

Based on the provided documentation, the `xxmx_hcm_person_ext_pkg` package appears to be a component of a software system that is responsible for performing specific data manipulation tasks in the context of Human Capital Management (HCM) within an Oracle database environment.

The package consists of two procedures:

- `transform_line`: This procedure is designed to apply specific data transformation rules to records within a database, typically following the copying of data from a staging table. It constructs and executes dynamic SQL statements for each record, based on the metadata obtained from related tables. The purpose is to perform transformations tailored to the specific needs of the data migration process. The procedure also incorporates extensive error handling, logging messages for successful operations, as well as capturing and logging errors when things go wrong.
- `update_DFF`: The second procedure, `update_DFF`, focuses on updating Descriptive Flexfields (DFF), which are a flexible mechanism in Oracle applications for capturing additional information. This procedure appears to be a simple, straightforward operation that sets a specific column `ASS_ATTRIBUTE20` to the value 'TEST' within the `XXMX_PER_ASSIGNMENTS_M_STG` table. Like `transform_line`, it contains error handling to ensure that any issues are logged and that transactions are rolled back in case of failures.

Both procedures receive similar input parameters, such as identifiers for the application suite, application, business entity, file set, and migration set, which suggest that they operate within a structured framework of the HCM application. They also both produce an output parameter `pv_o_ReturnStatus` to indicate whether the procedure has succeeded or failed.

Overall, the `xxmx_hcm_person_ext_pkg` package plays a role in facilitating the migration and transformation of HCM data within an Oracle application system, with a focus on ensuring data integrity and providing audit trails through logging and exception management.

## Update\_DFF

## xxmx\_hcm\_person\_ext\_pkg Package Documentation

### Overview

The `xxmx_hcm_person_ext_pkg` package is part of a software application designed to manage data migration and transformation within an Oracle database environment. It includes procedures that handle data transformation rules and updating Descriptive Flexfields (DFF) for the Human Capital Management (HCM) module. This documentation provides a detailed technical breakdown of the package's specification and its body.

### Package Specification

```
PACKAGE xmxm_hcm_person_ext_pkg AS
  PROCEDURE transform_line(...)
  PROCEDURE Update_DFF(...)
END xmxm_hcm_person_ext_pkg;
```

#### Procedures

- `transform_line`: A procedure that applies transformation rules after copying from STG (staging) and applying simple transforms. Accepts various parameters related to the application suite, business entity, and IDs associated with the file set and migration set.
- `update_DFF`: A procedure for updating Descriptive Flexfields related to assignments in the HCM module. It also takes parameters similar to `transform_line`.

### Package Body

The package body contains the implementation of the `transform_line` and `update_DFF` procedures.

#### Global Constants and Variables

The package body defines several global constants and variables, which include the following:

- `gcv_PackageName`: Constant storing the package name, set to 'xxmx\_hcm\_person\_ext\_pkg'.
- `gvv_ApplicationErrorMessage`: Variable for storing error messages, with a `VARCHAR2` data type and size 2048.

Other global variables are declared to store values related to data transformation such as counters, messages, and progress indicators.

#### Procedure: transform\_line

The `transform_line` procedure iterates over a cursor `TransformMetadata_cur` which fetches transformation metadata. It constructs a dynamic SQL statement for each record in the cursor, then executes the statement to update the corresponding transformation column.

##### Cursor: TransformMetadata\_cur

This cursor selects metadata from related tables to build the transformation SQL statement.

#### Dynamic SQL Execution

The constructed SQL statement in `gvt_sql` is executed using the `EXECUTE IMMEDIATE` statement. If successful, it logs a message using `xxmx_utilities_pkg.log_module_message` procedure. The return status is set to 'S' upon success.

#### Exception Handling

The procedure has an exception block that captures any errors during execution. There are two exceptions handled:

- `e_ModuleError`: A specific exception that may be raised within the package body.
- `WHEN OTHERS`: A general exception handler for unexpected errors.

In both cases, it rolls back any changes, logs an error message, and raises an application error with the message stored in `gvv_ApplicationErrorMessage`.

#### Procedure: Update\_DFF

The `update_DFF` procedure performs an update on the table `XXMX_PER_ASSIGNMENTS_M_STG`, specifically setting the `ASS_ATTRIBUTE20` column to 'TEST' for the provided `MigrationSetID`. It also uses exception handling similar to `transform_line`, ensuring error messages are logged and application errors are raised in case of failure.

#### Commit and Rollback

The `update_DFF` procedure commits the transaction upon successful update. Conversely, both `transform_line` and `update_DFF` perform a rollback in case an exception occurs.

#### Errors Display

At the end of the package body, the `SHOW ERRORS PACKAGE BODY xmxm_hcm_person_ext_pkg` command is called to display any compilation errors that might occur within the package body.

### Summary

This package provides mechanisms for transforming data during migration processes and updating Descriptive Flexfields in the HCM module. It emphasizes error handling and logging, ensuring that operations can be audited and issues traced. The use of dynamic SQL provides the flexibility to handle various transformations based on metadata-driven logic.

## transform\_line

## xxmx\_hcm\_person\_ext\_pkg Package Documentation

The provided PL/SQL code contains a package `xxmx_hcm_person_ext_pkg` along with its body which seems to be part of a larger software application related to Human Capital Management (HCM) extensions. Below is a detailed technical documentation for the source code. This documentation assumes a familiarity with Oracle PL/SQL language and concepts.

### Package Specification

The package specification `xxmx_hcm_person_ext_pkg` defines two procedures:

- `transform_line`
- `Update_DFF`

The code snippet below shows the procedure declarations from the package specification:

```
PROCEDURE transform_line(
  pt_i_ApplicationSuite IN xmxm_seeded_extensions.application_suite%TYPE,
  pt_i_Application      IN xmxm_seeded_extensions.application%TYPE,
  pt_i_BusinessEntity   IN xmxm_seeded_extensions.business_entity%TYPE,
  pt_i_StgPopulationMethod IN xmxm_core_parameters.parameter_value%TYPE,
  pt_i_FileSetID        IN xmxm_migration_headers.file_set_id%TYPE,
  pt_i_MigrationSetID    IN xmxm_migration_headers.migration_set_id%TYPE,
  pv_o_ReturnStatus      OUT VARCHAR2
);

PROCEDURE Update_DFF(
  pt_i_ApplicationSuite IN xmxm_seeded_extensions.application_suite%TYPE,
  pt_i_Application      IN xmxm_seeded_extensions.application%TYPE,
  pt_i_BusinessEntity   IN xmxm_seeded_extensions.business_entity%TYPE,
  pt_i_StgPopulationMethod IN xmxm_core_parameters.parameter_value%TYPE,
  pt_i_FileSetID        IN xmxm_migration_headers.file_set_id%TYPE,
  pt_i_MigrationSetID    IN xmxm_migration_headers.migration_set_id%TYPE,
  pv_o_ReturnStatus      OUT VARCHAR2
);
```

Both procedures accept similar parameters used for application suite, application, business entity, stage population method, file set ID, and migration set ID. They also provide an output parameter `pv_o_ReturnStatus` to indicate the status of the procedure execution.

### Package Body

The package body `xxmx_hcm_person_ext_pkg` provides implementations for the procedures `transform_line` and `update_DFF`. It includes a set of global constants, variables, a custom exception, and other local variables needed for the procedure implementations.

#### Global Constants and Variables

The package body starts with the declaration of global constants and variables, such as `gcv_PackageName`, which holds the name of the package, and other variables related to logging and error handling.

#### Procedure transform\_line

The `transform_line` procedure is described as a hook procedure to apply transformation rules after copying from a staging (STG) table and applying simple transforms.

##### Parameters

The procedure `transform_line` accepts the following parameters:

- `pt_i_ApplicationSuite`
- `pt_i_Application`
- `pt_i_BusinessEntity`
- `pt_i_StgPopulationMethod`
- `pt_i_FileSetID`
- `pt_i_MigrationSetID`
- `pv_o_ReturnStatus`

##### Cursor TransformMetadata\_cur

The procedure declares a cursor `TransformMetadata_cur`, which selects metadata information from multiple tables joined together. It uses this cursor to loop through the records and construct a dynamic SQL statement to update transformation-related columns.

#### Loop Logic and Dynamic SQL Execution

The main logic of the `transform_line` procedure is within a loop that iterates over records fetched by the `TransformMetadata_cur` cursor. It constructs and executes dynamic SQL to update transformation columns based on the retrieved metadata.

#### Logging and Error Handling

The procedure logs messages before and after the execution of the dynamic SQL using another package `xxmx_utilities_pkg.log_module_message`. It has exception handling blocks for both a custom exception `e_ModuleError` and the default `OTHERS` exception to manage errors and perform cleanup actions if necessary.

#### Procedure Update\_DFF

The `update_DFF` procedure is a stub with a similar parameter structure as `transform_line`. It performs an update on a table named `XXMX_PER_ASSIGNMENTS_M_STG` setting a column `ASS_ATTRIBUTE20` with a literal value 'TEST'.

##### Parameters

- `pt_i_ApplicationSuite`
- `pt_i_Application`
- `pt_i_BusinessEntity`
- `pt_i_StgPopulationMethod`
- `pt_i_FileSetID`
- `pt_i_MigrationSetID`
- `pv_o_ReturnStatus`

##### Error Handling

Like `transform_line`, `update_DFF` includes error handling blocks for a custom exception and the default `OTHERS` exception. It rolls back any changes and logs messages in case of errors.

### Exception Handling

The package body defines a custom exception `e_ModuleError` which is used within both procedures to manage known error conditions. The `OTHERS` exception clause is used to handle unexpected exceptions.

### Conclusion

The package `xxmx_hcm_person_ext_pkg` provides functionality to manipulate data transformations during an HCM data migration process. The `transform_line` procedure updates records with transformation logic, whereas `update_DFF` is a procedure stub for updating Descriptive Flex Fields (DFF). Both procedures are equipped with logging and error handling to ensure the integrity of the operation.