

Data – Oracle (MDA)

user guide V0.1



Contents

[1. Introduction 3](#_Toc23932885)

[2. Prerequisites 3](#_Toc23932886)

[3. Configuration 4](#_Toc23932887)

[4. Environment Variables 5](#_Toc23932888)

[5. Using the Business Object 5](#_Toc23932889)

[5.1. Configure 5](#_Toc23932890)

[5.2. Set Connection 5](#_Toc23932891)

[5.3. Begin Transaction 6](#_Toc23932892)

[5.4. Commit Transaction 6](#_Toc23932893)

[5.5. Rollback Transaction 6](#_Toc23932894)

[5.6. Execute 7](#_Toc23932895)

[5.7. Get Number 7](#_Toc23932896)

[5.8. Get Text 8](#_Toc23932897)

[5.9. Get Collection 8](#_Toc23932898)

[5.10. Get CSV File 9](#_Toc23932899)

[5.11. Get CSV 9](#_Toc23932900)

[5.12. Get User Name 9](#_Toc23932901)

[5.13. Get Machine Name 10](#_Toc23932902)

[5.14. Delete Rows - Where 10](#_Toc23932903)

[5.15. Drop Table 10](#_Toc23932904)

[5.16. Truncate Table 11](#_Toc23932905)

[6. Support 11](#_Toc23932906)

[7. Functional Tests 11](#_Toc23932907)

[8. Troubleshooting Guidelines 11](#_Toc23932908)

[9. Frequently Asked Questions 11](#_Toc23932909)

The information contained in this document is the proprietary and confidential information of Blue Prism Limited and should not be disclosed to a third party without the written consent of an authorised Blue Prism representative. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying without the written permission of Blue Prism Limited.

**© Blue Prism Limited, 2001 – 2019**®Blue Prism is a registered trademark of Blue Prism Limited

All trademarks are hereby acknowledged and are used to the benefit of their respective owners.  
Blue Prism is not responsible for the content of external websites referenced by this document.

Blue Prism Limited, Centrix House, Crow Lane East, Newton-le-Willows, WA12 9UY, United Kingdom  
Registered in England: Reg. No. 4260035. Tel: +44 870 879 3000. Web: [www.blueprism.com](file:///C:\Users\adutton\Documents\Rebranding\Templates\www.blueprism.com)

# Introduction

This asset is designed to provide connectivity from Blue Prism to the Oracle RDBMS using the ODP.NET (MDA) managed data access library. This VBO is a code wrapper around this library and this means that the stages have C# code that undertakes the actual work. Unless you wish to modify or add functionality to the VBO C# coding experience is not needed to make use of this VBO.

# Prerequisites

To use this asset, you require a licenced Blue Prism installation or trial. More information on Blue Prism can be found here.

<https://www.blueprism.com>

You will also require the Managed Data Access provider mentioned previously. The file that is required is **Oracle.ManagedDataAccess.dll**

This is available from our GitHub repository at <https://github.com/blue-prism/oracle-mda-utility> . Here you will find the release file, the Oracle .NET dll and documentation. Please note that the oracle .net dll, is a .net library and experience with coding in .net is not a prerequisite for the usage of this VBO.

The asset is created as a *bprelease* file, so that it may be reviewed and edited as you find necessary. It is maintained on a best endeavours’ basis by *Blue Prism* however, user feedback is appreciated, and this can be in relation to issues or errors found and feature requests.

# Configuration

To use this asset, import the ***.bprelease*** file which contains the following:

* Data – Oracle (MDA)
* Oracle MDA Test (Process)

Now the Oracle.ManagedDataAccess.dll file needs to be placed into the Blue Prism folder, which on our development machine is at:

C:\Program Files (x86)\Blue Prism Limited\Blue Prism Automate

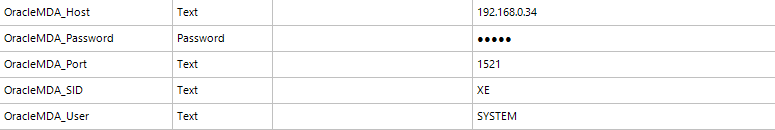
It will also be necessary to ensure that the Blue Prism development environment has access to this file for the execution of the stages the VBO contains. This is as follows:

Do note that the version number embedded in the file title may be different to what you have. This will depend on the maintenance cycle that Oracle have for this product. The above is necessary so that the code stages used can locate the code that they refer to. If your configuration is different in that Blue Prism is installed in a different location, it will be necessary to remove the Oracle ManagedDataAccess.dll entry in the External References section above and add a new one, making use of the Browse button to locate the file on your installation. Note that the namespace imports for *Oracle.ManagedDataAccess.Client* can be left as it is. This name is unlikely to change.

# Environment Variables

In order to simplify the usage of the asset, we have made available some environment variables that can be used in your processes.

These can be found in the System Tab, in the Processes/Objects group.



# Using the Business Object

The Visual Business Object contains the following actions:

## Configure

This action will allow a flag to be set to control the reporting of exceptions. Useful during development of a process.

Inputs:

| Name | Description | Data Type |
| --- | --- | --- |
| RaiseExceptions | This is a flag that sets whether exceptions are reported. | Flag |

## Set Connection

Create the connection string that is necessary to connect to the Oracle database.

Inputs:

| Name | Description | Data Type |
| --- | --- | --- |
| ConnectionName | This is the name of the Oracle installation or SID. | Text |
| Hostname | This is the IP address of the Oracle host. | Text |
| Port | This is the listening port that the Oracle listening service is observing. (Default 1521) | Text |
| Username | The username by which you will access the Oracle database system | Text |
| Password | The supplied password by which you will access the Oracle database system. | Password |

Outputs:

| Name | Description | Data Type |
| --- | --- | --- |
| ConnectionString | This is connectionstring that is used by the code stages of the VBO to access the Oracle database. It is stored in a data item on the Initialise page. | Text |

## Begin Transaction

This stage allows a set of database interactions to be encapsulated in a transaction. This allows a set of interactions to be completed or aborted if conditions are not met. Begin Transaction is followed by a Commit Transaction or a Rollback Transaction.

Outputs:

| Name | Description | Data Type |
| --- | --- | --- |
| Success | The flag indicating if the transaction was started correctly. | Flag |
| Message | The text message that holds any error information that may have occurred. | Text |

## Commit Transaction

This stage compliments the Begin Transaction stage and makes all the changes made to the database since the transaction began permanent.

Outputs:

| Name | Description | Data Type |
| --- | --- | --- |
| Success | The flag indicating if the transaction was committed correctly | Flag |
| Message | The text message that holds any error information that may have occurred. | Text |

## Rollback Transaction

This stage compliments the Begin/Commit Transaction as it allows the database to be returned to the stage prior to the commencement of the Begin Transaction. This would be the case should appropriate conditions or an exception occur.

Outputs:

| Name | Description | Data Type |
| --- | --- | --- |
| Success | The flag indicating if the transaction was rolled back correctly | Flag |
| Message | The text message that holds any error information that may have occurred | Message |

## Execute

This stage takes an SQL command and passes it to the database.

Inputs:

| Name | Description | Data Type |
| --- | --- | --- |
| Query | A valid Oracle SQL command script. | Text |

Outputs:

| Name | Description | Data Type |
| --- | --- | --- |
| Success | The flag indicating if the command was successful. | Flag |
| Message | The text message that holds any error information that may have occurred | Text |
| RecordsAffected | A numeric value detailing the number of records that the command parameter had an action upon. | Number |

## Get Number

This function executes an SQL script and returns a single numeric value.

Inputs:

| Name | Description | Data Type |
| --- | --- | --- |
| Query | A valid Oracle SQL command script. | Text |

Outputs:

| Name | Description | Data Type |
| --- | --- | --- |
| Result | The value returned from the supplied SQL query. | Number |
| Success | The value to explain if the query was successful. | Flag |
| Message | The exception message, if the query failed. | Test |

## Get Text

This function executes an SQL script and returns a text result.

Inputs:

| Name | Description | Data Type |
| --- | --- | --- |
| Query | A valid Oracle SQL command script. | Text |

Outputs:

| Name | Description | Data Type |
| --- | --- | --- |
| Result | The value returned from the supplied SQL query. | Text |
| Success | The value to explain if the query was successful. | Flag |
| Message | The exception message, if the query failed. | Text |

## Get Collection

This function executes a SQL *Select* statement and returns the resulting records in a Blue Prism collection data item.

Inputs:

| Name | Description | Data Type |
| --- | --- | --- |
| SelectQuery | A valid Oracle SQL select command script | Number |

Outputs:

| Name | Description | Data Type |
| --- | --- | --- |
| Result | The resulting collection data item containing the results of the select query. | Number |
| Success | The value to explain if the query was successful. | Flag |
| Message | The exception message, if the query failed. | Text |

## Get CSV File

This function takes a valid SQL Select query and returns the results to a CSV (Comma Separated Variable) file.

Inputs:

| Name | Description | Data Type |
| --- | --- | --- |
| SelectQuery | A valid Oracle SQL select command script. | Number |
| CSVFilename | The destination filename for where the results will be written to. | Text |

Outputs:

| Name | Description | Data Type |
| --- | --- | --- |
| Success | The value to explain if the query was successful. | Flag |
| Message | The exception message, if the query failed. | Text |

## Get CSV

This function takes a valid SQL Select query and returns the results to a string in comma separated format.

Inputs:

| Name | Description | Data Type |
| --- | --- | --- |
| SelectQuery | A valid Oracle SQL select command script. | Number |

Outputs:

| Name | Description | Data Type |
| --- | --- | --- |
| CSV | The text value containing the resulting data. | Text |
| Success | The value to explain if the query was successful. | Flag |
| Message | The exception message, if the query failed. | Text |

## Get User Name

This function returns the currently logged on user. This is an environment value and not a value from Oracle.

Outputs:

| Name | Description | Data Type |
| --- | --- | --- |
| User Name | The user name of the currently logged on user. | Text |

## Get Machine Name

This function returns the value of the current machine name where the user is logged on to. This is not a value from Oracle.

Outputs:

| Name | Description | Data Type |
| --- | --- | --- |
| Machine Name | The resulting degree angle value. | Text |

## Delete Rows - Where

This function will delete a set of records from a table where the where clause is satisfied. This function requires some database schema knowledge.

Inputs:

| Name | Description | Data Type |
| --- | --- | --- |
| TableToDeleteFrom | The table where the rows are to be removed from. | Text |
| WhereClause | The Boolean clause that must be satisfied. |  |

Outputs:

| Name | Description | Data Type |
| --- | --- | --- |
| Success | The value to explain if the query was successful. | Flag |
| Message | The exception message, if the query failed. | Text |

## Drop Table

This function takes table name and executes an SQL statement to drop the table. This is a destructive action. This function requires some database schema knowledge.

Inputs:

| Name | Description | Data Type |
| --- | --- | --- |
| TableToDrop | The table that is to be removed from the database schema. | Number |

Outputs:

| Name | Description | Data Type |
| --- | --- | --- |
| Success | The value to explain if the query was successful. | Flag |
| Message | The exception message, if the query failed. | Text |

## Truncate Table

This removes the complete set of rows in a table. This function requires some database schema knowledge.

Inputs:

| Name | Description | Data Type |
| --- | --- | --- |
| TableToTruncate | The table that is to have all its rows removed. | Text |

Outputs:

| Name | Description | Data Type |
| --- | --- | --- |
| Success | The value to explain if the query was successful. | Flag |
| Message | The exception message, if the query failed. | Text |

# Support

Support for this skill is provided via the Blue Prism Digital Exchange Community Forum. Post your questions here:

[Digital Exchange Community Forum](https://community.blueprism.com/communities/community-home?communitykey=1e516cfe-4d1f-4de9-a9eb-58d15bf38c81&tab=groupdetails)

# Functional Tests

There is a simple test process for each of the functions however the user should satisfy themselves that the results are both correct and satisfactory for their needs. Usage of these functions reflects that satisfaction. To make use of these tests, locate the Oracle MDA Test process and open it in process studio. Enter your values in the appropriate areas or make use of the provided environment variables. Your parameter areas are in a yellow block. Click on the start stage and then right click your mouse, from the drop-down choose *Set Next Stage,* the from the toolbar click the green triangle to run it.

# Troubleshooting Guidelines

There are no known commonly encountered issues at this stage, or corresponding resolutions for them. If users begin encountering issues, then this section will be updated with known resolutions.

It should be noted that this asset was built against the Oracle Express Edition. It has not been tested against any other version neither hosted or cloud.

# Frequently Asked Questions

There are no frequently asked questions at this stage.