agile™ How Products Become Profits™

Agile® e6.0

Agile e6 — Upgrade tool 3.0

Installation and Configuration Manual

Copyrights and Trademarks

The material in this document is for information only and is subject to change without notice. While reasonable efforts have been made in the preparation of this document to ensure its accuracy, Agile Software Corporation assumes no liability resulting from errors or omissions in this document or from the use of the information contained herein. Agile Software Corporation reserves the right to make changes in the product design without reservation and without notification to its users.

NOTICE OF RESTRICTED RIGHTS:
The Software is a “commercial item,” as that term is defined at 48 C.F.R. 2.101 (OCT 1995), consisting of “commercial computer software” and “commercial computer software documentation” as such terms are used in 48 C.F.R. 12.212 (SEPT 1995) and when provided to the U. S. Government, is provided (a) for acquisition by or on behalf of civilian agencies, consistent with the policy set forth in 48 C.F.R. 12.212; or (b) for acquisition by or on behalf of units of the Department of Defense, consistent with the policies set forth in 48 C.F.R. 227.7202-1 (JUN 1995) and 227.7202-4 (JUN 1995).

July 7, 2005

CONTENTS

Chapter 1

Introduction

This guide describes the installation and configuration of the Upgrade Tool 3.0 for migrating from Agile e5.1 to Agile e6.0.

The Upgrade Tool addresses experienced project engineers and PLM administrators with customizing and database experience. Do not use the tool without the necessary knowledge. Read the complete manual in order to get all necessary information. Do not attach to or even change the productive system. Always work on a copy of the productive database dump. Avoid working on productive computers to exclude any influence on the system. Never insert database connections of productive database users in any configuration file or script except for exporting the dump or a source for copying tables (Productive Database).

Do not use the software in any situation where significant damage to property or business could occur from a software error. In no event will AGILE or any other party who has been involved in the creation, production or delivery of the software be liable for special, direct, indirect, incidental, punitive or consequential damages, including loss of profits, revenue, business, goodwill, data or computer programs or inability to use the software, however caused and regardless of the theory of liability even if AGILE or such other party has been advised of the possibility of such damages.

Prerequisites

The Upgrade tool runs on the following software:

- ❑ Server platforms

  Windows 2003 Server

  Unix (all platforms supported by Agile e6.0)

- ❑ Memory

  The upgrade tool needs at least 512 MB of memory

- ❑ Databases

  Oracle 8.1.7 / 9.2.0.4 (Source database)

  Oracle 10.1.4 (Target database)

The best performance is reached when installing the upgrade tool on your database server. It is also possible to work on any machine in the LAN. The machine should have at least 1GB memory.

To install the Upgrade Tool you need at least

- ❑ 250 MB disk space for the software and generated log and data files
- ❑ sufficient disc space to store copies of your productive database and the reference dumps on your database.

Additional documentation

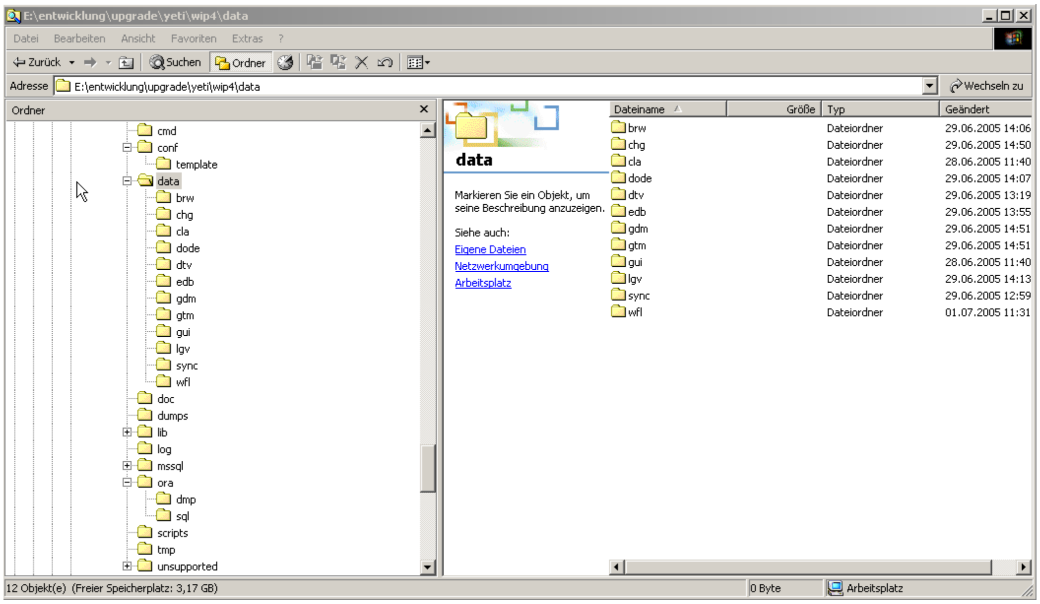
For general information about the upgrade process see the additional documentation Overview Upgrade Process in the directory **upgrade/doc/** or on the Support page.

For additional information and most up-to-date Upgrade information, check the Agile Support page at http://eignersupport.agilesoft.com/index.asp (you will need a password to enter the support website).

Installing the Upgrade Tool

1. Extract the software (upgrade.zip) to a folder *Upgrade* on your database server.

   The following structure is created:



E:\entwicklung\upgrade\yeti\wip4\data
Datei Bearbeiten Ansicht Favoriten Extras ?
Zurück
Suchen
Ordner
Adresse
Wechseln zu
Ordner
data
Markieren Sie ein Objekt, um seine Beschreibung anzuzeigen.
Siehe auch:
Eigene Dateien
Netzwerkumgebung
Arbeitsplatz
12 Objekt(e) (Freier Speicherplatz: 3,17 GB)
0 Byte
Arbeitsplatz

Adresse: E:\entwicklung\upgrade\yeti\wip4\data

```
cmd
conf
    template
data
    brw
    chg
    cla
    dode
    dtv
    edb
    gdm
    gtm
    gui
    lgv
    sync
    wfl
doc
dumps
lib
log
mssql
ora
    dmp
    sql
scripts
tmp
unsupported
```

Dateiname	Größe	Typ	Geändert
brw		Dateiordner	29.06.2005 14:06
chg		Dateiordner	29.06.2005 14:50
cla		Dateiordner	28.06.2005 11:40
dode		Dateiordner	29.06.2005 14:07
dtv		Dateiordner	29.06.2005 13:19
edb		Dateiordner	29.06.2005 13:55
gdm		Dateiordner	29.06.2005 14:51
gtm		Dateiordner	29.06.2005 14:51
gui		Dateiordner	28.06.2005 11:40
lgv		Dateiordner	29.06.2005 14:13
sync		Dateiordner	29.06.2005 12:59
wfl		Dateiordner	01.07.2005 11:31

   The file *upgrade\cmd\upg_env.cmd* (PC) or *upgrade/scripts/upg_env.sh* (Unix) contains the environment definition for your upgrade process.

2. Adapt the following environment definitions in upg_env.sh / upg_env.cmd:

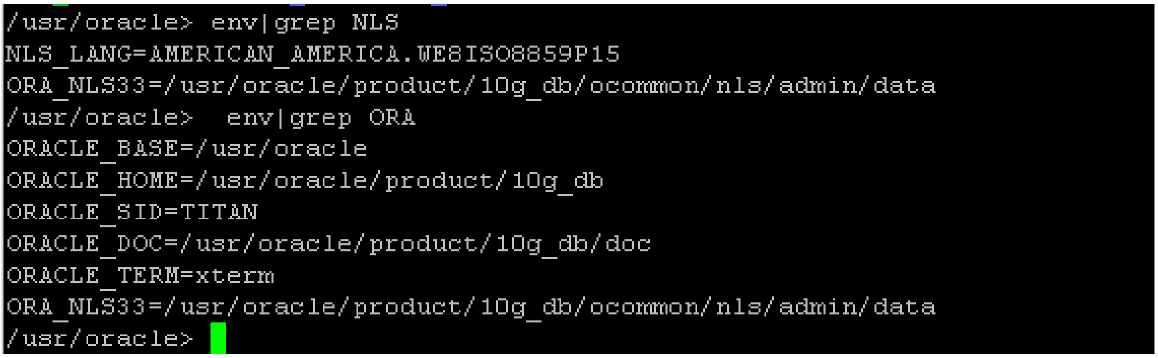
   - **JAVA_HOME**
     At least JRE 1.4.2 is required by upgrade tool.
     In the standard configuration of the file the JRE of the Agile e6.0 installation is used for that.

   - **ORACLE settings**
     Make that Oracle 10.1.4 environment is set before proceeding with upgrade.

   To check the environment for UNIX execute the following commands

   env|grep ORA

   Output should look like this:



```
/usr/oracle> env|grep NLS
NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P15
ORA_NLS33=/usr/oracle/product/10g_db/ocommon/nls/admin/data
/usr/oracle>  env|grep ORA
ORACLE_BASE=/usr/oracle
ORACLE_HOME=/usr/oracle/product/10g_db
ORACLE_SID=TITAN
ORACLE_DOC=/usr/oracle/product/10g_db/doc
ORACLE_TERM=xterm
ORA_NLS33=/usr/oracle/product/10g_db/ocommon/nls/admin/data
/usr/oracle> 
```

Chapter 3

Controll database settings

Control database settings in Oracle

The Upgrade Tool needs a well-configured database to provide a good performance. The Oracle standard database settings are not sufficient to run the program within the stated time.

1. Check the Oracle Parameter and verify that at least the following minimum values are set in your database instance:

   - ❑ db_cache_size >= 200 000 000 (200MB)
   - ❑ shared_pool_size >= 100 000 000 100Mbytes
   - ❑ log_buffer >= 163840 3*64 Kbytes

   If database memory consumption is too small, adapt the values.

   If you use the server parameter file **spfile** (like in the Agile e6 standard installation), execute the following commands to change the values of the initialization parameters.

   - ❑ Login into sqlplus as user sys
     SQL>ALTER SYSTEM SET <parameter name>=>Value> SCOPE=BOTH

     Note: Do not change the values of productive systems. Make a copy of the initialization file and adapt the values.
   - ❑ Read the Oracle online manuals and the Oracle10.1 installation manual from Agile in addition.

   Oracle needs physical memory. If the system starts swapping or paging, the Oracle performance degrades or causes errors. Examine your free physical memory and prevent the OS from swapping.

   Some UNIX systems have maximum values for shared memory. Refer to the installation instructions before changing any value.

2. Check the SQL Net configuration (Oracle only).

   The network domain is part of different oracle settings. Please check if the domain is consistently used for the following settings:

   - ❑ Global Database
   - ❑ Service name
   - ❑ Listener.ora
   - ❑ Default domain name

Global Database name

- □ Login as user sys and check the global database name.

The name should contain the network domain.

```
Sqlplus  <system>/<db_passowrd>@<db_service>
SQL>select * from global_name;
```

Example

```
GLOBAL_NAME -----------------------
PLM.WORLD
```

The example uses the default network domain in world. Also possible are values like agile.agilesoft.com.

Change the global database name login to SQL plus and execute the following commands:

```
SQL>alter database global name plm.agile.agilesoft.com
```

Service name

Service name in the SQL net configuration file tnsnames.ora network in the directory $ORACLE_HOME/network/admin

The service name must also include the network domain. Please check the seeting in the sqlnet configuration file tnsnames.ora

- □ Change to the directory $ORACLE_HOME/network/admin
- □ Open the file tnsnames ora and check if the service name is fully defined. That means the name contains the same network domain as the global database name.

```
PLM.WORLD =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = PLM.WORLD)
    )
  )
```

listener .ora

- □ Check if the global database name in the section SID_List of the listener configuration file contains also the same fully qualified global database name.

```
SID_LIST_LISTENER_PLM =
  (SID_LIST =
    (SID_DESC =
    (GLOBAL_DBNAME = PLM.WORLD)
    (SID_NAME = plm)
  )
 )

LISTENER_PLM =
 (DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
 )
```

Service name

Default domain name in SQL net configuration file is sqlnet.ora

The default setting for the network domain in the sqlnet.ora file should be the same.

- Change to the directory $ORACLE_HOME/network/admin and Open the file sqlnet.ora and check the default domain settings.

```
names.default_domain = world
```

Control database settings in SQL Server

The memory consumption of an SQL Server is dynamic.

1. Start the Enterprise Manager.
2. Select the server name and choose SQL-Server Properties in the Extra Menu.
3. Select memory folder in the new window and adjust the values.

Chapter 4

Prepare environments

Prepare reference environments

The upgrade tool needs the following database environments/users:

Source master	CADIM/EDB, axalant or Agile 5.x reference dump
Target master	Agile e6.0 reference dump
Customer	customer dump

A separate database user is needed for each environment.

1. Download the necessary reference dumps from Agile support website and unzip them in the upgrade/dumps directory.

   Note: Do not change the names of the downloaded reference dump files. The dumps cannot be imported automatically if a different name for the dump file is used.

2. Check if the following table spaces (Oracle) or file groups (SQL Server) exist in your database

   edb_tmp

   edb_tmpidx

   edb_lob

   If one of them does not exist they have to be created:

   Oracle

   Change to the directory *upgrade/ora/sql*

   Adapt file names, paths and file size in the script *cre_axa_tbs.sql*

   Login as user *system* to *sqlplus* and execute *cre_axa_tbs.sql*

   sqlplus system/<password>@agile

   SQL> @cre_axa _tbs.sql

   Note: Example how to call missing_f.cmd in the command line:
   missing_f.cmd[1] sa[2] password[3] ceqell\axa[4] edbprod[5] d:\mssql\data[6] d:\mssql\data[7] d:\mssql\data[8]

Parameters:

1 script name to be executed

2 database administrator login

3 database administrator password

4 instance name (e.g <hostname>\agile)

5 database name

6 path where edb_lob filegroup will be placed

7 path where edb_tmp filegroup will be placed

8 path where edb_tmpidx filegroup will be placed

3. Import reference and customer dumps

Note: For importing the dumps, do not change the table space names, because the created table statements on tables containing a blob clause will fail if the original table spaces EDB, EDB_IDX and EDB_LOB do not exist

Copy your customer dump file in the directory *upgrade/dumps*

Rename the file to <*db_user*>*.dmp* where <*db_user*> is the user name of your customer dump (e.g. *customer.dmp*)

Run *imp_dmp.cmd* (PC) or *imp_dmp.sh* (UNIX) to restore the original and target master databases (oracle and SQL Server) and the customer database (oracle only). To create your customer environment in SQL Server use Backup/Restore functionality of SQL Server.

4. Create statistics for all involved database schemas

Check Language settings

Because of an Oracle bug the setting for the environment variable NLS_LANG must be AMERICAN_AMERICA.WE8ISO8859P15. Otherwise statistics will not computed correctly.

Login as user with dba priviledge and perform analyzing.

```
sqlplus> EXECUTE  DBMS_STATS.GATHER_SCHEMA_STATS ('<schema>', N);
N is the sample %  for statistics collection(use 100%)
```

Chapter 5

General Configuration of the Upgrade Tool

Define database connections

The definition of the database connections is done in four steps:

Source Master:
Standard reference dump corresponding to the version of CADIM / axalant / Agile e-series customer dump

Target Master :
Standard reference dump for desired target version

Customer:
Database environment containing CADIM / axalant / Agile e5.x customer data

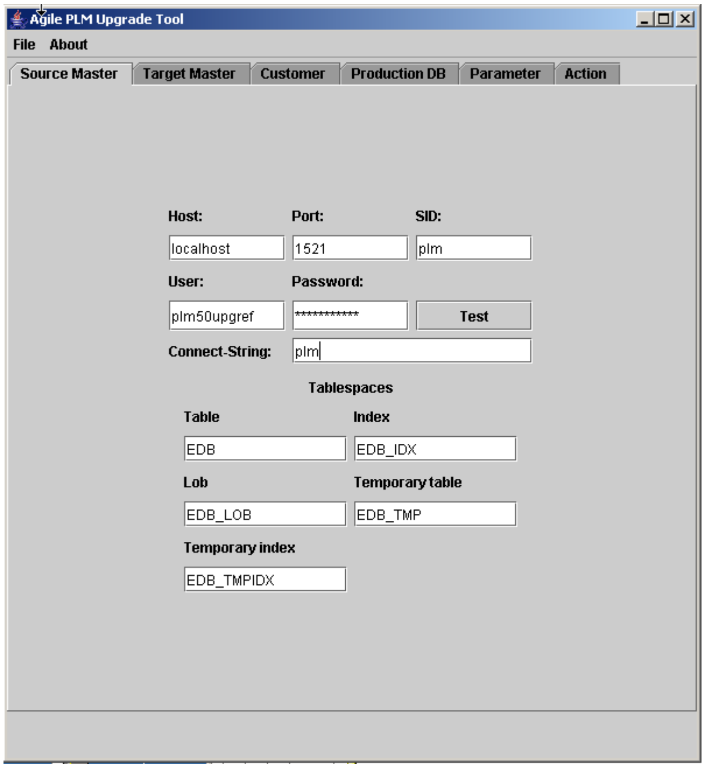
Productive Database:
Database environment containing productive CADIM / axalant / Agile e5.x . This connection is used as the source for taking over productive data

Start Upgrade Tool

1. Start the Upgrade tool

PC:	Run start_upg.cmd
UNIX:	Run start_upg.sh

The following screen will be opened:



Host: localhost
Port: 1521
SID: plm
User: plm50upgref
Password: ************
Connect-String: plm

Tablespaces

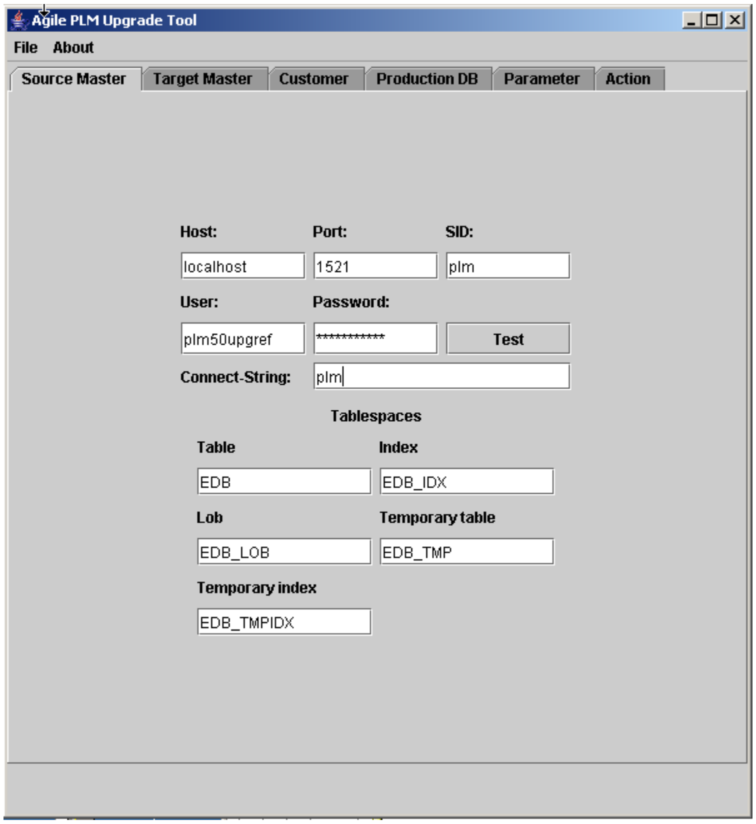
Table: EDB
Index: EDB_IDX
Lob: EDB_LOB
Temporary table: EDB_TMP
Temporary index: EDB_TMPIDX

Source Master

This is the connection to the CADIM, axalant or Agile 5.x reference dump.

1. Select the tab Source Master.

   Do not change this connection to a different CADIM /axalant / Eigner PLM dump. The reference dumps are modified. Use the database dumps delivered with the Upgrade Tool. Only the reference dumps have the Agile e6.0 table format but CADIM/axalant/Eigner PLM contents. *Standard LGV models are dropped!*



Host: localhost
Port: 1521
SID: plm
User: plm50upgref
Password: ***********
Connect-String: plm

Tablespaces

Table: EDB
Index: EDB_IDX
Lob: EDB_LOB
Temporary table: EDB_TMP
Temporary index: EDB_TMPIDX

2. Enter the following information:

Database selection	Select the desired version
Host	host name of database server
Port	port number of Oracle listener (default 1521) or SQL Server port number (default 1433)
SID	Oracle_SID (uppercase) or database name for SQL Server (lowercase)
User	database user name
Password	password of database user
Connection String	Service name, which is used to run SQL*PLUS commands on the machine the upgrade tool is installed on. Use fully qualified name including the network domain, f.e. plm.aile.agilesoft.com

Tablespaces:

Note: Name of used tablespaces (Oracle - uppercase) or file groups (SQL Server - lowercase)

Table	Default EDB[*]
Index	Default EDB_IDX[*]
LOB	Default EDB_LOB[*]
Temporary table	Default EDB_TMP[*] (edb on SQL Server until axalant 2000 SP3)
Temporary index	EDB_TMPIDX[*] (edb_idx on SQL Server until axalant 2000 Sp3)

[*] Lower case for SQL Server

Important: Press return after every change in a field. The color turns back from red to black. Otherwise the changes will be lost.

3. Test the connection using the Button "TEST

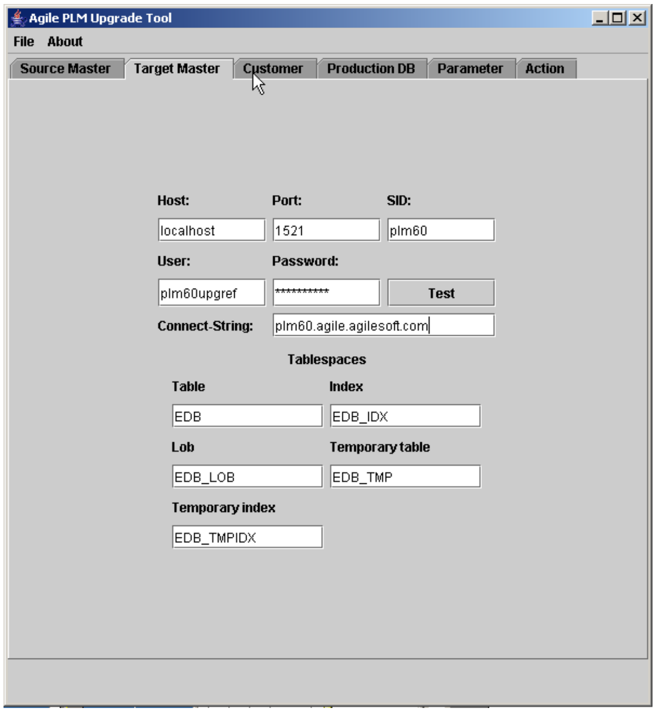
Note: The Java Connection can only be tested in Version 3.0.

Target Master

This is the database environment containing the Agile e6.0 reference dump.

1. Select the tab Target Master.
2. Fill out the connect parameters similar to Source Database.
3. Test the connection using the Button "TEST

Note: The Java Connection can only be tested in Version 3.0.



Host: localhost
Port: 1521
SID: plm60
User: plm60upgref
Password: **********
Connect-String: plm60.agile.agilesoft.com

Table: EDB
Index: EDB_IDX
Lob: EDB_LOB
Temporary table: EDB_TMP
Temporary index: EDB_TMPIDX

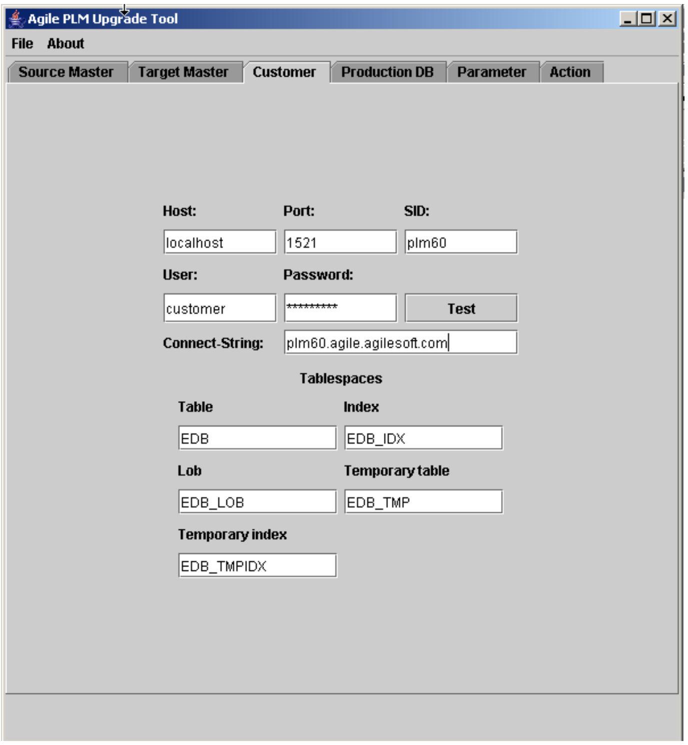
The Customer dump

This is the database connection to your customer dump. **B**

> Important: Make sure not to use a productive database dump!

1. Select the tab *Customer*.
2. Fill out the connect parameters similar to Source Database.
3. Test the connection using the Button “TEST”

> Note: The Java Connection can only be tested in Version 3.0.



Host: localhost
Port: 1521
SID: plm60
User: customer
Password: *********
Connect-String: plm60.agile.agilesoft.com

Table: EDB
Index: EDB_IDX
Lob: EDB_LOB
Temporary table: EDB_TMP
Temporary index: EDB_TMPIDX

Productive DB

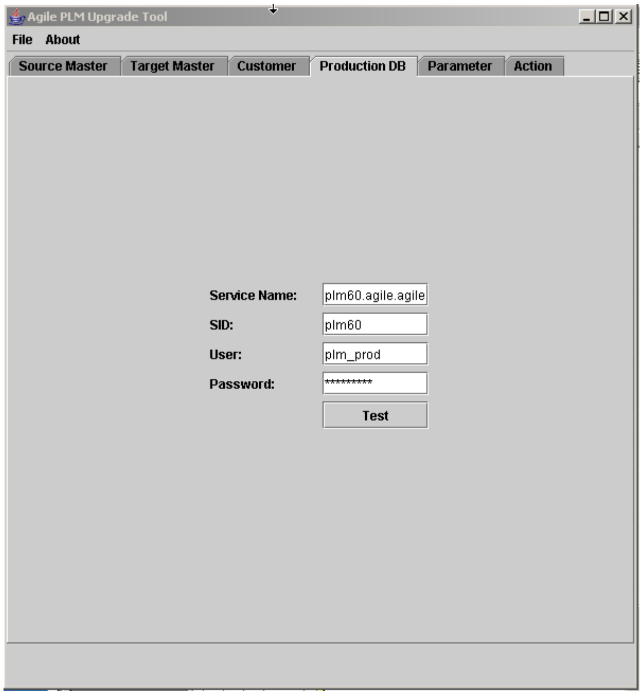
This is the database connection to the productive system. This connection is used as source for the transfer of productive data.

Compared to other connection definition only the service name of the sqlnet connection must be defined (defined in *tnsnames.ora* e.g. agile).

1. Select the tab *Production DB*
2. Make the following entries

ParameterName	Description
Service Name	Oracle service name including network domain e. g. AGILE.AGILESOFT.COM. Service name must be defined in tnsnames.ora
SID	Oracle_SID (uppercase) or database name for SQL Server (lowercase)
User	database user name
Password	password of database user

3. Test the connection using the Button “TEST



Service Name: plm60.agile.agile

SID: plm60

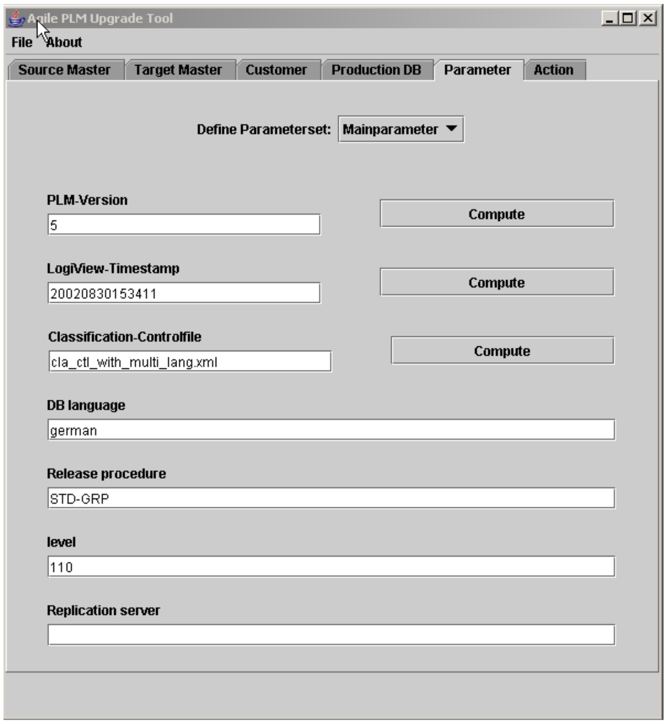
User: plm_prod

Password: *********

Parameters

Additional parameters which are available in the file ApplicationParamter.xml can be viewed and edited.

1. Select the tab *Parameter*



Define Parameterset: Mainparameter

PLM-Version: 5

LogiView-Timestamp: 20020830153411

Classification-Controlfile: cla_ctl_with_multi_lang.xml

DB language: german

Release procedure: STD-GRP

level: 110

Replication server:

2. Review and correct the entries if necessary and check the following table for valid entries.

   With the "Compute" button, the right values can be determined. Always check the computed values.

Parameter name	Description
PLM-Version	The customer dump version Following values are valid: 1 = CADIB/EDB 2.3.x 2 = AXALANT SP1 3 = AXALANT SP2 4 = AXALANT SP3 5 = PLM 5.0 6 = Agile e6.0
Logiview Timestamp	A Timestamp All logiview items with a change date after this time point will be deleted. You can adapt this value manually. Following values are possible: CADIM/EDB 2.3.2 – 19990329094555 CADIM/EDB 2.3#3 – 19990707174038 CADIM/EDB 2.3#4 – 19990707174038 CADIM/EDB 2.3#5 – 20000329161725 axalant2000 SP1 – 20001109140557 axalant2000 SP2 – 20010723102350 axalant2000 SP3 – 20011113092600 axa2000 SP3 PA1 – 20020808110309 Eigner PLM 5.0 - 20020830153411 The version must correspond to your customer dump version
Classification – Controlfile	A file name of the control file for the customer dump in the present case Valid entries are: cla_ctl.xml (used for CADIM/EDB and no multi language fields for classification attributes c_letter and Class cla_ctl_with_repl.xml used for CADIM/EDB and no multi language fields for classification attributes c_letter and Class, database replication is axctivated) cla_ctl_with_multi_lang.xml(axalant 2000 or higher ,c_letter_c_class defined as multi language fields) cla_ctl_with_multi_lang_repl.xml (axalant 2000 or higher ,c_letter_c_class defined as multi language fields; databse replication is activated)
Database Language	Language for the database dump. This influences the migration of the classification data. Values: German, English Default: German
Level	Status, that is set during classification upgrade for records in the tables t_cla_dat (pool attributes), t_group_dat(classes)
Replication server	Null or a valid name of the database server

	should be used of an implemented database replication to the environment be migrated

Chapter 6

Configure control and log files

A set of control and log files is defined for each upgrade step. The location of the files is stored in the main configuration file upgrade\conf\ApplicationParameter.xml.

Control and log files for comparing/updating repository tables

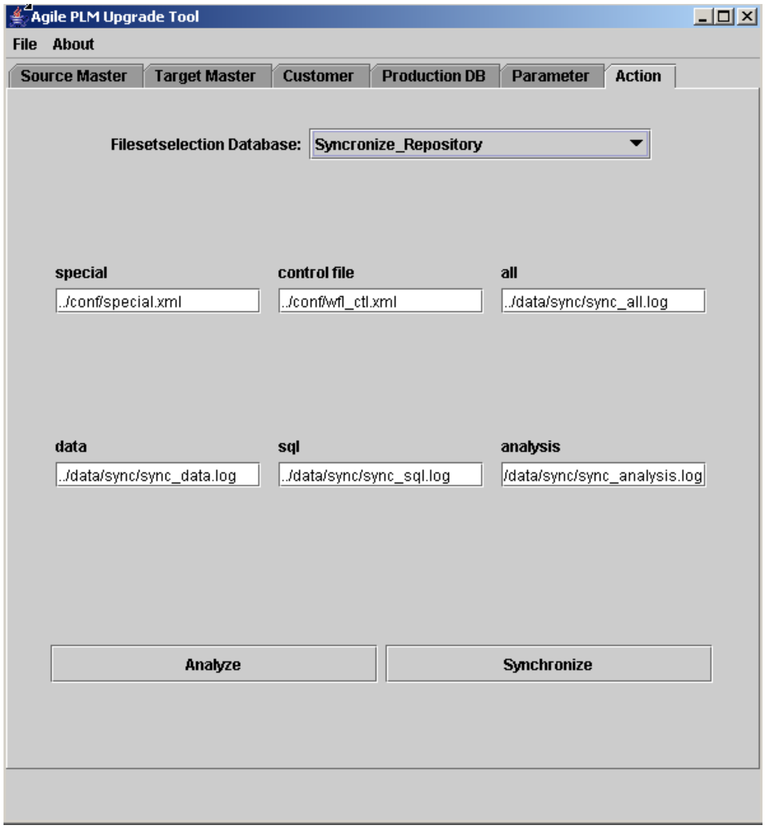
For all actions (see **Action** tab) that compare table content (Create files) and change repository information (Perform Insert, Update, Delete) a common set of control and log files is used.

Parameter Name	Description	Example
Datadictionary	Description of data model, defining the objects and their relationships for upgrade.(all Modules)	data\dtv\dtvDD.xml
Special.xml	Contains information about a table that must be renamed. For DTV- Upgrade only (DTV Upgrade only)	Conf\special.xml
Deletefile	Deleted records in xml format.(all Modules)	Data\dtv\dtv_del.xml
Insertfile	New records in xml format.(all Modules)	Data\dtv\dtv_ins.xml
Updatefile	Updated records in xml format.(all Modules)	Data\dtv\dtv_upd.xml
Errorlog	Errors(all Modules)	Data\dtv\dtv_err.xml
Infolog	Information(all modules)	Data\dtv\dtv_info.xml
Customizing	All conflicts for specific columns, which can be not handled by the upgrade tool automatically are written to customizing.log (DTV-Upgrade only)	Data\dtv\customizing.log
EdbID Replace	Generated during dtv upgrade, define the updates for the new foreign key references (based on new added EDB_ID's) (DTV-upgrade ans synchronize Repository)	Data\dtv\edb_id_replace.xml\

Note: If necessary update the name and the location of the control and logfiles for the actions.

Synchronize Repository

This step uses a set of specific control and log files. If the step is executed again, the log files will be saved and then overwritten. The saved log files extended with a consecutive number for every version (e.g. sync_all001.log — sync_all002.log — sync_all001.log)



Filesetselection Database: Syncronize_Repository

special: ../conf/special.xml

control file: ../conf/wfl_ctl.xml

all: ../data/sync/sync_all.log

data: ../data/sync/sync_data.log

sql: ../data/sync/sync_sql.log

analysis: /data/sync/sync_analysis.log

Parameter Name	Description	Example
sql	All SQL statements for creating and altering database objects are logged. This file is created in step “synchronize repository”.	upgrade\data\sync\sync_sql.lo g
all	Store all log information. This file is created during step “synchronize repository”.	upgrade\data\sync\sync_all.lo g
data	Table definition. This file is created during step “synchronize repository".	upgrade\data\sync\sync_data.l og
special	XML file containing special definitions for repository upgrade like move of fields. Default values for the new mandatory columns.	upgrade\conf\special.xml
analysis	Analysis.log store inconsistencies between DataView table definition and physical tables. This file is created in the step”analyze repository”	upgrade\data\sync\sync_analy sis
Control file	Not used for upgrade to Agile e6.0	

Note: If necessary update the name and the location of the control and logfiles for the actions

Configure special.xml for synchronizing repository

In the step Synchronize Repository, the following functionality is executed:

- Set static and dynamic default values for new mandatory columns or columns changed from null to **not null**
- Rename tables
- Move fields
- Change data type of a field

These functions need a specific configuration. This information is stored in upgrade\conf\special.xml. The delivery contains a preconfigured special.xml which define standard setting for all expected cases.

Very often the customer dump contains inconsistency, so that in the analyze mode the tool will add entries to the special.xml file. You then need to reviewed and adapted the configuration.

If the special.xml is damaged, copy a original special.xml file from the template directory (upgrade\conf\template) into the directory Upgrade\Conf and start the synchronization again.

Set Static and dynamic default values

Static values

To set a static value the configuration looks like

```
<FieldDefault>
<FieldName>T_TRE_DAT.CUR_FLAG</FieldName>
<FieldType>S</FieldType>
<FieldSize>1</FieldSize>
<DefaultValue>
<Value>n</Value>
</DefaultValue>
</FieldDefault>
```

In this case the column T_TRE_DAT_CUR_FLAG is set to ‘n’.

Dynamic values

The field values can be computed dynamically based on

- a Java function
  preconfigured function to use the number server to set values are available (see example)
- SQL Statement

Example:

sql statement is used to compute field value

```
<FieldDefault>
<FieldName>T_CTX_DAT.EDB_SEQ</FieldName>
<FieldType>I</FieldType>
<FieldSize>4</FieldSize>
```

```
<DefaultValue>
<Select>DISTINCT (SELECT COUNT(*) FROM T_CTX_DAT T WHERE T.C_ID &lt;=
thisRec.C_ID)*10</Select>
<Where>C_ID &gt; 0</Where>
</DefaultValue>
</FieldDefault>
```

Example

Java function is used to compute field values (get a new number from the number server and fill in the value)

```
<FieldDefault>
<FieldName>T_MASTER_DOC.EDB_ID</FieldName>
<FieldType>I</FieldType>
<FieldSize>10</FieldSize>
  <DefaultValue>
      <Function>GetNewEDBID(EDBEDBID)</Function>
  </DefaultValue>
</FieldDefault>
```

Rename tables

Specifies the old and the new name for tables.

If you have used DFM already, the following tables must be renamed (for upgrade from cadim to Agile 6 only).

T_EER_SIT

T_EER_SIT_STR

T_EER_SIT_MED

```
<RenameTable>
 <TableName>T_EER_SIT</TableName>
 <NewTableName>T_DDM_SIT</NewTableName>
</RenameTable>

<RenameTable>
 <TableName>T_EER_SIT_STR</TableName>
 <NewTableName>T_DDM_SIT_STR</NewTableName>
</RenameTable>
```

Move fields

This option allows to move a column of a table inclusive stored values to a new location. To move a field you have to specify:

- Source field (<table_name>.<column_name>)
- Target field (<table_name>.<column_name>)and
- Path (join condition between old and new table)

The sample configuration files show 3 different possibilities to move field values to a new location.

```
  <!-- Example transfer from typetable to entitytable. -->
   <MoveField>
    <SourceField>T_DOC_DRW.CRE_USER</SourceField>
    <Path>T_DOC_DRW.C_ID_2</Path>
    <Path>T_DOC_DAT.C_ID</Path>
    <DestField>T_DOC_DAT.CAX_CRE_SYSTEM</DestField>
   </MoveField>
  <!-- Example transfer from entitytable to entitytable via 
releationtable. -->
   <MoveField>
    <SourceField>T_MASTER_DAT.PART_ID</SourceField>
    <Path>T_MASTER_DAT.C_ID</Path>
    <Path>T_MASTER_DOC.C_ID_1</Path>
    <Path>T_MASTER_DOC.C_ID_2</Path>
    <Path>T_DOC_DAT.C_ID</Path>
    <DestField>T_DOC_DAT.CAX_CRE_SYSTEM</DestField>
   </MoveField>
  <!-- Example transfer in table. -->
   <MoveField>
    <SourceField>T_MASTER_DAT.PART_ID</SourceField>
    <DestField>T_MASTER_DAT.EDB_ICON</DestField>
   </MoveField>
</SpecialCases>
```

If you have a cax interface installed already, please check if one of the columns used to store cax specific information is defined as a document-type-table-column. These columns are now part of the standard axalant data-model and included in the document master table T_DOC_DAT.(migration from CADIM to Agile e6.0)

An example configuration file special_move.xml containing definition of moved fields are stored in the template directory …\upgrade\conf\template.

Change data type of a field

The upgrade tool allows changing the type definition of columns. The following type changes are possible:

- Integer→String

If the value of a column for all records is null then also incompatible data type changes can be executed, for example STRING-->Integer

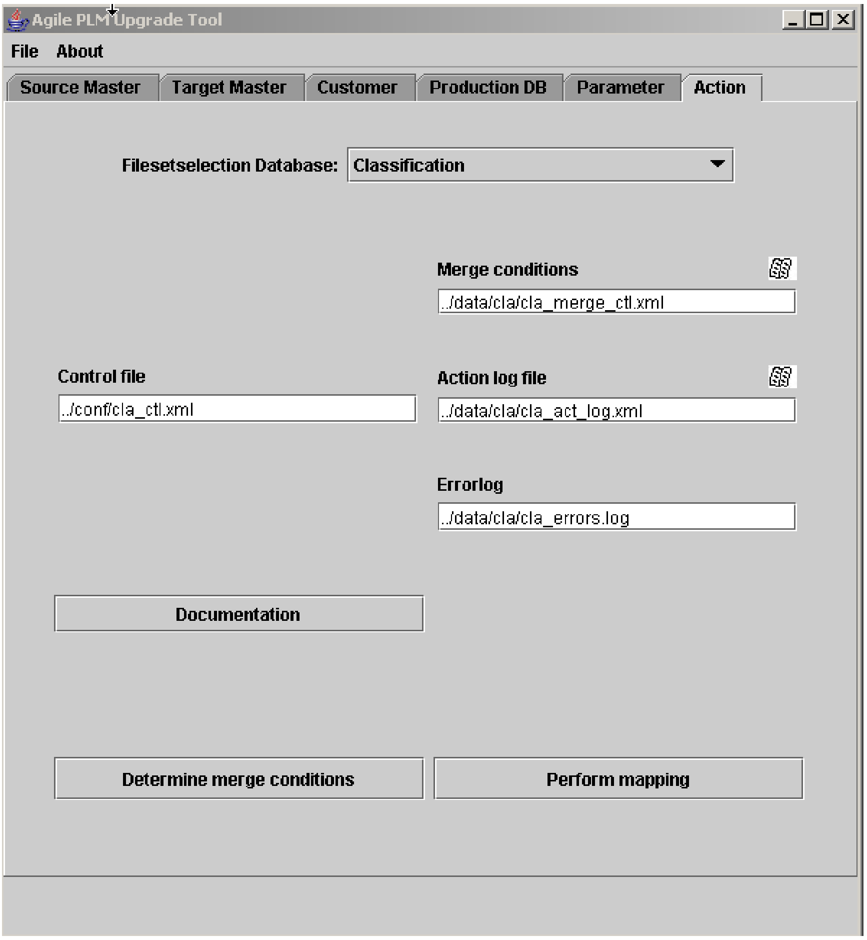
Cutting a string field is only possible if no record contains a longer value. Please check max length of stored values directly with SQL.

You have to replace “false” by “true” to confirm the change. The type definition “oldType” comes from the database; “newType” is the DataView definition. (T_FIELD. C_FORMAT)

```
FieldChange>
 <FieldName>T_DOC_DAT.FOO</FieldName>
 <ConfirmChange oldType="S80.0" newType="S40.0">false</ConfirmChange>
</FieldChange>
```

Classification

The following control and log files are used



Filesetselection Database: Classification

Merge conditions
../data/cla/cla_merge_ctl.xml

Control file
../conf/cla_ctl.xml

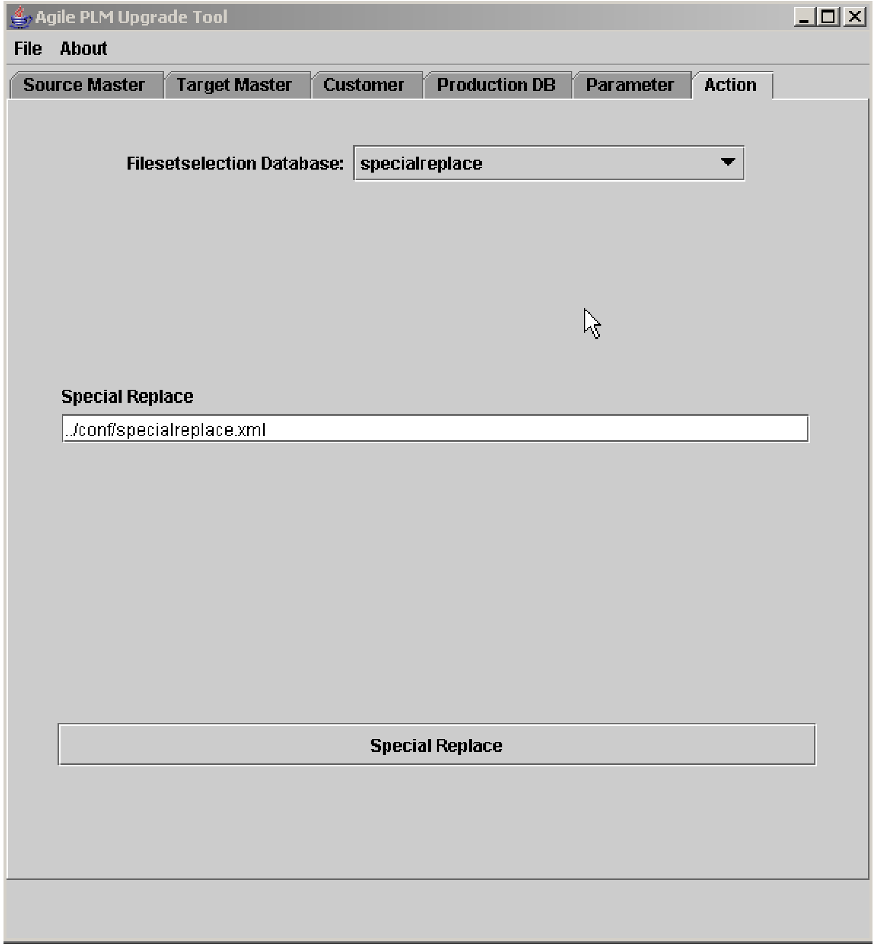
Action log file
../data/cla/cla_act_log.xml

Errorlog
../data/cla/cla_errors.log

Parameter Name	Description	example
Control file	cla_ctl.xml defines the rules for the classification upgrade like data field mapping, merge condition, etc.	upgrade\conf\cla_ctl.xml
Merge conditions	The file merge_ctl.xml is created during the step determine merge conditions and stores the new pool attributes and the original class specific attributes.	upgrade\data\cla\cla_merge_ctl.xml
Action log file	All actions are written to the log file act_log.xml.	upgrade\data\cla\cla_act_log.xml
Errorlog	Is not used yet.	upgrade\data\cla\cla_errors.log

Special replace

This configuration files contains definition of substrings in repository columns which should be replace by an other string



Filesetselection Database: specialreplace

Special Replace

../conf/specialreplace.xml

Example

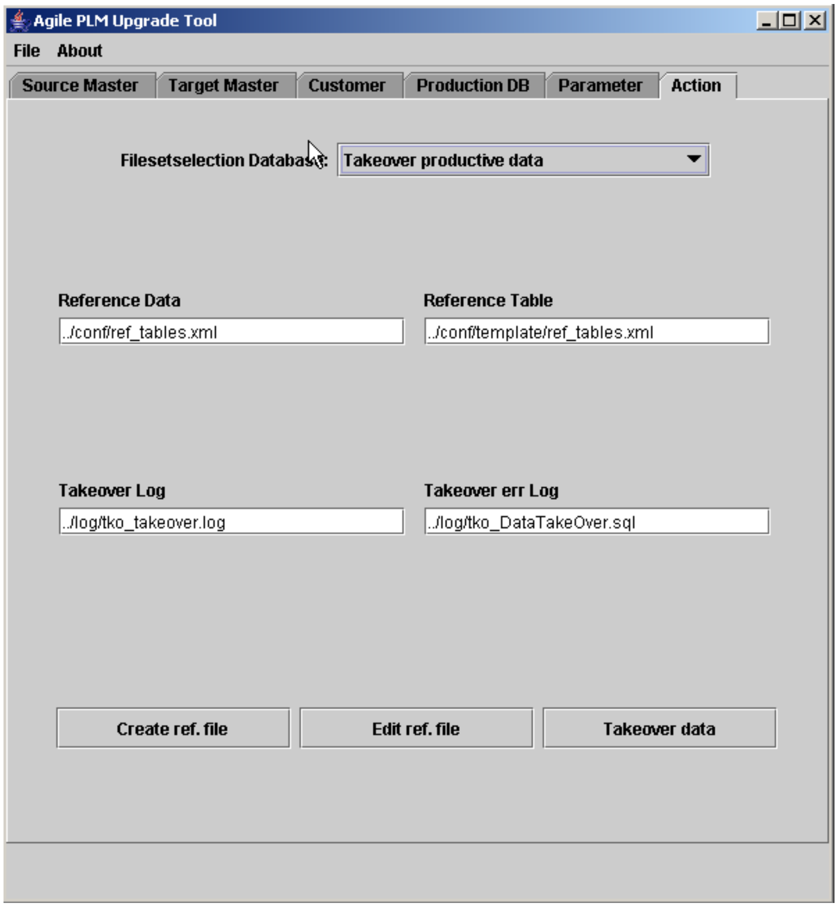
Uupdate the strings T_EER_SIT with T_DDM_SIT in LGC Code

```
<?xml version="1.0" encoding="UTF-8"?>
<special>
      <replace>
            <table>LV_DT_PRC</table>
            <field>MAIN</field>
            <example>T_EER_SIT</example>
            <replacewith>T_DDM_SIT</replacewith>
      </replace>
</special>
```

Chapter 7

Configure take over data from the production system

1. Select the tab Takeover productive data and review the configuration and Log files for this step



Filesetselection Database: Takeover productive data

Reference Data: ../conf/ref_tables.xml

Reference Table: ../conf/template/ref_tables.xml

Takeover Log: ../log/tko_takeover.log

Takeover err Log: ../log/tko_DataTakeOver.sql

Parameter Name	Description	example
Reference Data	Data Ref_data.xml defines which tables are proceeded during the step takeover productive data (s. chapter “Take over of reference data”).	upgrade\conf\ref_tables.xml
Reference tables	Ref_tables.xml contains the default list of reference tables.	upgrade\con\templates\ref_tables.xml
Takeover Log	Information about all executed commands.	upgrade\log\tko_takeover.log

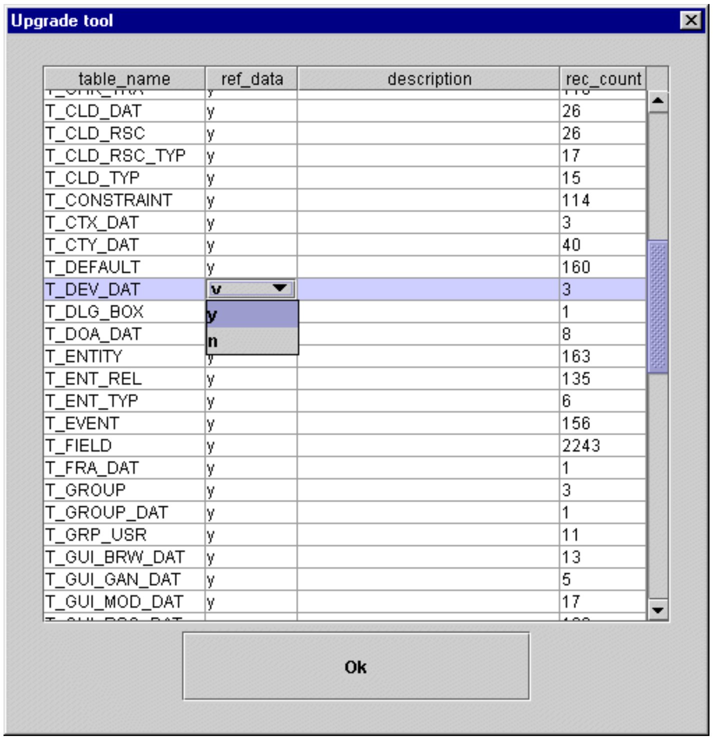
Takeover err Log	Information about occurred errors.	upgrade\log\tko_DataTakeover.sql

Define reference tables

1. Select folder **Takeover** in the Upgrade Tool and press the button **Create Ref File.**

   The Upgrade Tool will connect to the productive database, identify all tables and synchronize the information with the predefined list (ref_tables). Only tables with data will be written to the file.

2. To adapt the list, press **Edit ref. file**



table_name	ref_data	description	rec_count
T_CLD_DAT	y		26
T_CLD_RSC	y		26
T_CLD_RSC_TYP	y		17
T_CLD_TYP	y		15
T_CONSTRAINT	y		114
T_CTX_DAT	y		3
T_CTY_DAT	y		40
T_DEFAULT	y		160
T_DEV_DAT	y		3
T_DLG_BOX			1
T_DOA_DAT			8
T_ENTITY	y		163
T_ENT_REL	y		135
T_ENT_TYP	y		6
T_EVENT	y		156
T_FIELD	y		2243
T_FRA_DAT	y		1
T_GROUP	y		3
T_GROUP_DAT	y		1
T_GRP_USR	y		11
T_GUI_BRW_DAT	y		13
T_GUI_GAN_DAT	y		5
T_GUI_MOD_DAT	y		17

For each table you have to define, if it is a reference table. (ref_data=y.)The called reference tables will be droped in the customer dump and copied in from the production system using the connection to the production DB.

3. Select OK to save the XML file (*upgrade/data/ref_data_tab.xml*).

For detail information which tables should be copied please refer document Overview upgrade process.