

# Transformer Inspector Guide

***Prepared for:*** Morphis Customers/Partners

***Prepared by:*** Sales Team

[sales@morphis-tech.com](mailto:sales@morphis-tech.com)

***Document Ref:*** PCR\_SA\_001\_0001      ***Revision:*** 1.0

***Revised On:*** 05 April 2019



**morphistech**®

# TRANSFORMER INSPECTOR GUIDELINES

## CONTENTS

1.	Transformer inspector .....	3
1.1	Overview .....	3
1.2	How to use it.....	3
1.3	Transformer inspector prerequisites .....	5
2.	Running the analyses process on Linux.....	5
2.1	Introduction .....	5
2.2	Process.....	5
2.2.1	Extract text/xml from the binaries .....	5
2.2.2	Run morphis forms2xml (located in the inspector instalation folder).....	6
2.2.3	Run transformer inspector.....	6

## 1. TRANSFORMER INSPECTOR

### 1.1 OVERVIEW

The Transformer Inspector tool has been developed by Morphis and belongs to the Transformer migration solutions' family. It has been conceived for the early stage of the migration process to produce statistical information for any Oracle Forms/Reports application.

The output of the tool consists on a .ZIP file containing a XML file per analyzed module with statistics on the number of source objects, *Oracle@Forms* built-in functions, *Oracle@Reports* built-in functions and SQL statistics.

### 1.2 HOW TO USE IT

For Oracle Forms version 10g and 11g, Inspector utilizes *frmf2xml* and *frmcmp* tools to extract information from modules.

For Oracle Reports, 6i, 9i or 10g, Analyzer utilizes *rwcon60* (6i) or *rwconverter* (9i and 10g), to extract information from modules.

In order to analyze a particular module in both versions, that module must open in Oracle Forms/Reports Builder correctly, i.e., any referenced object in the module must be accessible through the Forms/Reports Builder. By using Oracle Forms/Reports API, Transformer Inspector avoids connecting to the application Database and compilation of the modules to be analyzed.

On start-up, Transformer Inspector obtains from the current Oracle Forms Builder setup the list of built-in functions available. The Transformer Inspector can still process the modules if this operation fails, however, calls to built-in functions may be misinterpreted as and wrongly considered calls to user defined functions. To avoid this, rename the file "*BuiltIns\_Forms60.xml*" or "*BuiltIns\_Forms90.xml*" to "*BuiltInsList.xml*".

Transformer Inspector has a very simple User Interface (Figure 1).

The user must select the Oracle Forms version (6i, 9i or 10g), the codepage used when the Oracle modules were created, add the module(s) to analyze, the output directory to which the output data should be written and a name for the output statistics file.

NOTE: Unexpected results may happen if the incorrect Oracle Forms version is used.

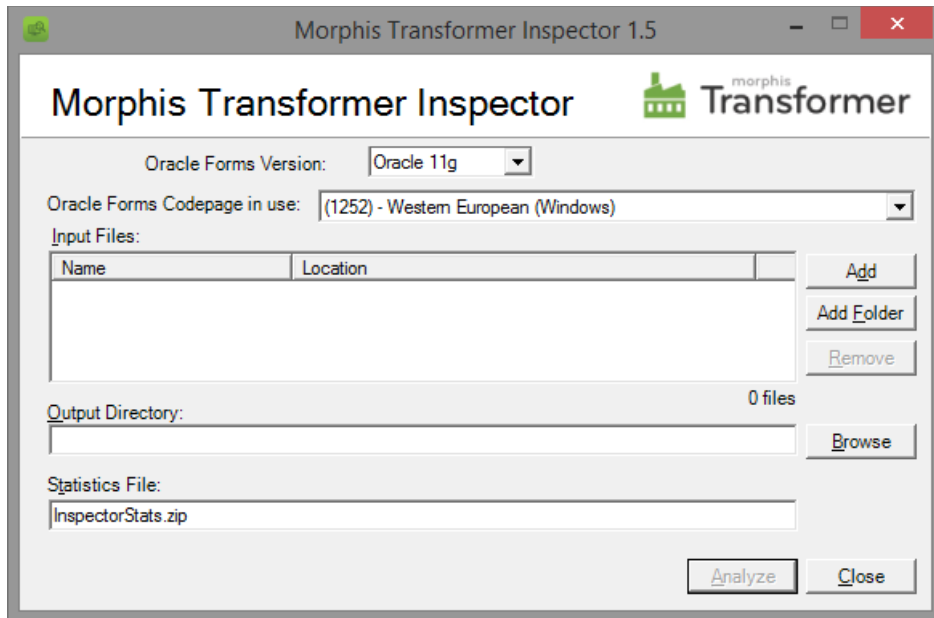


Figure 1 - Transformer Inspector

The option Fragment Statistics File creates several output files (indexed by 1,2...) up to the given maximum size.

The Transformer Inspector writes in the specified output directory a zip file with a XML file per analyzed module containing statistical information on the number of source objects, *Oracle@Forms* built-in functions used, *Oracle@Reports* built-in functions, and SQL statistics.

## Oracle Forms 6i

When using Transformer Inspector over Oracle Forms 6i an unpatched version of Oracle Forms Builder has to be used (Version 6.0.8.11.3)

## 1.3 TRANSFORMER INSPECTOR PREREQUISITES

- Microsoft .NET Framework 4.5 and 2.0;
- Microsoft Visual J# Version 2.0 Redistributable Package;
- Oracle Forms builder 6i, 9i, 10g or 11g;

### Forms 6i additional prerequisite

Microsoft Visual C++ 2008 SP1 Redistributable Package

## 2. RUNNING THE ANALYSES PROCESS ON LINUX

### 2.1 INTRODUCTION

It is possible to do the analysis process if *Oracle® Forms Builder* is only installed on Linux, although it is a more complex process. The following illustrates this process.

### 2.2 PROCESS

#### 2.2.1 EXTRACT TEXT/XML FROM THE BINARIES

On the Linux machine where *Oracle® Forms Builder* is installed, run the following commands:

#### Oracle 9i

```
- iff2xml90 DUMP=ALL OVERWRITE=YES <module_absolute_path>
  (applies to fmb, mmb and olb files)
- ifcmp90 module=<module_absolute_path> module_type=library
batch=yes script=yes logon=no window_state=minimize
  (applies to pll files)
```

Output files: \*\_fmb.xml, \*\_mmb.xml, \*\_olb.xml, \*.pdI

## Oracle 10g or 11g

```
- frmxmlsg (for 10g or 11g form schema generation)
- frmf2xml DUMP=OVERRIDEN OVERWRITE=YES USE_PROPERTY_IDS=YES
<module_absolute_path>
  (applies to fmb, mmb and olb files)
  - frmcmp module=<module_absolute_path> module_type=library
batch=yes script=yes logon=no window_state=minimize
  (applies to pll files)
```

Output files: \*\_fmb.xml, \*\_mmb.xml, \*\_olb.xml, \*.pdl+

### 2.2.2 RUN MORPHIS FORMS2XML (LOCATED IN THE INSPECTOR INSTALLATION FOLDER)

1. Move the output from the previous step to the Windows machine where Morphis Inspector is installed;
2. Run Forms2Xml on the file with the -ora modifier.

#### Forms2xml usage:

```
Forms2Xml.exe <module_absolute_path> -t <module_type> -c <codepage number> -v <version> [-d <output dir>] [-disableDefaults] [-disableInheritance] [-l <log file>]
options:
-t <module_type> = [F=FORM | M=MENU | O=OLB | L=LIB]
-c <codepage_number> = "Ex.: 1252 for Latin1"
-v <oracle_version> = [6i | 9i | 10g | 11g]
-d <output_dir>
-disableDefaults = disable default properties
-disableInheritance = disable inherited properties
-l <log_file> = log file name
-ora = when using this switch the module_absolute_path is the file extracted with oracle commands (applies only to 9i,10g, 11g)
```

Output files: \*.xfmb, \*.xmmb, \*.xpdl

### 2.2.3 RUN TRANSFORMER INSPECTOR

Use previous step output to run transformer inspector.



