



# Migration

Informatica MDM - Product 360

Version: 8.1.1

# 1 Table of Contents


1	Table of Contents .....	2
2	Pre-Migration Checklist .....	5
3	Control Center Migration .....	5
3.1	Updating Control Center .....	5
4	Repository Migration .....	6
4.1	Repository Merger .....	6
4.1.1	Installation and Execution .....	7
4.1.2	Troubleshooting .....	7
	Duplicate Elements .....	7
5	Database Migration .....	8
5.1	Media Manager Database Migration .....	8
5.1.1	Updating the Product 360 - Media Manager database .....	8
5.2	Product 360 Core Database Migration .....	9
5.2.1	Duplicate Article Attribute Names in the repository default language .....	9
	Problem/ Situation .....	9
	Solution 1 .....	10
	Solution 2 .....	10
5.2.2	ArticleTrading entries without existing Party entry are removed .....	10
	Problem/Situation .....	10
5.2.3	Invalid decimal formatting in Article Attribute Values or Structure Group Attribute Values .....	11
	Problem/ Situation .....	11
	Solution .....	11
5.2.4	Structure Migration .....	11
5.2.5	While migrating to 8.0.03 .....	12
5.2.6	Qualification permission info entries in the log .....	12
5.2.7	SQL scripts to check and migrate invalid decimal formatting in Article Attribute Values .....	13
5.2.8	SQL scripts to find and correct duplicate attribute names .....	15
6	Server and Desktop Migration .....	18
6.1	Installation of Hotfix, EBF or One-Off-EBF (=Patch) on Server .....	18

6.2	Installation of Hotfix, EBF or One-Off-EBF (=Patch) on Desktop .....	19
6.3	Structure Migration .....	19
6.3.1	Run Structure Migration Tool .....	19
6.4	XML parsing exception while loading export templates or other data graph objects.....	20
6.5	Entity IDs for ArticleLangType and ArticleExtensionType repository custom entities .....	21
6.5.1	Database .....	21
6.5.2	Repository .....	23
6.6	Server Jobs.....	23
6.7	Data Quality .....	23
6.7.1	Migration .....	23
6.7.2	Update .....	24
6.8	Server and Desktop Pre-Migration Checklist .....	24
6.8.1	Structure Migration Preparation .....	24
	Download Structure Migration Preparation Tool .....	25
	Install Structure Migration Preparation tool.....	25
	Run Structure Migration Preparation Tool .....	25
	Correct invalid item/variant/product mappings .....	26
<b>7</b>	<b>Media Manager Migration .....</b>	<b>26</b>
7.1	Update Media Manager .....	26
7.1.1	Updating Media Manager File Server .....	26
7.1.2	Updating the client modules .....	26
	Manually updating the client modules (Windows) .....	27
	Manually updating the client modules (OSX).....	27
7.1.3	Updating Funcd .....	27
7.1.4	Updating web front end .....	27
7.1.5	Deactivate old modules.....	28
	Deactivate Session Manager .....	28
	Deactivate Internet Funcd .....	28
<b>8</b>	<b>Supplier Portal Migration.....</b>	<b>28</b>
8.1	On the Product 360 Supplier Portal Server.....	28
<b>9</b>	<b>Web Search Migration .....</b>	<b>29</b>
9.1	On the WebSearch Server.....	29

9.2	On the Client .....	29
9.2.1	New properties in configuration.properties file: .....	29
<b>10</b>	<b>Business Process Management .....</b>	<b>31</b>
10.1	Workflows for Product 360 versions >= 8.0.00.....	31
10.2	Workflows for Product 360 versions < 8.0.00.....	31
<b>11</b>	<b>How to apply a Hotfix or EBF .....</b>	<b>31</b>
11.1	Preconditions. ....	31
11.2	Installation of Hotfix or EBF on Product 360 Server .....	32
11.3	Installation of Hotfix or EBF on Product 360 Desktop.....	32


Technical documentation about migration an installation to a new release can be found [here](#).

## 2 Pre-Migration Checklist

 **Always backup your system before updating! This especially includes the databases, file storage areas, configuration files and binaries distributables.**

Before beginning to migrate your Product 360 system, please check that:

- Your system meets the System Requirements

 Note: Please find the System Requirements in the corresponding version of PAM in MySupport portal.

- You must be able to use a command prompt to continue. If not, please contact your system administrator to assist.
- You have read the Release Notes of the version to be installed.

## 3 Control Center Migration

### 3.1 Updating Control Center

With each Hotfix and Major P360 release, there is also an update for the Control Center.

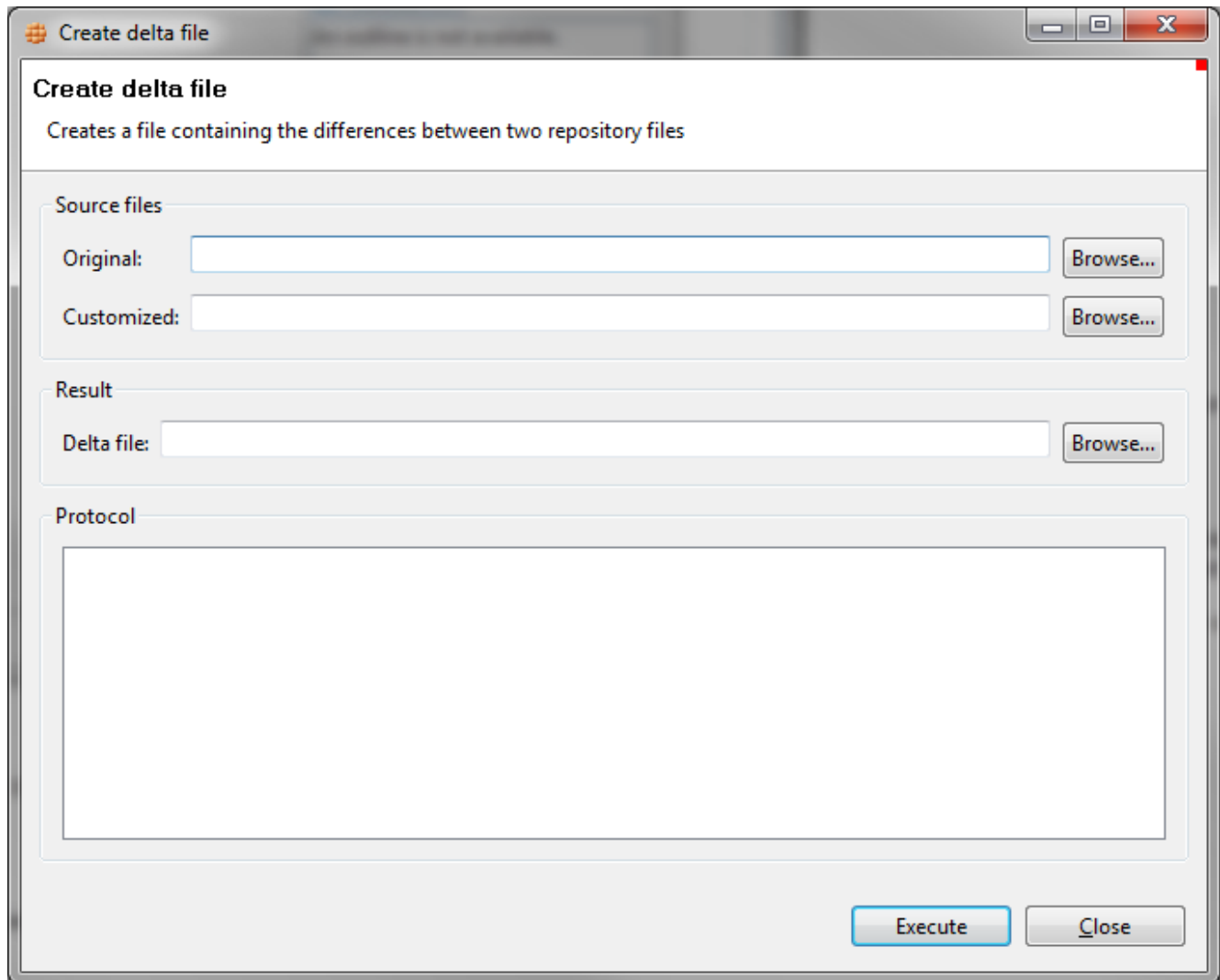
For updating the P360 Control Center follow the steps below.

1. Stop the Control Center service if still running.
2. Uninstall the service by executing the `uninstall.cmd` (Windows) or `uninstall.sh` (Linux)
3. Rename the root folder of your current running Control Center in something like `clusterix_old`.
4. Extract the new Control Center package in the same directory as the old one.
5. If you want to encrypt the passwords using standard implementation of encryption you should replace the Java JCE policy files in `jre\lib\security` folder. For more information please refer to chapter Encryption of secure information in the Server Installation manual.
6. If you made any adjustments to configuration files like `environment.conf` or `eclipse.ini` don't forget to adjust these files also. Be careful if you just copy these files from your old installation, since there might have been made some changes. It is safer to copy only your settings instead the whole file.
7. Inside the `configuration` folder re-configure the property files with your settings. It is recommended to **not** copy the old property files on the new ones since there might be changes or new properties. This also applies to property files such as `wrapper.conf` on which usually no modification is needed for an installation, but which might have changed from the standard product.
  - `server.properties`
  - `plugin_customization.ini`
  - network configurations
  - audit trail and `hmm / hsx` properties if needed.
8. If you had changes in Repository please apply them also to the Repository provided by the update package. See [Repository Migration](#) for more information.
9. Install the service again by executing `install.cmd` (Windows) or `install.sh` (Linux)
10. Start the Control Center 360 service again

## 4 Repository Migration

### 4.1 Repository Merger

The repository merger reduces the effort to migrate a customized repository to a new standard Product 360 release by automating most of that work. It is possible to create a delta report based on an original repository file and the customized repository file. The delta creation should always be performed with the **newest** merger for which you want to merge the Repository.



**Create delta file**

Creates a file containing the differences between two repository files

**Source files**

Original:

Customized:

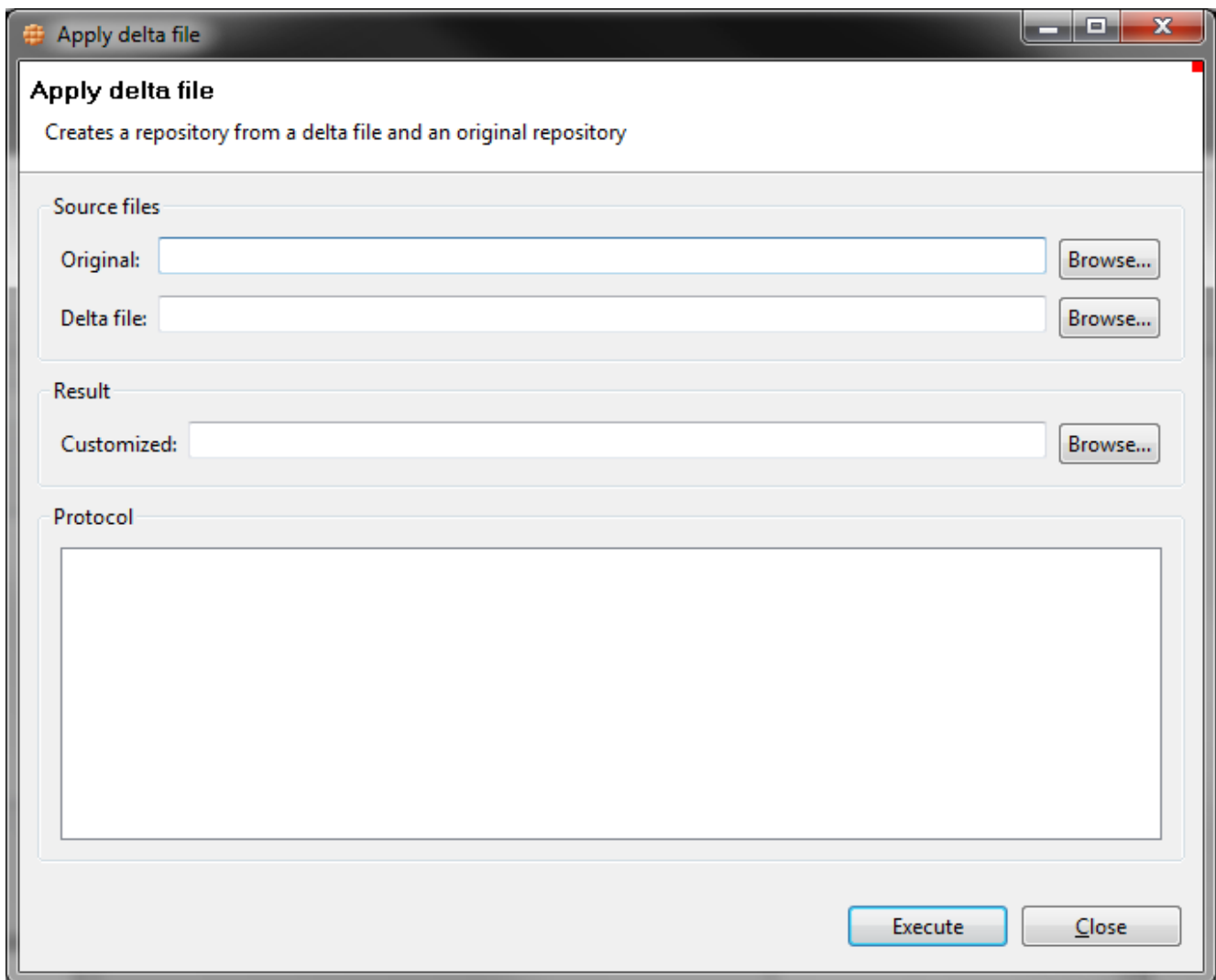
**Result**

Delta file:

**Protocol**

Dialog to create the delta file

Once you have the delta file, you can use the merger from the "target" Product Manager release to re-integrate the delta into the target's standard repository file.



Screen to apply previously created delta file

The result of the integration will be a CSV protocol file which can easily viewed in Excel or something similar. The protocol will only contain conflicts like "repository object not found" or "attribute value has been changed by customizing and standard". The entries in the protocol need to be checked manually, they can't be resolved automatically.

#### 4.1.1 Installation and Execution

The repository merger is being distributed together with the Repository Editor (starting from version 6.0.01, from version 7 it is called Repository Manager). You can execute it using the provided UI Dialog in the Repository Editor.

#### 4.1.2 Troubleshooting

Normally the repository merge works just fine - but there are some special situations. These and their solutions are described in the following:

##### Duplicate Elements

The repository contains duplicate elements (same element type, same identifier) which is not allowed in any case

This error occurs, if you have repository elements with the same identifier and the log message looks like this:

Log			
Severity	Type	Element	Message
ERROR	FIELD	EnumEntryLang.LanguageID	The repository contains duplicate elements (same element type, same identifier) which is not allowed in any case.

In former versions of Product 360 (a.k.a. HPM 5 and 6), it was allowed to have duplicate identifiers in the repository. Since version 7 this is not allowed anymore.

In the two pictures below you can see the two repository elements, that have the same field identifier.

```
<field identifier="EnumEntryLang.LanguageID" enum-ref="Enum.Language" field-type-ref="EnumEntryLangType.Language">
  <name>%field.EnumEntryLang.LanguageID.name</name>
  <description>%field.EnumEntryLang.LanguageID.description</description>
  <editable>false</editable>
  <visible>true</visible>
  <display-by-default>true</display-by-default>
  <mergeable>false</mergeable>
  <purpose>1</purpose>
  <exportPurpose>5</exportPurpose>
  <value>7</value>
  <documentation></documentation>
</field>

<field identifier="EnumEntryLang.LanguageID" enum-ref="Enum.Language.WithLanguageIndependent" field-type-ref="EnumEntryLangType.Language">
  <name>%field.EnumEntryLang.LanguageID.name</name>
  <description>%field.EnumEntryLang.LanguageID.description</description>
  <editable>false</editable>
  <visible>true</visible>
  <display-by-default>true</display-by-default>
  <mergeable>false</mergeable>
  <purpose>1</purpose>
  <exportPurpose>5</exportPurpose>
  <value>-1</value>
  <documentation></documentation>
</field>
```

## Solution

The solution of this error is quite simple. You have to **make the field identifiers unique** by renaming one of them. Identifiers need to be unique **in all repositories**, that are used for the migration (standard and custom).

That means if the standard repository contains duplicate identifiers too, you have to remove them as well.

*Please note that this issue is more likely to appear in migrations from HPM 5.x / 6.x to any version of Product 360.*

# 5 Database Migration

## 5.1 Media Manager Database Migration

### 5.1.1 Updating the Product 360 - Media Manager database



When you use an Oracle database and you want to update from version 5.4 to 5.5 it's necessary to run the long raw conversion before, otherwise the update program will refuse to commence. The conversion tool is located in the module Administration!



When you use an Oracle database and you want to update from version 5.4 to 5.5 the update program automatically converts every varchar2(3) column to a short integer column. This conversion may take a long time, depending on



the structure of your data. Because of that it is recommended that you plan your update window on a date when you can accept a certain down-time of your system. The procedure for updating an existing Product 360 - Media Manager database is as follows:

To update the Product 360 - Media Manager database, you require the file **PIM\_<Version>\_MediaManager.zip** from your Product 360 distribution.

The procedure for update your Product 360 - Media Manager database is as follows:

1. Uncompress the file **PIM\_<Version>\_MediaManager.zip** on your Windows computer.
2. Navigate to the folder **\\Setup\\win\\update database** of the uncompressed archive.
3. Run the program **HMM\_upd.exe**.
4. Select the file OPAS-G.ini from your local Product 360 - Media Manager installation.
5. Enter the administrator password and confirm by clicking on **OK**.
6. Click on **Start** to update and extend your database tables.


When the process is complete, you will find a log file with the name YYMMDD.TXT – where YY represents the year, MM the month and DD the day – in the update subfolder of your local Product 360 - Media Manager installation.

## 5.2 Product 360 Core Database Migration

All necessary Product 360 - Core Database migration scripts are included in the Product 360 Core Database setup, see Server Database. Normally no additional scripts need to be executed.

Nonetheless there are special situations. These and their solutions are described in the following:

### 5.2.1 Duplicate Article Attribute Names in the repository default language



 This problem only occurs, when migrating from PIM Version < 8.0.03 to PIM Version >= 8.0.03, and if the PIM database originally arises from a PIM Version <= 5.3.

#### Problem/ Situation

In PIM Versions 5.3 and older it was possible, to have duplicate attribute names in the repository default language for the same article in the MASTER and/or SUPPLIER database. In newer PIM versions, this situation does not occur anymore, since it is prevented through business logic. But if the PIM Database arises from an older PIM Version (<=5.3), their might still exist such duplicate attribute names.

With PIM Version 8.0.03 an unique index XAK2\_ArticleAttribute on the new column nameInKeyLanguage together with the deletionTimestamp and the articleRevisionID is introduced in the PIM Database. The creation of this index fails, if duplicate attribute names in the repository language do exist (see above). This situation becomes noticeable through the following issue:

- [for MSSQL] the Database Setup shows one of the following errors for update script PCM\_MASTER\_Upd\_V3.0.5.63 or PCM\_SUPPLIER\_Upd\_V3.0.5.63:
  - Msg 1505, Level 16, State 1, ...  
The CREATE UNIQUE INDEX statement terminated because a duplicate key was found for the object name 'dbo.ArticleAttribute' and the index name 'XAK2\_ArticleAttribute'. The duplicate key value is (attribute1, Dec 31 9999 12:00AM, 1).
  - Msg 50000, Level 18, State 44, ...  
1 duplicate article attribute name(s) in the repository key language (languageID=9) is/are existing!
- [for MSSQL] the Database Setup shows one of the following errors for script 'shared\_PPM4\_Rep\_Copy':

- Invalid column name 'Origin'.
-  **In this special situation it is necessary to rollback to the PIM database backup you have made before you started the PIM Core Database Setup, then first of all execute the steps described under "Solution" below and after that proceed with the migration.** 
- This may also occur if the `repository.default.language` is another than in the old environment. In this case please see solution 2.
- [for Oracle] the Database Setup shows the following error for update script PCM\_MASTER\_Upd\_V3.0.5.68 or PCM\_SUPPLIER\_Upd\_V3.0.5.68:
  - ORA-01452: cannot CREATE UNIQUE INDEX; duplicate keys found
- In order to proactively find and correct duplicate attribute names, please use the corresponding SQL script in the chapter "[SQL scripts to find and correct duplicate attribute names](#)" below.

### Solution 1

The duplicate article attribute names in the repository language have to be corrected in the PIM Database.

1. Therefor execute the SQL script to find duplicate attribute names (see chapter "[SQL scripts to find and correct duplicate attribute names](#)" below) in the MASTER and SUPPLIER database in order to find all duplicate attribute names and the corresponding Item/Variant/Product. Save the results of this SQL script, so you are able to find those Items/Variants/Products later on.
2. Then execute the SQL script to correct duplicate attribute names (see chapter "[SQL scripts to find and correct duplicate attribute names](#)" below) in the MASTER and SUPPLIER database in order to automatically rename all these duplicate attribute names to '<attributeName> (<counter>)', e.g. 'Length (2)'
3. Re execute the Database Setup. It shall now run through without the above mentioned error.
4. Correct the attribute names (that were found in step 1 and renamed in step 2) to your needs through PIM Desktop, PIM Web or Rest API.

If you have problems with the SQL scripts, please contact the Informatica Global Customer Support.


### Solution 2

That kind of issue is also shown if the `repository.default.language` inside the `server.properties` is not set correct.

For example: The old environment was running in `de_DE` and now there is `en_US` set.

Ensure that the `repository.default language` is still the same like in the old environment.

## 5.2.2 ArticleTrading entries without existing Party entry are removed

 This problem only occurs when migrating to a PIM version  $\geq 8.0.00.00$ .

### Problem/Situation

Since PIM 8.0.00.00 an update script exists, which runs every time the Database Setup is executed and (soft-)deletes all ArticleTrading objects, where the corresponding Party is (soft-)deleted (=not existing).

This update script also deletes **ArticleTrading** objects which meet the following conditions:

The ArticleTrading logical key for **ArticleTradingType.LK.PartyMS** is defined with a not editable default value (aka "fixed value"), for example 1001 **and** in the database MAIN.Party table is no record for this ID


(This can actually only happen in case the repository has been misconfigured. to disable a Party logical key one should always use the ID for the Public party.)

## Solution

A record in the MAIN.Party table needs to be created which has the ID which was used in the repository, BEFORE the database setup is executed! It's up to the Partner or Customer DBA to create this record with a simple Insert statement. Please make sure that the ID has the needed value from the repository (in our example: 1001)

This issue is also described in the Chapter "Domain Model (Repository)" of the Product 360 Development Guide, and here under "Logical Key".

### 5.2.3 Invalid decimal formatting in Article Attribute Values or Structure Group Attribute Values

 This problem only occurs, when migrating from Product 360 Version prior to 5.1

#### Problem/ Situation

The attribute value is persisted as a string value in the database. In former Product 360 versions there was no datatype validation and no consistent format for decimal values. Values were persisted as they have been typed in. This means, that decimal values could have been entered (and persisted in the database) in different formats, e. g. "1,2" (with "," as decimal separator) or "1.000,23" (with "." as thousands separator) or 1,000.23 (with "," as thousands separator and "." as decimal separator).

In the current Product 360 Version, decimal values are persisted in a language-independent format, with "." as decimal separator and no thousands separator.

Having old values in the database and interpreting them with a current client leads to validation errors in the Product 360 Desktop Client and in worst cases to wrong interpretation of these values.

#### Solution

The correct solution for this problem depends strongly on the individual customer situation. Because of this it is important in the first step to analyze the article attribute values and structure group attribute values in the customers database and then decide, how to migrate the "invalid" values in the database with a sql script.

Here you can find an example sql script for the following situation:

- there are no article attribute values and structure group attribute values with thousands separator (like "1.000,23")
- in all attribute values with comma (like "1,2"), the comma can always be interpreted as decimal separator

Find the migration sql scripts for this situation in the chapter ["SQL scripts to check and migrate invalid decimal formatting in Article Attribute Values"](#) below.

If you have problems in creating the correct migration sql scripts for your current customer situation, please contact the support department.

### 5.2.4 Structure Migration

First of all, please read the following knowledge base article, in order to understand the impacts of the new structure paradigm: Structure Types

All structures in the Product 360 System are in the first step migrated automatically through the Core Database setup in the following way:

Structure type before migration (id)	Structure type after migration (id)
Catalog structure (1)	Catalog structure (1)
Standard classification system (2)	Standard classification system (2)

Structure type before migration (id)	Structure type after migration (id)
User-defined structure (3)	Output structure (3)
Print structure (4)	Output structure (3)
-	Primary maintenance structure (4)
-	Secondary maintenance structure (5)


After this automatic migration, please refer to the structure migration steps described in chapter [Product 360 Server and Desktop Migration](#).

### 5.2.5 While migrating to 8.0.03

The following database model changes have been made with the hotfix 8.0.03:

Table	Change description
<all tables of versionable root entities (e.g. Article, Structure, StructureGroup, ...)>	The columns <code>AcLID</code> and <code>AcLFlag</code> have been moved from the root table (e.g. Article) to the respective revision table (e.g. ArticleRevision)
ArticleAttribute	The new column <code>NameInKeyLanguage</code> was added.
<all tables with <code>DeletionTimestamp</code> column>	For deleted objects, the <code>DeletionTimestamp</code> is now '9999-12-31 00:00:00' instead of null.
Catalog	The column <code>SupplierID</code> has been moved to the <code>CatalogRevision</code> table.

### 5.2.6 Qualification permission info entries in the log

 This problem can only occur when migrating Product 360 Version < 8.0.05 to Product Version >= 8.0.05 and having GDSN deactivated

With Product 360 version 8.0.05 the "PublicationStatusEntry" entity will be deactivated if GDSN is not enabled in the `application_modules.properties`. If you have maintained any qualified field rights for a user group in a version previous 8.0.05 and you are not having GDSN activated, then you might have a similar entry in the log:

```
INFO | jvm 1 | 2017/07/19 14:06:26 | 14:06:25,993 WARN [Worker 2] [QualificationPermissionElementImpl]
INFO | jvm 1 | 2017/07/19 14:06:26 | No enum provider has been found for any logical key with the qualification
permission identifier "gdsnMessageType".
INFO | jvm 1 | 2017/07/19 14:06:26 | May be this identifier is not anymore used for any logical key in the repository
INFO | jvm 1 | 2017/07/19 14:06:26 | or no enumeration is defined for the logical keys that use this identifier.
INFO | jvm 1 | 2017/07/19 14:06:26 | The permissions for this identifier will be reset when the permissions are changed
the next time.
```

You can check with following SQL statements which user groups are affected:

#### Main database - Oracle

```
SELECT * FROM "UserGroup" WHERE "QualiPermModel" LIKE '%identifier="gdsnMessageType"%';
```

#### Main database - MSSQL

```
SELECT * FROM [dbo].[UserGroup] WHERE [QualiPermModel] LIKE '%identifier="gdsnMessageType"%';
```

### Solution:

There are two ways to solve this problem. The first one is by using the desktop client. As soon as you change any qualified field right for a structure group, the invalid entry "gdsnMessageType" will be removed. The second approach is to change the saved data directly on the database. Please use following SQL statements:

#### Main database - Oracle

```
UPDATE "UserGroup"
SET "QualiPermModel" = REPLACE("QualiPermModel", '<element sys-id="1" sys-type="dp" sys-
class="com.heiler.ppm.std.core.internal.permission.QualificationPermissionElementImpl" identifier="gdsnMessageType" key-
class="java.lang.String"/>', '')
WHERE ID IN ( SELECT ID FROM "UserGroup" WHERE "QualiPermModel" LIKE '%identifier="gdsnMessageType" key-
class="java.lang.String"/>%');
COMMIT;
```

#### Main database - MSSQL

```
UPDATE [dbo].[UserGroup]
SET [QualiPermModel] = CAST(REPLACE(CAST([QualiPermModel] AS VARCHAR(MAX)), '<element sys-id="1" sys-type="dp" sys-
class="com.heiler.ppm.std.core.internal.permission.QualificationPermissionElementImpl" identifier="gdsnMessageType" key-
class="java.lang.String"/>', '') AS NTEXT)
WHERE ID IN (SELECT ID FROM [dbo].[UserGroup] WHERE [QualiPermModel] LIKE '%identifier="gdsnMessageType" key-
class="java.lang.String"/>%');
```

## 5.2.7 SQL scripts to check and migrate invalid decimal formatting in Article Attribute Values

#### for MASTER and SUPPLIER database

```
-- This script can be used after migration of a pim database older than version 5 in order to correct old formatted article
attribute values.
--
-- It searches for invalid formatted decimal values in ArticleAttributeValue.Value and ArticleAttributeValue.ValueMax
-- which have only one comma and only digits in it, and replaces the comma with a dot
-- f.e.:
--   - "1,2" becomes "1.2"
--   - "10000,23" becomes "10000.23"
--
-- Limitations:
--   Old values with thousands separator (f.e. "1.000,2") are not migrated. Also values with characters other than digits
--   and "," (f.e. "1,2 mm") are not migrated.
-- Execute the following statements once on the MASTER database and once on the SUPPLIER database.
-- 1. search for invalid values (here you have the ability to recheck, if these values should be migrated)
SELECT ar.Identifier, aav.ID, aav.Value, aav.ValueMax
```

```

FROM ArticleRevision ar
INNER JOIN ArticleAttribute aa ON aa.ArticleRevisionID = ar.ID
INNER JOIN ArticleAttributeValue aav ON aa.ID = aav.ArticleAttributeID
WHERE aa.DataTypeID in (2, 3, 7) AND
      ((len(aav.Value) - len(replace(aav.Value, ',', ''))) = 1 AND aav.Value not like '%[^0-9,]%') OR
      ((len(aav.ValueMax) - len(replace(aav.ValueMax, ',', ''))) = 1 AND aav.ValueMax not like '%[^0-9,]%'))
-- 2. correct invalid values in ArticleAttributeValue.Value
UPDATE ArticleAttributeValue set Value = replace (Value, ',', '.')
WHERE ID in (
SELECT aav.ID
FROM ArticleAttribute aa
INNER JOIN ArticleAttributeValue aav ON aa.ID = aav.ArticleAttributeID
WHERE aa.DataTypeID in (2, 3, 7)
AND (len(aav.Value) - len(replace(aav.Value, ',', ''))) = 1
AND aav.Value not like '%[^0-9,]%'
)
-- 2. correct invalid values in ArticleAttributeValue.ValueMax
UPDATE ArticleAttributeValue set ValueMax = replace (ValueMax, ',', '.')
WHERE ID in (
SELECT aav.ID
FROM ArticleAttribute aa
INNER JOIN ArticleAttributeValue aav ON aa.ID = aav.ArticleAttributeID
WHERE aa.DataTypeID in (2, 3, 7)
AND (len(aav.ValueMax) - len(replace(aav.ValueMax, ',', ''))) = 1
AND aav.ValueMax not like '%[^0-9,]%'
)

```

#### for MAIN database

```

-- This script can be used after migration of a pim database older than version 5 in order to correct old formatted
structure group attribute values.
--
-- It searches for invalid formatted decimal values in StructureGroupAttributeVal.Value and
StructureGroupAttributeVal.ValueMax
-- which have only one comma and only digits in it, and replaces the comma with a dot
-- f.e.:
-- - "1,2" becomes "1.2"
-- - "10000,23" becomes "10000.23"
--
-- Limitations:
-- Old values with thousands separator (f.e. "1.000,2") are not migrated. Also values with characters other than digits
and ",", (f.e. "1,2 mm") are not migrated.
-- Execute the following statements on the MAIN database.
-- 1. search for invalid values (here you have the ability to recheck, if these values should be migrated)
SELECT sgav.Identifier, sgav.ID, sgav.Value, sgav.ValueMax
FROM StructureGroupAttribute sga
INNER JOIN StructureGroupAttributeVal sgav ON sgav.StructureGroupAttributeID = sga.ID
WHERE DataTypeID in (2, 3, 7) AND
      ((len(sgav.Value) - len(replace(sgav.Value, ',', ''))) = 1 AND sgav.Value not like '%[^0-9,]%') OR
      ((len(sgav.ValueMax) - len(replace(sgav.ValueMax, ',', ''))) = 1 AND sgav.ValueMax not like '%[^0-9,]%'))
-- 2. correct invalid values in StructureGroupAttributeVal.Value
UPDATE StructureGroupAttributeVal set Value = replace (Value, ',', '.')
WHERE ID in (
SELECT sgav.ID
FROM StructureGroupAttribute sga
INNER JOIN StructureGroupAttributeVal sgav ON sgav.StructureGroupAttributeID = sga.ID
WHERE DataTypeID in (2, 3, 7)

```

```

AND (len(sgav.Value) - len(replace(sgav.Value, ',', ''))) = 1
AND sgav.Value not like '%[^0-9,%]'
)
-- 2. correct invalid values in StructureGroupAttributeVal.ValueMax
UPDATE StructureGroupAttributeVal set ValueMax = replace (ValueMax, ',', '.')
WHERE ID in (
SELECT sgav.ID
FROM StructureGroupAttribute sga
INNER JOIN StructureGroupAttributeVal sgav ON sgav.StructureGroupAttributeID = sga.ID
WHERE DataTypeID in (2, 3, 7)
AND (len(sgav.ValueMax) - len(replace(sgav.ValueMax, ',', ''))) = 1
AND sgav.ValueMax not like '%[^0-9,%]'
)

```

## 5.2.8 SQL scripts to find and correct duplicate attribute names

### MSSQL - find duplicate attribute names

```

-- find all duplicate attributes in the key language
DECLARE @KeyLanguageID BIGINT
SET @KeyLanguageID = 9 -- TODO: change to the repository key language ID (9 = en_US, 7 = de_DE)
SELECT AR.[ArticleID] AS [ArticleID], AR.[Identifier] AS [ItemNo], AR.[CatalogID], AR.EntityID AS [Article.EntityID], AA.[ArticleRevisionID], AAL.ID AS [ArticleAttributeLangID], AAL.[Name]
FROM [ArticleRevision] AR
INNER JOIN [ArticleAttribute] AA
ON AR.[ID] = AA.[ArticleRevisionID]
AND AR.[DeletionTimestamp] = AA.[DeletionTimestamp]
INNER JOIN [ArticleAttributeLang] AAL
ON AA.[ID] = AAL.[ArticleAttributeID]
AND AA.[DeletionTimestamp] = AAL.[DeletionTimestamp]
INNER JOIN
(
SELECT DISTINCT XAA.[ArticleRevisionID] AS [ArticleID], XAAL.[Name] AS [Name], MAX(XAA.[ID]) AS [MaxArticleAttributeID]
FROM [ArticleAttributeLang] XAAL
INNER JOIN [ArticleAttribute] XAA
ON XAA.[ID] = XAAL.[ArticleAttributeID]
AND XAA.[DeletionTimestamp] = XAAL.[DeletionTimestamp]
AND XAAL.[LanguageID] = @KeyLanguageID
WHERE XAAL.[LanguageID] = @KeyLanguageID
GROUP BY XAA.[ArticleRevisionID], XAAL.[Name]
HAVING COUNT(*) > 1
) x
ON x.[ArticleID] = AA.[ArticleRevisionID]
AND x.[Name] = AAL.[Name]
AND x.[MaxArticleAttributeID] <> AA.ID
WHERE AAL.[LanguageID] = @KeyLanguageID
ORDER BY AR.[CatalogID], AR.[Identifier], AAL.[Name];
GO

```

### Oracle - find duplicate attribute names

```

-- find all duplicate attributes in the key language
SELECT AR."ArticleID" AS "ArticleID", AR."Identifier" AS "ItemNo", AR."CatalogID", AR."EntityID" AS "Article.EntityID",
AA."ArticleRevisionID", AAL.ID AS "ArticleAttributeLangID", AAL."Name"

```

```

FROM "ArticleRevision" AR
INNER JOIN "ArticleAttribute" AA
  ON AR."ID" = AA."ArticleRevisionID"
  AND DECODE( AR."DeletionTimestamp", AA."DeletionTimestamp", 1, 0 ) = 1
INNER JOIN "ArticleAttributeLang" AAL
  ON AA."ID" = AAL."ArticleAttributeID"
  AND DECODE( AA."DeletionTimestamp", AAL."DeletionTimestamp", 1, 0 ) = 1
INNER JOIN
(
  SELECT DISTINCT XAA."ArticleRevisionID" AS "ArticleID", XAAL."Name" AS "Name", MAX(XAA."ID") AS "MaxArticleAttributeID"
  FROM "ArticleAttributeLang" XAAL
  INNER JOIN "ArticleAttribute" XAA
    ON XAA."ID" = XAAL."ArticleAttributeID"
    AND DECODE( XAA."DeletionTimestamp", XAAL."DeletionTimestamp", 1, 0 ) = 1
    AND XAAL."LanguageID" = 9 -- TODO: change to the repository key language (9 = en_US, 7 = de_DE)
  WHERE XAAL."LanguageID" = 9 -- TODO: change to the repository key language (9 = en_US, 7 = de_DE)
  GROUP BY XAA."ArticleRevisionID", XAAL."Name"
  HAVING COUNT(*) > 1
) x
ON x."ArticleID" = AA."ArticleRevisionID"
AND x."Name" = AAL."Name"
AND x."MaxArticleAttributeID" <> AA.ID
WHERE AAL."LanguageID" = 9 -- TODO: change to the repository key language (9 = en_US, 7 = de_DE)
ORDER BY AR."CatalogID", AR."Identifier", AAL."Name"

```

#### MSSQL - correct duplicate attribute names

```

-- correct all duplicate attributes in the key language
DECLARE @KeyLanguageID BIGINT
SET @KeyLanguageID = 9 -- TODO: change to the repository key language ID (9 = en_US, 7 = de_DE)
DECLARE REPLACE_CURSOR CURSOR
FOR
  SELECT AAL.ID, AA.ArticleRevisionID
  FROM [ArticleAttribute] AA
  INNER JOIN [ArticleAttributeLang] AAL
    ON AA.[ID] = AAL.[ArticleAttributeID]
    AND AA.[DeletionTimestamp] = AAL.[DeletionTimestamp]
  INNER JOIN
  (
    SELECT DISTINCT XAA.[ArticleRevisionID] AS [ArticleID], XAAL.[Name] AS [Name], MAX(XAA.[ID]) AS
[MaxArticleAttributeID]
    FROM [ArticleAttributeLang] XAAL
    INNER JOIN [ArticleAttribute] XAA
      ON XAA.[ID] = XAAL.[ArticleAttributeID]
      AND XAA.[DeletionTimestamp] = XAAL.[DeletionTimestamp]
      AND XAAL.[LanguageID] = @KeyLanguageID
    WHERE XAAL.[LanguageID] = @KeyLanguageID
    GROUP BY XAA.[ArticleRevisionID], XAAL.[Name]
    HAVING COUNT(*) > 1
  ) x
ON x.[ArticleID] = AA.[ArticleRevisionID]
AND x.[Name] = AAL.[Name]
AND x.[MaxArticleAttributeID] <> AA.ID
WHERE AAL.[LanguageID] = @KeyLanguageID
ORDER BY AA.ArticleRevisionID;

```



```

DECLARE @ID BIGINT, @ArticleRevisionID BIGINT, @LastArticleRevisionID BIGINT, @Counter INT, @NameAddon NVARCHAR(5),
@SQLSCRIPT NVARCHAR(1000);
SET @LastArticleRevisionID = -1
OPEN REPLACE_CURSOR;
FETCH NEXT FROM REPLACE_CURSOR INTO @ID, @ArticleRevisionID
WHILE ( @@FETCH_STATUS = 0 )
BEGIN
    IF @LastArticleRevisionID <> @ArticleRevisionID
    BEGIN
        SET @LastArticleRevisionID = @ArticleRevisionID
        SET @Counter = 1
    END;
    SET @Counter = @Counter + 1
    SET @NameAddon = ' (' + CAST(@Counter AS NVARCHAR(3)) + ')'
    SET @SQLSCRIPT = 'UPDATE dbo.[ArticleAttributeLang] SET [Name] = (SUBSTRING([Name], 1, 244) + ''' + @NameAddon + ''')
WHERE [ID] = ' + CAST(@ID AS NVARCHAR(25))
    PRINT @SQLSCRIPT
    EXEC sys.sp_executesql @SQLSCRIPT
    FETCH NEXT FROM REPLACE_CURSOR INTO @ID, @ArticleRevisionID
END;
CLOSE REPLACE_CURSOR;
DEALLOCATE REPLACE_CURSOR;
GO

```

#### Oracle - correct duplicate attribute names

```

-- correct all duplicate attributes in the key language
SET SERVEROUTPUT ON
DECLARE
    KeyLanguageID          NUMBER := 9; -- TODO: change to the repository key language (9 = en_US, 7 = de_DE)
    ZID                    NUMBER;
    ArticleRevisionID      NUMBER;
    LastArticleRevisionID  NUMBER := -1;
    Counter                NUMBER;
    NameAddon              NVARCHAR2(5);
    SQLSCRIPT              VARCHAR2(1000);

    CURSOR C1 (vKeyLanguageID IN NUMBER) IS SELECT AAL."ID", AA."ArticleRevisionID"
FROM "ArticleAttribute" AA
    INNER JOIN "ArticleAttributeLang" AAL
        ON AA."ID" = AAL."ArticleAttributeID"
        AND DECODE( AA."DeletionTimestamp", AAL."DeletionTimestamp", 1, 0 ) = 1
    INNER JOIN
    (
        SELECT DISTINCT XAA."ArticleRevisionID" AS "ArticleID", XAAL."Name" AS "Name",
MAX(XAA."ID") AS "MaxArticleAttributeID"
FROM "ArticleAttributeLang" XAAL
    INNER JOIN "ArticleAttribute" XAA
        ON XAA."ID" = XAAL."ArticleAttributeID"
        AND DECODE( XAA."DeletionTimestamp", XAAL."DeletionTimestamp", 1, 0 ) = 1
        AND XAAL."LanguageID" = vKeyLanguageID
WHERE XAAL."LanguageID" = vKeyLanguageID
GROUP BY XAA."ArticleRevisionID", XAAL."Name"
HAVING COUNT(*) > 1
    ) X
    ON x."ArticleID" = AA."ArticleRevisionID"
    AND x."Name" = AAL."Name"

```

```

AND x."MaxArticleAttributeID" <> AA.ID
WHERE AAL."LanguageID" = vKeyLanguageID
ORDER BY AA."ArticleRevisionID";

BEGIN
OPEN C1(KeyLanguageID);
LOOP
FETCH C1 INTO ZID, ArticleRevisionID;
EXIT WHEN C1%NOTFOUND;

BEGIN
IF (LastArticleRevisionID <> ArticleRevisionID) THEN
BEGIN
LastArticleRevisionID := ArticleRevisionID;
Counter := 1;
END;
END IF;

Counter := Counter + 1;
NameAddon := ' (' || Counter || ')';

SQLSCRIPT := 'UPDATE "ArticleAttributeLang" SET "Name" = (SUBSTR("Name", 1, 244) || ''' || NameAddon || ''') WHERE
"ID" = ' || ZID;
DBMS_OUTPUT.PUT_LINE(SQLSCRIPT);
EXECUTE IMMEDIATE SQLSCRIPT;
END;
END LOOP;
CLOSE C1;
END;
/

```

## 6 Server and Desktop Migration

### 6.1 Installation of Hotfix, EBF or One-Off-EBF (=Patch) on Server


**i** EBF packages and One-Off-EBF (=Patch) packages contain only delta plugins which have been changed. In order to recognize such packages, the suffix **delta** is contained in its name. Hotfixes and Major releases have the suffix **full** in its name.

Generally this is an easy process. Please consider the following steps in order to migrate your PIM to a newer version.

- Consider pre-migration checklists
- Before applying an Hotfix or Major update, you have to update the Control Center itself. See chapter "Control Center Migration" of this Migration guide.
- Login into the Control Center
- Navigate to the installation tab
- With the upload functionality, upload all packages you want to deploy. Do not rename or extract any of the packages that are delivered.
- If you want to deploy also accelerator plugins or custom plugins, you can also upload them. Custom plugins have to be uploaded separately as jar file.
- Please note that all uploaded files will be deployed and overwrite the existing files. But all files will be backedup in a folder, so you can revert them if something went wrong.
- By clicking the upload button, the update process will start and all configured servers will be updated. All configuration files on all servers will be also replaced with the ones that are located in the Control Center.

- Execute steps in chapter "Product 360 Core Database Migration" by running setup to your existing database connection in order to update existing database.

Generally, you should run your custom JUnit tests in order to ensure that your custom functionality is still working.

 If you want to encrypt the passwords using standard implementation please refer to chapter Encryption of secure information in the Server Installation manual. Updating to newest Hotfix you should replace the Java JCE policy files in `jre\lib\security` folders of all Product 360 components with recommended ones. If you install an EBF, then this step is not needed.

## 6.2 Installation of Hotfix, EBF or One-Off-EBF (=Patch) on Desktop

- Before updating the Desktop you should execute the migration of the Server first.
- For **Hotfix** or **Major** updates:
  - Remove content of PIM Desktop Installation Folder and unzip the content of the new release package
  - Update the configuration files if you have any customizings here. Instead of copying the whole file, rather copy only your changes, since there might be some new settings which then will be overwritten.
- For **EBF's** and **One-Off-EBF's** (=patches):
  - Remove the existing plugins with old revision and then place the plugin with new revision which are part of the EBF resp. One-Off-EBF.

## 6.3 Structure Migration

First of all, please read the following knowledge base article, in order to understand the impacts of the new structure paradigm: Structure Types

The Structure Migration Tool is part of the standard Product 360 8.0 Installation, so you do not have to install anything separately. With the Structure Migration Tool you can migrate structures to maintenance structures. Therefore this tool first of all validates the given structures in means of the new structure paradigm. The validation (and also correction of invalid items) works the same as the Structure Migration Preparation Tool (see [Product 360 Server and Desktop Pre-Migration Checklist](#)). If the structures are valid they are migrated to maintenance structures.

### Versioning

If you are using the Versioning functionality of Product 360, please be aware, that the selected structures will be migrated to maintenance structures only in the working version. All other versions affected by these structures will be closed, because possibly invalid object mappings in versions other than the working version are not correctable.

The following permissions are needed, when using the Structure Migration Tool:

- Interface visibility "Structure systems"
- For validation of structures:
  - Action right "Create item/variant/product assortments" (depending on which data types are used in the Product 360 system)
  - Action right "Delete item/variant/product assortments" (depending on which data types are used in the Product 360 system)
- For migration of structures:
  - Action right "Edit structure systems"
  - Action right "Close versions" (only when using versioning)

### 6.3.1 Run Structure Migration Tool

- Open the view "Structure systems".
- Select all the structures you want to migrate to maintenance structures.

- Right-click and choose "Migrate structures..."
- You have the possibility, to recheck your selection and probably correct it.
- Then click on "Migrate".
- The migration process is scheduled as a background server job.
- You can view the results of this job in the "Process overview" perspective. Here choose the job category "[Data maintenance](#)", the job type "Migrate structures" and then your currently scheduled job.
- In the "Log" view you can see the results of the migration job.
- If all the selected structures can be migrated to maintenance structures, you see a corresponding info and they are migrated.
- If the selected structures can not be migrated, you see a corresponding warning. Please then continue with the steps mentioned in chapter "Correct invalid item/variant/product mappings" on this page: [Product 360 Server and Desktop Pre-Migration Checklist](#)
- After correction of all invalid object mapping, restart the Structure Migration Tool for the same structures as before, by executing again the steps mentioned above.

## 6.4 XML parsing exception while loading export templates or other data graph objects

If you experience unexpected errors during the serialization or deserialization of data graphs after an upgrade or change of your local environment, the following hints might be helpful.

In the past we had problems with the SDK deployment mechanism, specifically with the target platform refreshing. When you define the target platform in the SDK by using the update-site, eclipse is copying all bundles of the update site to an internal cache directory.

Now, when you switch to a different target platform release, eclipse is updating its internal cache **only** in case the bundles have changed. The fact that a bundle has changed is determined by the bundle name and the version incl. build identifier suffix of the version. If none of these changed, the bundle is **not** refreshed in the internal cache which leads to `ClassNotFoundException` as soon as someone wants to access a new class of that bundle.

In most of our cases there is no issue with this since nearly all bundles have a build identifier which changes with every build.

Most of them, but not all. The `com.heiler.ppm.xml` bundle has no build identifier. This bundle contains the XML parser implementations which must be used with Product 360 - those parsers are also used within the Java SDK since we're using the so called "endorsed-directory" mechanism. During launch (either by cmd, by service or by launch config of the SDK) we provide the full path to the `com.heiler.ppm.xml/endorsed` directory to java.

So, in case the directory is not there, or the path is not correct in the launch scenarios, the default XML parser will be used which does currently not work with the used XML serialization of the EMF based data graph objects.

Strange XML parsing exceptions are the symptoms of this problem.

With Product 360 version 7.0.04 we increased the version number of the `com.heiler.ppm.xml` bundle from 5 to 6. So you need to make sure you also update all configuration files (incl. `wrapper.conf`) as well as your launch configs - unless you will also experience those exceptions.

A typical stacktrace of such exception can look like this:

```
Caused by: org.eclipse.emf.ecore.xmi.IllegalValueException: Value ';' is not legal. (file:///D:/informatica/pim/client/all.datagraph,7, 21)
at org.eclipse.emf.ecore.xmi.impl.XMLHandler.setFeatureValue(XMLHandler.java:2648)
at org.eclipse.emf.ecore.xmi.impl.XMLHandler.setAttribValue(XMLHandler.java:2702)
at org.eclipse.emf.ecore.xmi.impl.SAXXMLHandler.handleObjectAttribs(SAXXMLHandler.java:83)
at org.eclipse.emf.ecore.xmi.impl.XMLHandler.createObjectFromFactory(XMLHandler.java:2178)
at
org.eclipse.emf.ecore.sdo.util.DataGraphResourceFactoryImpl$DataGraphResourceImpl$LoadImpl$1.createObjectFromFactory(DataGraphResourceFactoryImpl.java:670)
at org.eclipse.emf.ecore.xmi.impl.XMLHandler.createObjectByType(XMLHandler.java:1316)
```

```

at
org.eclipse.emf.ecore.sdo.util.DataGraphResourceFactoryImpl$DataGraphResourceImpl$LoadImpl$1.handleFeature(DataGraphResource
FactoryImpl.java:554)
at org.eclipse.emf.ecore.xmi.impl.XMLHandler.processElement(XMLHandler.java:1023)
at org.eclipse.emf.ecore.xmi.impl.XMLHandler.startElement(XMLHandler.java:1001)
at org.eclipse.emf.ecore.xmi.impl.XMLHandler.startElement(XMLHandler.java:712)
at com.sun.org.apache.xerces.internal.parsers.AbstractSAXParser.startElement(Unknown Source)
at com.sun.org.apache.xerces.internal.impl.XMLDocumentFragmentScannerImpl.scanStartElement(Unknown Source)
at com.sun.org.apache.xerces.internal.impl.XMLDocumentFragmentScannerImpl$FragmentContentDriver.next(Unknown Source)
at com.sun.org.apache.xerces.internal.impl.XMLDocumentFragmentScannerImpl.scanDocument(Unknown Source)
at com.sun.org.apache.xerces.internal.parsers.XML11Configuration.parse(Unknown Source)
at com.sun.org.apache.xerces.internal.parsers.XML11Configuration.parse(Unknown Source)
at com.sun.org.apache.xerces.internal.parsers.XMLParser.parse(Unknown Source)
at com.sun.org.apache.xerces.internal.parsers.AbstractSAXParser.parse(Unknown Source)
at com.sun.org.apache.xerces.internal.jaxp.SAXParserImpl$JAXPSAXParser.parse(Unknown Source)
at com.sun.org.apache.xerces.internal.jaxp.SAXParserImpl.parse(Unknown Source)
at org.eclipse.emf.ecore.xmi.impl.XMLLoadImpl.load(XMLLoadImpl.java:181)
... 70 more

Caused by: java.lang.NumberFormatException: For input string: ";"
at java.lang.NumberFormatException.forInputString(Unknown Source)
at java.lang.Long.parseLong(Unknown Source)
at java.lang.Long.valueOf(Unknown Source)
at org.eclipse.emf.ecore.impl.EcoreFactoryImpl.createELongObjectFromString(EcoreFactoryImpl.java:958)
at org.eclipse.emf.ecore.impl.EcoreFactoryImpl.createFromString(EcoreFactoryImpl.java:157)
at org.eclipse.emf.ecore.xmi.impl.XMLHelperImpl.createFromString(XMLHelperImpl.java:1613)
at org.eclipse.emf.ecore.xmi.impl.XMLHelperImpl.setValue(XMLHelperImpl.java:1154)
at org.eclipse.emf.ecore.xmi.impl.XMLHandler.setFeatureValue(XMLHandler.java:2643)

```

## 6.5 Entity IDs for ArticleLangType and ArticleExtensionType repository custom entities

With Product 360 7.1.04.00 a new data base column "EntityID" has been introduced for data base tables "ArticleLang" and "ArticleExtension". The content of this column has to be the same as in the corresponding repository logical key for custom entities based on "ArticleLangType" and "ArticleExtensionType" repository entity types.

The "ArticleExtensionType" based standard repository entities cannot be migrated automatically because it's not possible to determine the entity of the data base entries. Those entities got a valid Entity ID but use "0" as corresponding logical key value.

### 6.5.1 Database

All "ArticleLangType" based standard and custom repository entities have been considered in corresponding data base update scripts.

If you have to migrate own custom entities based on "ArticleExtensionType" repository entity type, you have to migrate the corresponding data base entries manually. Please use the following sql scripts.

#### MSSQL - Migrate ArticleExtension entries

```

-- 1st step -----
-- are there entries to be migrated?
SELECT COUNT(ID) AS 'Count of entries to be migrated' FROM ArticleExtension WHERE EntityID = 0

-- 2nd step -----

```

```

-- migrate
DECLARE @ArticleEntityID          varchar(10)
DECLARE @ArticleExtensionEntityID varchar(10)
/* TODO set appropriate value */
SET @ArticleEntityID              = <entity id used for your ArticleType based custom entity>      -- example: 1000 for
"Article" repository entity
/* TODO set appropriate value */
SET @ArticleExtensionEntityID = <entity id used for your ArticleExtensionType based custom entity> -- example: 1050 for
"ArticleExtension.EANUCC" repository entity
EXEC ('UPDATE ae SET EntityID=' + @ArticleExtensionEntityID +
      ' FROM ArticleExtension AS ae ' +
      ' INNER JOIN ArticleRevision AS ar ON ae.ArticleRevisionID = ar.ID ' +
      ' WHERE ar.EntityID = ' + @ArticleEntityID +

      /* only not converted values */
      ' AND ae.EntityID = 0' +

      /* TODO set specific values for logical key columns */
      ' AND ae.ExtensionType IN (''Extension1_a'', ''Extension1_b'')' +
      ' AND ae.BuyerID in (1, 2, 3)' +
      ' AND ae.LanguageID < 0' +
      ' AND ae.Territory IN (''WORLD'', ''US'')')

```

#### ORACLE - Migrate ArticleExtension entries

```

-- 1st step -----
-- are there entries to be migrated?
SELECT COUNT(ID) AS "Entries to be migrated" FROM "ArticleExtension" WHERE "EntityID" = 0;

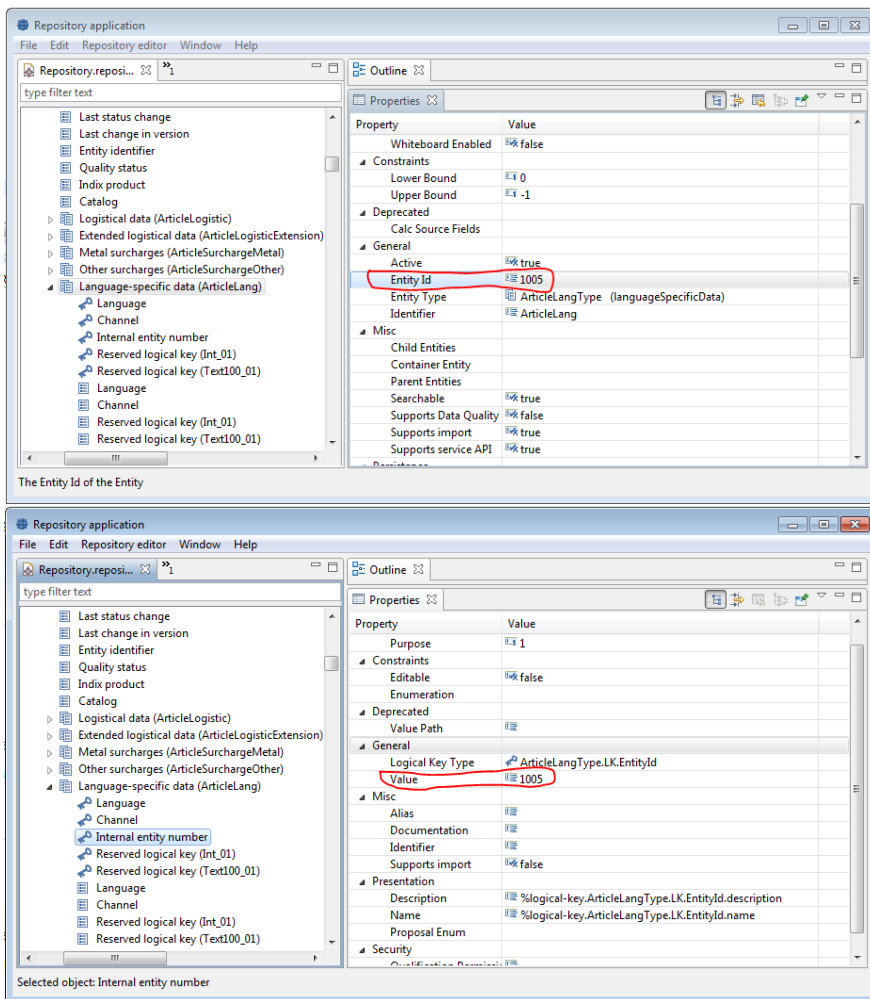
-- 2nd step -----
-- migrate
DECLARE
/* TODO set appropriate value */
ArticleEntityID          NUMBER          :=<entity id used for your ArticleType based custom entity>;      -- example:
1000 for "Article" repository entity
/* TODO set appropriate value */
ArticleExtensionEntityID NUMBER          :=<entity id used for your ArticleExtensionType based custom entity>; -- example:
1050 for "ArticleExtension.EANUCC" repository entity

BEGIN
EXECUTE IMMEDIATE
  'UPDATE (SELECT "ArticleExtension"."EntityID"
            FROM "ArticleExtension"
            INNER JOIN "ArticleRevision"
              ON "ArticleExtension"."ArticleRevisionID" = "ArticleRevision"."ID"
            WHERE "ArticleRevision"."EntityID" = ' || ArticleEntityID ||
            /* only not converted values */
            ' AND "ArticleExtension"."EntityID" = 0 ' ||
            /* TODO set specific values for logical key columns */
            ' AND "ArticleExtension"."ExtensionType" IN (''Extension1_a'', ''Extension1_b'')
            AND "ArticleExtension"."BuyerID" in (1, 2, 3)
            AND "ArticleExtension"."LanguageID" < 0
            AND "ArticleExtension"."Territory" IN (''WORLD'', ''US'') )
            SET "EntityID" = ' || ArticleExtensionEntityID;
END;

```

## 6.5.2 Repository

Ensure all affected custom entities have the corresponding logical key with an appropriate value.



## 6.6 Server Jobs

With version 8.0.5 the preference for defining the number of threads for server jobs in the `quartz.properties` has become obsolete.

If changes have been made to this preference, the number should be set to the new default, which is 3.

```
org.quartz.threadPool.threadCount=3
```

The maximum number of running jobs can now be configured in the `plugin_customization.ini` with the preference:

```
com.heiler.ppm.job.server/maxRunningServerJobs = 40
```

## 6.7 Data Quality

### 6.7.1 Migration

To migrate dq rules, configurations and dictionaries from one system to another,

1. Make a backup of the dataquality shared folder in the target system before migrating the new packages.
2. ensure to copy the shared server folder 'dataquality' to the target system.
3. In order to also have a synced version between custom dictionaries in the database and the files in the dataquality/dictionaries folder, zip the complete dataquality/dictionaries folder.  
Or make a zip file containing only the custom dictionaries (latter is recommended and should be also the way customers/partners build their custom packages).  
This ensures that not only dictionaries from OOB standard are considered, but also customer dictionaries that are used for custom rules mapplets.

## 6.7.2 Update

For a full update, updating of the rule mapplet files, the dictionaries and in case of GDSN predefined rule configurations is needed.

Please refer to the appropriate sections in the New & Noteworthy and release notes for further specific details of a dataquality update.

If there is no specific information in the New & Noteworthy, it is recommended to do a full update using standard steps and default settings during dictionary update described as followed

1. **Make a backup** of the dataquality shared folder before updating the new packages.
2. **Update Rule mapplets:**  
Following rule files must be updated: 'Informatica\_PIM\_Content.xml' and depending of whether GDSN accelerator is used also 'Informatica\_PIM\_GDSN.xml'. If both files are needed these are part of the GDSN accelerator package. There is also a standalone version of the standard rule mapplet file 'Informatica\_PIM\_Content.xml' file in the dqRuleset zip artefact.  
For an update make sure that these the two rule files are deployed into the shared server folder 'dataquality/rules'. Do this by either  
- uploading the new version to the server via upload mapplet feature in the 'Data Quality' Perspective of the Desktop Client (strongly recommended).  
- copying the new version to that folder prior to server start
3. **Update rule configurations:**  
There are only OOB predelivered rule configurations for GDSN Accelerator package. The file 'StandardDataQualityMappingProfile.xml' is located in the GDSN accelerator package.  
For an update make sure to copy that file to the shared server folder 'dataquality/config' prior to server start. Overwrite the file if already existing.
4. **Update dictionaries:**  
Following dictionaries must be updated: Dictionary information contained in 'Informatica\_PIM\_Content.zip' and depending of whether GDSN accelerator is used also 'Informatica\_PIM\_GDSN.zip'. If both files are needed these are part of the GDSN accelerator package. There is also a standalone version of the 'Informatica\_PIM\_Content.zip' file. Customer dictionaries are updated to the target system by providing a zip file containing the complete content (see migration steps above) or only the custom dictionary files (latter is recommended).

## 6.8 Server and Desktop Pre-Migration Checklist

### 6.8.1 Structure Migration Preparation

First of all, please read the following knowledge base article, in order to understand the impacts of the new structure paradigm: Structure Types

It is important, that the following steps are executed **before** the migration to Version 8.0 of the Product 360 system itself. The reason for this is, that with Product 360 8.0 former existing structures are handled as output structures by default, and output structures in Product 360 8.0 do not have the same characteristics and behaviour as structures in former Product 360 Versions (especially concerning the inheritance of structure features to items/variants/products, see also the linked article about the new structure paradigm above).

With the Structure Migration Preparation Tool you can validate structures to be migrated to maintenance structures in Product 360 8.0. Therefor this tool generates assortments for invalid item/variant/product mappings. In the next step any



user can use these assortments for correcting the invalid object mappings (also on those clients without having the Structure Migration Preparation Tool installed).

### Download Structure Migration Preparation Tool

Structure Migration Preparation Tool is available for the newest hotfix release of Product 360 7.1. Find the download packages on the corresponding Hotfix wiki pages. The following artifacts are needed:

- *HPM\_x.x.xx.xx\_Rev-xxxxx\_server\_migration\_structure.zip*
- *HPM\_x.x.xx.xx\_Rev-xxxxx\_client\_migration\_structure.zip*

**i** Please be aware that this tool was tested and released only for the newest hotfix release of Product 360 7.1. You can use this tool also with older Product 360 hotfix releases 7.0 and 7.1, it should be downwards compatible for these major versions. If you run into problems thereby, please contact Product 360 Support. Please also contact Product 360 Support, if you need this tool for Product 360 5.3 or Product 360 6.0. Thanks.

**i** For custom ArticleType-based entities please ensure, that there is also a corresponding AssortmentType-based entity existing.

### Install Structure Migration Preparation tool

The Structure Migration Preparation Tool consists of a Product 360 server feature and a Product 360 Desktop client feature. It is a tool for administrative purpose only. Because of that it is recommended to install the client feature only on those Product 360 client(s), which are used by the users performing the structure migration itself.

- The Product 360 server feature can simply be installed by copying content of the *HPM\_x.x.xx.xx\_Rev-xxxxx\_server\_migration\_structure.zip* package into Product 360 Server folder (into features and plugins folders respectively). After this you have to restart the Product 360 Server.
- The Product 360 client feature can simply be installed by copying content of the *HPM\_x.x.xx.xx\_Rev-xxxxx\_client\_migration\_structure.zip* package into Product 360 Desktop Client folder (into features and plugins folders respectively). After this you have to restart the Product 360 Desktop Client.

The following permissions are needed, when using the Structure Migration Preparation tool:

- Interface visibility "Structure systems"
- Action right "Create item/variant/product assortments" (depending on which data types are used in the Product 360 system)
- Action right "Delete item/variant/product assortments" (depending on which data types are used in the Product 360 system)

### Run Structure Migration Preparation Tool

- Open the view "Structure systems".
- Select all the structures you want to migrate to maintenance structures.
- Right-click and choose "Validate for migration"
- The validation process is scheduled as a background server job.
- You can view the results of this job in the "Process overview" perspective. Here choose the job category "[Data maintenance](#)", the job type "Migrate structures" and then your currently scheduled job.
- In the "Log" view you can see the results of the validation job.
- If all the selected structures can be migrated to maintenance structures, you see a corresponding info and everything is fine. You can then proceed with the migration of the Product 360 System.
- If the selected structures can not be migrated, you see a corresponding warning. Please then continue with the following step:

## Correct invalid item/variant/product mappings

If objects (items/variants/products) are existing, which have invalid mappings to structure groups and structure group features in the selected structures (see last step above), several assortments with these objects are created.

The assortment name has the following format:

**<catalog\_label>\_<object\_type>\_<cause>**

For each supplier catalog and the master catalog separate assortments are created. Also for each object type (**item**, **variant**, **product**) separate assortments are created.

And at last for each of the following causes (for invalid mappings) separate assortments are created:

- **attributemapping**: objects, that have attribute(s) with multiple mappings to structure group features within the given structures.
- **nodelevel**: objects, that are classified to node-level structure groups of the given structures.
- **multiclassified**: objects, that are multi-classified to one of the given structures.

All assortments are created in an assortment category named "\_\_MIGRATION\_STRUCTURE\_VALIDATION" and are visible for all users (also on those clients without having the Structure Migration Preparation Tool installed).

**Now, the task is, to go through each assortment, and correct the invalid mappings.**

### Versioning

Structures are only validated in the working version, because the migration is only done in the working version. This is because possibly invalid object mappings in a version other than the working versions are not correctable. All affected versions (other than the working version) will later be closed by the Structure Migration Tool (see also [Product 360 Server and Desktop Migration](#)).

You can recheck the structures as often as you want by simply running the Structure Migration Preparation Tool again with the same structures (see last chapter above) until eventually you get a positive result of the validation job. But please be aware, that all existing assortments in the category "\_\_MIGRATION\_STRUCTURE\_VALIDATION" will be automatically deleted, when executing the "Validate structures for migration" action again.

## 7 Media Manager Migration

### 7.1 Update Media Manager

#### 7.1.1 Updating Media Manager File Server

To update the Product 360 - Media Manager File Server, you require the file **PIM\_<Version>\_MediaManager.zip** from your Product 360 distribution.

1. Uncompress the file **PIM\_<Version>\_MediaManager.zip** on your computer.
2. Navigate to the folder **\Setup\HMM** of the uncompressed archive.
3. Copy the **help%0** and **update** directories to **Volume0** on your file server, to the **opasdata** directory.

#### 7.1.2 Updating the client modules

Client modules are updated automatically on the first start after updating Product 360 - Media Manager. If problems occur the following steps can be performed manually.



If you are installing new client modules or updating the version, you must re-activate Product 360 - Media Manager. For more details, refer to Activating Product 360 - Media Manager, defining volumes & setting up Funcd.

### Manually updating the client modules (Windows)

1. Switch to the directory **opasdata/update/win/ocln** on your Volume0.
2. Run **OPAS\_cln.exe**.
3. Follow the subsequent instructions.

### Manually updating the client modules (OSX)

1. Switch to the directory **opasdata/update/osx** on your Volume0.
2. Mount the image **IMM.install.dmg**.
3. Run the **included package installer**.
4. Follow the subsequent instructions.

### 7.1.3 Updating Funcd

To update the Product 360 - Media Manager Funcd, you require the file **PIM\_<Version>\_ThirdPartySoftware.zip** from your Product 360 distribution.



You should always update the Funcd to get the last versions of the third party tools. To update the Funcd and the third party tools proceed as follows:

1. Uninstall Funcd and the third party tools.
2. Do the installation like described in Installing Funcd.

### 7.1.4 Updating web front end

To update the Product 360 - Media Manager web application, you require the file **PIM\_<Version>\_MediaManager.zip** from your Product 360 distribution.

The procedure for updating your web front end is as follows:

1. Uncompress the file **PIM\_<Version>\_MediaManager.zip** on your computer.
2. Stop the Tomcat server.
3. Make a backup of the web application by renaming e.g. **C:\OpasGWebServer** to **C:\OpasGWebServer\_backup**.
4. Navigate to the folder **\Setup\webapp package\full** of the uncompressed archive.
5. Unpack the **OpasGWebServer.zip** to **C:\**.
6. Copy and overwrite the configuration files from **C:\OpasGWebServer\_backup\XOBSessionManager\conf** to **C:\OpasGWebServer\XOBSessionManager\conf**.
7. Copy and overwrite configuration files from **C:\OpasGWebServer\_backup\Tomcat\conf** to **C:\OpasGWebServer\Tomcat\conf**.
8. Delete **C:\OpasGWebServer\Tomcat\webapps**.
9. Copy **C:\OpasGWebServer\_backup\Tomcat\webapps** to **C:\OpasGWebServer\Tomcat\webapps**.
10. Delete the directory **C:\OpasGWebServer\Tomcat\webapps\opas\WEB-INF\lib**.
11. Delete the directory **C:\OpasGWebServer\Tomcat\webapps\opas\WEB-INF\classes**.
12. Delete the directory **C:\OpasGWebServer\Tomcat\webapps\opas\script**.
13. Navigate to the folder **\Setup\webapp package\update** of the uncompressed archive.
14. Unpack **opas.zip** to **C:\OpasGWebServer\Tomcat\webapps\opas** and replace existing files.
15. When you are updating from a version < 8.0 you have to make the following changes in the file **C:\OpasGWebServer\Tomcat\webapps\opas\Base.cfg** to define the database connection parameters.
 

```
<DATABASE_URL>jdbc:jtds:sqlserver://localhost:1433/opasdb</DATABASE_URL>
<DATABASE_DIALECT>org.hibernate.dialect.SQLServer2012Dialect</DATABASE_DIALECT>
<DATABASE_USER>Username</DATABASE_USER>
<DATABASE_PASSWORD>Password</DATABASE_PASSWORD>
```
16. Restart the Tomcat server.

❗ Do copy and not replace the existing opas directory with the archive, because the update package contains only the changed parts of the web front end.

i When changing the layout you have to update the layout profiles and re-assign them to the users. You can use the layout distribution in the "Administration" area in the web front end.

### 7.1.5 Deactivate old modules

The modules: Informatica Media Manager Session Manager, Informatica Media Manager Functd and Informatica Media Manager Internet are no longer needed for versions >= 8.0. Therefore these services can be deactivated.

#### Deactivate Session Manager

Open the "**Windows Services**" and search the service "Informatica Media Manager Session Manger" (Depending on you preinstalled version, the name can differ for example Heiler Media Manager Session Manager). Stop the service if it is running and afterwards set the Startup type to disabled or manual.

#### Deactivate Internet Functd

- If the "Informatica Media Manager Process Watcher" is running as Window service you have to stop this service.
- Open the Process Watcher module.
- Stop the automatic mode by clicking cancel.
- Select **Processwatcher > Processwatcher default settings**
- Disable the monitoring of the Internet functd in the second pane.
- Save and close the application.
- If the "Informatica Media Manager Process Watcher" was running as Window service you have to start this service again.

## 8 Supplier Portal Migration

To update an existing installation of Product 360 Supplier Portal, please run the following steps:

### 8.1 On the Product 360 Supplier Portal Server

- Stop your Product 360 Supplier Portal Windows service "Informatica Product 360 - Supplier Portal"
- Uninstall the old service by running following command in a Windows console ( replace <SERVICE\_NAME> with your Services Service Name, default: **ISP** )

```
<INSTALLATION ROOT>/uninstall.bat <SERVICE_NAME>
```

- Make a backup of your current <INSTALLATION ROOT>/configuration and <INSTALLATION ROOT>/filestorage directories.
- Unzip the installation package "PIM\_<Version>\_SupplierPortal.zip" into your new NEW\_INSTALLATION ROOT directory.
  1. After unzipping merge the new <INSTALLATION ROOT>/configuration/configuration.properties with the configuration file backed up previously. See Product 360 Supplier Portal Release Notes for new or updated configuration options that may be added, changed or removed. Database migrations are applied during the first server start, so please ensure your database connection parameters are still valid.
  2. Replace the new <INSTALLATION ROOT>/configuration/logback.xml with the one you back up previously.
  3. Replace the new <INSTALLATION ROOT>/filestorage directory with the one you back up previously.
- Install the new service by running following command in a Windows console ( replace <SERVICE\_NAME> with your Services Service Name, default: **ISP** )

```
<INSTALLATION_ROOT>/install.bat <SERVICE_NAME>
```


Check the windows services control panel and start the service "Informatica Product 360 - Supplier Portal" if not starting automatically. Verify the startup type is set to "Automatic".

## 9 Web Search Migration

To update an existing installation of Product 360 Web Search, please run the following steps:

### 9.1 On the WebSearch Server

- Stop the service "Informatica Product 360 Search"
- Uninstall the old service by running "service\_HPS\_uninstall.cmd"
- Unzip the installation package "PIM\_xx\_WebSearch.zip" into a temp folder
- In the old installation remove following folders and replace with the same folder from the distribution package
  - apache-tomcat-xxx
  - internal
  - indexExamples
- The solr folder must not be replaced as it contains the index configuration settings
- Compare the new `configuration.properties` file with the old one and add some new property like the REST credentials of PIM WebSearch Server and optional PIM Server. The new properties are listed below.

 If you want to protect the passwords using standard implementation of encryption you should replace the Java JCE policy files in `jre\lib\security` folder. For more information please refer to chapter Encryption of secure information in the Server Installation manual.

- Run the command "configure.cmd" to update configuration files
- Run the command "service\_HPS\_install.cmd" to install Windows service again

Check the windows services control panel and start the service "Informatica Product 360 Search" if not starting automatically. Verify the startup type is set to "Automatic".

### 9.2 On the Client


Log in at Product 360 Desktop and open the View "Search Indices". Run an update on all indices to make sure all indices are searchable with the updated version.


Log in at Product 360 Web and run a test search to verify proper functionality.

#### 9.2.1 New properties in `configuration.properties` file:


Property	Default Value	Description	Remarks
<b>Settings for customized EncryptionService</b>			
ppm.encryptionService	< empty >	Full classname of a customized EncryptionService implementation ( <b>optional</b> )	


## WebSearch REST server settings


 WebSearch REST credential is now set in `configuration.properties` file. In previous version it was stored in file: `<PIM_WEBSEARCH_INSTALLATION_ROOT>\apache-tomcat-7.0.14\webapps\hps-web\WEB-INF\spring-security.properties`.

fulltextsearch.rest.url	http://localhost:18090/hps-web/rest2	Base Url for REST.	
fulltextsearch.rest.username	config	Username of the REST user of WebSearch server	
fulltextsearch.rest.password	Heiler33!	Password of the REST user of WebSearch server	 If you want to encrypt the password please refer to chapter Encryption of secure information in the Server Installation manual.

## HPM REST server settings:

 HPM REST credential is set in `configuration.properties` file instead in each search index.

hpm.rest.url	http://localhost:1512/rest	Base Url for REST.	
hpm.rest.username	rest	Username of the REST Product 360 user which has Service API access permissions.	
hpm.rest.password	heiler	Password of the REST Product 360 user	 If you want to encrypt the password please refer to chapter Encryption of secure information in the Server Installation manual.

Settings for tomcat container: Password can be hashed now to secure tomcat password			
container.admin.user.password	heiler	Password of tomcat user	 The password can be hashed. Please refer to chapter Web Container Password Hashing for further information.
<b>Solr settings</b>			
solr.default.core.size	100000	This defines the maximum number of documents that will be indexed per solr core.	This is an important property for high volume installations. According to this property multiple solr cores will be created to keep the size of a single core small. The search on multiple cores will be performed with the solr distributed search.
solr.cluster.node.list		Definition of a comma separated list of solr nodes used for indexing multiple cores in round robin approach. The list has to be in the format server1:18090,server2:18090,server3:18090	This is an important property for high volume installations in combination with the property <i>solr.default.core.size</i> to spread multiple solr cores over different server nodes.

## 10 Business Process Management

### 10.1 Workflows for Product 360 versions >= 8.0.00

Workflow processes designed with the Informatica BPM Designer for the Product 360 versions 8.0.00 to 8.0.03 do not have to be migrated. However, a new default workflow is required for the new "Terminate workflow" functionality and needs to be deployed to the Informatica BPM server (see Informatica BPM Installation).

### 10.2 Workflows for Product 360 versions < 8.0.00

Workflow processes designed with the jBPM workflow engine for Product 360 versions earlier than version 8.0.00 cannot automatically be migrated and deployed to the Informatica BPM server.

## 11 How to apply a Hotfix or EBF

### 11.1 Preconditions.

- You must be able to use a command prompt to continue. If not, please contact your system administrator to assist.
- Please read the Release Notes of the version to be installed.
- **It is highly recommended to backup your system before updating. This especially includes the databases, file storage areas, configuration files and binaries distributables.**

## 11.2 Installation of Hotfix or EBF on Product 360 Server

**i** EBF packages contain only **delta** plugins which have been changed. In order to recognize such packages, the suffix delta is contained in its name. Hotfixes and Major releases have the suffix **full** in its name.

Before applying an Hotfix or Major update, you have to update the Control Center itself. See [P360 - Control Center Migration](#)

- Login into the Control Center
- Navigate to the installation tab
- With the upload functionality, upload all packages you want to deploy. Do not rename or extract any of the packages that are delivered.
- If you want to deploy also accelerator plugins or custom plugins, you can also upload them. Custom plugins have to be uploaded separately as jar file.
- Please note that all uploaded files will be deployed and overwrite the existing files. But all files will be backedup in a folder, so you can revert them if something went wrong.
- By clicking the upload button, the update process will start and all configured servers will be updated. All configuration files on the server will be also replaced with the ones that are located in the Control Center package.
- Execute steps in [PIM - Core Database Migration](#) by running setup to your existing database connection in order to update existing database.

Generally, you should run your custom JUnit tests in order to ensure that your custom functionality is still working.

## 11.3 Installation of Hotfix or EBF on Product 360 Desktop

- Before updating the Desktop you should execute the migration of the Server first.
- For **Hotfix** or **Major** updates:
  - Remove content of Product 360 Desktop Installation Folder and unzip the content of the new release package (EBF only apply delta.)
  - Update the configuration files if you have any customizings here. Instead of copying the whole file, rather copy only your changes, since there might be some new settings which then will be overwritten.
- For **EBF**'s:
  - Remove the existing plugins with old revision and then place the plugin with new revision which are part of the EBF.