Emoveo Version 2.8.1

PPMetrics – Configuration Guide

Document Version: 2.6
Document Release Date: June 2023

Table of Contents

1. Introduction

This document provides information about the configuration of the Emoveo application. It was written for:

- Emoveo configurators

Related documents:

- Installation and Administration Guide
- Report Configuration Guide
- User Guide

2. Emoveo Overview

Emoveo is an application that automates the IT organizational process of code deployment. It enables organizations to define their custom migration processes, then execute them in compliant and efficient manners. Emoveo is a web-based tool with a friendly user interface, designed to take the complexity out of this critical yet cumbersome process.

3. Emoveo Objects and Concepts

- **Migration** - Migration, or deployment, is the organizational process of transferring one or more application components through a structured process (e.g., file, script).
- **Environment** - A server or a database instance which the migration processes may transfer components from and to.
- **Object Type** - A series of technical steps required to execute as part of the migration process.
- **Special Command** – Particular reusable function embedded in one or more object types and commands.
- **Package Line** – A single application component to be migrated (e.g. AOL or SQL). This is typically migrated as a part of larger migration or package (defined below).
- **Package** - A group of application components or package lines which are being migrated as part of the same migration.
- **Workflow** - A structured flow, consisting of a user-defined number of steps, defining a migration process and followed throughout its instances.
- **Custom Fields** - Various Emoveo objects allow configurators to augment the canned fields with custom ones.
- **Tokens** – Variables which Emoveo objects such as special commands and object types may reference during their execution. The information referenced through tokens is undefined until the actual execution in a specific context.

4. Configuring Environments

As part of the migration process, Emoveo logs onto remote machines using well established protocols, in the same manner as users normally would. While any user account may be used to access the environments, it is recommended to create and utilize dedicated accounts for the migration processes.

4.1. The Environments Screen

To view the environments already configured and perform general operations on them, go to the menu item Configure -> Environments. The following operations are available on this screen:

Operation	Method
Search environment(s)	Enter any part of a search string into the text box to the right of the magnifying glass and hit SEARCH. Alternatively, utilize the search pane on the right to filter for environment(s) based on several attributes
Edit environment	Click the bolded environment name
Copy environment(s)	Select one or more environments to copy by checking the box to their left. Then from the 'Action' drop down select 'Copy selected Environments' and hit GO
Delete environment(s)	Select one or more environments to copy by checking the box to their left. Then from the 'Action' drop down select 'Delete selected Environments' and hit GO. Please note that a workflow that has already been used by an associated package cannot be deleted.
Add an environment	Click the ADD ENVIRONMENT button below the environments table

Table 1- Environment screen operations

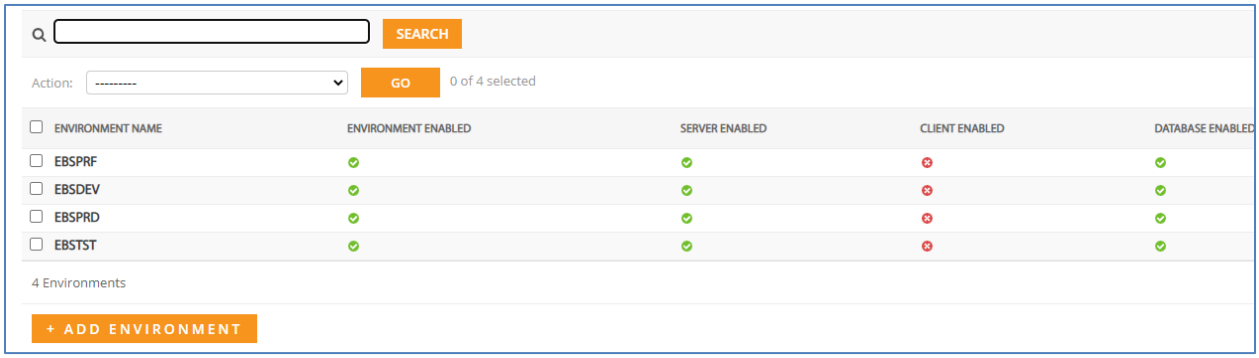

Fig. 1 - The environments screen

4.2. Updating Basic Environment Information

The environment screen contains a set of required and optional fields to fill out for the environment across two tabs. Below is a description of the three out-of-the-box sections of the environment screen. Please note that if custom environment fields have been configured, they will appear in a fourth section at the bottom.

Prompt	Description
Environment Name	Unique name for the environment. Typically, one part of the environment's name describes the role of the environment in the migration chain (e.g., DEV)
Reference Code	Unique string representation of the environment, which should normally not be modified post creation
Description	A brief description of how the environment is used
Environment Enabled	Whether to enable the use of the environment throughout Emoveo. By default, new environments are set to be enabled by default

Table 2 - General environment fields

Prompt	Description
Enabled	Whether to enable this element of the environment
Hostname	The DNS name or IP address of the machine
OS Type	The type of operating system used by the machine
NT Domain	The domain name to use if the OS type is a Windows server
Username	The username that Emoveo uses to log onto the server/client to perform the deployment operations
Password, Confirm Password	The password of the supplied username
Base Path	The path for applications on this computer, typically the home directory of the defined username. When Emoveo logs in to the environment, it changes directories to the base path as the first step

Table 3 - Server and client environment fields

Prompt	Description
Database Enabled	Whether to enable this element of the environment
Hostname	The DNS name or IP address of the machine running the database
Database Type	The database's product name
Connection String	The connection string used to connect to this database from a remote system
Username	The database username to be used by Emoveo to make remote database connections
Password, Confirm Password	The password of the supplied username

Oracle SID	For Oracle databases only, the SID of the database
Database Link	Reference to a remote Oracle database, if applicable
Port Number	The port number of the database's listener
JDBC URL	For Oracle databases only, an alternative connection method may be provided using JDBC technology
Version	The version number of the database. Used for informational purposes

Table 4 - Database environment fields

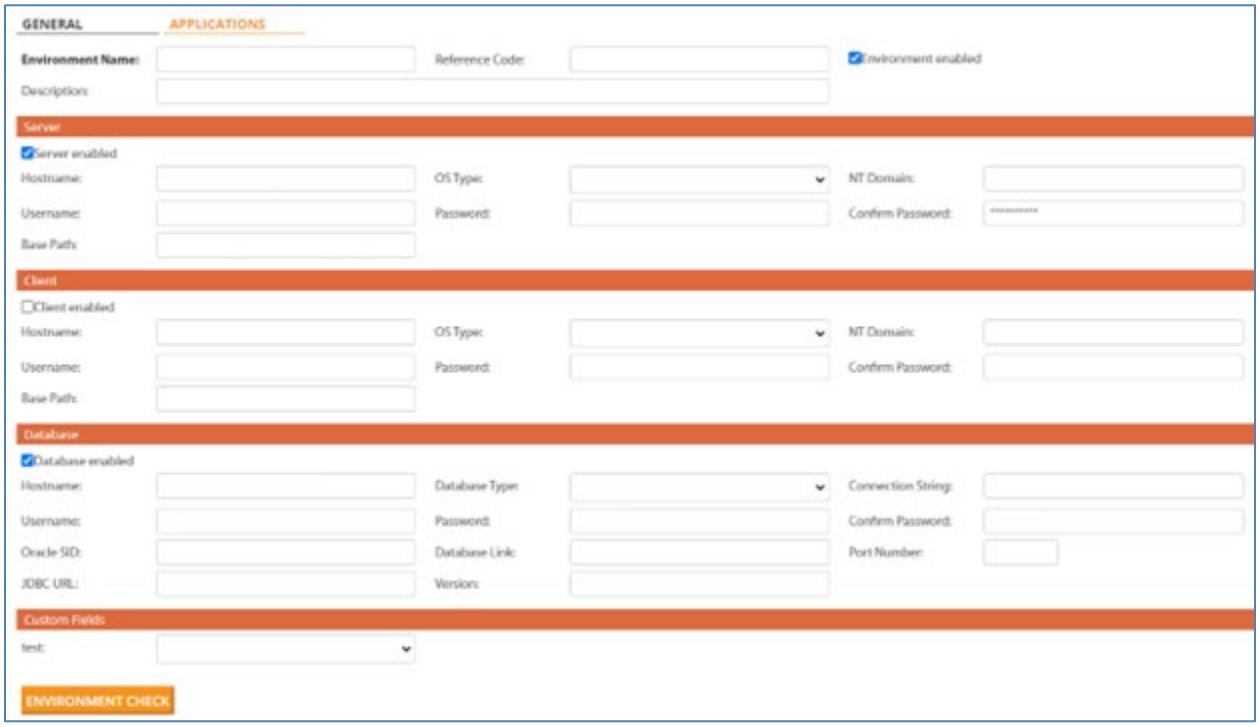

Fig. 2 - Editing environment information

4.3. Updating Environment's Applications Information

It is often the case that a given environment hosts more than one application involved in the migration process. Consequently, the actual commands executed as part of the package processing often need to reference a specific application within the environment. For that reason, Emoveo allows configurators to define any number of applications for each configured environment. Please note the following with respect to entry of application information in Emoveo's environments:

- To access the applications of a given environment, toggle to the APPLICATIONS section from the top of the environment screen.
- The applications section allows you edit, create, and delete applications for the given environment.
- The actual fields to populate are similar to the environment fields and need to be filled out when containing different values than the former.

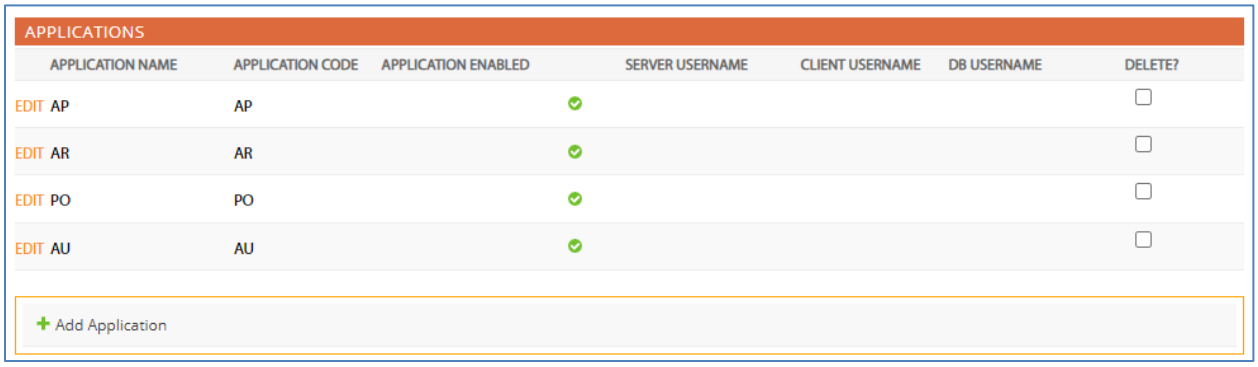

Fig. 3 - The applications screen

4.4. Checking Environments

Emoveo provides a straightforward way of validating the information supplied to create the environments. From either tab of the environment screen, click on ENVIRONMENT CHECK to launch the checker. On the checker screen, check the boxes next to each test you would like to perform, and click on PERFORM SELECTED CHECKS. Emoveo will then display the actual commands it executed, as well as the results of the check.

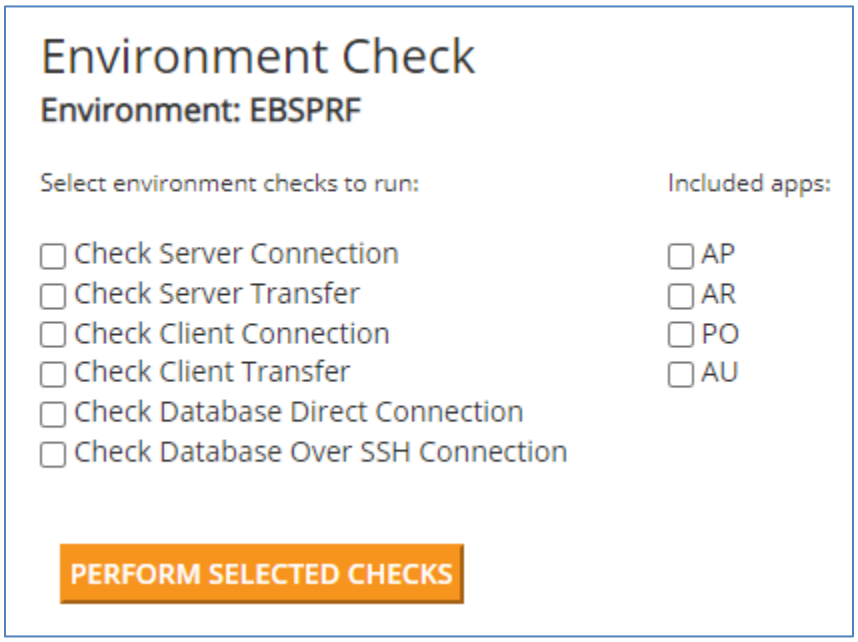

Fig. 4 - The environment checker

5. Configuring Special Commands

Special commands are configurable Emoveo objects, serving as reusable functions embedded in one or more object types and commands. The Emoveo system comes with a series of canned special commands, covering the most common functions used in migration processes. Additionally, custom special commands may be added to the system to address more environment specific situations.

5.1. The Special Commands Screen

To access the special commands screen, go to the menu item Configure -> Special Commands. The following operations are available from this screen:

Operation	Method
Search commands	Enter any part of search string into the text box to the right of the magnifying glass and hit SEARCH. Alternatively, utilize the search pane on the right to filter for command(s) based on their status
Edit command	Click the bolded special command name
Delete command(s)	Check the box to the left of every command to delete, then from the 'Action' drop-down select 'Delete selected Special Commands' and hit GO
Copy command(s)	Check the box to the left of every command to copy, then from the 'Action' drop-down select 'Copy selected Special Commands' and hit GO

Table 5 - Special commands screen operations

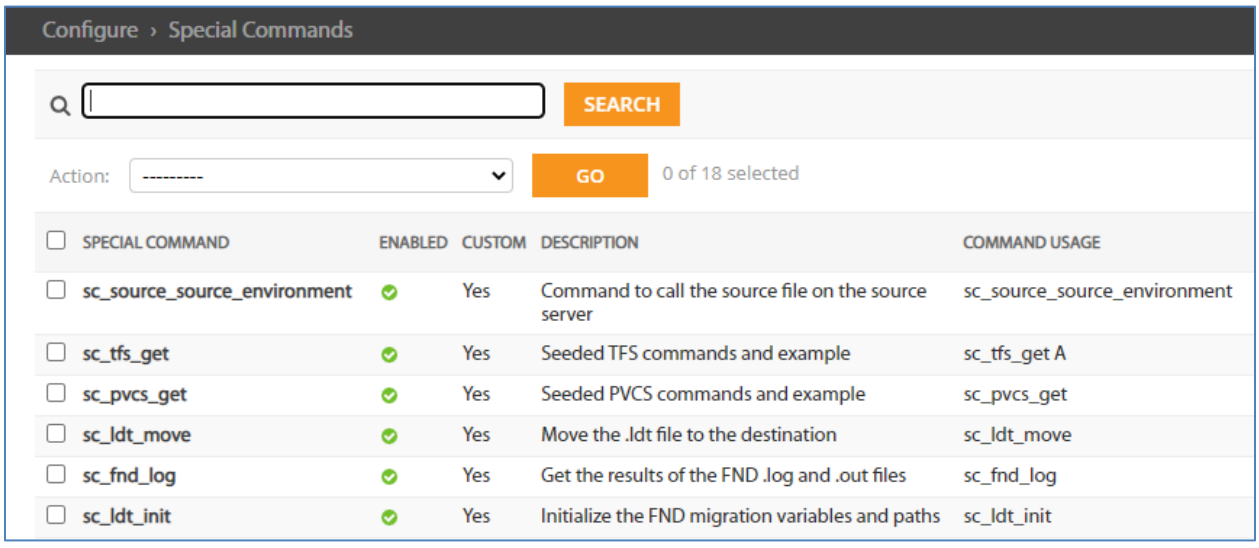

Fig. 5 - The special commands screen

5.2. Editing Special Command Information

The special commands screen contains three sections: General, parameters, and commands. The general section covers basic command information, the parameters section optionally defines values passed to the special command during execution time, and the commands section covers the actual steps which the special command executes when invoked.

Section	Prompt	Description
General	Special Command	A required unique name of the special command, typically in lower cases and underscores
General	Description	Brief optional description of the special command
General	Enabled	Whether to enable the command for use throughout Emoveo
Parameters	Parameter Name	Required name of the parameter
Parameters	Default Token	Required default token name
Parameters	Description	Optional brief description of the parameter
Parameters	Order	The order of the parameter relative to other parameters in the same special command
Commands	Order	The processing order of the command relative to other commands in the same special command
Commands	Command Name	A required name for the command, typically representing its operation
Commands	Timeout (seconds)	The number of seconds Emoveo will allow the command to be executed before a timeout error occurs and the execution halted
Commands	Enabled	Whether to execute the command when the special command is invoked
Commands	Description	Optional brief description of the command
Commands	Condition (SQL)	An optional SQL based condition evaluated during the command's execution and determines whether to execute the command steps
Commands	Command Steps	The command line commands to execute, which may involve operating system commands, other special commands, and tokens

Table 6 - Special command fields

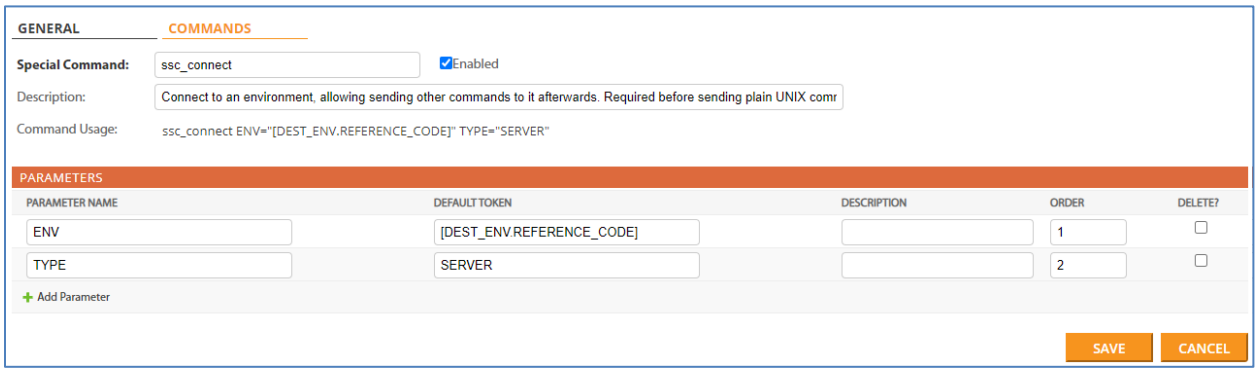

Fig. 6 - Editing special command information

6. Configuring Object Types

Object types are configurable Emoveo objects containing a series of technical steps executed as part of the migration process. Typically, each object type represents a specific type of application component to be migrated. The Emoveo system comes with a series of canned object types, covering the most common functions used in migration processes. Additionally, custom object types may be added to the system to address more environment specific situations.

6.1. The Object Types Screen

To access the object types screen, go to the menu item Configure -> Object Types. The following operations are available on this screen:

Operation	Method
Search object types	Enter any part of search string into the text box to the right of the magnifying glass and hit SEARCH. Alternatively, utilize the search pane on the right to filter for object types(s) based on their status
Edit object type	Click the bolded object type name
Delete object type (s)	Check the box to the left of every object type to delete, then from the 'Action' drop-down select 'Delete selected Object Types' and hit GO
Copy object type (s)	Check the box to the left of every object type to copy, then from the 'Action' drop-down select 'Copy selected Object Types' and hit GO
Add object type	Click the button ADD OBJECT TYPE at the bottom of the screen

Table 7 - Object types screen operations

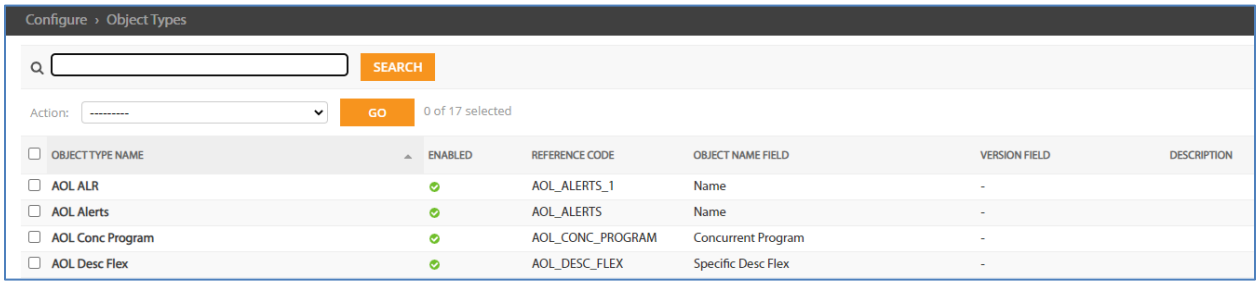

Fig. 7 - The object types screen

6.2. Editing Object Type Information

The object types screen contains three sections: General, fields, and object commands. The general section covers basic object type information, the fields section optionally defines values passed to the object type during execution time, and the object commands section defines the actual steps which the object type executes when invoked.

Section	Prompt	Description
General	Object Type Name	A required unique name of the object type
General	Reference Code	Unique string representation of the object type, which should normally not be modified post creation
General	Enabled	Whether to enable the command for use throughout Emoveo
General	Description	Brief optional description of the object type
General	Object Name Field	The name of the object displayed in package lines that utilize it, and linked to one of the object type's fields
General	Version Field	Typically used when Emoveo is integrated with a version control system. Indicates which field represents the revision number for the object.
Fields	Field Prompt	Required label of the field
Fields	Field Token	Required token (variable) representation of the field
Fields	Description	Optional brief description of the field
Fields	Control Type	The type of control to be used by users when populating the field
Fields	Storage	The database column to store the value of the field. Use 'Auto' to let Emoveo pick the first available column
Fields	Enabled	Whether the field should be available for use
Fields	Visible	Whether to make the field visible to users
Fields	Editable	Whether to make the field editable to users
Fields	Required	Whether to make the field mandatory to users
Fields	Order	The order of the field, relative to other fields in the same object type
Object Commands	Order	The sequence of the command's execution relative to other commands in the same object type
Object Commands	Timeout (seconds)	The number of seconds Emoveo will allow the command to be executed before a timeout error occurs and the execution halted
Object Commands	Enabled	Whether to process the command
Object Commands	Description	Optional brief description of the command
Object Commands	Condition (SQL)	In situations where the command's execution depends on the state of the data, an SQL based condition may be entered and reference tokens. For example: *'[P.EXEC_AS_APPS_YN]' = 'N'*
Object Commands	Command Steps	The command line commands to execute, which may involve operating system commands, special commands, and tokens

Table 8 - Object type fields

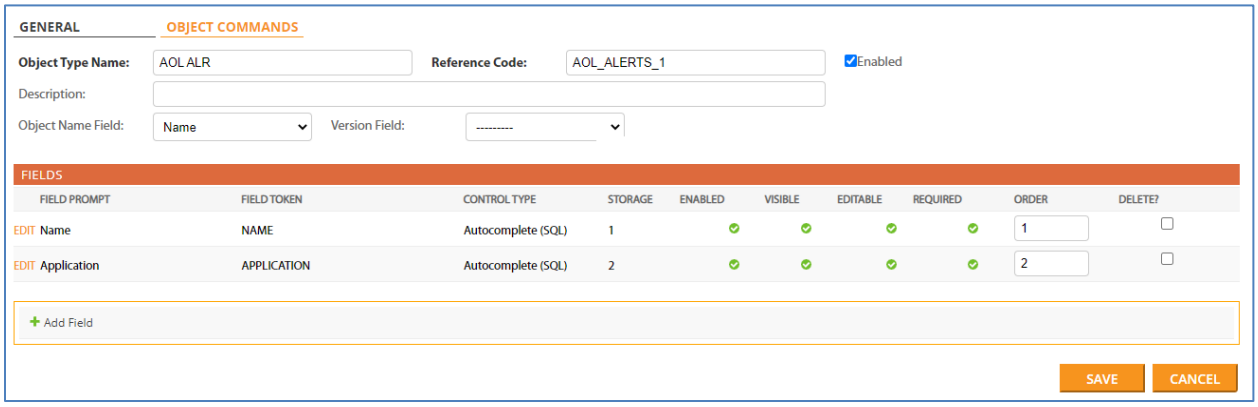

Fig. 8 - Editing object type information

7. Configuring Workflows

An Emoveo workflow is a configurable object, representing a structured flow defining a migration process and followed throughout. Every workflow consists of one or most steps, which may be technical ‘execution steps’, or ‘decision steps’, in which authorized user(s) decide on the workflow’s progression.

7.1. The Workflows Screen

To access the workflows screen, go to the menu item Configure -> Workflows. The following operations are available on this screen:

Operation	Method
Search workflows	Enter any part of search string into the text box to the right of the magnifying glass and hit SEARCH. Alternatively, utilize the search pane on the right to filter for workflow(s) based on their status
Edit workflow	Click the bolded workflow name
Delete workflow(s)	Check the box to the left of every workflow to delete, then from the ‘Action’ drop-down select ‘Delete selected Workflows” and hit GO
Copy workflow(s)	Check the box to the left of every workflow to copy, then from the ‘Action’ drop-down select ‘Copy selected Workflows’ and hit GO
Add workflow	Click the button ADD WORKFLOW at the bottom of the screen

Table 9 - Workflow screen operations

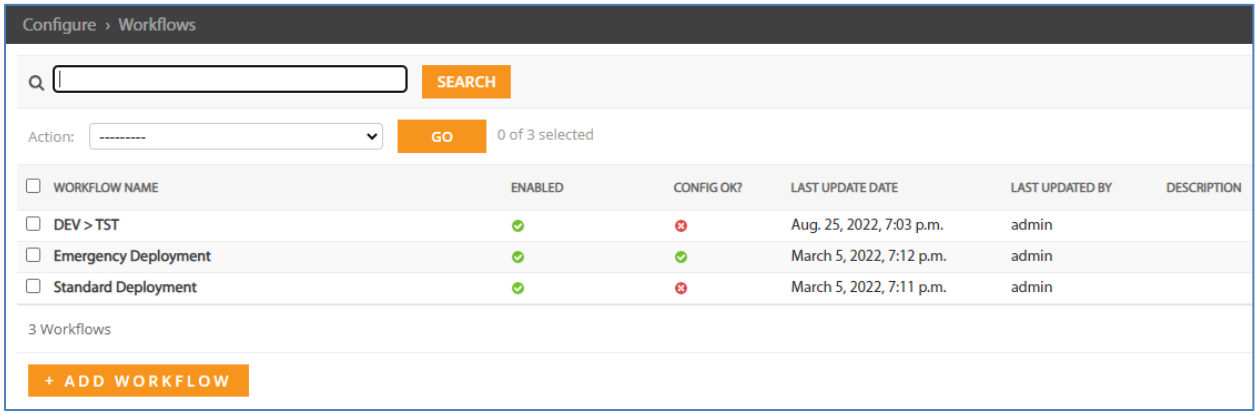

Fig. 9 - The workflows screen

7.2. Editing Workflow Information

The object type screen contains three sections: General, workflow steps, and object types. The general section covers basic workflow information, the workflow steps section defines the building blocks of the workflow, and the object types section defines those available for use in the workflow.

Section	Prompt	Description
General	Workflow Name	Required unique name of the workflow
General	Reference Code	Unique string representation of the workflow, which should normally not be modified post creation. If left blank, Emoveo will pick a name automatically
General	Enabled	Whether to make the workflow available for use
General	Description	Optional brief description of the workflow
Object Types	Chosen object types	Use the lists on both sides of the screen to determine which object type(s) should be allowed for use in instances of the workflow
Workflow steps	On success step	Should be populated after the workflow steps have been defined. Specifies which step the workflow should transition to in case of a successful execution or a decision to move forward
Workflow steps	On failure step	Should be populated after the workflow steps have been defined. Specifies which step the workflow should transition to in case of an unsuccessful execution or a decision not to move forward
Workflow steps	Order	Should be populated after the workflow steps have been defined. Emoveo starts workflow execution from the lowest order value of all the steps. Order of subsequent steps is governed by the transitions defined, although reports and logs will still make use of the order values defined

Table 10 - Workflow fields

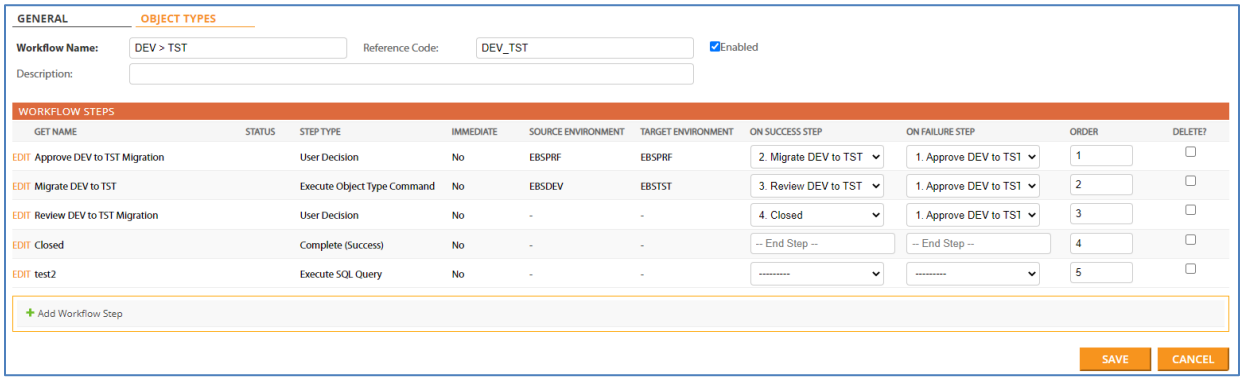

Fig. 10 - Editing workflow information

Prior to configuring the workflow steps and its transitions, it is recommended to draw the entire process on paper or using a diagramming tool such as Microsoft Visio. One characteristic of a well-defined workflow is that all the transitions are specified, and Emoveo will warn the configurator upon save if that is not the case.

7.3. Editing Workflow Step Information

Workflows steps are the main building blocks of workflows and may incorporate heavy business logic. The workflow step screen consists of four sections: General, step security, step commands, and notifications. The general section covers the basic attributes of the step, the step security section defines the user(s) authorized to act on it, the step commands section defines the technical commands which

may execute when the step is traversed, and the notification sections defines the details of the emails which should optionally go out under certain circumstances.

Section	Prompt	Description
General	Step Type	The type of workflow step, out of the following: • User Decision – Step in which an authorized user decides whether to progress the workflow • Execute Workflow Step Command – Step in which Emoveo executes the command(s) defined for the step • Execute Object Type Command – Step in which Emoveo executes the command(s) associated with step's object type • Execute SQL Query – Step in which Emoveo executes an SQL query, whereas its results determine whether to define the step's result as successful or not • Complete (Success) – A terminal workflow step defining the entire workflow instance as successful • Complete (Failure) – A terminal workflow step defining the entire workflow instance as unsuccessful
General	Step Name	A unique name within the step's workflow
General	Immediate	For execution steps, define whether Emoveo should automatically proceed with the execution when the step becomes active, or whether an authorized user should have to manually order Emoveo to initiate the execution
General	Description	Optional brief description of the step
General	Status	Determines the status designation of the workflow instance when the step becomes active
General	Source Environment	For execution steps performing migrations between environments, the source environment in the process
General	Target Environment	For execution steps performing migrations between environments, the target environment in the process
Notifications	Trigger on	The event which should trigger the email notification: • Eligible – The step becomes active • Success – The step completed processing with a positive outcome • Failure – The step completed processing with a negative outcome
Notifications	Enabled	Whether to process the email notification. Disabled notifications are ignored during package execution
Notifications	Grouping	Whether to group emails associated with the step at various levels. The options are: • Do not group (Always send) – An enabled notification which meets the 'Trigger on' criterion will launch notifications at the individual step level, any number of times and for any number of package lines • Once per package line – An enabled notification which meets the 'Trigger on' criterion will launch notifications only on the first such occurrence of each package line • Once per line group – An enabled notification which meets the 'Trigger on' criterion will launch notifications only on the first such occurrence at the line group level

		• Once per package – An enabled notification which meets the 'Trigger on' criterion will launch notifications only on the first such occurrence at the entire package level
Notifications	Email to	Email address(es) of the recipient(s) or tokens resolving to the same. Email addresses should be separated by commas
Notifications	Subject	The subject line of the email, which may also include tokens
Notifications	Notification Text	Contents of the email, which may also include tokens
Step Security	User Step Security	One or more individual users authorized to act on the step
Step Security	Group Step Security	One or more user groups authorized to act on the step
Step Commands	Command Information	For execution steps of type 'Execute Workflow Step Command', one or more commands to be executed. Please refer to the special commands section of this document for an explanation of the different fields

Table 11 - Workflow step fields

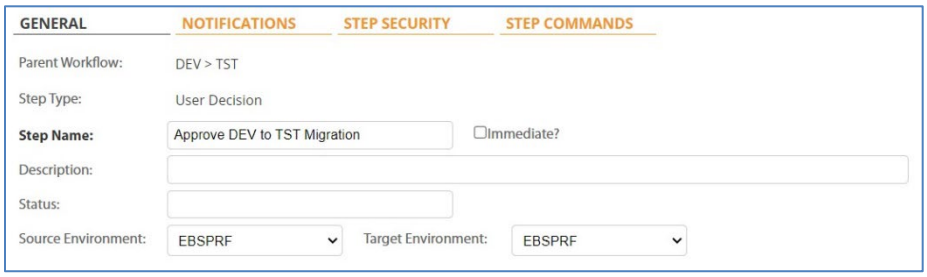

Fig. 11 - Editing workflow step information

8. Configuring Custom Fields

While the Emoveo default environment and package fields contain all the required fields for the actual migration processes, often, especially in large organizations, there is a need to augment these standard fields with custom ones. These custom fields serve two primary purposes:

1. Information and reporting purposes.
2. Referenced in command execution to accomplish tasks specific to the organization and the environment.

To configure custom fields of *environments,* go to the menu item Configure -> Environment Fields, and for *packages* go to Configure -> Package Fields. The following operations are available on the custom fields screen:

Operation	Method
Edit field	Click the EDIT link to the left of an existing custom field
Delete field	Check the DELETE box to the right of every existing custom field, then click SAVE
Add field	Click on the 'Add Custom Field' link at the bottom left of the screen
Set field order	Once the different fields have been configured, their display order on the screen may be set using the order field

Table 12 - Custom fields screen operations

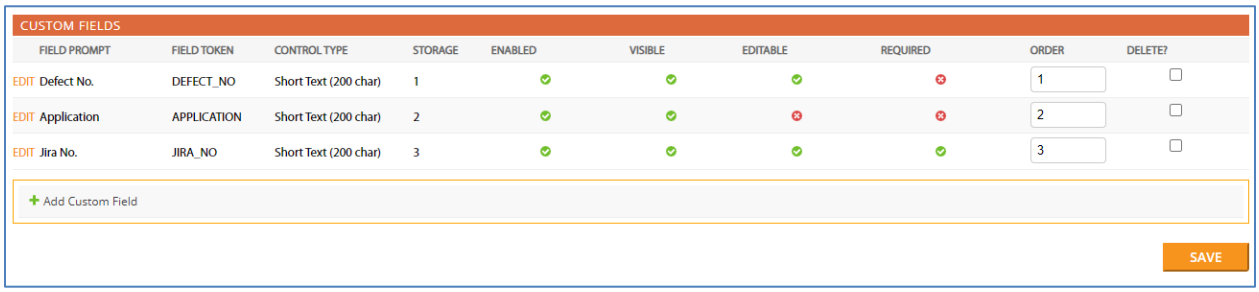

Fig. 12 - The custom fields screen

While editing or adding a custom field, the following required and optional fields are available for data entry:

Prompt	Description
Field Prompt	The visible label of the field as seen on the environment screen
Field Token	A unique code for the field, which may be referenced in commands
Description	Optional brief description of the purpose of the field
Control Type	The type of control used to populate this field
Storage	The database column used to house the content of this field. Choose the default 'Auto' to have Emoveo use the first available column. Please note that for every custom field value entered, Emoveo captures internally in its database two values, hidden and visible.

Validation Code	For controls of type ‘Drop Down (SQL)’ and ‘Autocomplete (SQL)’ enter an Oracle SELECT statement to be dynamically executed when the environment screen is launched. The SELECT statement must retrieve a hidden value followed by a visible value. For controls of type ‘Drop Down (List)’, enter one choice per line, with the hidden and visible columns separated by a colon
Hidden Default	The default field value for controls of a limited list selection
Enabled	Whether to put the custom field to use throughout Emoveo
Visible	Whether to display the custom field
Editable	Whether to allow users to edit the contents of the field
Required	Whether to require users to input a value into the field

Table 13 - Fields of custom fields

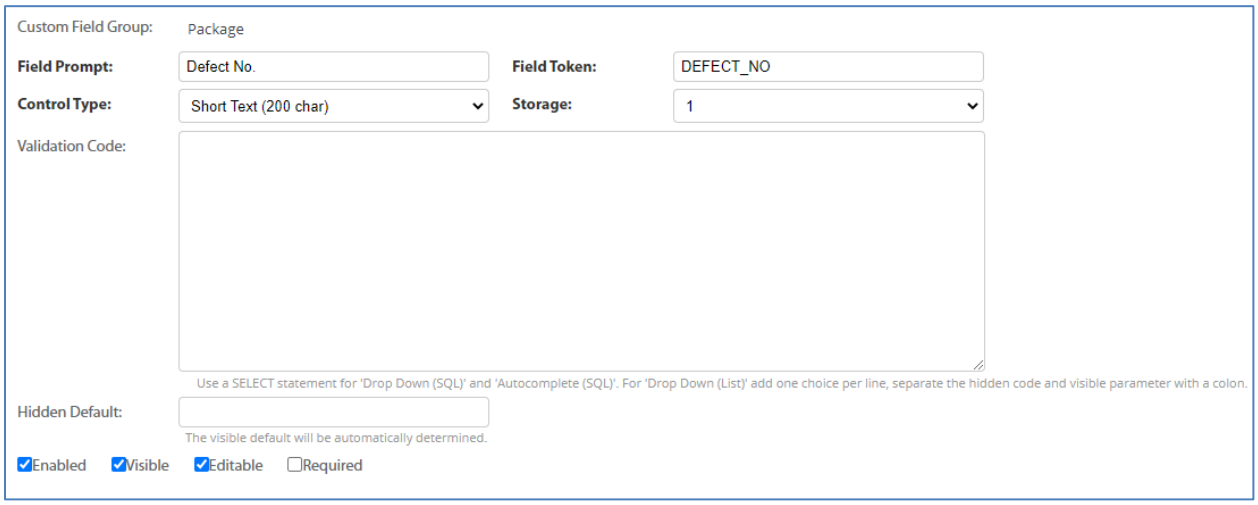

Fig. 13 - Editing custom field information

9. Tokens and the Token Builder

When configurators create objects such as object types or special commands, they often utilize tokens to reference variables which are populated with values in specific contexts. Tokens are coded using a specific format, which must be adhered to for Emoveo to properly process them:

1. Tokens are always enclosed in brackets and written in capital letters.
2. All tokens are prefixed with their category (typically three letters).
3. The category portion of the token may then contain a parameter value which limits the token's coverage to a specific object and designated through an equals symbol with the parameter's value in double quotes.
4. Some tokens are then followed by a sub-category designation, enclosed in dots.
5. Tokens representing custom fields have their category designation be followed by the letter F enclosed in dots.
6. All tokens are then followed by the name of the variable referenced within the context defined in the previous steps.

Examples:

1. To reference the client base path of the environment "EBSPRD", use:

   [ENV="EBSPRD". CLIENT_BASE_PATH]

   - The ENV represents the token's category, environments in this case.
   - The token's category is then followed by a parameter value, in this case the code of a specific environment.
   - Finally, the actual variable name within the specific context defined.

2. To reference the custom package field "Defect No.", use:

   [PKG.F.DEFECT_NO]

   - The PKG represents the token's category, packages in this case.
   - The token's category is then followed by a custom field designation.
   - Finally, the token of the custom field is used.

To simplify the process of utilizing tokens, Emoveo includes the token builder tool, which eliminates the need to remember the specifics of the token language. To access the token builder, go to the menu item Configure -> Token Builder.

The token builder contains the following elements:

1. The token's category and sub-category are selected from the left side of the screen.
2. The token details section which displays, and enables copying of, the constructed token based on the configurator's selections.
3. If the selected token supports parameters, Emoveo enables the user to select a specific parameter value in the token details section.

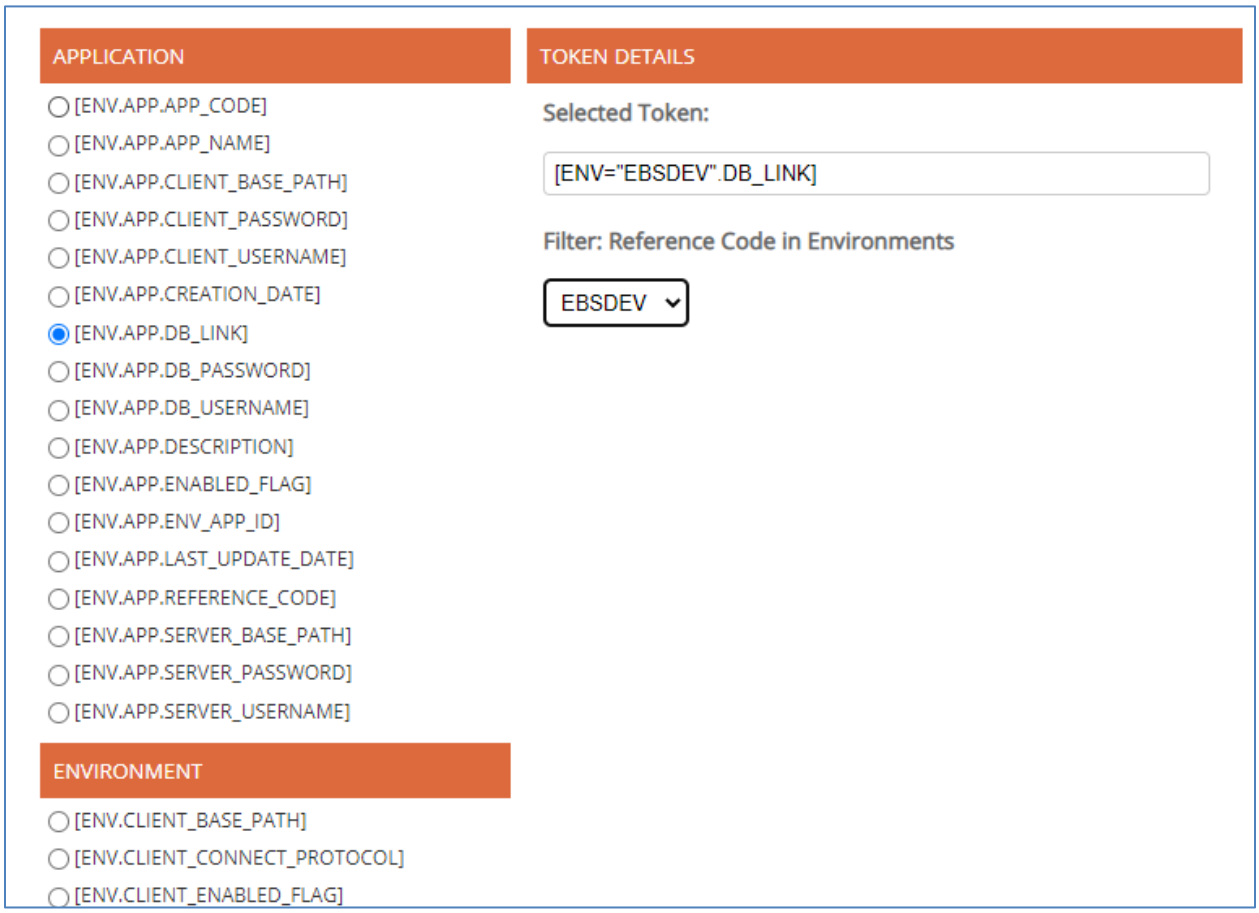

Fig. 14 - The token builder screen