



Exor to AgileAssets Maintenance and PMS Interface Installation and User Guide

January 2014



Contents

1.0 INTRODUCTION 3

2.0 DESCRIPTION OF THE PROJECT 3

3.0 USING THE SOLUTION..... 3

 3.1 ROAD NETWORK INFORMATION 3

 3.1.1 View XAA_ROUTE 3

 3.1.2 View XAA_ROUTE_SDO..... 3

 3.1.3 XAA_SPATIAL_AUDIT 4

 3.2 EVENT INFORMATION 4

 3.3 ASSET INFORMATION 6

 3.4 REPORTS..... 7

4.0 INSTALLATION AND REMOVAL 8

 4.1 INSTALLATION 8

 4.2 REMOVAL..... 8

5.0 SUMMARY AND CONCLUSION 8

CONFIDENTIALITY STATEMENT

The contents of this document, including system ideas and concepts, are confidential and proprietary in nature and are not to be distributed in any form without the prior written consent of Bentley Systems

1.0 Introduction

The purpose of this document is to provide the Kentucky Transportation Cabinet (KYTC) with support documentation for the Exor to AgileAssets Maintenance and PMS Interface project. This document is intended to explain how to install and use the various objects required in this project. This document does not attempt to describe what the objects should do. For that information see the current version of: “Exor to Agile Assets interface Specification” or “Draft Exor to Agile Assets interface Design.” This document should be read in its entirety prior to installing and running the objects related to the Exor to AgileAssets Maintenance and PMS Interface project.

2.0 Description of the project

Bentley Systems has developed a set of tools that utilizes both existing functionally and custom oracle objects to provide a solution to meet the requirements discussed in the Specification and Design documentation mentioned above. Various Oracle objects were created to support the following items: Road Network Information, Event Information, Asset Information and Reports. Since AgileAssets will be interacting with some of these objects, a new schema was created to interact with. The name of this schema is exor_to_aa and is defined with the default password of exor_to_aa.

3.0 Using the solution

As mentioned above the solution created by Bentley Systems can be broken into several sections.

3.1 Road Network Information

The solution needs to be able to provide a list of the current version of the road system. This list needs to have road sections break at distance break points and roll-up continuous sections. A history of changes to route spatial objects also needs to be maintained and reported on.

3.1.1 View XAA_ROUTE

Type:	View
Location:	Exor Schema, Alias created for exor_to_aa
Ran Via:	Select Statement
Input Accessed	None
Output Accessed	Select Statement of XAA_ROUTE
Related Objects	None

This view provides the lists of the current version of the road system.

The View XAA_ROUTE can be accessed via the command “select * from XAA_ROUTE” The SQL can be modified to provide information only for a particular route by adding filters to the query.

3.1.2 View XAA_ROUTE_SDO

Type:	View
Location:	Exor Schema, Alias created for exor_to_aa
Ran Via:	Select Statement
Input Accessed	None
Output Accessed	Select Statement of XAA_ROUTE_SDO
Related Objects	None

This view contains spatial data and provides the list of the current version of the road system’s geometry.

The View XAA_ROUTE_SDO can be accessed via the command “select * from XAA_ROUTE_SDO” The SQL can be modified to provide information only for a particular route id by adding filters to the query.

CONFIDENTIALITY STATEMENT

The contents of this document, including system ideas and concepts, are confidential and proprietary in nature and are not to be distributed in any form without the prior written consent of Bentley Systems

3.1.3 XAA_SPATIAL_AUDIT

Type:	Table
Location:	Exor Schema, Alias created for exor_to_aa
Ran Via:	Select Statement
Input Accessed	None
Output Accessed	Select Statement of XAA_SPATIAL_AUDIT
Related Objects	xaa_spatial_audit_trg

The XAA_SPATIAL_AUDIT table contains the adds and deletes as a result of changes to the spatial representation of a route.

The XAA_SPATIAL_AUDIT table can be accessed via the command "select * from XAA_SPATIAL_AUDIT"
The SQL can be modified to provide information only for a particular route id or date by adding filters to the query.

3.2 Event Information

The AgileAssets Systems needs a way to find out if the location of objects stored in the system has changed due to a network operation on the road network.

Type:	Procedure
Location:	Exor Schema, Alias created for exor_to_aa
Ran Via:	exec xky_hig_to_aa.process_route_events;
Input Accessed	Table: XAA_LOC_IDENT
Output Accessed	Table: XAA_NET_REF
Related Objects	xaa_route, xaa_route_all, xaa_route_temp_sql

The procedure process_route_events allows AgileAssets to query the exor system to see if any routes have changed from the date that it has in its system.

To you this procedure the table XAA_LOC_IDENT needs to have the following information filled in:

HISTORIC_DATE	Date	Date of the locations coming from the AgileAssets System
LOC_IDENT	INTEGER	AgileAssets Location ID #
ROUTE_NAME	VARCHAR2 (30 Byte)	Route Name in the form of an Exor System UNIQUE
OFFSET_FROM	NUMBER (22,4)	From Milepoint
OFFSET_TO	NUMBER (22,4)	To Milepoint
SOURCE_TABLE	VARCHAR2 (32 Byte)	Raw Data Table Name - Informational Item that lists our source data table name

CONFIDENTIALITY STATEMENT

The contents of this document, including system ideas and concepts, are confidential and proprietary in nature and are not to be distributed in any form without the prior written consent of Bentley Systems

Once this information is filled in the process xky_hig_to_aa.process_route_events is executed. This process will clear the XAA_NET_REF table and then repopulate it with the following additional information:

NEW_DATE	Date	Date of the locations coming from the Exor System (filled by Exor)
NEW_ROUTE_NAME	VARCHAR2 (30 Byte)	New Route Name (filled by EXOR) If the route is closed or invalid this value can be null
NEW_OFFSET_FROM	NUMBER (22,4)	New From Milepoint (filled by EXOR) If the route is closed or invalid this value can be null
NEW_OFFSET_TO	NUMBER (22,4)	New To Milepoint (filled by EXOR) If the route is closed or invalid this value can be null
PROCESS_MSG	VARCHAR(100)	Messages and Errors produced during the process. Identified possible errors are: <ul style="list-style-type: none">• Old Route Location Not Found• Route Location Not Changed• Route Location Changed• Route Location Closed

There are four possible process messages:

- Old Route Location Not Found
 - This can occur if the input offsets never match historic Exor values or if the input offsets were never continuous, i.e. there was a gap in the system in between those points.
- Route Location Not Changed
 - If the route location offsets and name have remained the same, this message will be generated. It is possible that the date has changed though.
- Route Location Changed
 - This message is generated if the name changes or if the mile points of the offsets are no longer the same as the input offsets.
- Route Location Closed
 - If the route location has been end dated in the Exor system then this process message is displayed.

It is up to the user to decide whether or not this process is ran manually or via a job scheduler.

CONFIDENTIALITY STATEMENT

The contents of this document, including system ideas and concepts, are confidential and proprietary in nature and are not to be distributed in any form without the prior written consent of Bentley Systems

3.3 Asset Information

This procedure allows the user to create tables of Inventory items with only desired attribute data. Like the road network the route is segment to break on distance breaks or attribute data changes.

Type:	Procedure
Location:	Exor Schema, Alias created for exor_to_aa
Ran Via:	As HIG_ADMIN: exec xky_hig_to_aa.generate_asset_info;
Input Accessed	Table: EXOR.XAA_ASSET_ATTRIB, EXOR.XAA_ASSET_TYPE
Output Accessed	Various Tables with select granted to exor_to_aa
Related Objects	

To use is procedure, the two configuration tables need to be populated. The first is XAA_ASSET_TYPE, this table has 3 columns to configure. The procedure uses this table to know which asset to look for in the XAA_ASSET_ATTRIB table, what the output table name should be and what route type to run the asset against.

Column	Null	Data Type	Comment
ASSET_TYPE	N	VARCHAR2(4)	The four letter code for the asset type.
TABLE_NAME	N	VARCHAR2(30)	The name of the table that will hold the resulting data
ROUTE_TYPE	N	VARCHAR2(4)	The group type that the asset should be reported on. For KYTC this will always be 'RT'

The second table XAA_ASSET_ATTRIB is used to tell the procedure which attributes to include in the output table, what order the attributes should be in and what name to give that attribute in the output table.

Column	Null	Data Type	Comment
ASSET_TYPE	N	VARCHAR2(4)	The four letter code for the asset type.
COLUMN_SEQ	N	NUMBER	Order of the columns in the table
COLUMN_NAME	N	VARCHAR2(30)	Name of the column name in the table defined by XAA_ASSET_TYPE.TABLE_NAME
COLUMN_DERIVATION	N	VARCHAR2(80)	An asset 'VIEW_COLUMN' attribute value or a function using a 'VIEW_COLUMN' attribute or a simple function or a literal value. This information is used in a select statement from the view V_NM_XXXX_NW where XXXX is the four letter asset code.

After the initial install these tables are populated with the attribute information located in appendix A of the Interface Design Document.

To access the output use a select statement to query the table names that was inputted into the xaa_asset_type table.

It should be noted that this procedure needs to be ran as the hig_admin (KTYC username is EXOR), if the permissions and aliases for the output tables are to be created for exor_to_aa. Running it as exor_to_aa will cause the output tables to be empty due to internal exor security.

CONFIDENTIALITY STATEMENT

The contents of this document, including system ideas and concepts, are confidential and proprietary in nature and are not to be distributed in any form without the prior written consent of Bentley Systems

3.4 Reports

A report is needed so that staff can review what road network changes have occurred and check that the appropriate changes have been applied.

Type:	Procedure
Location:	Exor Schema, Alias created for exor_to_aa
Ran Via:	exec xky_hig_to_aa.generate_report_info;
Input Accessed	none
Output Accessed	XAA_LENGTH_CHANGE
Related Objects	XAA_SPATIAL_AUDIT

When ran this procedure processes length changes since the last date that it was ran. The last date is determined from the maximum date found in the column xaa_length_change.len_change_process_date. All dates used in this procedure are truncated so the time would be 00:00 for each date used.

This procedure has two optional parameters:

- Last run date override
 - Type: date
 - Default: null
 - This parameter allows the user to manually enter when the changes should be processed from. This is extremely useful during the testing phase.
 - It is highly recommended that this parameter be filled to create a last run date in the xaa_length_change table. Otherwise, the procedure will process changes from 01-JAN-1950 and may take a long while to complete.
- Process spatial changes
 - Type: Boolean
 - Default: true
 - This parameter allows the user to disable processing of the spatial audit table. It is mostly just useful during the initial testing to only look at the network changes.

Example of forced last run use: `exec xky_hig_to_aa.generate_report_info(to_date('01-DEC-2013'),true);`

This will process all changes from December 1st, 2013 and process the special data.

Normal use would be: `exec xky_hig_to_aa.generate_report_info;`

This will process all changes from the last date ran and process the special data.

To get only the latest results:

```
select * from xaa_length_change
where report_run_date= (select max(report_run_date) from xaa_length_change);
```

This will show only the results from the last time the procedure was run.

CONFIDENTIALITY STATEMENT

The contents of this document, including system ideas and concepts, are confidential and proprietary in nature and are not to be distributed in any form without the prior written consent of Bentley Systems

4.0 Installation and Removal

After unpacking the installation package into its own directory. We recommend doing this in your exor\stage directory with a new directory created with the name exor_2_aa. The user will see the following directory structure:

- admin
- docs
- installs

The admin directory is where all the scripts are kept for the installer to use.

The docs directory contains this document and any related end-user material.

The installs directory contains scripts for installing and removing the product.

4.1 Installation

From a command window go into the installs directory. Start SQLPLUS as EXOR and use the command:
start exor_to_aa_install.sql

From a command window go into the installs directory. Start SQLPLUS as a user that can create table spaces and users; then use the command:

```
start exor_to_aa_install_sys.sql
```

This will install the solution and configure the input tables for the Asset Information and create the EXOR_TO_AA schema.

4.2 Removal

From a command window go into the installs directory. Start SQLPLUS as a user that can remove table spaces and users; then use the command:

```
start exor_to_aa_uninstall_sys.sql
```

From a command window go into the installs directory. Start SQLPLUS as EXOR and use the command:
start exor_to_aa_uninstall.sql

It is recommended that custom changes made to the report configuration tables be exported before removal is ran, especially if the user plans to reinstall this solution.

5.0 Summary and Conclusion

The solution created by running the accompanying script will meet the needs outlined in the Specification and Design documentation. Since, depend items are only views or a table populated by a trigger, all parts of the solution can be ran as needed without worrying about co-dependencies.

CONFIDENTIALITY STATEMENT

The contents of this document, including system ideas and concepts, are confidential and proprietary in nature and are not to be distributed in any form without the prior written consent of Bentley Systems