

Network Manager

**Fix Release Notes**

4.7.0.0 Fix 55

Table of Contents

2. Introduction 2

3. Fix Details 2

4. Log No. Summary 3

5. List of New and Amended Files 7

6. Notes on the use of Oracle pathing tools 10

7. Notes on reversible XSPs 12

7.1 LB\_XRPT 12

7.2 Standard Exor XSP model 12

7.3 Location Bridge Asset XSPs 13

7.4 XSP and Derived Geometry 14

7.5 XSP Retrieval and Transformation 14

7.6 Examples relating to XSP 14

7.6.1 Available XSPs over an extent of network. 14

7.6.2 XSP Retrieval and Transformations 15

8. Materialized view of asset locations 17

9. Notes on units 18

9.1 Unit conversion of table functions 18

9.2 Unit compatibility with Oracle Spatial data 18

9.3 Unit conversion improvement 18

9.4 Units and rounding 18

9.5 Changes to unit registration procedure 18

10. Snapping or Net-Coding 19

# Introduction

This document defines the changes made to the Location Bridge component of the Network Manager product. It is bundled as Fix 55 for 4.7.0.0. This incorporates all the changes that were issued in LB 4.2 (exnm04070001en\_updt47), 4.3 (exnm04070001en\_updt48), 4.4 (exnm0407001en\_updt49), 4.5 (exnm040007001en\_updt51) and is intended to be installed directly after the 4.1 upgrade if required. It is also intended to be installed on Location Bridge which has already been upgraded to 4.2, 4.3, 4.4 or 4.5 through the fixes numbered above. From scratch, LB should be installed from the lb\_v4\_1\_1 zip file and immediately updated using this fix.

From Build 4 onwards of this fix also contains a SQL file which will de-install the LB objects. The script is drop\_lb.sql and can be found under the admin\util folder.

Build 10 of fix 55 (exnm04070010en\_updt55) includes changes to the method of computing paths for the generation of linear locations. These changes are highlighted in their own section.

Build 12 of the fix (exnm04070012en\_updt55) includes change to allow retrieval and reversal of relative XSPs and again, these changes are highlighted in their own section.

Builds 13 and 14 include some bug fixes and changes to the units.

A script to materialize all asset locations relative to the datums on which they are persisted and aggregated over all their potential route systems is included under the \admin\views sub-folder. The inclusion of this materialized view is currently optional and so it is not installed automatically. See the relevant chapter below.

The upgrade contains procedures that match the usual network edit procedures that are available in an Exor system. Note that to keep LB asset locations in-sync with Exor network edits,( build 2 or later) of the core fix 56 must be installed. A warning is issued if this is not the case.

After reading through this document, should you have any further training or consultancy requirements then please contact your Bentley account manager.

Please ensure that all listeners, map servers, scheduler processes and dbms\_jobs are disconnected prior to the installation of this fix.

# Fix Details

|  |  |
| --- | --- |
| Fix Details Baseline Release | 4.7.0.0 |
| Fix Description | This fix provides changed and improved functionality in the Location Bridge component of Network Manager. |
| Prerequisites | NM3 at version 4.0.7.1 and Location Bridge version 4.1, 4.2, 4.3, 4.4 or 4.5. Systems that utilise the LB modules may also need to access Exor data through a proxy user and may therefore have dependencies on the series of fixes which allow the setting of context variables to provide proxy user security, namely 4.7 fixes 29 and its pre-requisites. Network edits have an operational dependency on build 2 of fix 56 although LB will compile if this is not present. |
| Implementation Instructions | Ensure that the system is not in use before upgrading with this fix release.  The staging folder is the location of the folder that exnm04070010en\_updt55.zip was extracted to (the folder containing this readme).  Log onto SQL\*PLUS as the Highways Owner with the staging folder as the working directory.  At the prompt type START exnm04070010en\_updt55.sql and press return.  Exit SQL\*Plus |
| Limitations | None known |
| Configuration Information | None |
| How To Test | Tests to be performed from client ARS modules |
| Rollback Strategy | Initially implement on a test environment |

# Log No. Summary

This chapter summarises all software issues that have been addressed by this fix.

For issues raised by users, Bentley Technical Support Group (TSG) Service Request Numbers are cross referenced where applicable.

|  |  |  |
| --- | --- | --- |
| Details | Internal Reference | TSG Service Request |
| Change v\_network\_types to include networks without locatable asset types | Enhancement 525376  Task 525896 |  |
| Location based predicates | Enhancement 316415 |  |
| Failure in route-based aggregation | Defect 433726 |  |
| Location Bridge query relative to route fails to operate with NULL measures | Defect 563403 |  |
| Problems in computation of aggregated geometry | Task 188739 |  |
| Problems with NULL aggregated geometry | Task 527600 |  |
| Add validation of measures on the load-by linear range function | Defect 524768 |  |
| Location Bridge generates incorrect results in the set operation MINUS (LB\_OPS.RPT\_MINUS) | Defect 563409 |  |
| Get whether an on-network asset type is Point or Continuous | Enhancement 316426 |  |
| Corrections and various improvements in the registration/de-registration of asset-types. | Defect 563412 |  |
| Enhancement to provide list of possible XSPs over a network location | Enhancement 523312 |  |
| Enhancement to provide a flag to indicate if an asset-type is point or continuous in respect of its location. The enhancement includes a view which can provide further flags relating to the asset-type and the network. | Enhancement 524308 |  |
| Added function to retrieve lists of XSPs for an asset and location | Enhancement 592172 Task 592173 |  |
| Problems with unit translations on route to datum conversions and vice versa | Defect 590818 |  |
| Modified range query to include the flag to return locations that are wholly within the search group or range |  |  |
| Minor performance issue in the search for assets over a group of groups. | Defect 568113 |  |
| Improvement on updates to JXP (allows the setting to NULL) |  |  |
| Added contiguity check |  |  |
| Added some exceptions to cater for re-registration of same asset types - preventing failure. |  |  |
| Unit translations can fail when units are the same. SQL accesses nit conversions where no data is found. | Defect 615163 |  |
| Load locations fails with an Oracle error ORA-30625: method dispatch on NULL SELF argument is disallowed | Defect 614192 |  |
| Load locations fails with an Oracle error ORA-01858: a non-numeric character was found where a numeric was expected | Defect 614128 |  |
| Location Bridge unit translation on load gives problems as used in createlinearange | Defect 614127 |  |
| Problems in use of an outer-join on exor unit translations | Defect 615231 |  |
| Range queries with the whole-only flag set do not work with different unit systems | Defect 592678 |  |
| LB allows registration of asset type on group-based network. | Defect 645688 |  |
| LB allows loading of empty location | Defect 645253 |  |
| Lateral offset is not computed on the aggregated geometry | Defect 626149 |  |
| LB module to supply list of available XSPs over a linear range will break on sub-class variations. | Defect 615198 |  |
| investigate and remove occurrences of combination geometries | Task 524553 |  |
| LB Registration silently ignores non-datum Network Types | Defect 528777 |  |
| Save Continuous Linear Location gives point/line reference error | Defect 670669 |  |
| Dynamic SQL formatting issue encountered in an RTD instance so path geometry failed to compute | Defect 675217 |  |
| Add UNIQUE constraint to EXOR\_UNIT\_ID column of LB\_UNITS table | Enhancement 675816 |  |
| JXP integrity | Defect 691141 |  |
| Theme removal on the de-installation of LB | Defect 691172 |  |
| Add the asset location start and end dates on to returned ref-cursors in selected functions (GetAssetLinearLocations and GetNetworkLinearLocations) | Enhancement 691212 |  |
| Add the asset location start and end dates on to returned table data in functions GetAssetLinearLocationsTtab and GetNetworkLinearLocationstab | Enhancement 691212 |  |
| Synonyms to eB Interface functions are missing in the original install scripts. These synonyms are corrected in the build 8 (onwards) of this fix. It works around the TFS entry shown but does not solve the issue in the original install script. This problem is fixed within the original install script which is available outside of this fix. | Defect 734037 |  |
| Problem in merge of location data in cases where location is wholly contained in second of two elements | Defect 740329 |  |
| Create an asset view which is based on the aggregated geometry of a specific asset type | Defect 741236 |  |
| Problem in the aggregation of offset linear geometries due to incorrect arc-tolerance in the arc-densification process. | Defect 743634 |  |
| Delete asset location API retains some location related data | Defect 741221 |  |
| Registration of Oracle network data can use nodes of the incorrect type | Defect 744554 |  |
| Incorrect order of location data from path. | Defect 672425 |  |
| Unable to compute a continuous network location when the start and end positions are on the same linear element | Defect 759869 |  |
| Unit translations can fail when units are the same | Defect 615163 |  |
| 'Linear type does not support aggregation' error message is thrown when translating references to the original linear type | Defect 732744, 655864 |  |
| ComputeLinearLocation query fails with error ORA-29532  This is caused by having registered metadata and Oracle Network data in a schema different to the Exor Highways owner. Although the upgrade will handle any missing synonyms, the method of generating the objects used in the pathing procedures differs over previous releases so this issue no longer arises. | Defect 670673 |  |
| Add continuous location on ESU Save location throws error - ‘From’ measure greater than the ‘To’ measure | Defect 759545 |  |
| Synonym and object mis-match problems causes difficulty in dropping a network and in the application of build 9 of fix 55 (location Bridge) | Defect 775038 |  |
| Pathing from exact end of single network element fails | Defect 671058 |  |
| Trace from start of first network element to mid-point of directly connected network element fails | Defect 670075 |  |
| Trace from mid-point of first network element to end of directly connected network element incorrect | Defect 670071 |  |
| Trace spanning mid-point of single network element fails | Defect 670065 |  |
| Mid-point to Mid-point traces return incorrect results | Defect 670057 |  |
| Retrieval of preferred location tab fails to return sequence | Defect 774809 |  |
| Installing LB, LB\_UNITS table does not get populated which makes Exor Integration database package deployment fail. | Defect 773923 |  |
| Support for retrieval and aggregation of relative XSPs | Tasks 820337, 820347, 872786, 766168  Feature 775511  Defects 655330, |  |
| Problems in location queries such as get\_obj\_id\_as\_rpt\_tab can fail in cases where object type is a road group type expressed as an Exor Foreign Table | Defect 635058  Task 893775 |  |
| Dropping LB now fails after system options are deployed. Options are now removed prior to the removal of LB in the hig\_products table. | Defect 916707 |  |
| Prevent update of key from sequence generator when key is supplied during insert | Defect 909096 |  |
| Reporting materialized view results in NULL XSP when XSP reversals do not exist | Defect 910054 |  |
| Problem with aggregation on non-exclusive partial linear route type | Defect 913849 |  |
| LB\_GET.GET\_LB\_RPT\_TAB doesn't respect units | Defect 655329 |  |
| Drop LB script can fail if executed immediately after install and upgrade | Defect 917586 |  |
| Location Bridge network edits undo operations can fail due to a constraint violation | Defect 922119 |  |
| Location Bridge network edits can fail when no aggregated geometry is present | Defect 921623 |  |
| Query of point data can fail when point extent data is used | Defect 920068 |  |
| Exor core network editing issues | Defect 922068 |  |
| Location bridge materialized view for reporting hub fails due to a scalar subquery returning multiple rows | Defect 923497 |  |
| Added in-line documentation to some key procedures. Note that this is an ongoing process. | Defect 670082 |  |
| Added further checks in procedures of LB\_REG package | Defect 924307 |  |
| lb\_loc.get\_asset\_location\_tab can produce error due to scalar sub-query returning multiple rows. | Defect 925982 |  |
| LB\_GET.GET\_LB\_RPT\_TAB ignores p\_m\_unit argument | Defect 925148 |  |
| In-line documentation has tabs | Defect 925041 |  |
| lb\_loc.get\_asset\_location\_tab doesn't populate the SEQ\_ID column (unlike lb\_loc.get\_pref\_location\_tab) | Defect 927313 |  |

# List of New and Amended Files

|  |  |
| --- | --- |
| Filename | Version |
| exnm0407015en\_updt55.sql | 1.1 |
| lb\_ref.pkh | 1.9 |
| lb\_ref.pkb | 1.9 |
| lb\_get.pkh | 1.21 |
| lb\_get.pkb | 1.54 |
| lb\_reg.pkh | 1.6 |
| lb\_reg.pkb | 1.15 |
| lb\_load.pkh | 1.9 |
| lb\_load.pkb | 1.31 |
| lb\_ops.pkh | 1.8 |
| lb\_ops.pkb | 1.13 |
| lb\_loc.pkh | 1.5 |
| lb\_loc.pkb | 1.13 |
| lb\_nw\_edit.pkh | 1.1 |
| lb\_nw\_edit.pkb | 1.9 |
| lb\_net\_code.pkh | 1.3 |
| lb\_net\_code.pkb | 1.2 |
| v\_network\_types.sql | 1.3 |
| v\_lb\_directed\_path\_links.vw | 1.0 |
| v\_lb\_path\_links.vw | 1.0 |
| v\_lb\_type\_nw\_flags.vw | 1.0 |
| v\_network\_elements.vw | 1.0 |
| v\_nm\_inv\_on\_network.vw | 1.1 |
| lb\_path.pkh | 1.10 |
| lb\_path.pkb | 1.14 |
| lb\_path\_reg.pkh | 1.2 |
| lb\_path\_reg.pkb | 1.9 |
| create\_nlt\_geometry\_view.prc | 1.5 |
| GetNetworkLinearLocationsTab.fnc | 1.2 |
| GetLinearElementTypes.prc | 1.1 |
| GetAssetLinearLocationsTab.fnc | 1.2 |
| GetAssetLinearLocations.fnc | 1.1 |
| GetNetworkLinearLocations.fnc | 1.1 |
| make\_nw\_from\_lrefs.prc | 1.1 |
| v\_nm\_datum\_themes.vw | 1.2 |
| v\_nm\_element\_xsp.vw | 1.1 |
| v\_nm\_element\_xsp\_rvrs.vw | 1.0 |
| v\_nlt\_xsps.vw | 1.1 |
| v\_nlt\_xsp\_rvrs.vw | 1.0 |
| v\_nlt\_element\_xsps.vw | 1.2 |
| v\_nm\_nlt\_data.vw | 1.0 |
| V\_LB\_PATH\_BETWEEN\_POINTS.vw | 1.6 |
| v\_lb\_rep\_asset\_locations.vw | 1.9 |
| drop\_lb.sql | 1.20 |
| lb\_dependencies.vw | 1.0 |
| najx\_id\_seq\_trg.trg | 1.0 |
| nal\_asset\_locations\_all\_who.trg | 1.1 |
| nal\_jxp\_validtion.trg | 1.0 |
| njxt\_id\_seq\_trg.trg | 1.0 |
| njx\_id\_seq\_trg.trg | 1.0 |
| nm\_asset\_geometry\_all\_trg.trg | 1.0 |
| nm\_location\_geometry\_trg.trg | 1.0 |
| nm\_loc\_id\_seq\_usage\_who.trg | 1.0 |
| nm\_locations\_full.vw | 1.0 |
|  |  |
|  |  |

# Notes on the use of Oracle pathing tools

There follows a short description of the options available to use the Oracle pathing tools.

The methods used to generate the network property graph have changed from those outlined in previous fixes, particularly the methods outlined in the previous builds of fix 55 (<= build 9). In previous versions, the network metadata name was supplied and the network was constructed and persisted in objects described inside the metadata. The new release constructs property graphs for least-cost-path dynamically within Oracle global temporary tables. The whole procedure of generating the graph and using the path should be performed within the same session. This lends itself to dynamic graphs being assembled and disposed of and is not suitable to large-scale networks.

Code within the package lb\_path\_reg is now obsolete and will be removed in a future release. Also, certain code within the lb\_path package such as lb\_path.set\_network is also obsolete and will be retired in a future release. They are retained to avoid compilation issues for the short-term.

The network property graph may be assembled through the spatial intersection of a buffer around a series of linear references. Other methods to generate the graph will be available in the next build.

Since the graph is assembled dynamically over a selected area, the need for a user to be able to specify the network name no longer exists. The network metadata will now be based on a single network named LB\_NETWORK and other network definitions will be removed during the upgrade.

The graph is instantiated over the network through reference to the procedure lb\_path.make\_nw\_from\_lrefs which uses an array of linear references and an optional asset type. If an asset type is supplied, the network graph is constrained to datum types over which the specified asset type may be placed. If the asset type is not specified, all datum network types are possibly included in the graph. A stand-alone version of this procedure has been made available which takes a set of linear references and no asset type. This will be removed in a future build and has been made available for compatibility with a demonstration system.

The image below is a representation of the network defined by the supply of 9 linear references, each of which is converted to a geometry; the series of geometries are assembled into a polyline which gives rise to a buffer from which the network is defined. The diagram shows a path from the first to the last of the supplied linear references. The buffer size defaults to 200 meters but may be modified by changing the value in the system option LBNWBUFFER. Note that this is cached and changes will only take effect within new sessions or after the lb\_path package is recompiled.

It is anticipated that the network graph may be constructed from a variety of sources in future releases such as polygon/grid square and possibly over the full network.



# Notes on reversible XSPs

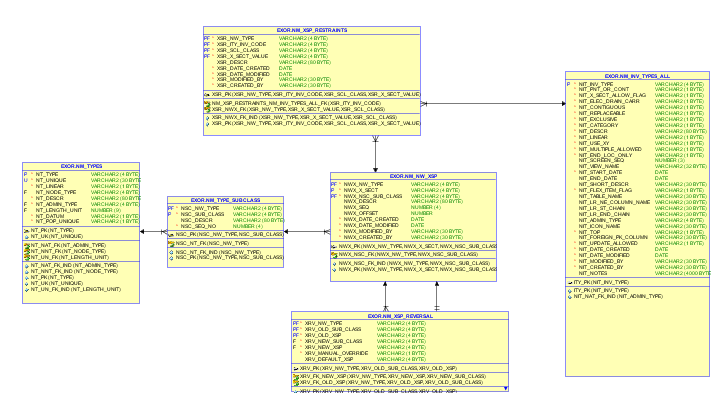
## LB\_XRPT

This upgrade includes a new object definition of LB\_XRPT with a table LB\_XRPT\_TAB to act as a collection mechanism. The data type LB\_XRPT is based on the LB\_RPT object but includes an XSP element and dates. Location data may be extracted into the table representation of this object and may be translated just as the LB\_RPT data type with tools such as aggregation. The data types differ in the way that the XSP is reversed when expressed relative to a route in an opposing direction. See example of retrieval of asset event data with reversed XSP.

The start and end-dates of locations within the object relate to the start and end-dates of the persisted record on retrieval directly from the database or they relate to the minimum start date and maximum end date of individual locations when aggregated relative to a route.

## Standard Exor XSP model

Cross-Sectional Positions or XSPs offer a description of the possible lateral positions for a specified network and sub-class combination. For example, an interstate or state-owned road highway would be very different to a local road. Each of these combinations could be configured to indicate a typical and realistic range of lateral descriptors. There may be differences in the number of lanes and the structure of off-pavement locations such as the verge. XSPs are handled inside the Exor metadata. Each network and sub-class combination may be allowed to support a range of ordered XSP codes. Methods of recording XSPs may be directional such as using descriptors of “Left” or “Right”; these directional indicators are usually expressed relative to the direction of ascending measure of the linear referent. They may be reversed during the reversal of the direction of measure. The reversal of the XSPs is performed according to the business rules created in the NM\_XSP\_REVERSAL table. The standard Exor core XSP model in shown below.



The data for the objects above are generally added from the Exor client modules. The list of available XSPs for each network and sub-class combination is contained in the table NM\_NW\_XSP. Assets that may be placed on a specified network may also be restricted to sub-sets of the available XSPs. This allows asset types such as signs and guard-rails for example to be restricted to those XSPs that are off the pavement surface.

Asset XSP validation is made according to the rules that exist for the network reference relating to the persisted location record. Asset locations are generally persisted as references to a datum network element and in some cases, the datum model may not support a sub-class. Under these circumstances, the sub-class data may be stored under a linear road group and derived at the datum level. The views XSP\_RESTRAINTS and XSP\_REVERSAL provide a description of the XSPs and their reversal values which may be available for a specific asset type across the persisted data and those that may be inherited through linear group membership.

The standard Exor Inventory XSP model allows an asset type to be flagged as having an XSP and this forces a single attribute value to be held on each asset record of the specified type. The asset locations may span multiple datums, which for many route systems may have a direction of ascending measure which either coincides or opposes that of the route. Assets may be placed on the datum directly or through a translation from a route reference and the source referent is not recorded. Hence, it is impossible to know if the XSP was recorded relative to the datum or the route. This means the system cannot provide asset XSPs as a lateral offset since the recording of direction is lost.

## Location Bridge Asset XSPs

Unlike the persisted XSP values in an Exor asset system, Location Bridge holds the XSP and offset value in the location data. LB Assets may span different network and sub-classes so the range of possible XSPs may vary depending on the intended location of the asset. Hence, more sophisticated tools to generate the list of available XSPs over a range of network or over a path are made available. See example of [list of XSPs over an extent of network](#_Available_XSPs_over)

The data design is such that it allows each network sub-class and XSP combination to be supplied with an offset value (in meters) which can be used to provide an approximation to the lateral position of the XSP and assets that are placed there. It can be used as a spatial offset from the road centerline for dynamically segmented geometry. Each XSP has a sequence which indicates the order of the lateral positions.

Also note that the network or path used for the asset location may give rise to multiple sub-classes and network types which in turn give rise to the potential of different offset being used over different segments of the asset location.

## XSP and Derived Geometry

In cases where the XSP is configured with an offset value, any derived geometry will be parallel to the network centreline at a lateral offset equal to the offset value in meters. Negative offset values are generated on the right of the network relative to the direction of increasing measure.

## XSP Retrieval and Transformation

The order of XSPs is quite important as each XSP may be relative to the direction travelled along the network. For example, what may be apparent as a left lane in one direction is a right lane when travelling in the opposite direction. The XSP codes are held on each datum location along with a direction flag which indicates if the XSP was recorded in the direction of the datum ascending measure or the opposing direction. This allows an XSP to be reversed when an asset location is viewed relative to a route which traverses the datum in an opposing direction. There is a dependency on the reversal rules being configured in the Exor metadata. If the reversal rules do not exist, only the direction indicator will be reversed. [See the example on XSP retrieval and transformation.](#_XSP_Retrieval_and)

## Examples relating to XSP

### Available XSPs over an extent of network.

First, it is worth looking at the list of available XSPs for a specific asset type over the extents of networks being used. The extent of network is contained in the LB\_RPT\_TAB nested table. This can be generated using the API to decompose a route-based reference to a datum-based reference. The code below provides the table of XSP s and descriptions that are valid for the asset type XSPT over the extent of network on route 767 between 250 and 600 units.

SELECT \*

FROM TABLE (lb\_ref.get\_xsp\_on\_lb\_rpt\_tab (lb\_get.get\_lb\_rpt\_d\_tab (

lb\_rpt\_tab (lb\_rpt (767,

8,

NULL,

NULL,

NULL,

NULL,

1,

250,

600,

1))),

'XSPT'));

Returning:

|  |  |
| --- | --- |
| **XSP** | **XSP\_DESCR** |
| R | Right |
| L | Left |

Hence, it is valid to use either the L or R XSP value. Had the value used not been in this list, the placement of the asset would fail with an error as below. The count relates to the number of datums over which the XSP is invalid.

ORA-20001: XSP is invalid - count = 8

ORA-06512: at "EXOR.LB\_LOAD", line 272

ORA-06512: at "EXOR.LB\_LOAD", line 233

ORA-06512: at line 5

Having established that the XSPs are valid, the assets may be placed.

### XSP Retrieval and Transformations

Consider having generated two asset locations on two different ranges of the route for example using code below.

, the asset

declare

l\_nal\_id integer;

begin

l\_nal\_id := LB\_LOAD.LD\_NAL('XSPT', 98, 'Test of XSP', NULL, 'Y', 'N', trunc(sysdate), NULL);

LB\_LOAD.LB\_LD\_RANGE(l\_nal\_id, 'XSPT', 767, 'G', 8, 250, 600, 1, 'R', trunc(sysdate), 1, NULL);

end;

declare

l\_nal\_id integer;

begin

l\_nal\_id := LB\_LOAD.LD\_NAL('XSPT', 99, 'Test of XSP', NULL, 'Y', 'N', trunc(sysdate), NULL);

LB\_LOAD.LB\_LD\_RANGE(l\_nal\_id, 'XSPT', 767, 'G', 8, 0, 360, 1, 'L', trunc(sysdate), 1, NULL);

end;

The asset locations are decomposed from the route to the datum with the XSP value as entered. The direction flag is inherited from the route system. This allows the same set of asset locations to be extracted with a reversed XSP in cases where a different route system traverses the location in an opposing direction or if the relative XSP is extracted relative to a specific datum which is in an opposing direction to that of the route through which the data was entered.

The following code shows the methods with which the asset location data is extracted and the method of translation of the location data from its persisted state to one relative to the route system, including any XSP transformations.

Using the example asset with ID of 99 from above, the assets locations may be extracted from the database as events relative to the datum. Note that this asset was placed with an XSP of L relative to route 767.

select ne\_id, ne\_unique, obj\_type, obj\_id, dir\_flag, start\_m, end\_m, xsp, offset

from table(lb\_loc.get\_asset\_location\_tab(99, 'XSPT', NULL, 1)), nm\_elements

where ne\_id = refnt;

Returning:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **NE\_ID** | **NE\_UNIQUE** | **OBJ\_TYPE** | **OBJ\_ID** | **DIR\_FLAG** | **START\_M** | **END\_M** | **XSP** | **OFFSET** |
| 73967 | 7892 | XSPT | 102 | 1 | 0 | 30.79 | L | 5 |
| 75324 | 9249 | XSPT | 102 | 1 | 0 | 30.49 | L | 5 |
| 75331 | 9256 | XSPT | 102 | -1 | 0 | 51.22 | R | -5 |
| 75330 | 9255 | XSPT | 102 | -1 | 0 | 48.22 | R | -5 |
| 73973 | 7898 | XSPT | 102 | -1 | 0 | 39.54 | R | -5 |
| 280413 | 40984 | XSPT | 102 | -1 | 0 | 89.17 | R | -5 |
| 74227 | 8152 | XSPT | 102 | -1 | 0 | 56.14 | R | -5 |
| 74340 | 8265 | XSPT | 102 | 1 | 0 | 14.43 | L | 5 |

Note that where the direction flag is reversed, the XSP and offsets are reversed.

Expressing the asset locations as events relative to the route system containing route 767:

select ne\_id, ne\_unique, obj\_type, obj\_id, dir\_flag, start\_m, end\_m, xsp, offset

from table(lb\_loc. get\_asset\_location\_tab(99, 'XSPT', NULL, 8)), nm\_elements

where ne\_id = refnt;

Returning the original XSP.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **NE\_ID** | **NE\_UNIQUE** | **OBJ\_TYPE** | **OBJ\_ID** | **DIR\_FLAG** | **START\_M** | **END\_M** | **XSP** | **OFFSET** |
| 767 | 0015A446 3/00040 | XSPT | 102 | 1 | 0 | 360 | L | 5 |

Reversing the route:

begin

nm3rvrs.reverse\_route(767, trunc(sysdate));

end;

On reversal, the measures and XSP are:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **NE\_ID** | **NE\_UNIQUE** | **OBJ\_TYPE** | **OBJ\_ID** | **DIR\_FLAG** | **START\_M** | **END\_M** | **XSP** | **OFFSET** |
| 767 | 0015A446 3/00040 | XSPT | 102 | 1 | 254.09 | 614.09 | R | -5 |

# Materialized view of asset locations

A materialized view script is included for those who wish to generate it. Currently, its generation is optional. The script is provided under the admin\views folder and is named V\_LB\_REP\_ASSET\_LOCATIONS.VW. The script will drop and recreate the materialized view. DBAs may wish to add details such as storage clauses. The view will provide the locations of all Exor inventory and LB assets as reference point locations relative to all relevant network references – routes and datums.

To execute the script, log in to SQL\*Plus using the account of the Highways owner and execute the script. The materialized view may take an hour or so to execute depending on the number of standard Exor inventory and LB asset locations and depending on the number of route systems. Note that the definition of the view may change in subsequent releases.

Once completed and properly indexed, queries can be made on asset locations by route or by asset-based keys. A text based index on the concatenation of the network unique key and description is available making it convenient to query (say) all asset locations on network elements where either the unique key or the description (or both) contain the phrase “High Street”. This can be done by using a text-based operator as shown below:

select \* from V\_LB\_REP\_ASSET\_LOCATIONS

where contains(network\_name\_descr, 'high street') > 0

# Notes on units

## Unit conversion of table functions

The package LB\_OPS is now equipped with two new functions each transforming an LB\_RPT\_TAB object. One will “normalise” the input lb\_rpt\_table by translating the units to the normal form where units are held in the natural units of the network that the array element relates to. Another will transform unit values into any specified Exor unit.

These two procedures allow query tools such as get\_lb\_RPt\_tab and translation tools such as get\_lb\_RPt\_R\_tab and get\_lb\_RPt\_D\_tab to normalise the input data into a form which suits the query on the database and extraction from the database and allows conversion of the output.

## Unit compatibility with Oracle Spatial data

The upgrade script will change names of the Exor units of measure to ensure they match the unit names inside the MDSYS unit metadata. This allows Exor units to be mapped onto SDO units which makes SDO operators such as within distance operate seamlessly by passing SDO unit names into the operator as strings.

## Unit conversion improvement

The upgrade script will create new unit conversion data to allow a NULL conversion when two items of data are in the same system of units. This allows the SQL code within the package procedures to be simplified without the need for complex case statements or IF-THEN-ELSE structures in PL/SQL blocks.

## Units and rounding

LB data measures are stored in an un-rounded format. This gives better precision during arithmetic operations such as translations. Certain unit conversion factors are not exact integers and the Exor route measure data is persisted in a rounded state as defined by the units system and this can lead to rounding errors as a route measure is translated to datum and back again. Due to this, during certain operations it is necessary to perform a range-based query which is adjusted for any lack of precision.

## Changes to unit registration procedure

The unit registration procedure lb\_reg.register\_unit has changed under defect 924307. The change is minor and should not affect any existing usage as the exor\_unit\_id argument is now an integer and not a generic number.

# Snapping or Net-Coding

Snapping is the term used to describe generating a network reference or list of references from a spatial reference. Code originally designed within the Exor core returns Ids, distance-from and the measure along the network element. Current code expects that the input geometry or XY coordinates and the units of measure of the buffer-distance or search-radius are in the same units as the network type to which the geometry is to be snapped. However, more recent requirements are such that the code is expected to operate using an XY or geometry in a user-specified coordinate system and a search-radius in any unit of measure. It is also expected that the output measures and geometry items should be generated in units of measure and coordinate system specified by the user. To this end a new package has been included in this build. It includes a single function which transforms a simple XY coordinate (in a given SRID) and produces a list of network references and related information. It uses new object types of LB\_SNAP and its table object lb\_snap\_tab

The function specification is:

FUNCTION lb\_snap\_xy\_to\_nw (pi\_x IN NUMBER,

pi\_y IN NUMBER,

pi\_in\_srid IN NUMBER,

pi\_buffer IN NUMBER,

pi\_in\_uol IN nm\_units.un\_unit\_id%TYPE,

pi\_out\_srid IN NUMBER,

pi\_out\_uol IN nm\_units.un\_unit\_id%TYPE,

pi\_themes IN nm\_theme\_array\_type,

CARDINALITY IN INTEGER)

RETURN lb\_snap\_tab;

Where the arguments are described below.

|  |  |  |
| --- | --- | --- |
| pi\_x | Number | X ordinate of source point |
| pi\_y | Number | Y ordinate of source point |
| pi\_in\_srid | Number | Coordinate system of the source XY expressed as a SRID (eb 27700 for UK national grid) |
| pi\_buffer | Number | Buffer or search radius around the input point to define the candidate area for selection of the network to which the XY will be snapped. |
| pi\_in\_uol | nm\_units.un\_unit\_id%TYPE,  (or integer) | The Exor unit of length used in the buffer distance |
| pi\_out\_srid | Number | Coordinate system of returned geometry |
| pi\_out\_uol | nm\_units.un\_unit\_id%TYPE,  (or integer) | The Exor unit of length used in the returned values of distance-from and measure |
| pi\_themes | nm\_theme\_array\_type | The list of themes to which the XY will be snapped |
| cardinality | Integer | An estimate of the number of returned items (used in optimiser statistics in some versions of server) |

The function returns an array or table of lb\_snap object data comprising:

|  |  |  |
| --- | --- | --- |
| ne\_id | integer | Primary key of network element to which the XY is snapped. |
| nlt\_id | integer | A pointer to the linear type of network to which the XY is snapped |
| ne\_nt\_type | varchar2(4) | The network type of the element |
| ne\_gty\_group\_type | varchar2(4) | The group-type of the element (NULL if a datum) |
| ne\_unique | varchar2(30) | The unique reference of the element (unique in conjunction with ne\_nt\_type) |
| ne\_descr | varchar2(240 | The description of the element. |
| measure | number | The measure along the element at the projection point. |
| distance\_from | number | The minimum distance from the projection point to the input point. |
| unit\_id | integer | The unit of length used in returned values |
| unit\_name | varchar2(30) | The name of the unit of length used in returned values |
| nw\_geom | sdo\_geometry | The geometry of the candidate element (as use din highlighting) |
| proj\_pt | sdo\_geometry | The geometry of the projection point. |

Below is an example of its use. XY data is passed in UK National Grid (27700) and resultant geometry (projection point and element geometry) are returned in WGS84 (SRID 4326) which is used in Google maps. The search radius or buffer is entered as a length in the Exor unit 1 indicating metres. The value 2 is used to denote that the units on the output are in kilometres. Note that the projection point is returned as an LRS geometry with the XY and a measure along the element. This is the same as that returned in the measure attribute but without unit conversion. Any measures returned in the element geometry are also returned in their native state and are not translated.

SELECT t.ne\_id,

t.nlt\_id,

t.ne\_unique,

t.measure,

t.distance\_from,

t.unit\_id,

t.unit\_name,

p.x,

p.y,

p.z

FROM TABLE (lb\_net\_code.lb\_snap\_xy\_to\_nw (

216000,

109934,

27700,

200,

1,

4326,

2,

nm\_theme\_array\_type (nm\_theme\_entry (11),

nm\_theme\_entry (9),

nm\_theme\_entry (7)),

10)) t,

TABLE (SDO\_UTIL.getvertices (proj\_pt)) p

ORDER BY distance\_from;

Returning:

