILES	7
<i>UNIX/Linux</i>	7
<i>Windows</i>	8
<i>Java Prerequisites:</i>	8
RUNNING THE INSTALLATION PROGRAM	8
<i>UNIX/Linux</i>	8
<i>Windows</i>	8
STANDALONE (CLIENT-SIDE) INSTALLATION	9
POST-INSTALLATION ADJUSTMENTS	10
ADAPT THE CONFIGURATION FILES	10
<i>File: conf/fifiase.conf</i>	10
<i>File: conf/fifi.conf</i>	11
<i>File: conf/cfw.conf</i>	11
<i>File: conf/fifi.d/tikaserver.conf</i>	12
<i>File: conf/fifi.d/icepdf.conf</i>	13
MODIFY TRIPSYSTEM CONFIGURATION (TDBS.CONF)	13
<i>Mandatory Settings</i>	13
REINSTALL IMPORT CONNECTOR SERVICES ON WINDOWS	14
CONFIGURING THE METADATA COPYING ASE	14

Introduction

About this document

This document describes the installation procedure of TRIPcof as done by the installation programs and scripts provided for it on the different supported platforms.

Installation Types

TRIPcof can be installed in two variants:

- Server-side
- Standalone (client-side, Windows only)

Server-Side Installation

The server-side installation integrates TRIPcof with TRIPsystem using ASE (Application Software Exit) libraries.

Version 8.4 of TRIPsystem must be installed before TRIPcof can be installed.

In server-side operation, TRIPcof is invoked from TRIPnpx or TRIPjxp who calls upon TRIPsystem to launch the ASE procedure for TRIPcof.

Client-Side Installation

A standalone (or client-side) installation can be done on a Windows machine where TRIPsystem is not present. The Windows installation program supports this scenario. It may ask for confirmation in uncertain cases. A standalone, client-side installation can be used directly from version 8.2 or later of TRIPnpx and TRIPjxp.

Note that TRIPnpx and TRIPjxp can also directly use a server-side installation of TRIPcof, which makes it possible to deploy TRIPsystem, TRIPcof and TRIPjxp/TRIPnpx all on a single machine.

Compatibility with other TRIP SDK products

Although previous / pre-release information has indicated otherwise, TRIPcof is not compatible with the TRIPview interfaces in TRIPclient and TRIPjtk.

TRIPcof is not supported for deployment on a machine where either TRIPview-C or TRIPview is installed. If this such a deployment nevertheless is performed, TRIPcof will in most cases yield to TRIPview/-C for text extraction and HTML conversion operations. This may result in unpredictable results.

To use HTML conversion with hit highlighting in standalone (client-side) operation, version 8.2 or later of TRIPnpx or TRIPjxp is required.

Java Dependency

A 64-bit Java runtime environment (JRE) is required. Use version 17 or later. The java.exe program must be available on the system PATH. If a Java VM is not present or is of the wrong architecture (e.g. 32-bit), many file formats supported by TRIPcof will not be possible to process.

One way to check the version and address length of your (default) Java installation is by executing the command "java -version". You should see information similar to this:

```
C:\>java -version
openjdk version "17.0.8" 2023-07-18 LTS
OpenJDK Runtime Environment Corretto-17.0.8.7.1
OpenJDK 64-Bit Server VM Corretto-17.0.8.7.1
```

The example above shows that the machine has an installation of a 64-bit Amazon Corretto Java VM of version 17.

For UNIX/Linux, a script is provided called "showjvm.sh". This is found in the TRIPcof 'bin' directory. It outputs the path to the installed Java VM library if one is present. For example:

```
tripbox:~> /opt/trip/cof/v152/bin/showjvm.sh
/usr/lib/jvm/java-17-openjdk-17.0.9.0/jre/lib/amd64/server/libjvm.so
```

If the JRE is installed after TRIPcof, or later upgraded so that the JRE is installed in a different location, you must modify the JavaVM property in the configuration files `fifi.conf` and `cfw.conf`. Failure to do so may cause TRIPcof to stop working properly.

Other Dependencies

To use HTML conversion of files other than PDF, TRIPcof requires that LibreOffice 4.4 or later is installed on the same machine with TRIPcof.

For installation on Windows, the Microsoft .NET Framework version 4.6 or 4.7 is required.

Directory Structure

UNIX/Linux

The directory structure of TRIPcof on UNIX/Linux systems is the following:

```

$TRIPCOF_HOME
| - bin                (scripts and executables)
| - conf               (main configuration files)
|   | - cfw.connectors.d (import connector config files)
|   | - cfw.importtds.d  (config files for connector data
sources)
|   | - fifi.d           (file filter adapter config files)
|   | - samples.cfw      (sample custom connector config files)
|   | - samples.fifi     (sample custom adapter config files)
| - doc                (documentation)
| - include             (C include files for custom adapters)
| - lib                 (TRIPcof shared libraries)
|   | - java            (Java JAR files TRIPcof and adapters)
| - log                 (default log TRIPcof)
| - samples             (sample custom adapters and connectors)
| - tmp                 (scratch/working directory)

```

Windows

The directory structure of TRIPcof on Windows systems is the following:

```

%TRIPCOF_HOME%
| - bin                (executables and DLLs)
| - conf               (main configuration files)
|   | - cfw.connectors.d (import connector config files)
|   | - cfw.importtds.d  (config files for connector data
sources)
|   | - fifi.d           (file filter adapter config files)
|   | - samples.cfw      (sample custom connector config files)
|   | - samples.fifi     (sample custom adapter config files)
| - doc                (documentation)
| - include             (C include files for custom adapters)
| - lib                 (C lib files for custom adapters)
|   | - java            (Java JAR files TRIPcof and adapters)
| - log                 (default log directory for TRIPcof)
| - samples             (sample custom adapters and connectors)
| - scripts             (scripts used by adapters and
connectors)
| - tmp                 (scratch/working directory)

```

Server-Side Installation

Overview

This section describes the steps for performing a server-side installation of TRIPcof.

The steps in brief:

1. Uninstall TRIPview and TRIPview-C
2. Unpack the files
3. Adapt the configuration files
4. Modify the TRIPsystem configuration file `tdbs.conf`

Uninstall TRIPview

If TRIPcof and TRIPview (or TRIPview-C) are both installed, TRIPcof will always yield to TRIPview for text extraction and HTML conversion operations. If you have a pre-existing TRIPview installation on the target machine and wish to use TRIPcof instead, you must uninstall TRIPview before you can use TRIPcof.

TRIPview uninstallation on UNIX/Linux

TRIPview has no uninstall script for UNIX/Linux, so you must do this manually.

- Delete the `/usr/local/trip/view` directory tree.
- Open the TRIPsystem configuration file (`tdbs.conf`) in a text editor and remove all remaining references to TRIPview. This especially includes `TDBS_VIEWSHR` and its inclusion in the `TDBS_ASELIBS` list.
- Optionally delete the installed TRIPview files.

TRIPview uninstallation on Windows

Uninstall TRIPview from the Programs and Features applet in the Windows control panel. When done, open the TRIPsystem configuration file (`tdbs.conf`) in a text editor and remove all remaining references to TRIPview. This especially includes `TDBS_VIEWSHR` and its inclusion in the `TDBS_ASELIBS` list.

Stop import connector services on Windows

If you are performing an upgrade installation and you have manually registered the `cfwimport` program to run as a one or more services, you must stop all such services before you run the installer. The installation program will otherwise not be able to proceed.

The `cfwimport` based services can be found listed in the Services app that you can find in the Administrative Tools section of the Windows control panel. The `cfwimport` services have names beginning with the phrase “TRIP import connector”.

Unpack the Files

UNIX/Linux

On UNIX/Linux, TRIPcof comes in a distribution archive that is a tar file, optionally compressed with `gzip`. The top directory of the file hierarchy contained therein is named after the version (“v152” for version 1.5-2).

Perform the following steps from a command prompt.

1. Create a directory where you want your files
2. Change the current directory to the newly created directory
3. Unpack the distribution archive with the tar command.

Example: UNIX/Linux shell commands

```
mkdir -p /opt/trip/cof  
cd /opt/trip/cof  
tar xf ~/tripcof_rhel9-x86-64_v152.tar
```

Windows

TRIPcof is available as an MSI-based installation program. Use this if you wish install TRIPcof for direct, local use from TRIPjxp or TRIPnxp only without having to have a local installation of TRIPsystem.

Run the installation program of your choice and follow the instructions.

Java Prerequisites:

A 64-bit Java runtime version 17 is needed in order to be able to use the Java-based file filter adapters.

Running the Installation Program**UNIX/Linux**

Installation on UNIX and Linux platforms is done by running the `install_cof.sh` shell script found in the `$TRIPCOF_HOME/bin` directory.

This script modifies the ownership of the TRIPcof files and modifies the TRIPsystem configuration file (`tdbs.conf`). Please also note that post-install modification of the TRIPcof configuration files is necessary. See section "Adapt the Configuration Files" below for more details.

Windows

The installer for TRIPcof is provided as an `.msi` file. Start the installation by running the MSI file and follow the instructions displayed.

Please note that from the Windows installer for TRIPcof requires the .NET Framework 4.6 or 4.7. If your computer lacks this, the installation will fail. If this happens, you must install the prerequisite software before you can proceed with the TRIPcof installation.

Please also note that post-install modification of the TRIPcof configuration files is necessary. See section "Adapt the Configuration Files" below for more details.

Standalone (client-side) Installation

TRIPjxp and TRIPnpx supports client-side access to TRIPcof for text extraction and HTML conversion. Older TRIP SDK products (TRIPclient and TRIPjtk) cannot be used in client-side operation with TRIPcof.

The only installation consideration is that any pre-existing TRIPview installation must be removed before text extraction and HTML conversion via TRIPcof can be used. See page 4 for more details.

Note that connectors cannot be used with client-side installations since they require a local TRIPsystem installation.

For usage information, please refer to the TRIPcof Operations Manual.



Post-installation Adjustments

Adapt the Configuration Files

After having installed the TRIPcof distribution files onto the local file system, you need to review and edit the configuration files under the `conf` and `conf/fifi.d` directories. Most settings will work as is, but there are a few that you may need to modify.

Note: all relative paths are relative to the directory in which the configuration file is located.

File: `conf/fifiase.conf`

This is the main configuration file for the TRIPcof ASE that exposes text extraction and HTML conversion behavior similar to that of TRIPview.

Although it is not necessary to make any changes to this file, you may modify it if you wish.

DefaultTextAdapter	<p>Preferred file filter adapter for text extraction.</p> <p>If defined, TRIPcof will only try to use this file filter adapter to perform extraction of plain text and document properties from files.</p> <p>If undefined, TRIPcof will attempt to use all installed file filter adapters. Results will be output from the first file filter adapter that reports success.</p> <p>Default: (undefined)</p>
DefaultHtmlAdapter	<p>Preferred file filter adapter for HTML conversion.</p> <p>If defined, TRIPcof will only try to use this file filter adapter to perform the conversion of files to HTML.</p> <p>If undefined, TRIPcof will attempt to use all installed file filter adapters. Results will be output from the first file filter adapter that reports success.</p> <p>Default: (undefined)</p>
AseLogLevel	<p>Log level.</p> <ul style="list-style-type: none"> 0 Off 1 Fatal errors only 2 Errors 3 Errors and warnings 4 Errors, warnings and information 5 Errors, warnings, information and debug statements <p>Default: 0</p>
AseLogDirectory	<p>The directory where log files are written.</p> <p>Default on regular installations: <code>../log</code></p> <p>Default on Docker: <code>/var/lib/trip/log/cof</code></p>

RequestTimeout	<p>The number of seconds a request (e.g. for a text extraction) may take before it is aborted.</p> <p>Default: 60</p>
IOTimeout	<p>The max number of seconds the ASE will wait for communications from the underlying file filter adapter before assuming it has hanged or otherwise ceased to operate normally.</p> <p>Default: 20</p>

File: conf/fifi.conf

This is the main configuration file for the TRIPcof file filter adapters that implement logic for the conversion of one file format to another.

NOTE: You must verify the **JavaVM** property manually! When you upgrade your Java installation, you may also need to modify this property again.

JavaVM	<p>The Java virtual machine library to use.</p> <p>The value of this property is the <i>fully qualified path</i> to the Java virtual machine shared library on you system. Remember that for a 64-bit TRIPcof you need a 64-bit Java VM.</p> <p><i>The default value is only provided as an example. You will need to modify it to suit your system!</i></p> <p>The installation program (Windows) and script (UNIX/Linux) will attempt to detect a Java VM and provide a value for this property. However, if the installation program is unable to detect a VM or has detected the wrong one, you still need to verify and potentially modify the value of this property.</p>
JavaJarDir	<p>The directory that contains the TRIPcof JAR files.</p> <p>Default: ../lib/java</p>
JavaLibDir	<p>The directory that contains the TRIPcof shared libraries.</p> <p>Default (UNIX/Linux): ../lib</p> <p>Default (Windows): ../bin</p>

File: conf/cfw.conf

This is the main configuration file for the TRIPcof import connector functionality. Since this file contains the username and password for the TRIP user used by TRIPcof to read and write TRIP databases that the connectors populate, the file access permissions should be set so that only the user (and possibly group) that run the cfwimport program have read access to the file.

NOTE: You must verify the **JavaVM** property manually! When you upgrade your Java installation, you may also need to modify this property again.

JavaVM	<p>The Java virtual machine library to use.</p> <p>The value of this property is the <i>fully qualified path</i> to the Java virtual machine shared library on your system. Remember that for a 64-bit TRIPcof you need a 64-bit Java VM.</p> <p><i>The default value is only provided as an example. You will need to modify it to suit your system!</i></p> <p>The installation program (Windows) and script (UNIX/Linux) will attempt to detect a Java VM and provide a value for this property. However, if the installation program is unable to detect a VM or has detected the wrong one, you still need to verify and potentially modify the value of this property.</p>
JavaJarDir	<p>The directory that contains the TRIPcof JAR files.</p> <p>Default: ../lib/java</p>
JavaLibDir	<p>The directory that contains the TRIPcof shared libraries.</p> <p>Default (UNIX/Linux): ../lib</p> <p>Default (Windows): ../bin</p>
RegistryUser	<p>The name of the TRIP user that TRIPcof will use to read and write databases populated by connectors.</p> <p>Default: SYSTEM</p>
RegistryPass	<p>The password for the TRIP user that TRIPcof will use to read and write databases populated by connectors.</p> <p>Default: Z</p>
IndexAfterImport	<p>Determines if cfwimport should index the TRIP connector database after a successful import run.</p> <p>Default: True</p>

File: conf/fifi.d/tikaserver.conf

This configuration file controls the behavior of the file filter adapter for Apache Tika. Most of the properties in this file must not be modified at all. Only those listed in the table below may be adjusted.

TikaServerHeapSize	<p>This value controls how much heap memory will be available to the Apache Tika server process. The value is expressed according to the syntax of the "-Xmx" Java parameter.</p> <p>For processing large documents, you should increase this value to 2G or more.</p> <p>Default: 512m</p>
---------------------------	---

File: conf/fifi.d/icepdf.conf

This configuration file controls the behavior of the file filter adapter for ICEpdf. Most of the properties in this file must not be modified at all. Only those listed in the table below may be adjusted.

EnableFileCache	<p>This value controls how the ICEpdf library uses temporary files during conversion from PDF to HTML. If unset or set to True, use of temporary files will be enabled. If set to False, temporary file use will be disabled.</p> <p>Default: True</p>
------------------------	--

Modify TRIPsystem Configuration (tdbs.conf)

Although TRIPcof keeps its settings in its own set of configuration files, the tdbs.conf file still needs to be modified so that TRIPsystem can find the correct ASE library to invoke.

The normal installation procedure on Windows, UNIX and Linux involves automatic modification of the TRIPsystem configuration file. No manual post-installation steps are required.

The following table specifies the property values in the NonPrivileged section relevant to TRIPcof.

Mandatory Settings

COF_ASE_FIFI	The fully qualified path to the text extraction and HTML conversion ASE library (libfifiase.so on UNIX/Linux or fifiase.dll on Windows).
COF_CONFIG_DIR	<p>The fully qualified path to the configuration directory of TRIPcof. Must be set to the "conf" subdirectory of the TRIPcof installation (e.g. /opt/tripcof/v140/conf).</p> <p>This property is not used within Docker containers based on the TRIP Docker image provided by DVG.</p>
TDBS_ASELIBS	<p>This TRIPsystem property is a comma-separated list of names of properties whose values are the fully qualified paths of the ASE libraries that should be loadable by TRIPsystem.</p> <p>Add the value "COF_ASE_FIFI" to this property.</p>

Settings in tdbs.conf - UNIX/Linux Example

```
COF_ASE_FIFI=/opt/trip/cof/v140/lib/libfifiase.so
```

```
COF_CONFIG_DIR=/opt/trip/cof/v140/conf
```

```
TDBS_ASELIBS=COF_ASE_FIFI
```

Reinstall Import Connector Services on Windows

When performing an upgrade installation of TRIPcof on Windows, any import connector services (hosted by the cfwimport.exe program) will be uninstalled as part of the upgrade.

Any import connector service that should be retained must be separately reinstalled after the main upgrade installation has completed. Refer to the TRIPcof Operating Manual for details.

Configuring the Metadata Copying ASE

The metadata copying ASE routine copies selected data from TRIPcof-extracted document property key/value fields (e.g. PROP_NAME and PROP_VALUE) into specific fields for particular properties (such as AUTHOR, TITLE, etc).

Note: *this behavior is separate from the document property mapping functionality available for import connectors. This ASE is provided for convenience purposes only, e.g. for use in applications that use TRIPcof for text extraction directly, without using connectors.*

The ASE, with the function name “cof_metacopy”, is written as a LOAD ASE and can therefore be used with TFORM loading, or programmatically from a TRIPnpx or TRIPjpx application.

Programmatic use from TRIPjpx or TRIPnpx involves adding an TdbAseCall instance referring to the “cof_metacopy” ASE to either the InsertAseList property (for inserting new records) or the UpdateAseList property (for updating existing records).

The ASE routine requires a file “metaase.conf” to be present in the \$COF_CONFIG_DIR directory. This file should be populated with contents like in the following example:

```
[DBNAME]
__PROP_NAME=PROP_NAME
__PROP_VALUE=PROP_VALUE
AUTHOR=Author,Primary Author,Last-Author,dc:creator,meta:author
TITLE=Title,dc:title
```

Each section (“[DBNAME]” in the above example) is the name of a database.

The property names in this file (AUTHOR and TITLE in the above example) represent the database’s field that should receive the properties represented by the values. The property values represent the extracted property names, as stored by TRIPcof.

Each value must be delimited by a comma only, and represent an alternate name of the property as found in the document files.

The default names of the document property key/value pair fields are PROP_NAME and PROP_VALUE. If different field names are used, the special tags __PROP_NAME and __PROP_VALUE can be used to inform the ASE about this.

Note that different document formats store properties under different names, and sometimes use multiple properties to represent the same thing. Although this means that there may be duplicates, the ASE is written to ignore duplicate property values. So, make sure to specify all the known names for a particular property.