



Version 1.0  
User Manual

# Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>4</b>
1.1	What is Trinem Project Manager? .....	4
1.2	About This Book .....	4
1.3	Versions Supported.....	4
1.4	Typographical Conventions .....	4
1.5	Symbols Used in This Book .....	5
<b>2</b>	<b>Project Manager – Overview.....</b>	<b>6</b>
<b>3</b>	<b>Installation Guide .....</b>	<b>7</b>
3.1	Installing Project Manager .....	7
3.1.1	Windows .....	7
3.1.2	Unix .....	7
3.2	Licensing Project Manager .....	8
<b>4</b>	<b>Running Trinem Project Manager .....</b>	<b>9</b>
4.1	Specifying the Mode.....	9
4.2	Specifying the Database Credentials .....	10
4.2.1	Using Database Username and Password .....	10
4.2.2	Using OS Authentication.....	10
4.2.3	Using Encrypted Username/Password .....	10
4.3	Specifying the Export File .....	11
4.4	Exporting a Project .....	11
4.5	Importing a Project.....	12
4.6	Password Protecting the Export File .....	12
<b>5</b>	<b>How Trinem Project Manager Works .....</b>	<b>14</b>
5.1	Export.....	14
5.2	Import .....	14
5.2.1	Overview.....	14
5.2.2	Shadow Tables.....	15
5.2.3	Renaming of Objects .....	15
5.2.4	Object Ids .....	16
5.2.5	Sequences.....	16
5.2.6	Users.....	17
<b>6</b>	<b>Using setodbc .....</b>	<b>18</b>
<b>7</b>	<b>Using with Trilogy – Empower Your Users.....</b>	<b>19</b>

7.1	Introduction.....	19
7.2	Creating the Dialog.....	19
7.3	Create the trilogy.conf entries.....	20
7.4	End User Dialog.....	20

# 1 Introduction

## 1.1 What is Trinem Project Manager?

*Trinem Project Manager* is a command-line tool which allows a Harvest or CA-SCM administrator to export a Project to a single binary file. This file can then be imported to any other CA-SCM instance of the same release or later, recreating the exported project on this new broker in its entirety.

The ability to migrate a single project in this way gives CA-SCM administrators much greater flexibility in how brokers, teams and projects are structured. By using *Trinem Project Manager*, you can copy projects from off-shore development teams, load balance CA-SCM servers by moving high-capacity projects from broker to broker, backup individual projects so that they can be restored later without regressing other development teams and so on.

## 1.2 About This Book

This book is aimed at CA-SCM Administrators. A degree of knowledge of CA-SCM, its terminology and its internal data structures is assumed.

## 1.3 Versions Supported

*Trinem Project Manager* supports Harvest r7.0 and r7.1 and also CA-SCM r12.0 and r12.1. You can export from a 7.x release of CA-SCM and import to a 12.x release but the reverse is not true – you cannot export from r12.x and import into r7.x.

For the r12.x release, the product underwent a name change from *Harvest* to *CA-SCM*. For the sake of clarity, this manual will refer only to *CA-SCM*. Any such references can be taken to also refer to *Harvest* unless the text explicitly mentions otherwise.

## 1.4 Typographical Conventions

Throughout this manual, the following typographical conventions apply:

<code>Courier</code>	Text shown in this font indicates computer output or program listing.
<b>Courier Bold</b>	Text shown in this font indicates text that should be entered by the user.

## 1.5 Symbols Used in This Book

The following symbols are used to highlight areas of text that are of particular interest:



Text Highlighted with this symbol refers to a point raised elsewhere in the manual.

---



Text Highlighted with this symbol refers to important points

---

## 2 Project Manager – Overview

*Trinem Project Manager* is a command-line tool which communicates directly with the CA-SCM Database via ODBC. It needs to be installed on the same machine as the CA-SCM broker software. Once installed, *Trinem Project Manager* can extract the data from the CA-SCM database and create an export file. Conversely, it can read an export file and recreate the project by inserting data into the CA-SCM database.

The export file contains all the information necessary to recreate the project in its entirety on another broker. This includes the project lifecycle (including all states, views, processes and access permissions), the repository or repositories referenced by the project, all snapshots, any users referenced by the project, all packages, package groups and associated forms (including any custom forms), any form attachments and – of course – all versioned data.

The exported file is in binary format and is encrypted. Only *Trinem Project Manager* can read the file to import the project. For additional security it is possible to add a password to the export file. Only a user with knowledge of the password can then import the file.

The export file is machine and database independent. You can export a project from a Windows / SQL Server based CA-SCM instance and import it to a Unix based Oracle CA-SCM instance without issue.



*Trinem Project Manager* can export projects from Harvest 7.x databases. When importing such projects to CA-SCM r12.x schemas, the data is upgraded automatically.

---

## 3 Installation Guide

### 3.1 Installing Trinem Project Manager

#### 3.1.1 Windows

The Windows version of *Trinem Project Manager* is delivered as a single ZIP file containing two binary executables:

```
tripm.exe  
setodbc.exe
```

There are no dependencies on any external libraries (dlls) for either executable (apart from standard Windows Operating System libraries held in C:\windows\system32), nor does the application ship with any dlls of its own. Therefore, installing *Trinem Project Manager* is as simple as unzipping the file into a location where you wish the product to reside.

#### 3.1.2 Unix

The Unix version of *Trinem Project Manager* is delivered a single TAR file containing two binary executables:

```
tripm  
setodbc
```

Create a directory where you wish to install Trinem Project Manager and copy the tar file into it. Extract the contents of the tar file by entering the following command:

```
tar xvf <tarfilename>
```

This will extract the tripm and setodbc executables into the directory. You can then delete the tar file.

There is a dependency on the CA-SCM ODBC drivers for the tripm application. Therefore you will need to set the library path environment variable to the location of the ODBC libraries (namely libodbc). This will be either LD\_LIBRARY\_PATH, SHLIB\_PATH or LIBPATH depending on your Unix variant.

You will also need to set the environment variable ODBC\_HOME to point to this location as this is required by the CA-SCM ODBC drivers.

## 3.2 Licensing Project Manager

*Trinem Project Manager* requires a license key in order to run the export/import functions. The license key is called `pm.lic` and needs to reside in the same directory as the *Trinem Project Manager* executable. If this file can not be located then *Trinem Project Manager* will give an error message.

---



You can still use the `-header` option to determine the contents of an export file without requiring a license key. This means that you can put *Trinem Project Manager* on any machine on which you want to see the contents of an export file without necessarily needing a license key for the machine.

---

Licensing covers the expiry date of the license and the hostname of the machine on which *Trinem Project Manager* is installed.

Here is an example `pm.lic`:

```
LICENSE KEY:  ZEVO-EOTP-ZGIY-WIDQ-LBYB-UEIC-AQCO
CUSTOMER NAME: Trinem Consulting Ltd
EXPIRY DATE:  31/12/12
HOSTNAME:     cascm-serv-01
```

---



The expiration date is specified in European Date Format (DD/MM/YYYY)

---

The customer name, expiry date and hostname are encrypted in the license key. Therefore, changes to any of these fields require a new license key to be generated. Trinem Consulting will supply an appropriate `pm.lic` file.

Do not attempt to edit the `pm.lic` file. If the license key is not recoded to match the other fields, the license file will be considered invalid and *Trinem Project Manager* will not start.

## 4 Running Trinem Project Manager

*Trinem Project Manager* is invoked from the command line as the CA-SCM user (the user who owns the CA-SCM application). On windows servers, you will find it necessary to open a command prompt window (sometimes known as a DOS box). On Unix servers you need to login as the CA-SCM user and navigate to the directory where *Trinem Project Manager* is installed.

To invoke Trinem Project Manager you enter the command:

```
tripm <options>
```

The <options> specified will depend on how you wish to connect to the CA-SCM (or Harvest) database and whether you want to do an export or an import.



You can enter:

```
tripm -h
```

to get help text and a list of available options.

---

### 4.1 Specifying the Mode

You need to specify one of three options in order to tell *Trinem Project Manager* which function to perform. These are

-exp	Perform an export
-imp	Perform an import
-header	Show the content of an export file (when it was created, the project it contains, the baseline and repository information, who performed the export and so on).

## 4.2 Specifying the Database Credentials

You can use a number of options to specify the logon credentials to the CA-SCM/Harvest database.



---

The Data Source Name (DSN) is derived automatically from the information contained in the `HServer.arg` file in `$CA_SCM_HOME`. This is the reason why *Trinem Project Manager* needs to be installed on the CA-SCM server and not on the Database Server itself.

---

### 4.2.1 Using Database Username and Password

If you don't mind specifying the username and password for the database in clear text you can specify them using the options `-usr <username>` and `-pw <password>`.

The command line would then be:

```
tripm -usr <username> -pw <password> <options>
```

### 4.2.2 Using OS Authentication

If you are using Operating System Authentication with SQL Server then you can use the option `-osauth`.

The command line would then be:

```
tripm -osuath <options>
```

### 4.2.3 Using Encrypted Username/Password

If you want to encrypt the username and password so that they are not exposed on the command line then you can use the `setodbc` utility described later in this manual to create an encrypted file. Once this is done you can reference it by using the `-eh <odbcfile>` option where `<odbcfile>` is the name of the file created by `setodbc`.

The command line would then be:

```
tripm -eh <odbcfile>
```



If you have installed Trinem's RTI utility to allow agentless deployments or in order to support deployments to iSeries/Tandem/OpenVMS etc, then you can use the same ODBC file as you created for the RTI server.

---

### 4.3 Specifying the Export File

The `-f <filename>` option specifies the name of the export file.

When doing an export, this file will be created and will hold all the data for the export.

When doing an import, this file will be read in order to recreate the project.

When using the `-header` option, this file will be read and the header information will be extracted and displayed.

This flag is mandatory and needs to be supplied for every mode (`-imp`, `-exp` and `-header`).



You can use the special filename `"-"` to send export data to standard output stream (in `-exp` mode) or to read export data from standard input stream (in `-imp` and `-header` modes).

---

### 4.4 Exporting a Project

To export a project, specify the `-env <project name>` option to tell *Trinem Project Manager* the name of the project you want to export.

A typical command line to perform an export would look like this:

```
tripm -eh <odbcfile> -exp -env <project name> -f <filename>
```

This will use the encrypted username and password held in the `<odbcfile>` to connect to the CA-SCM database via ODBC. It will then export the project named `<project name>` to the file `<filename>`.

## 4.5 Importing a Project

To import a project, specify the `-imp` flag and the `-f <filename>` directive to tell *Trinem Project Manager* which file should be imported.

A typical command line to perform an import would look like this:

```
tripm -eh <odbcfile> -imp -f <filename>
```

This will use the encrypted username and password held in the `<odbcfile>` to connect to the CA-SCM database via ODBC. It will then import the project contained in the export file `<filename>`.

Note, the project will be imported with the same name as it was exported. If a project of that name already exists, then the imported project will automatically be renamed to `<original name>-NEW`. If a project with *that* name already exists then the project will automatically be renamed to `<original name>-NEW-1` and so on.

Form Names have to be unique across projects. If an imported form shares the same name with an existing form then it is automatically renamed. If this happens then an informational message is printed.



More details on how an import works is given later in this document.

---

## 4.6 Password Protecting the Export File

If the export file contains sensitive information you may wish to protect it with a password. Providing a password during an export operation creates an export file which is protected. When importing such a file, *Trinem Project Manager* will require the same password to be entered. Failure to enter the correct password will result in the import being denied.

There are two ways of providing a password.

```
-p <password>
```

Specifying this option on the command line provides a clear-text password with no further human interaction.

If you do not wish the password to be exposed on the command line you can use the following option:

```
-prompt
```

This will prompt you to enter a password which will then be hidden (every key stroke will be echoed as a \* character).

The same options are available during both export and import operations.

During an export operation, the password specified is what will be used to protect the export file. The same password will need to be given on import. If you use the `-prompt` option during an export then you will be asked to type the password twice and both entered passwords have to be identical before the export will proceed.

During an import operation, the password specified is what will be used to validate that you are allowed to import the project from the file. If you use the `-prompt` option during an import then you will only be asked to type the password once. If the password does not match what was entered during export, the import will immediately be terminated.



Do not lose or forget the password. The password is encrypted using a one-way hashing function which cannot be reversed.

---

## 5 How Trinem Project Manager Works

### 5.1 Export

During an export operation, *Trinem Project Manager* runs queries against the CA-SCM database tables and writes the results directly to the specified export file. An export operation does not create or update any tables in the CA-SCM database.

Each table is queried to extract the subset of data relevant to the project being exported. The results of these queries are then written to the export file. The only exception to this is for users – all users are extracted and are included in the export file. This is because it is not possible to determine which users are referenced from forms.

The export data is encrypted using a rolling encryption algorithm. It is not possible to view exported data by opening the file.



The rolling encryption algorithm means that exporting the same project twice will result in two completely different files with different checksums. You cannot compare the two files for equality.

---

During an export, you will see a series of information messages detailing the name of a table along with the number of rows included in that table. These table names are only relevant during import – they are not created during an export operation.

### 5.2 Import

#### 5.2.1 Overview

During an import operation, the following steps are performed:

- The import file is read and used to create a series of “shadow” tables. These shadow tables start with the characters TRI- (to differentiate them from the CA-SCM database tables which all begin HAR-) and are usually (but not always) named similarly to the equivalent table within the CA-SCM schema.
- If the imported data is from a Harvest 7.0 or 7.1 schema and the target database is for CA-SCM r12.x, then the appropriate shadow tables are converted to the new format.
- Object IDs of all objects contained or referenced within the shadow tables are updated so that they lie outside the range of the object IDs in the target CA-SCM schema.

- Form and Project Names that already exist in the target CA-SCM schema are renamed so that they are unique.
- Any Custom Form types that exist with identical structure in the target database schema are removed from the shadow tables.
- User Object IDs and Group Object IDs are analyzed and amended so that users and groups referenced in the shadow tables that exist in the target database schema are adjusted to point to the existing user or group (and the equivalent user or group is removed from the shadow tables).
- Baseline data in the shadow tables is adjusted to take into account whether the target database has the repository / snapshot combination held in the source system.
- Once the shadow tables have been amended, a merge takes place. The data from the shadow tables is copied into the CA-SCM tables. This process operates under a single transaction so that any failures (for example, a constraint violation) will rollback the entire transaction, leaving the target database unaffected.

## 5.2.2 Shadow Tables

Shadow tables are created during an import operation. Each shadow table begins TRI- and is usually named similarly to the equivalent CA-SCM schema table to which it is to be merged after its object IDs have been amended. For example, TRIENVIRONMENT is the equivalent of HARENVIRONMENT; TRISTATE is the equivalent of HARSTATE and so on.

There are a few exceptions to this rule. For example, there are shadow tables created called TRIWORKINGVIEW and TRIBASELINEVIEW. These do not have direct equivalents in CA-SCM and are actually merged into HARVIEW. However, they are kept separate for the ease of processing before the merge takes place.

During an import operation, the shadow table will be automatically dropped if it already exists before it is recreated and populated with the data contained within the export file. Informational messages during an import will tell you if the table has been dropped and recreated or simply created.

## 5.2.3 Renaming of Objects

After all the shadow tables have been populated from the export file, then a check is made to ensure that any project or form names contained with the shadow tables do not have a matching name in the CA-SCM tables. If they do then they are renamed and an informational message is printed. This is to prevent primary key violations during the merge process.

## 5.2.4 Object Ids

All object IDs in the shadow tables are amended so that they lie outside the range of those contained within the CA-SCM tables. The following object IDs are amended:

- ATTACHMENTOBJID
- ATTRID
- ENVOBJID
- FORMOBJID
- FORMTYPEOBJID
- ITEMOBJID
- NAMEOBJID
- PACKAGEOBJID
- PKGGRPOBJID
- PROCESSOBJID
- REPOSITOBJID
- STATEOBJID
- USRGRPOBJID
- USROBJID
- VERSIONDATAOBJID
- VERSIONOBJID
- VIEWOBJID

## 5.2.5 Sequences

Sequences are amended so that, following the final merge operation, any new objects will be created correctly without clashing with the newly merged rows from the shadow tables.

The following sequences are amended:

- HARBRANCHSEQ
- HARENVIRONMENTSEQ
- HARFORMATTACHMENTSEQ
- HARFORMSEQ
- HARITEMNAMESEQ
- HARITEMSSEQ
- HARPACKAGEGROUPSEQ
- HARPACKAGESEQ
- HARPROCESSEQ
- HARREPOSITORYSEQ
- HARSTATESEQ
- HARUSERGROUPSEQ
- HARUSERSEQ
- HARVERSIONDATASEQ
- HARVERSIONSSEQ
- HARVIEWSEQ

## 5.2.6 Users

Users included in the export file which do not exist in the target database schema are automatically created as active users (that is, they have entries in both HARALLUSERS and HARUSER) with the *Disabled* attribute set. This allows them to be referenced from forms (and to act as package assignees) without them having the ability to log in to the target broker. This mechanism can be overridden by including the flag `-nodisable` in the command line options during an import. This creates users without the *Disabled* attribute set and therefore allows them to log in with their existing credentials into the target broker.

If the source database was Harvest 7.0 or 7.1 then this flag does nothing and therefore the *Disabled* attribute is always set. This is because the password encryption algorithm is different in Harvest 7.x and CA-SCM r12.x (changing from CACrypto to ETPKI) and therefore, it would not be possible for a user to login to the target broker.

## 6 Using setodbc

If you do not wish to expose the username and password to the CA-SCM database on the command line, then you can choose to store the credentials encrypted in a file which you can then pass to *Trinem Project Manager* with the `-eh` option. This is done with the `setodbc` command which is included with the distribution.

`setodbc` is invoked with the following options:

```
setodbc [ -usr <username> ] [ -pw <password> ] <filename>
```

If you specify `-usr <username>` then this is taken to be the username of the CA-SCM database owner.

If you specify `-pw <password>` then this is taken to be the password of the CA-SCM database owner.

If either `<username>` or `<password>` is not given, then you will be prompted to supply it interactively. In the case of a password, this will be done with local echo disabled (each typed character echoes a `*` character) and you will be prompted to enter the password twice.

The final parameter is the name of the file which will contain the encrypted username and password.



Note that `setodbc` also ships with Trinem's RTI (Remote Transport Interface) product which is used for agentless deployments and to control code on platforms currently unsupported by CA-SCM Agents (iSeries, OpenVMS, Tandem etc). If you have already encrypted the username/password for use with the RTI Server then you can re-use this file, there is no need to do this again for Project Manager.

---

## 7 Using with Trilogy – Empower Your Users

### 7.1 Introduction

*Trinem Project Manager* is a server-side, command line tool, designed to be used by CA-SCM administrators. However, there may be occasions when it would be desirable to allow end-users to create exports of their own projects. For example, you may wish to allow some project teams to backup their own projects periodically so that they can restore them at a later date if necessary (for example, if a team's repository is accidentally deleted). This puts the onus on to the development teams, reduces the amount of work on CA-SCM administrators and allows full database backups to be used purely for disaster recovery scenarios.

Ideally, teams granted such abilities will need access to their own export file but they should not have update access to the CA-SCM database.

Using *Trilogy* allows this to be done. *Trilogy* is a client-server tool which allows end-users to interact with scripts contained on the server. By installing *Trilogy* Client on your end-user's workstations, they can bring up dialogs which allow them to invoke scripts on the server which can then run *Trinem Deployment Manager*. By sending the export file to standard output (by using "-" as the target filename) and by specifying "filechooser" as the target for standard output in the *Trilogy* job, the export file can be routed back to the client and saved on their local filesystem.

There is insufficient room in this manual to detail all of *Trilogy's* features. The remainder of this explanation will assume working knowledge of the *Trilogy* tool. You can download the *Trilogy* manual at [www.trinem.com](http://www.trinem.com)

### 7.2 Creating the Dialog

Create a dialog file on the server containing the fields you want the end-user to see when they invoke the *Trilogy Job*. The following is a typical example:

```
- Project to Export
Project Name      {
- Encryption
Password          [*
-
```

### 7.3 Create the trilogy.conf entries

Create a Trilogy job in the server-side trilogy.conf file. Enter the following directives:

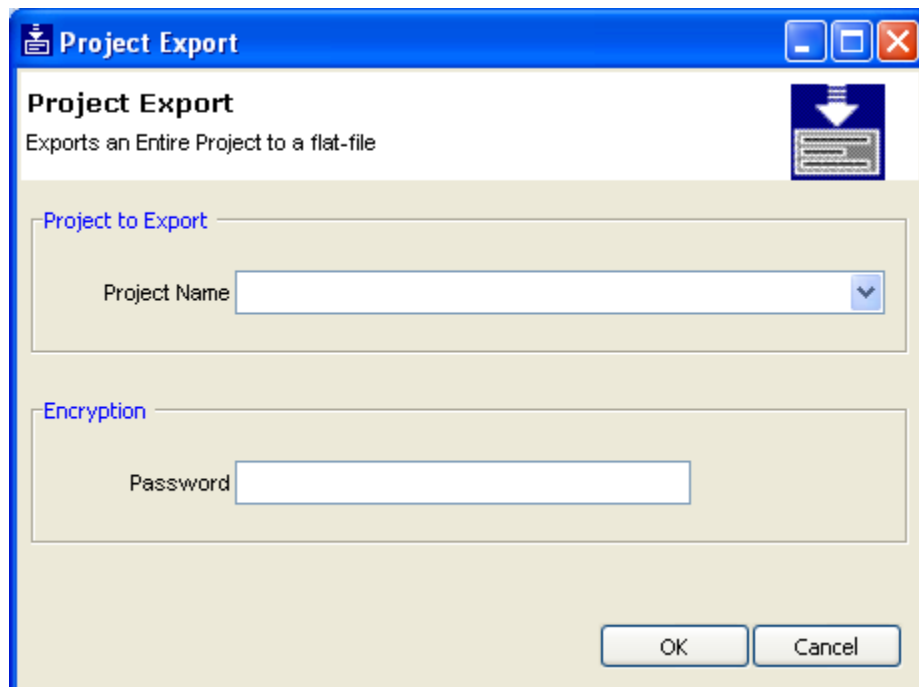
```
EXPORT:  
  Banner=On  
  BannerHeading=Project Export  
  BannerText=Exports an Entire Project to a flat-file  
  Title=Project Export  
  Dialog=$TRILOGYHOME/SCREENS/export.scn  
  Stderr=Report  
  Stdout=filechooser  
  Program=$TRILOGYHOME/scripts/export.bat  
  PopulateField1With=$TRILOGYHOME/scripts/GetProjects.bat
```

The Project Name drop-down list is populated by the server-side script "GetProjects.bat". This script should present a list of projects to which the invoking user has access.

The Standard Out stream is sent to a filechooser dialog (Stdout=filechooser). The "export.bat" script (which is invoked when the user submits the dialog) will send the export file to standard output (using -f - option to *Trinem Project Manager*). This will automatically cause the informational messages to be routed to standard error so we capture those into a Report Window at the client end (Stderr=Report).

### 7.4 End User Dialog

When presented at the client, the dialog will look like this:



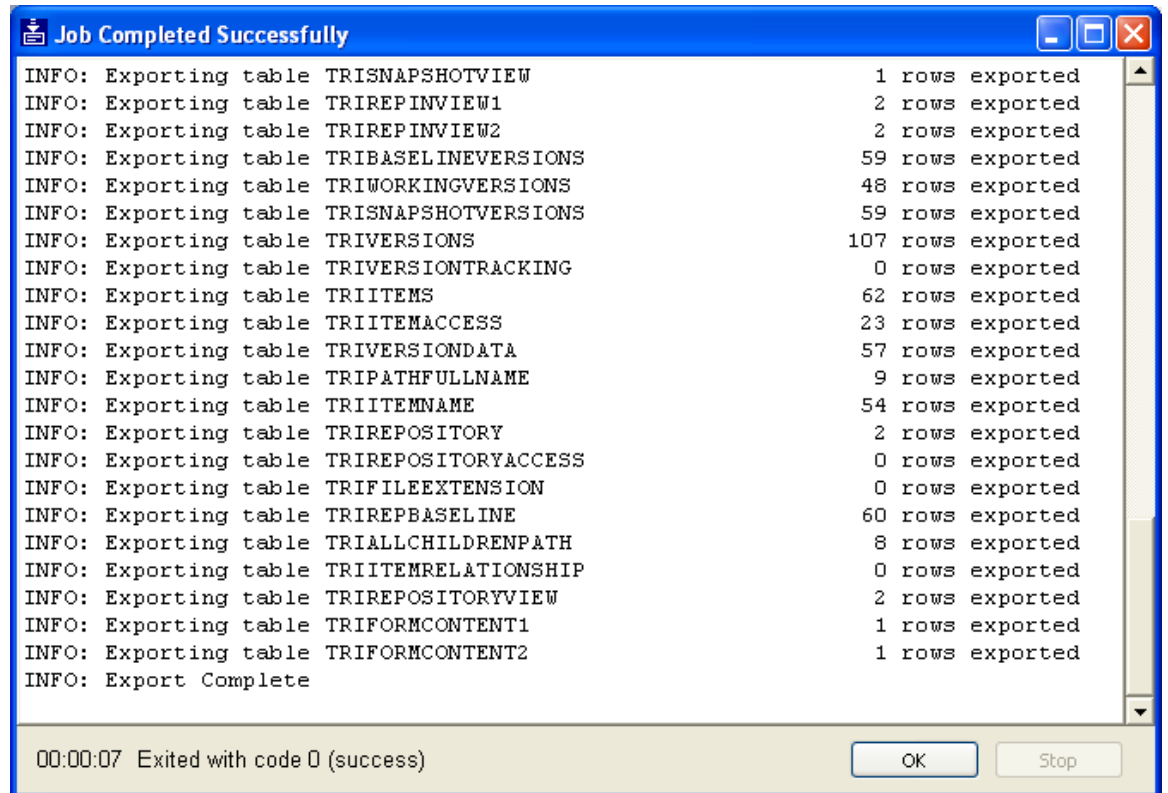
When OK is clicked the server-side script "export.bat" will be invoked.

export.bat should look something like this:

```
@echo off
triexport.exe" -eh pm.odbc -exp -env "%TRIFIELD1%" -p "%TRIFIELD2%" -f -
```

When this runs, the project to export will be taken from the environment variable TRIFIELD1 and the password to encrypt the export with will be taken from the environment variable TRIFIELD2. Both of these are set by *Trilogy* and represent the content of the client-side dialog (field 1 being the project to export and field 2 being the password).

At the point where the dialog is submitted, a filechooser dialog will open, asking for the name and location to where the export file should be saved. Once this filechooser dialog is submitted, the server-side script runs and a Report Window opens giving the output of standard error, thus:



At the conclusion of this process, the client end-user will have the export file saved on their client workstation.

## Appendix A – Command Line Options

-exp	Perform an export of the specified project.
-imp	Import the project from the specified file.
-header	Show the header information contained in the specified export file and exit without attempting an import.
-f <file>	File to use for export, import or header (Use - to send export to standard output or to read export from standard input).
-usr <user>	Username used to connect to the database.
-pw <pass>	Password used to connect to the database.
-dbauth	Database Connection is authenticated via OS
-eh <db pass file>	Filename containing the encrypted database username/password. Created with setodbc. Use this instead of -usr and -pw or -osauth.
-env <proj>	Specifies name of project to export (This is mandatory if -exp is specified).
-fto2	Send informational messages to stderr stream rather than stdout. Note, if -f - is used, then this option is set automatically
-p <file password>	Password protecting the export file. On export, this is the password with which to protect the file. On import, this is the password with which to validate access to the file.
-prompt	As -p but prompts interactively for the password.
-logfile <file>	Write all log output to logfile (as well as stdout/stderr)
-nodisable	Normally, new users are created with the "Disabled" marker set to prevent them logging in to to the target broker. Using this flag creates the users as "active". Note, if the export was taken from a r7.x system then the user is automatically disabled and this flag has no effect (due to the change in password encryption)

## Appendix B – License Terms and Conditions

### B.1 Terms Used in this License

"We", "us" and "our" refers to Trinem Consulting Ltd. "You" and "your" refers to the individual or entity that has ordered the programs from us. "Programs" refers to the various software components that make up the entire *Trinem Project Manager* application suite and includes the documentation. "License" refers to your right to use the programs under the terms of this agreement. The laws of the United Kingdom govern this agreement. You and Trinem Consulting Ltd agree to submit to the exclusive jurisdiction of, and venue in, the courts of the United Kingdom in any dispute relating to this agreement.

### B.2 License Overview

We are willing to license the programs to you only upon the condition that you accept all the terms contained in this agreement. Read the terms carefully as the installation of any portion of the *Trinem Project Manager* software on one or more of your organization's computers confirms your acceptance. If you are not willing to be bound by these terms please do not install any portion of the *Trinem Project Manager* software.

### B.3 License Rights

We grant you a non-exclusive, non-transferable license to use the programs only for the purposes described in the documentation. *Trinem Project Manager* is licensed separately for each server on which it is installed. You should not install *Trinem Project Manager* on more workstations than are governed by your purchase order. Your purchase order will list the servers on which you are permitted to install *Trinem Project Manager*. If you wish to install the *Trinem Project Manager* on more than this number of servers then you must purchase additional licenses. Contact Trinem Consulting Ltd for more information.

### B.4 Ownership and Restrictions

We retain all ownership and Intellectual Property Rights in the programs (excluding those open-source components supplied by external organisations – see below). The programs may be installed on your computers only and must not be installed on any other organisation's computers. You may make one copy of the original media for backup purposes.

You must not:

- 1) Remove or modify any program markings or any notice of our proprietary rights;
- 2) Make the programs available in any manner to any third party;
- 3) Use the programs to provide third-party training;
- 4) Assign this agreement or give or transfer the programs or an interest in them to another individual or entity;
- 5) Cause or permit reverse engineering or de-compilation of the programs.
- 6) Disclose results of any program operation, functionality or benchmark tests without our prior consent
- 7) Use any Trinem Consulting Ltd name, trademark or logo.

## **B.5 Export**

You agree that United Kingdom export control laws and other application export and import laws govern your use of the programs, including technical data. You agree that neither the programs nor any direct product thereof will be exported, directly or indirectly, in violation of these laws, or will be used for any purpose prohibited by these laws including, without limitation, nuclear, chemical, or biological weapons proliferation.

## **B.6 Disclaimer of Warranty and Exclusive Remedies**

THE PROGRAMS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND. WE FURTHER DISCLAIM ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

IN NO EVENT SHALL WE BE LIABLE FOR ANY INDIRECT, INCIDENTAL, SPECIAL, PUNITIVE OR CONSEQUENTIAL DAMAGES, OR DAMAGES FOR LOSS OR PROFITS, REVENUE, DATA OR DATA USE, INCURRED BY YOU OR ANY THIRD PARTY, WHETHER IN AN ACTION IN CONTRACT OR TORT, EVEN IF WE HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. OUR ENTIRE LIABILITY FOR DAMAGES HEREUNDER SHALL IN NO EVENT EXCEED THE PURCHASE PRICE OF THE PROGRAMS.

## **B.7 Technical Support**

Our technical support team will provide web and email based technical support and updates to you for the programs licensed under this agreement for the duration of any technical support agreement detailed in the purchase order.

## **B.8 End of Agreement**

You may terminate this agreement by destroying all copies of the programs. We have the right to terminate your right to use the programs if you fail to comply with any of the terms of this agreement, in which case you shall destroy all copies of the programs.

## **B.9 Relationship Between the Parties**

The relationship between you and us is that of licensee/licensor. Neither party will represent that it has any authority to assume or create any obligation, express or implied, on behalf of the other party, nor to represent the other party as agent, employee, franchisee, or in any other capacity. Nothing in this agreement shall be construed to limit either party's right to independently develop or distribute software that is functionally similar to the other party's products, so long as proprietary information of the other party is not included in such software.

## **B.10 Entire Agreement**

You agree that this agreement is the complete agreement for the programs and licenses, and the agreement supersedes all prior or contemporaneous agreements or representations. If any term of this agreement is found to be invalid or unenforceable, the remaining provisions will remain effective.