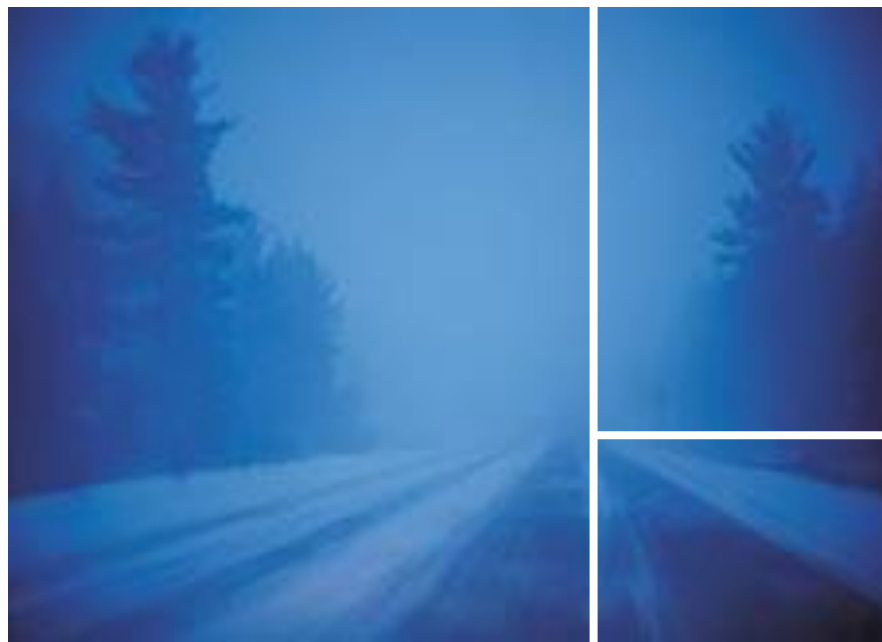


Exor Leading the way in Infrastructure Asset Management Solutions



# Exor Corporation Limited



## *Spatial Server System Admin v4.4*



*The Global Leader in  
Infrastructure Asset Management*



This page intentionally blank

## Quality Assurance Statement

Document Details	
<b>File:</b> Spatial Server Admin 4.4.docx	<b>Prepared by:</b> P. Voudouris
<b>Manual Name:</b> Spatial Server Admin V4.4	<b>Reviewed by:</b> Colin Stewart
<b>Version:</b> 1.1	<b>Approved for issue by:</b> Colin Stewart
<b>Date of Issue:</b> July 2011	<b>Product Manager:</b>
<b>File Name:</b> Document Centre/Exor/Product Manuals/4.4.0.0 Product Manuals/4.4.0.0 completed documentation/Spatial Server Admin v4.4.docx	

Document Version Control				
Revision	Date	By	Chapter / Page	Description
1.1	July 2011	PV		
1.0				Original

### Notice to User

The information contained within this manual pertains to the following versions:

<b>Highways by Exor:</b>	<b>Release 4.4</b>
<b>ESRI © ArcSDE ©, ArcGIS©</b>	<b>9.3.1</b>
<b>Oracle© Enterprise Edition</b>	<b>11g</b>

# CONTENTS

Quality Assurance Statement .....	3
Creating Oracle Spatial (SDO) layers .....	6
Creating the datum layer .....	7
Setting up the SDE metadata .....	7
Importing shapefiles .....	8
Register the datum layer in the NM3 metadata .....	14
Update the SDO (MDSYS user) metadata .....	16
Creating the required indexes .....	18
Troubleshooting shapefile imports .....	19
Create the datum view .....	19
Creating Layers-The GIS0020 form .....	20
Creating the nodes layer .....	24
Creating Group Layers .....	26
Non-Linear groups and Locator .....	28
Creating Asset Layers .....	31
Creating Dynsegged Assets .....	31
Creating off-network assets .....	33
Dropping SDO layers .....	36
Creating Document Layers .....	39
Creating Accident Layers .....	41
Creating Street Lighting layers .....	43
Creating Enquiries Layers .....	45
Creating Defects Layers .....	47
Creating Work Order Layers .....	49
Creating Structures Layers .....	50
Creating Streetworks Sites Layer .....	52
Creating Street Gazetteer Manager Layers .....	53
Creating Structural Schemes Layers .....	55
Creating TMA Layers .....	56
Creating Custom Layers .....	61
Register a spatial table as a theme .....	64
GIS0020 form Utilities .....	66
View Spatial Metadata .....	66
Rebuild spatial index .....	67
View current Theme .....	68
Analyze feature table .....	68
Refresh Metadata .....	68
Additional utilities .....	69
Configuring Maps with the Oracle MapBuilder Tool .....	73
Starting Mapbuilder .....	74
Mapbuilder Connection Page .....	74
Map Builder Themes .....	77
Map Builder Base Maps .....	83
Importing Existing data .....	88
Registering the spatial tables and views .....	90
Addressing problems with individual layers .....	92
Step 1- registering the datum layer .....	92
Step 2- registering the points spatial table .....	92
Step 3- registering the node spatial views .....	92
Step 4 – Setting the extents of the layers in the sde metadata .....	93
Step 5 – Group and Asset layers .....	93

Using the GIS0010 form .....	94
Introduction.....	95
General Information.....	95
GIS Themes .....	96
Roles Tab.....	105
Geometry Types Tab.....	105
Base Themes Tab .....	107
Snapping Tab.....	107
LRef Details Tab.....	108
Area Details Tab.....	109
Asset Details Tab .....	110

**CHAPTER**

**1**

# Creating Oracle Spatial (SDO) layers

The document is meant for users working with Exor version 4.4 software running ArcSDE 9.3.1 and Oracle 11g.

This chapter describes the workflow to create and register a network datum layer as an Oracle Spatial table using a shapefile and then create all the resulting layers (assets and groups) based on the datum layer. This chapter also contains information on how to create layers for other exor products such as Accidents, Street Lighting, Structures, Streets (NSG), etc.

## Creating the datum layer

This section describes how to convert a shapefile to an oracle spatial table and register it in the NM3 and ArcSDE metadata.

**Note for NSG Users:** The spatial table that you create for the ESU network type MUST be named NM\_NSG\_ESU\_SHAPES

Most of the examples use the following setup and example data:

- The NW metadata has been set up for a NW type called **MDAT**
- The node type for the MDAT NW is called **ROAD**
- The input shapefile contains measured shapes and is called **SDO\_NETWORK.shp**
- The spatial table to be created will be called **SDO\_NETWORK** with a PK column called **NE\_ID**

## Setting up the SDE metadata

First you need to ensure that your SDE schema does have the correct metadata for handling Oracle Spatial layers.

This is controlled by the DBTUNE table in the SDE schema and the dbtune file under \$SDEHOME\etc\dbtune.sde

Log in to SQL and type:

```
select count(*) from dbtune where
config_string='SDO_GEOMETRY'
```

If the count is zero the SDE instance does not support Oracle Spatial data types. You will need to follow these steps:

Navigate to the \$SDEHOME\etc folder and open the **dbtune.sde** file in notepad. To be sure do a search on SDO\_GEOMETRY in this file. If the string is not found copy the following into the dbtune.sde file:

```
##SDO_GEOMETRY
GEOMETRY_STORAGE      "SDO_GEOMETRY"
ATTRIBUTE_BINARY       "LONGRAW"

UI_TEXT                "User Interface text description for
                        SDO_GEOMETRY keyword"

UI_NETWORK_TEXT        "User Interface network text
                        description for SDO_GEOMETRY keyword"

COMMENT                "Any general comment for SDO_GEOMETRY
                        keyword"

END
```

It doesn't really matter where you copy this section as long as it is outside of an existing dbtune 'section' (Starting with '##' and ending with 'END')

Now you will need to import the new keyword into the **sde schema** by using the `sdedbtune` command from a DOS window:

```
sdedbtune -o import -f dbtune.sde -u sde -p sde
```

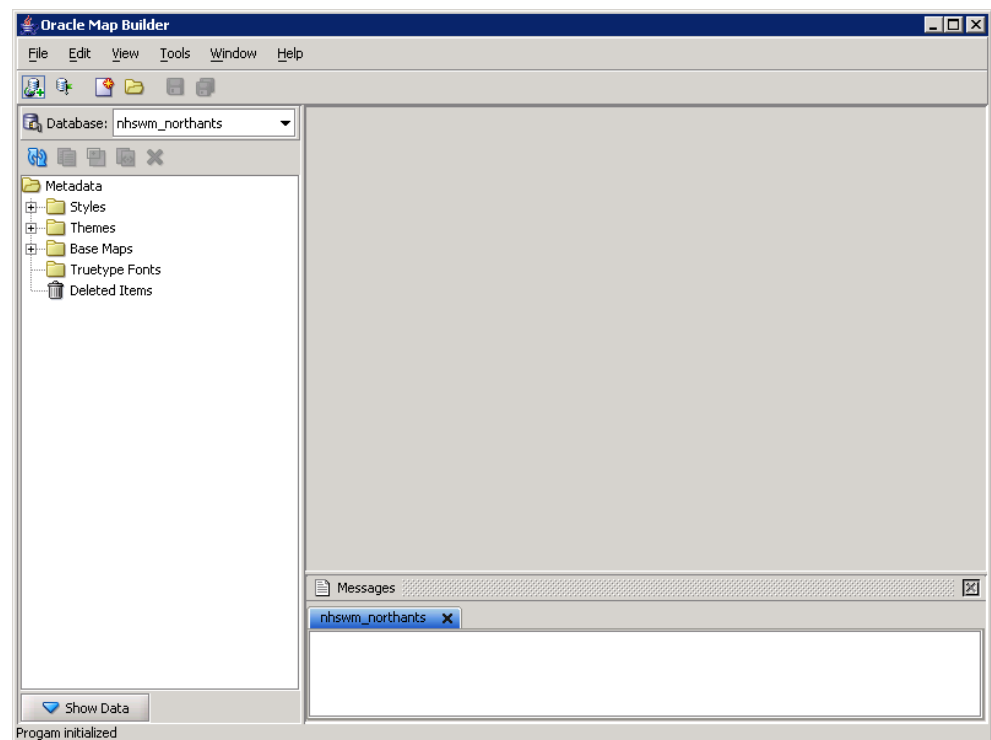
## Importing shapefiles

Three methods are discussed. The first one uses Mapbuilder, the second, the Oracle shp2sdo utility and the third uses ArcSDE commands.

### Using Oracle Map Builder

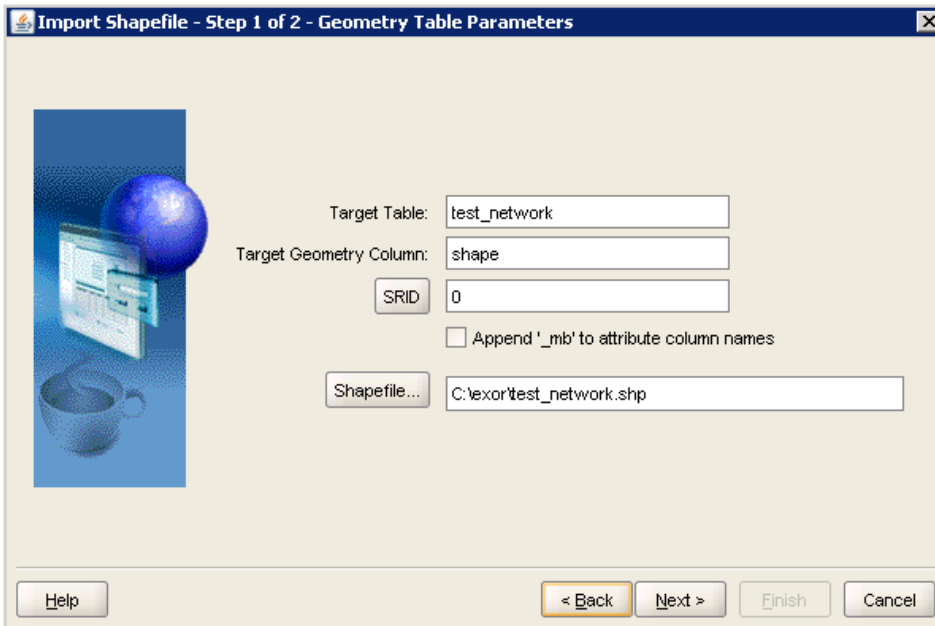
For the purposes of this document, we are using an example shapefile “county.shp” and I’m importing into an example table of “shp\_county”, with a geometry column of “shape”.

- Open Map Builder and connect to the relevant database.



- Select Tools -> Import Shapefile
- Complete the form accordingly.  
i.e. give it a table name to import into (e.g. TEST\_NETWORK),  
advise on what column the geometry data is going into (e.g. SHAPE),  
ensure the SRID is set to 0,  
browse for the relevant shapefile.  
Then select Next.





**Import Shapefile - Step 1 of 2 - Geometry Table Parameters**

Target Table:

Target Geometry Column:

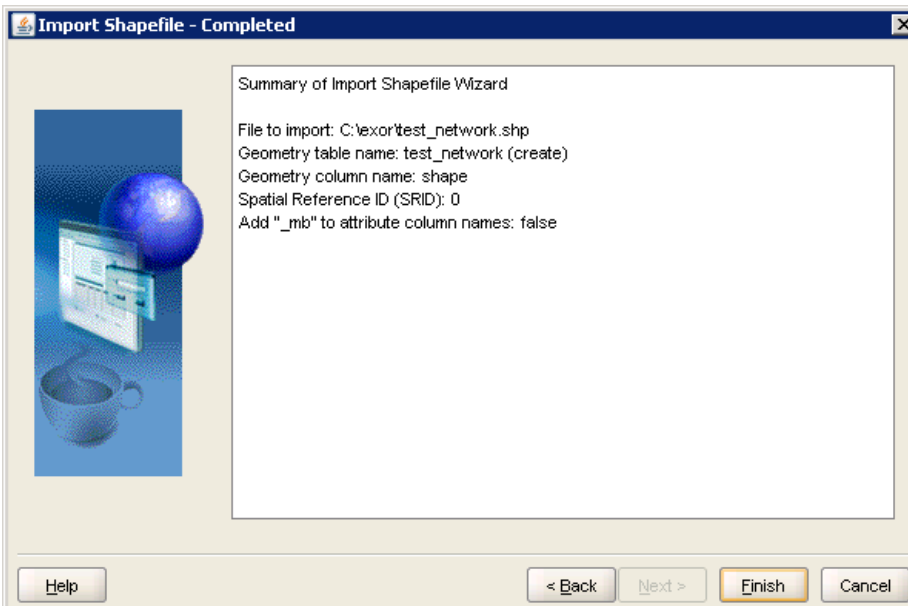
SRID:

☐ Append '\_mb' to attribute column names

Shapefile...

Help < Back Next > Finish Cancel

- Select Next for Step 2 of 2, as we are not creating a Predefined theme.
- You will now be presented with the Wizard summary, to check what you have selected.



**Import Shapefile - Completed**

Summary of Import Shapefile Wizard

File to import: C:\exor\test\_network.shp  
 Geometry table name: test\_network (create)  
 Geometry column name: shape  
 Spatial Reference ID (SRID): 0  
 Add "\_mb" to attribute column names: false

Help < Back Next > Finish Cancel

NB. If the table name already exists, it will say "create" in brackets, otherwise (as in this example) it will say "create" since it's a new table.

On completion of the import, you need to double check what the geometry type is for the shapefile. This can be done by using the following example SQL:

```
SQL> select a.shape.sdo_gtype
      from test_network a
      group by a.shape.sdo_gtype;
```

SHAPE.SDO\_GTYPE

-----

3302

At this point ensure there are two indexes on the table - one spatial and one primary key. Also check the table structure and ensure that there is a unique number(38) column that can act as a unique primary key column. If one doesn't exist, then just add a column to the table and then use the example SQL below, to populate the column (ensure to change the test\_network and ne\_id values accordingly):

```
declare
  l_count number;
begin
  l_count:=0;
  for csrec in (select rowid from test_network) loop
    l_count:=l_count+1;
    update test_network set ne_id = l_count
      where rowid = csrec.rowid;
  end loop;
end;
/
commit
```

- Map Builder should automatically create a spatial index on the newly created table, If not, run the following:

```
SQL> exec nm3sdo.create_spatial_idx('TEST_NETWORK','SHAPE',);
```

- Once this is done, we need to tell Highways that it exists. This is “normally” done via the GIS Layer Tool, but as we are creating a layer from a shapefile, we need to populate the GIS Themes form manually (GIS0010). You can either fill the form in manually, or you can run a procedure to populate the majority of it:

e.g.

```
exec nm3sdo.register_SDO_table_as_theme
( p_table      => 'TEST_NETWORK'
, p_pk_col     => 'NE_ID'
, p_fk_col     => NULL
, p_shape_col  => 'SHAPE'
, p_cre_idx    => 'Y'
, p_estimate_new_tol => 'Y');
```

- This should now populate GIS Themes (GIS0010, Exor -> GIS Data -> GIS Themes) with the relevant info.

### Using Oracle's shp2sdo utility

---

The shp2sdo utility is available for download from Oracle Metalink:

<http://www.oracle.com/technology/software/products/spatial/files/shp2sdo.zip>

### Ensure that your shapefile contains Measured Lines (PolyLineM shapes).

To convert your shapefile to an SDO table:

Open a command window and run shp2sdo from the command line.

```
D:\>shp2sdo
```

```
shp2sdo - Shapefile(r) To Oracle Spatial Converter
Version 2.15 21-May-2004
```

Copyright 1997,2004 Oracle Corporation  
For use with Oracle Spatial.

Input shapefile (no extension):

*Type in the name of the shapefile without the extension (.shp) e.g. sdo\_network. The program will display the number of records in the shapefile and will prompt you for the output table:*

Shape file sdo\_network.shp contains 11360 linestrings

Output table [sdo\_network]:

*Type in the output table name e.g. sdo\_network\_sample*

Output data model [O]:

*Accept the default by pressing <enter>*

Geometry column [GEOM]:

*Type in 'SHAPE'*

ID column []:

*Accept the default by pressing <enter>*

Use a spatial reference system ID (SRID) ? [N]:

*Accept the default by pressing <enter>*

Change tolerance value from the default (0.00000005) ?

[N]: Y

*Type in 'Y'*

Please enter a new tolerance value: 0.0005

*Type in '0.0005'*

Generate data inside control files ? [N]:

*Type in 'Y'*

Target database Oracle8i? [N]:

*Accept the default by pressing <enter>*

Spatial Data requires more than 6 digits precision? [N]:

*Accept the default by pressing <enter>*

Bounds: X=[-102.049947,-94.607096]

Y=[36.994170,40.002960]

Override ? [N]:

*Note that these numbers will be different depending on your shapefile. You can change the MBR of the layer or accept the default (shapefile extent) by pressing <enter>*

*The program will now create two files. One is an SQL script to create the table ready for the data, the other is the control file with the data embedded:*

Processing shapefile sdo\_network into spatial table

SDO\_NETWORK

Data model is object-relational

```

Geometry column is SHAPE
Points stored in SDO_POINT attributes
Data is in the control file(s)
Control file generation for Oracle10g or higher
Spatial data loaded with 6 digits of precision
Tolerance set to 0.000500
Conversion complete : 11360 linestrings processed
The following files have been created:
  sdo_network_sample.sql :      SQL script to create the
table
  sdo_network_sample.ctl :      Control file for loading
the table

D:\ >

```

Check the SQL script and make sure it is sensible. Things to check are that sensible columns are being created, that the shape column is correct and that the primary key column is of type **NUMBER(38)**

**The above paragraphs assume that the shape file already contains the NE\_ID column. If it doesn't then load it anyway and match it in the database after the data has been loaded.**

Run the sql script as the highways owner to create the empty spatial table and populate the USER\_SDO\_GEOM\_METADATA table.

**Make sure to check the contents of the data in the ctl file (or .dat file if you opted for separate files after all). The geometry type must be 3302 If not, open the .ctl file in a text editor and do a search and replace if required.**

Load the data, by running SQL Loader as follows substituting appropriate values:

```
sqlldr <username>/<password> <control file>.ctl
```

Once the data is loaded ensure that the primary key is established

### Layer Registration

You will now need to register the layer within the Network Manager layer metamodel (NM\_THEMES\_ALL) and SDE.

To do this start a SQL session, login as the highways owner and run the following the nm3sdo.register\_sdo\_table\_as theme procedure. This procedure will register the newly created datum layer in the NM\_THEMES\_ALL table as a new them. Furthermore if the REGSDELAY product option is set to 'Y' then the layer will be registered in the SDE metadata.

The syntax is shown below.

```

BEGIN
  nm3sdo.register_sdo_table_as_theme
    ( p_table           => <FEATURE_TABLE_NAME>
    , p_pk_col          => <FEATURE_PK_COL>

```

```

, p_fk_col           => NULL
, p_shape_col        => <FEATURE_SHAPE_COL>
, p_tol              => <SDO TOLERANCE VALUE>
, p_cre_idx          => 'Y/N'
, p_estimate_new_tol => 'Y/N'
, p_override_sdo_meta => 'Y/N'
);
END;
```

Where:

**p\_table:** The spatial table name (nth\_feature\_table)  
**p\_pk\_col:** The primary key column of the spatial table (nth\_feature\_pk\_column) **Make sure that the Primary key column (e.g. NE\_ID) is defined as a NUMBER(38). If not you will need to create a new column as a number(38) and populate it with the values from the original column.**

**p\_fk\_col:** The feature fk column-Only applies to dynsegged themes (nth\_feature\_fk\_column)

**p\_shape\_col:** The column in the spatial table that holds the geometries (nth\_feature\_table)

**p\_tol:** The SDO tolerance. This is the tolerance value that will be used by Oracle and it largely depends on your data (XY coordinate values). If you know the tolerance value of your data enter it here (e.g. 0.005). Otherwise specify it as 'NULL' and enter 'Y' in the **p\_estimate\_new\_tol** parameter to let the system calculate the tolerance. For more information on tolerance values see also the discussion in the DIMINFO Description section in this document

**p\_cre\_idx:** Specify whether you want a spatial index to be created automatically

**p\_estimate\_new\_tol:** Specify whether you want to let the procedure estimate the tolerance values of your data.

**p\_override\_sdo\_meta:** Specify whether you want to let the procedure to re-create the entries in the user\_sdo\_geom\_metadata table

Assuming our spatial table is called TEST\_NETWORK with a Primary Key column of NE\_ID and a Spatial column called SHAPE, you need to run:

```

BEGIN
  nm3sdo.register_sdo_table_as_theme
  ( p_table           => TEST_NETWORK'
  , p_pk_col          => 'NE_ID'
  , p_fk_col          => NULL
  , p_shape_col       => 'SHAPE'
  , p_tol             => NULL
  , p_cre_idx         => 'Y'
  , p_estimate_new_tol => 'Y'
  , p_override_sdo_meta => 'Y'
  );
END;
```

You should now be able to view the new spatial table in ArcGIS.

If for any reason an error occurs during this process you can re-run it by first removing any metadata entries that have been created:

```
delete nm_themes_all
where nth_feature_table = 'TEST_NETWORK'
```

(you will need to query the SDE.LAYERS table to retrieve the layer\_id for the TEST\_NETWORK table)

```
begin
  nm3sde.drop_layer( pi_layer_id from sde.layers );
end;
```

### Using ArcSDE commands

Use shp2sde to import the shapefile using the new keyword (SDO\_GEOMETRY). **Make sure that the Primary key column (e.g. NE\_ID) is defined as a NUMBER(38). If not you will need to create a new column as a number(38) and populate it with the values from the original column.**

**NOTE: This paragraph assumes that an ArcSDE license is available and the ArcSDE service is running against the schema you are loading the data in.**

```
shp2sde -o create -l sdo_network,shape -f nstp2_sdo_mdat -a all -C
ne_id,USER -k ORACLE_SPATIAL -g RTREE,6 -e 1M -u nstp -p nstp
```

**NOTE: For NSG Users the spatial table MUST be named NM\_NSG\_ESU\_SHAPES e.g.**

```
shp2sde -o create -l nm_nsg_esu_shapes,shape -f nstp2_sdo_mdat
-a all -C ne_id,USER -k ORACLE_SPATIAL -g RTREE -e 1sM+ -u
nstp -p nstp
```

If the measure values are in miles or km you will have to use the -m parameter to define the precision scale for measures e.g.

```
shp2sde -o create -l SDO_NETWORK,shape -f nstp2_sdo_mdat -m 0,1000 -a
all -C ne_id,USER -k ORACLE_SPATIAL -g RTREE -e 1M -u nstp -p nstp
```

This will hold up to three decimal points in the M values

Now perform an sdelayer -o describe\_long command on the newly imported layer. Take a note of the values in the layer envelope (Min X, Min Y, Max X and Max Y) you will need these later in this procedure.

The datum network layer should now be successfully loaded in the database and registered with SDE. You will then need to register it with the NM3 metadata.

### **Register the datum layer in the NM3 metadata**

You will first need to insert a new road into NM\_THEMES\_ALL using this PL/SQL block:

```
DECLARE
```

```

l_rec_nth nm_themes_all%ROWTYPE;
BEGIN
  l_rec_nth.nth_theme_id              :=
nm3seq.next_nth_theme
e_id_seq;
  l_rec_nth.nth_theme_name            := 'THEME_NAME';
  l_rec_nth.nth_table_name            :=
'NM_ELEMENTS';
  l_rec_nth.nth_where                  := NULL;
  l_rec_nth.nth_pk_column              := 'NE_ID';
  l_rec_nth.nth_label_column          := 'NE_UNIQUE';
  l_rec_nth.nth_rse_table_name        :=
'NM_ELEMENTS';
  l_rec_nth.nth_rse_fk_column          := 'NE_ID';
  l_rec_nth.nth_st_chain_column        := NULL;
  l_rec_nth.nth_end_chain_column      := NULL;
  l_rec_nth.nth_x_column              := NULL;
  l_rec_nth.nth_y_column              := NULL;
  l_rec_nth.nth_offset_field          := NULL;
  l_rec_nth.nth_feature_table         :=
'FEATURE_TABLE';
  l_rec_nth.nth_feature_pk_column      :=
'FEATURE_PK_COLUMN';
  l_rec_nth.nth_feature_fk_column      := NULL;
  l_rec_nth.nth_xsp_column            := NULL;
  l_rec_nth.nth_feature_shape_column  :=
'FEATURE_SHAPE_COLUMN';
  l_rec_nth.nth_hpr_product            := 'NET';
  l_rec_nth.nth_location_updatable    := 'N';
  l_rec_nth.nth_theme_type            := 'SDO';
  l_rec_nth.nth_dependency            := 'I';
  l_rec_nth.nth_storage               := 'S';
  l_rec_nth.nth_update_on_edit        := 'N';
  l_rec_nth.nth_use_history            := 'N';
  l_rec_nth.nth_start_date_column     := NULL;
  l_rec_nth.nth_end_date_column       := NULL;
  l_rec_nth.nth_base_table_theme      := NULL;
  l_rec_nth.nth_sequence_name         := NULL;
  l_rec_nth.nth_snap_to_theme         := 'N';
  l_rec_nth.nth_lref_mandatory        := 'N';
  l_rec_nth.nth_tolerance             := 10;
  l_rec_nth.nth_tol_units             := 1;
  l_rec_nth.nth_dynamic_theme         := 'N';
  --
  nm3ins.ins_nth (l_rec_nth);
  --
END;
COMMIT;

```

You will then need to update the lookup table NM\_NW\_THEMES. This table only contains two columns with references to NM\_LINEAR\_TYPES and NM\_THEMES\_ALL:

```

insert into nm_nw_themes
select nlt_id ,nth_theme_id
from nm_themes_all,nm_linear_types
where nth_feature_table='TEST_NETWORK'
and nlt_nt_type='<network_type>';

COMMIT;

```

### ***Update the SDO (MDSYS user) metadata***

The ArcSDE registration sometimes-seems to enter invalid entries the sdo metadata entries in the USER\_SDO\_GEOM\_METADATA table. To check this type in SQL:

```

SQL> select diminfo from user_sdo_geom_metadata where
table_name='TEST_NETWORK';

DIMINFO(SDO_DIMNAME, SDO_LB, SDO_UB, SDO_TOLERANCE)
-----
-----
SDO_DIM_ARRAY(SDO_DIM_ELEMENT(NULL, 0, 2147483647, .0005),
SDO_DIM_ELEMENT(NULL,
0, 2147483647, .0005), SDO_DIM_ELEMENT('M', 0, 2147483647, .00005))

```



## DIMINFO Description



The SDO\_DIM\_ELEMENT type is defined as:

```
Create Type SDO_DIM_ELEMENT as OBJECT (
  SDO_DIMNAME VARCHAR2(64),
  SDO_LB NUMBER,
  SDO_UB NUMBER,
  SDO_TOLERANCE NUMBER);
```

Where:

- SDO\_DIMNAME is the dimension name. Typically the values will be either 'X', 'Y' 'M' or 'Z'.
- SDO\_LB is the Lower Boundary of the dimension (e.g. Xmin)
- SDO\_UB is the Upper Boundary of the dimension (e.g. Ymin)
- SDO\_TOLERANCE\_NUMBER is a value used by some Oracle Spatial functions to determine whether two ordinate values in the same dimension are equivalent. For example, in functions that determine whether two line segments share the same endpoint, the ordinate values of the endpoints of the two line segments are compared. Rather than checking for equal coordinate values, a range check is done to determine equivalency. Conceptually, the comparison of two two-dimensional endpoints ((x1, y1) and (x2, y2)) looks like this:
  1. If (x1 <= (x2 + xTolerance)) and (x1 >= (x2 - xTolerance)), then x1 is equivalent to x2.
  2. If (y1 <= (y2 + yTolerance)) and (y1 >= (y2 - yTolerance)), then y1 is equivalent to y2.
  3. If x1 is equivalent to x2 and y1 is equivalent to y2, then the points (x1, y1) and (x2, y2) are equivalent in Oracle Spatial geometry functions.

To compare data exactly to the level of your data's accuracy, specify .[00000]5 where the number of zeroes after the decimal point and before the digit 5 is the same as the number of digits of accuracy of your data.

**Consequently, if when attempting to merge in SM you get an error that geometries do not connect (although the start/end vertices have the same coordinates) the first thing you can try is to increase the tolerance value in the DIMINFO column**

The SDO\_DIM\_ARRAY instance is of size n if there are n dimensions. That is, DIMINFO contains 2 SDO\_DIM\_ELEMENT instances for 2-dimensional geometries, 3 instances for 3-dimensional geometries, and 4 instances for 4-dimensional geometries. Each SDO\_DIM\_ELEMENT instance in the array must have valid (not null) values for the SDO\_LB, SDO\_UB, and SDO\_TOLERANCE attributes.

You can think of the SDO\_DIM\_ARRAY as the equivalent of the SDE entity types (nlsMpa+)

In our example, notice the NULL values for the SDO\_DIM\_ELEMENT type and the RLVs (Ridiculously Large Values) for the X,Y,M dimensions boundaries. Interestingly ArcView or ArcMap don't seem to care about this as long as the SDE metadata and entity types are correct.

If the SDO\_DIM element values appear correct you can skip this step.

A good way to find the correct extents of the datum layer is to run the following procedure:

```
SQL> select
nm3sdo.calculate_table_diminfo('TEST_NETWORK', 'SHAPE')
from dual;
```

```
NM3SDO.CALCULATE_TABLE_DIMINFO('TEST_NETWORK','SHAPE')(SD
O_DIMNAME, SDO_LB, SDO_U
```

```
-----
SDO_DIM_ARRAY(SDO_DIM_ELEMENT('X', 891011.951,
6003237.07, .005), SDO_DIM_ELEMENT('Y', 1831221.31,
4302916.29, .005), SDO_DIM_ELEMENT('Z', 0, 13.556, .005))
```

This will calculate the Minimum Bounding Rectangle (MBR) of all the geometries in the layer and return it as a DIM\_ARRAY type that you can then use to update the DIMINFO column in the MDSYS metadata.

**Note that these values represent the MBR of existing data. If the extent of your network is likely to change you will need to enter the maximum X,Y values.**

To update the USER\_SDO\_GEOM\_METADATA table use the values from the previous function:

```
update user_sdo_geom_metadata set diminfo =
MDSYS.SDO_DIM_ARRAY
  (MDSYS.SDO_DIM_ELEMENT('X', 891012.951, 6003238.07,
.005)
, MDSYS.SDO_DIM_ELEMENT('Y', 1831222.31, 4302918.29,
.005)
, MDSYS.SDO_DIM_ELEMENT('M', 0, 14.556, .005))
where table_name='TEST_NETWORK'
/
```

```
commit;
```

Now have a look in the SDE.LAYERS table and note the values in the MINX, MINY, MAXX, MAXY columns for the datum layer. If the values are not the same as the ones in the SDO\_GEOM\_METADATA you will need to update them:

```
update sde.layers set
  MINX=891012.951
, MINY=1831222.31
, MAXX=6003238.07
, MAXY=4302919.29
where table_name='TEST_NETWORK'
/
```

```
commit;
```

## Creating the required indexes

If not already performed in the previous steps, you will also need to:

1. Create a unique key index on the NE\_ID column:

```
CREATE UNIQUE INDEX sdo_network_pk
ON TEST_NETWORK(NE_ID)
```

2. You will also need to analyze the spatial table and indexes:

```
exec DBMS_STATS.GATHER_TABLE_STATS (OWNNAME => NULL,
TABNAME => 'TEST_NETWORK', CASCADE => TRUE);
```

The spatial table needs to be analyzed otherwise panning and zooming is very slow.

### 3. Create a spatial index::

```
CREATE INDEX SDO_NETWORK_SPIDX ON
TEST_NETWORK (SHAPE)
INDEXTYPE IS MDSYS.SPATIAL_INDEX;
```

### *Troubleshooting shapefile imports*

When running the shp2sde command you may get a ‘duplicate columns’ oracle error. This is probably because the shapefile attribute file (\*.dbf) may have duplicate fields (dBase files allow this). The easiest way to resolve this is to create a text file- you can call it any name e.g. cols.txt which will contain a single line:

<source\_shapefile\_pk\_field> <target\_sdo\_datum\_layer\_primary\_column> ,

e.g:

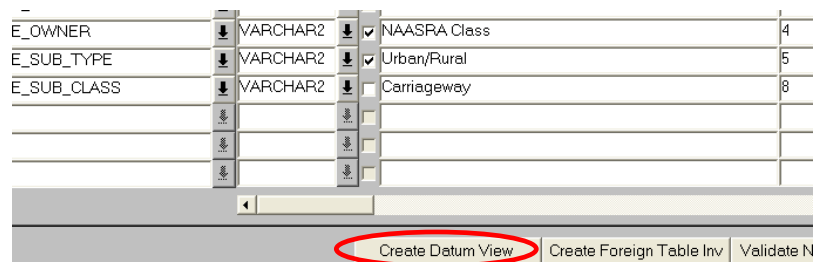
rse\_he\_id ne\_id

Then instead of using the “-a all” use the -a file=cols.txt in the shp2sde command

## Create the datum view

Make sure you click on the [Create Datum View] in the NM0002 form for the datum type. Failing to do this will produce errors in Spatial Manager when using the Identify tool.

Figure 1-NM0002 form-Create Datum view



E_OWNER	↓	VARCHAR2	↓	✓	NAASRA Class	4
E_SUB_TYPE	↓	VARCHAR2	↓	✓	Urban/Rural	5
E_SUB_CLASS	↓	VARCHAR2	↓	✓	Carriageway	8
	↓		↓			
	↓		↓			
	↓		↓			

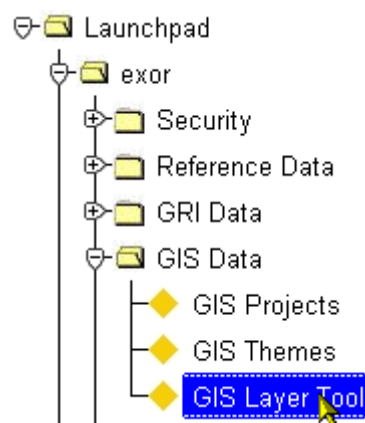
☒ Create Datum View
 ☐ Create Foreign Table Inv
 ☐ Validate N

## Creating Layers-The GIS0020 form

After creating your first datum layer you can proceed on creating all the other layers for the products you have installed. The **GIS0020-GIS Layer tool** form will help you achieve this.

The form is introduced in v 4.0 and it provides a set of wizards to create the relevant SDO layers for each Exor product you have installed.

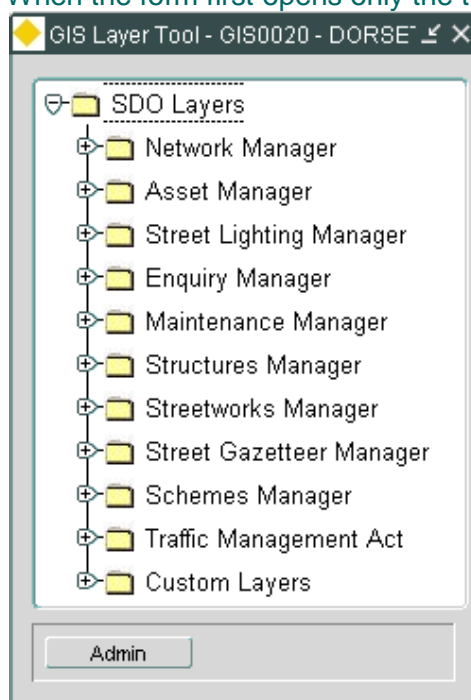
The form is called from the highways launchpad as shown in the next figure:



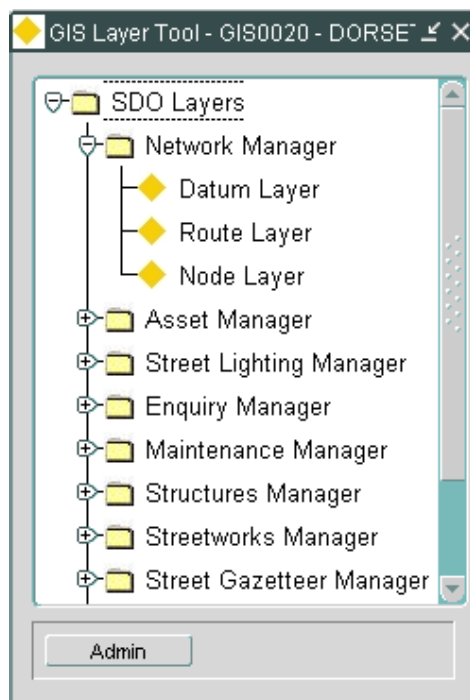
The form is divided into 3 areas:

1. **Tree area:** This will contain in a tree-like view /menu of all the products installed in your system and the available SDO layers
2. **Wizard area:** This is the dynamic part of the form which will change depending on the layer selected from the tree view.
3. **Details area:** This will display in a grid the theme(s) details for each layer.

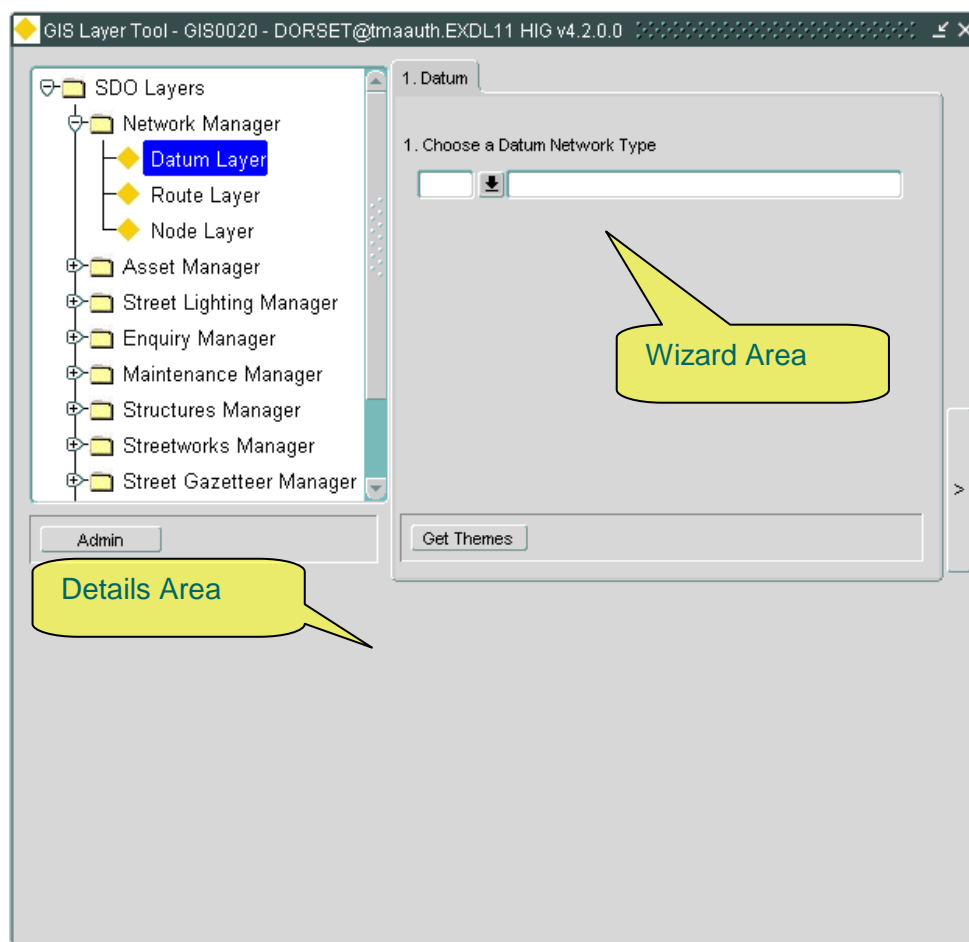
When the form first opens only the tree area is displayed.



You can click on the relevant product node in the tree to expand and display the available layers



You can now click on each layer to expand the form and display the Wizard and Detail areas. The next figure shows the form after we selected the Datum Layer tree node:



Note that the Details area is not populated. You will need to first select from the Wizard area an item type (in our example a Datum type) and then click on the **[Get Themes]** button to display the relevant data in the Details area:

GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01.HIG v4.4.0.0

SDO Layers

- Network Manager
  - Datum Layer**
  - Route Layer
  - Node Layer
- Asset Manager
- Document Manager
- Enquiry Manager
- Maintenance Manager
  - Defects Layer
  - Work Order Lines Layer

1. Datum

1. Choose a Datum Network Type

ESU

Admin

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column
ESU	NM_ELEMENTS	NE_ID	NM_NSG_ESU_SHAPES	SHAPE
ESU_TAB	NM_ELEMENTS	NE_ID	NM_NSG_ESU_SHAPES_TABL	SHAPE
ESU_JOIN	NM_ELEMENTS	NE_ID	NM_NSG_ESU_SHAPES_JOIN	SHAPE

Geometry Type

3302

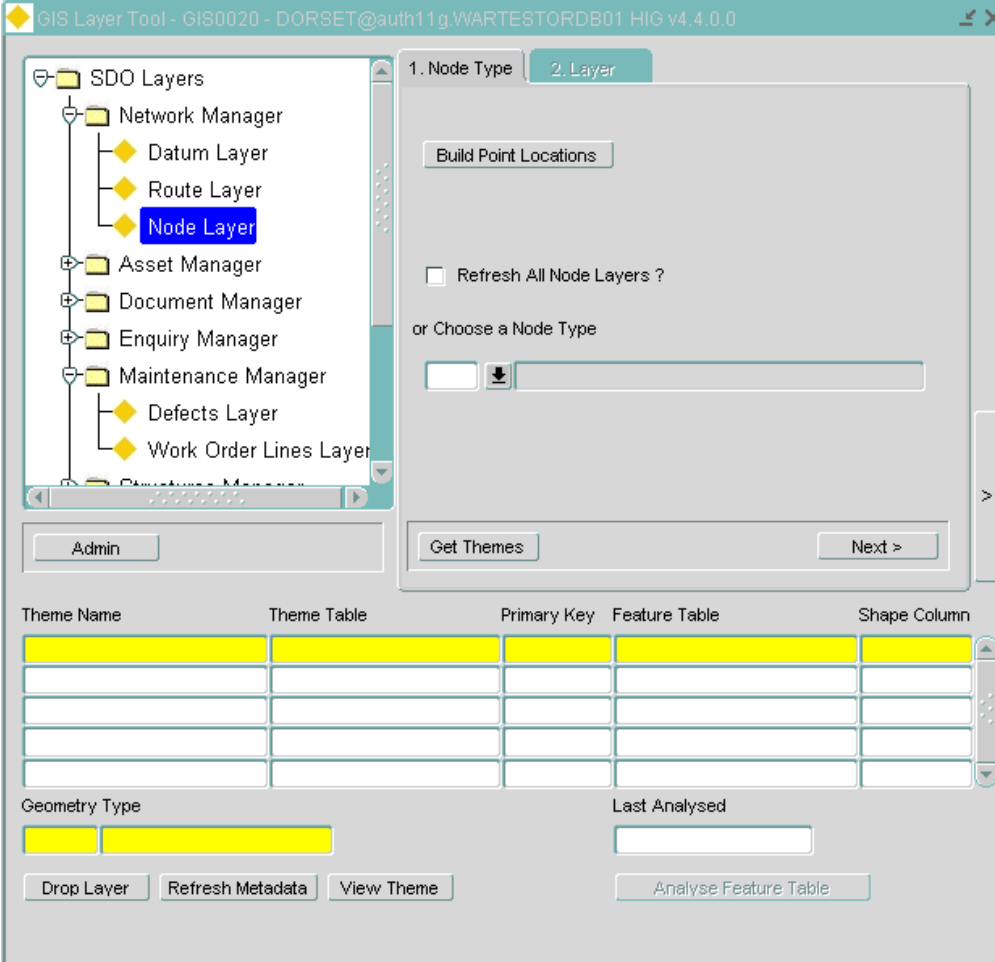
Last Analysed

The following paragraphs discuss in detail the use of GIS0020 form and creation of spatial layers for each product.

## Creating the nodes layer

To create the node layer:

1) Choose “Network Manager” > “Node Layer” from the Tree menu in the GIS0020 form:



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01.HIG v4.4.0.0

SDO Layers

- Network Manager
  - Datum Layer
  - Route Layer
  - Node Layer**
- Asset Manager
- Document Manager
- Enquiry Manager
- Maintenance Manager
  - Defects Layer
  - Work Order Lines Layer


Admin

1. Node Type | 2. Layer

Build Point Locations

☐ Refresh All Node Layers ?

or Choose a Node Type



Get Themes

Next >

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

Geometry Type

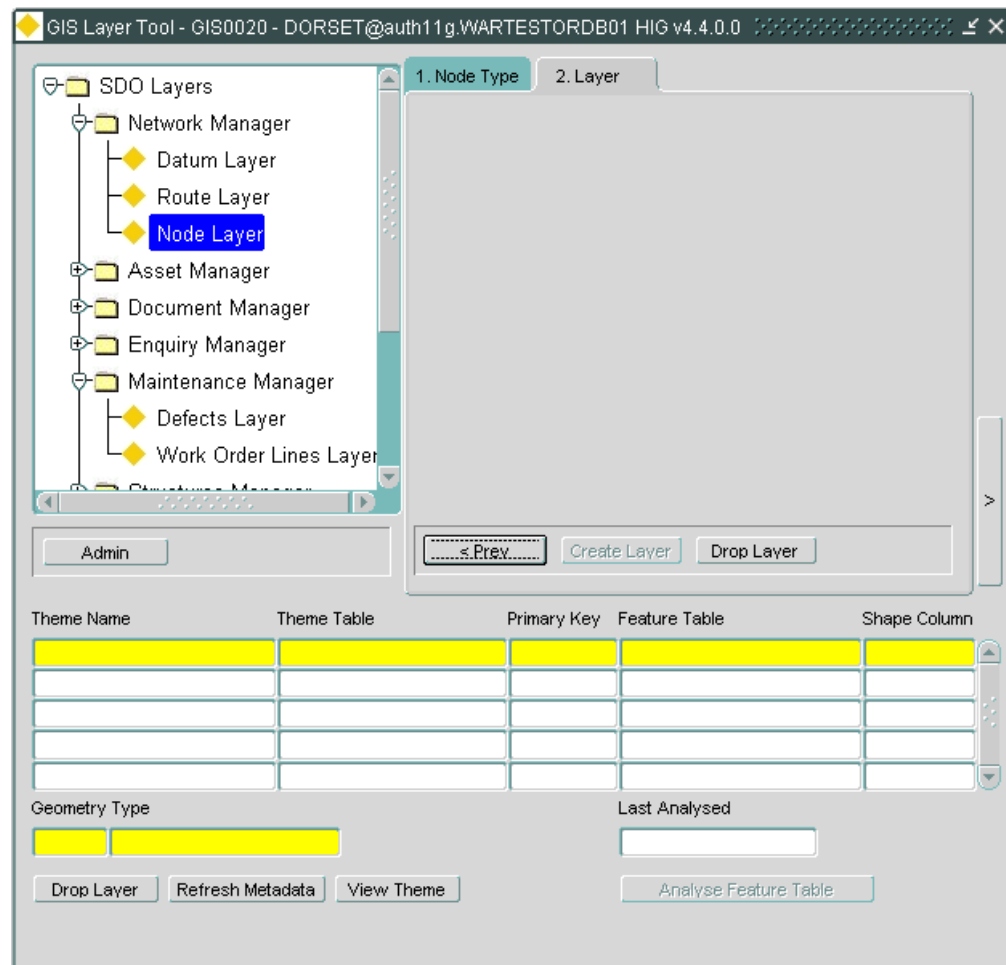
Last Analysed

Drop Layer Refresh Metadata View Theme Analyse Feature Table

2) You can either choose a specific Node type, or choose to refresh All node type layers. For a specific Node type, you can see the Themes already set up for it.



3) Go to next tab to create the layer (use “Create Layer” button or “Drop Layer”)



4) If you choose to refresh All node layers then you can expect this:

- A base table theme created for Point Locations named NM\_POINT\_LOCATIONS
- View themes created for EACH node type available. The views will be named as V\_NM\_NO\_<node\_type>\_SDO.

To populate the NM\_POINT\_LOCATIONS table with the spatial representation of the points you can click on the **[Build Point Locations]**

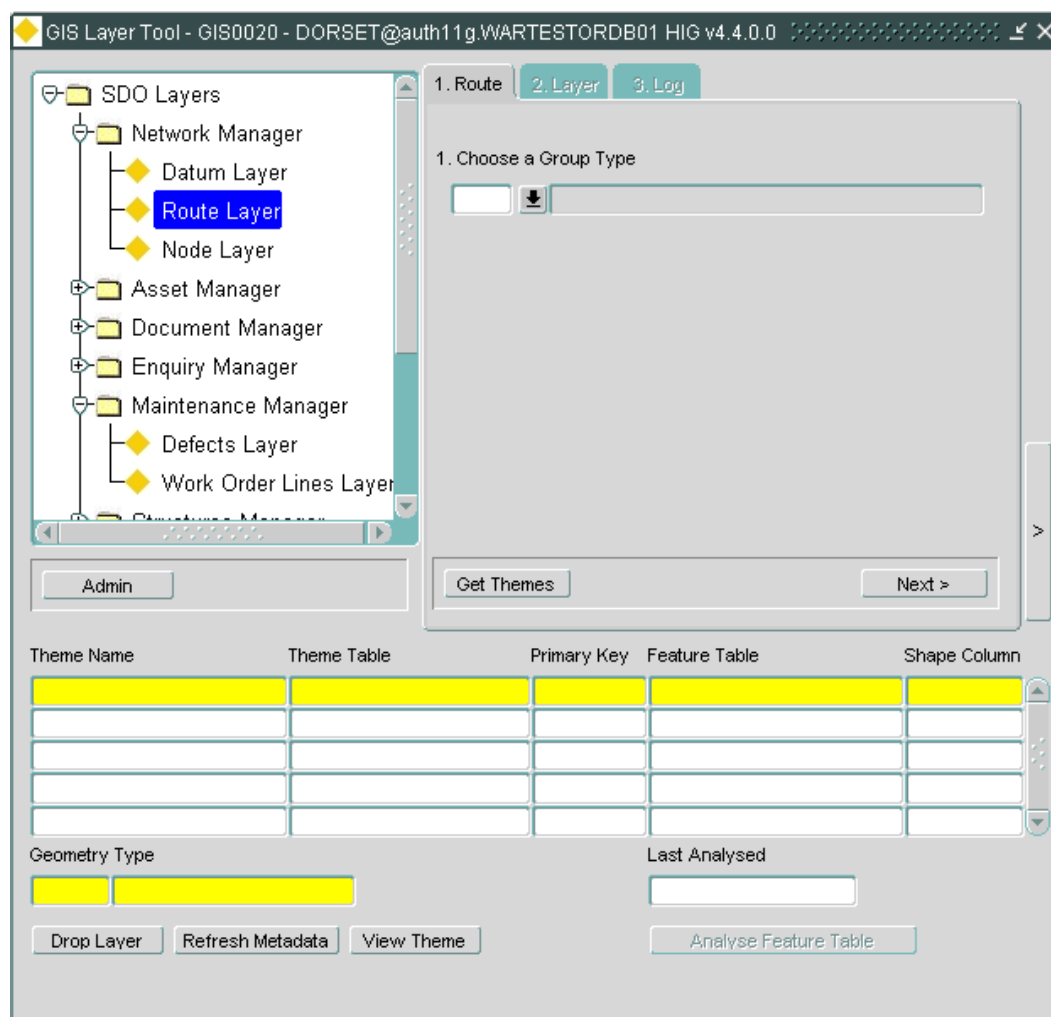
Clicking on this button will :

- Create all entries in NM\_POINT\_LOCATIONS for all points that don't have a shape.
- Drop and recreate spatial index (if present)
- Register the NM\_POINT\_LOCATIONS table in the Oracle metadata (USER\_SDO\_GEOM\_METADATA)
- Analyze the NM\_POINT\_LOCATIONS table

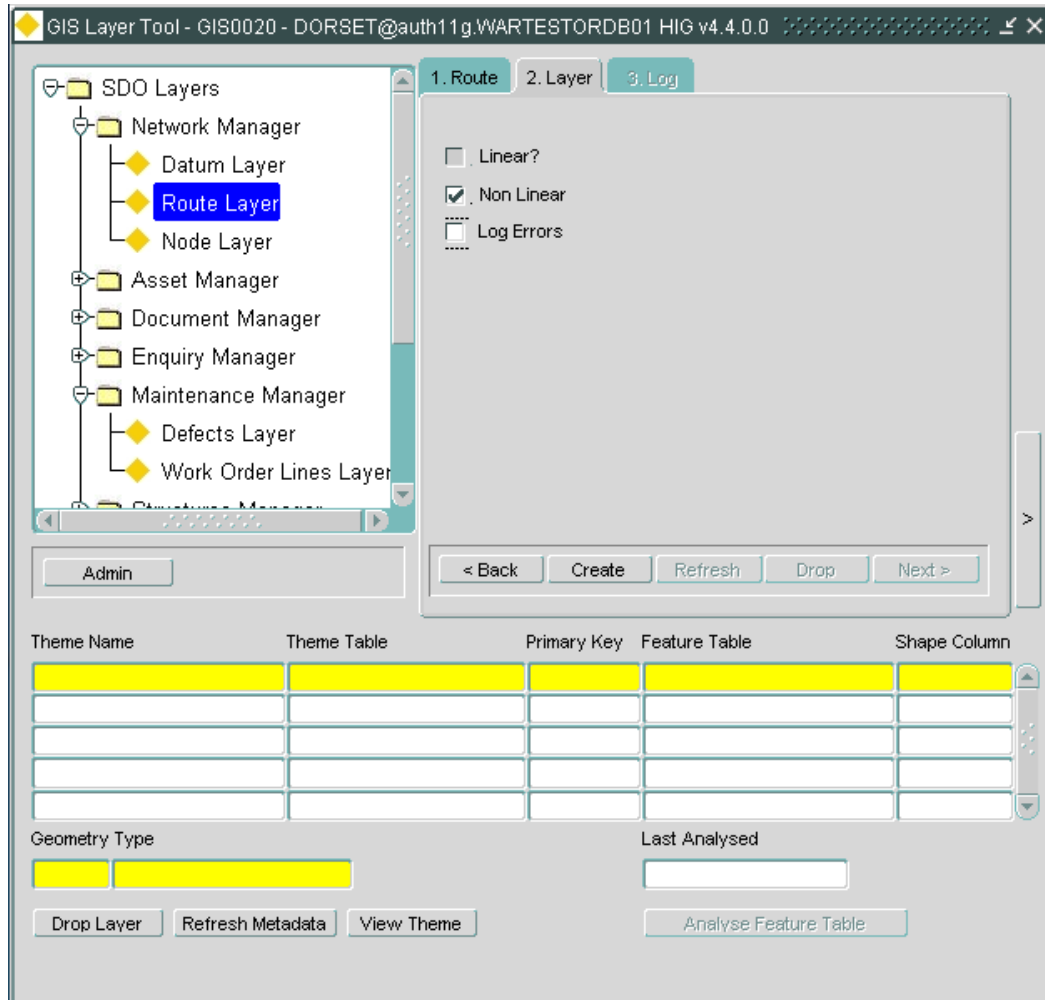
## Creating Group Layers

To create a Group (Route) layer:

1. Choose “Network Manager” > “Route Layer” from the Tree menu:



2. In the Wizard area, choose a Group type from the drop down menu.
3. If layers already exist, they will be displayed in the Details area. Note that you have the option to drop them using [Drop Layer] button. Click on [Next]



4. Select to create a linear or non-linear layer as appropriate. Note that if the Group type is non-linear anyway the [Linear] check box will be disabled. You can also choose to Log any errors encountered during layer creation (dynamic segmentation process) by checking the [Log Errors] checkbox.
5. Click on the [Create Layer] button to create the layer

For linear Group types this procedure will generate the spatial data by tracing and concatenating the datum elements and will create one table and two views for each Group Type. These are:

- NM\_NLT\_<nt\_type>\_<gty\_type>\_SDO -the base spatial table containing all the group shape
- V\_NM\_NLT\_<nt\_type>\_<gty\_type>\_SDO – View containing date-tracked group shapes
- V\_NM\_NLT\_<nt\_type>\_<gty\_type>\_SDO\_DT - View containing date-tracked shapes and network attributes

Note that when a linear group layer is created each group is represented by a single measured geometry which is the result of the concatenation of the member datum element shapes calibrated to the group's length and units.

For non-linear Group types the tables and views generated will follow these naming conventions:

- NM\_NAT\_<nt\_type>\_<gty\_type>\_SDO -the base spatial table containing all the group shape
- V\_NM\_NAT\_<nt\_type>\_<gty\_type>\_SDO – View containing date-tracked group shapes
- V\_NM\_NAT\_<nt\_type>\_<gty\_type>\_SDO\_DT - View containing date-tracked shapes and network attributes

The tables and views are registered in the SDE and MDSYS metadata tables and all relevant spatial indexes are created.

#### Important Notes:

- The procedure will fail if the table or views already exist. To drop the spatial table and views see the Dropping SDO Layers section.
- The \* \_DT view will only be created if the system option SDODATEVW is set to 'Y'. This **MUST** be the default value if the site uses SM since it only uses the date-tracked views.
- In order for the spatial tables and views to be registered with the SDE metadata the product option REGSDELAY must be set to 'Y' (yes)
- To be able to see the layers in SM you will need to assign roles to the layers using the GIS0010 form For more information on using the GIS0010 form please refer to the *Using the GIS0010 form chapter*.

## Non-Linear groups and Locator

Recent changes in Atlas for generation of Group layers means that Group SDO views use the Group type view

e.g. V\_NM\_NSGN\_OFFN

instead of the Network type (datum) view

V\_NM\_NSGN

The Group type view is non-key preserved. On the other hand, Locator displays layers using the ROWID column as the default key column and this will cause ORA-01445 errors:

ORA-01445: cannot select ROWID from a join view without a key-preserved table

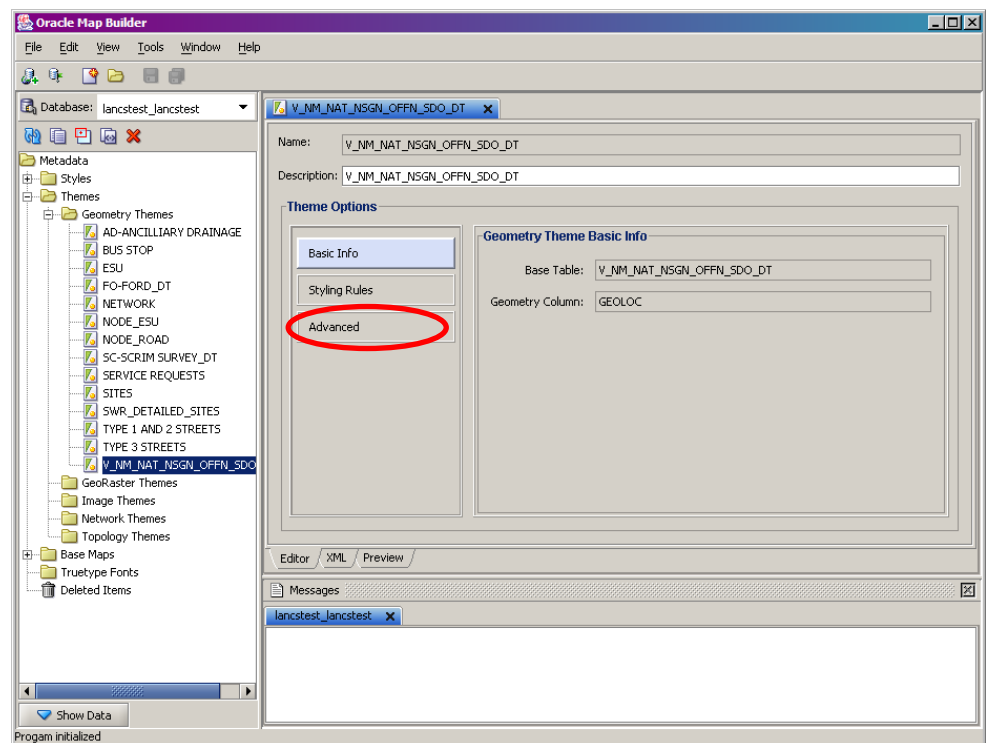
To fix this you will need to manually update the Key column attribute for the Theme(s) in Mapbuilder. To do this:

Start Mapbuilder

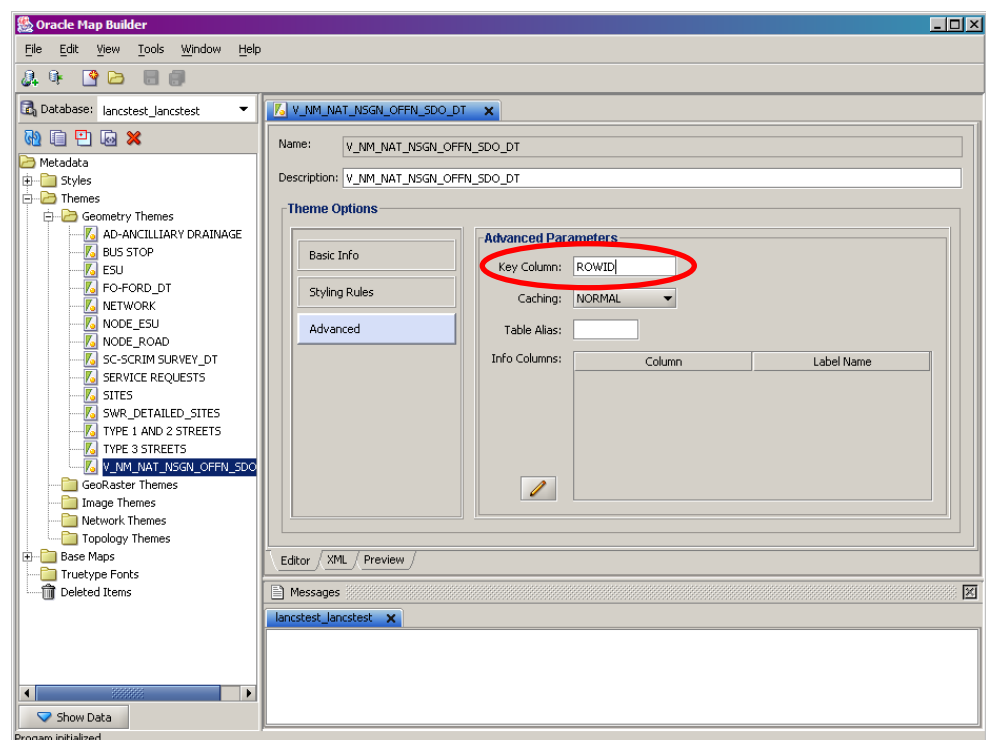
From the tree menu select Metadata->Themes->Geometry Themes-><Theme Name e.g. V\_NM\_NAT\_NSGN\_OFFN\_DT>

Right-click on the theme name and select [Open]

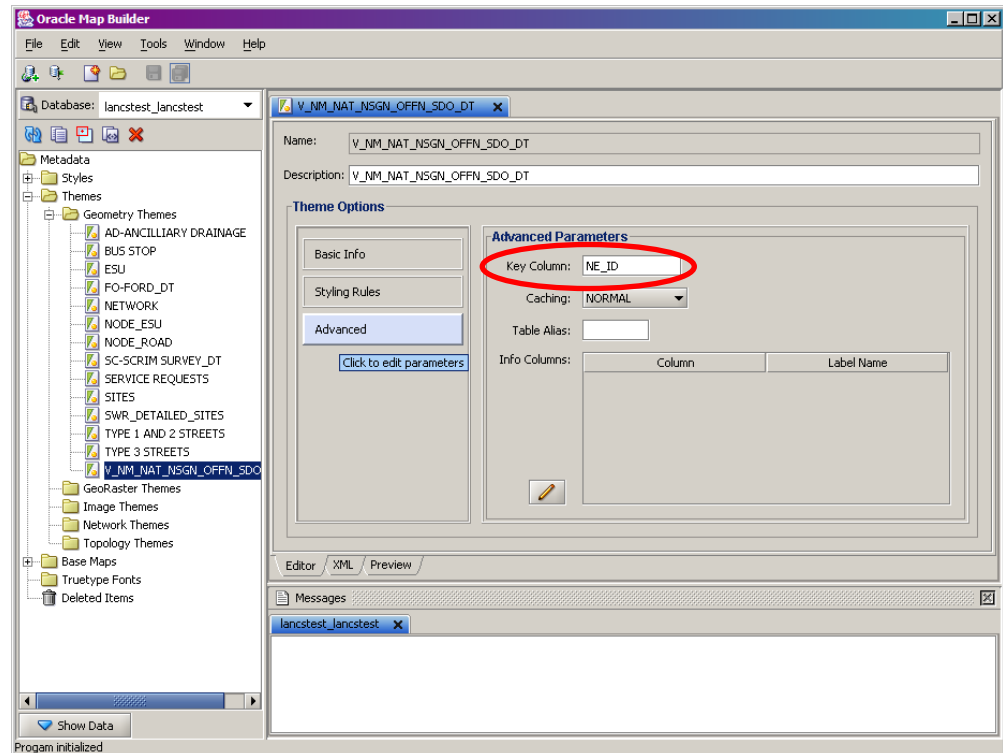
The Theme Properties Tab will display as shown below:



Click on The [Advanced] button



Click on the Key column field and overtype ROWID with NE\_ID



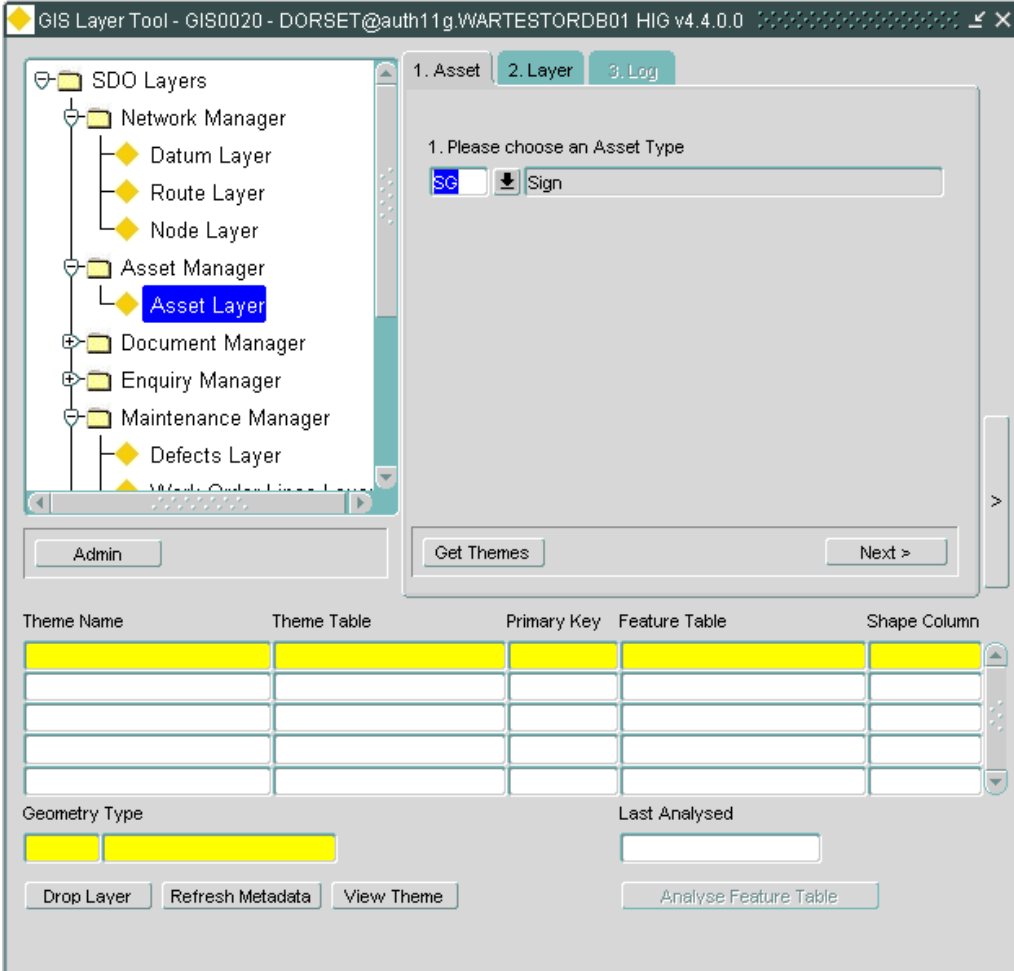
Click on the [Save] button and exit Mapviewer.

## Creating Asset Layers

You can create two types of asset layers. The first is the 'standard' asset layer created using dynamic segmentation routines. The assets will get created along the datum centerline based on their start and end positions henceforth called '**dynsegged assets**'. Or, you may create an asset which does not have a network location- an '**off-network**' asset. The following paragraphs will guide you through the creation of both types using new GIS0020 form.

### Creating Dynsegged Assets

- 1) From the GIS0020 form select "Asset Manager" > "Asset Layer" from the Tree menu:
- 2) Choose Asset Type from the drop-down list. Press [Tab] on your keyboard to enable the [Next] button and press on [Next]



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01 HIG v4.4.0.0

1. Asset 2. Layer 3. Log

1. Please choose an Asset Type

SC Sign

Get Themes Next >

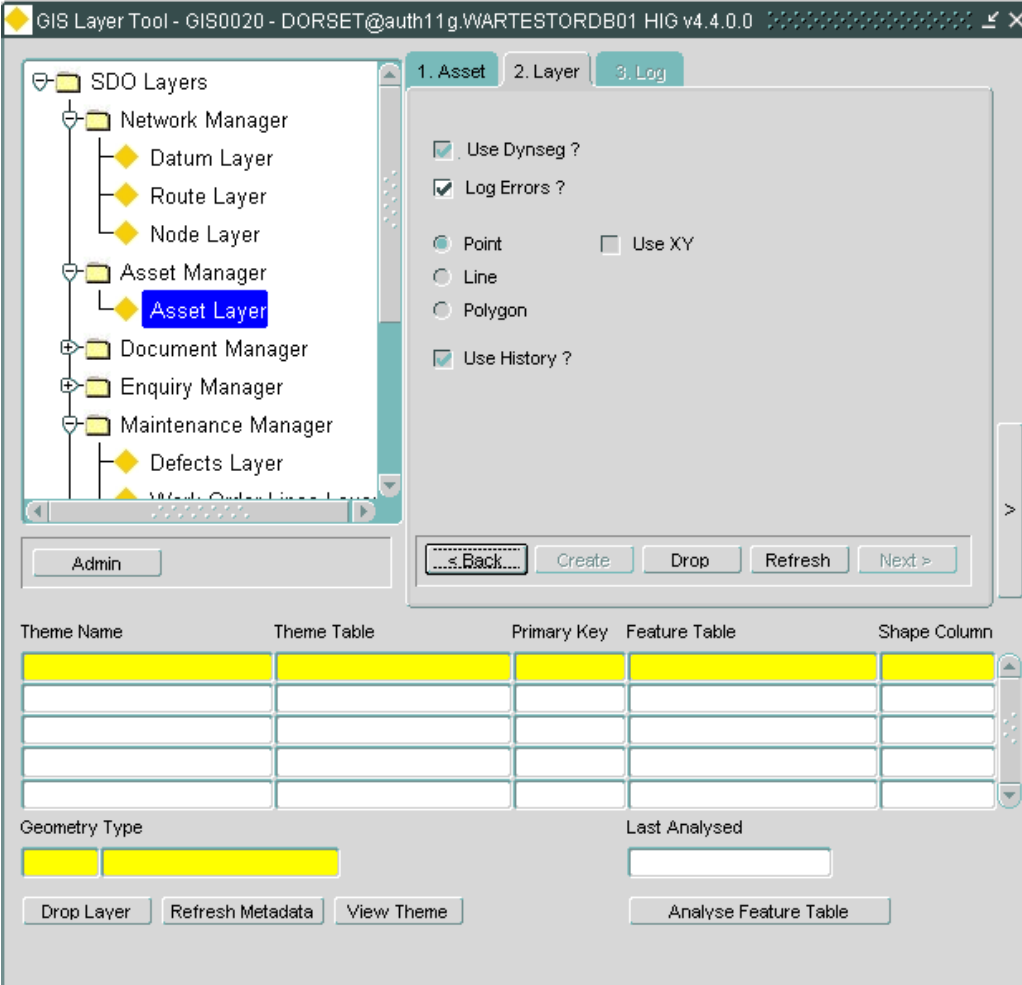
Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

Geometry Type Last Analysed

Drop Layer Refresh Metadata View Theme Analyse Feature Table

- 3) Go to next tab and check 'Use Dynseg'. Note that you will be prevented from entering the layer type (Point, Line or Polygon) as this is already defined by the Asset type – Point or Continuous. You can

also check the [Log Errors] check box to create a log file containing any errors that may occur during the dynamic segmentation process.



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01 HIG v4.4.0.0

SDO Layers

- Network Manager
  - Datum Layer
  - Route Layer
  - Node Layer
- Asset Manager
  - Asset Layer**
- Document Manager
- Enquiry Manager
- Maintenance Manager
  - Defects Layer

1. Asset 2. Layer 3. Log

☒ Use Dynseg ?

☒ Log Errors ?

☒ Point ☐ Use XY

☐ Line

☐ Polygon

☒ Use History ?

Admin < Back Create Drop Refresh Next >

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

Geometry Type Last Analysed

Drop Layer Refresh Metadata View Theme Analyse Feature Table

4) Click on the [Create Layer] button to build the SDO Layer.

The system will generate three new layers:

- NM\_NIT\_<asset\_type>\_SDO – the base spatial table containing all the asset shapes
- V\_ NM\_NIT\_<asset\_type>\_SDO - View containing date tracked asset shapes
- V\_ NM\_NIT\_<asset\_type>\_SDO\_DT - View containing date tracked asset shapes and asset attributes

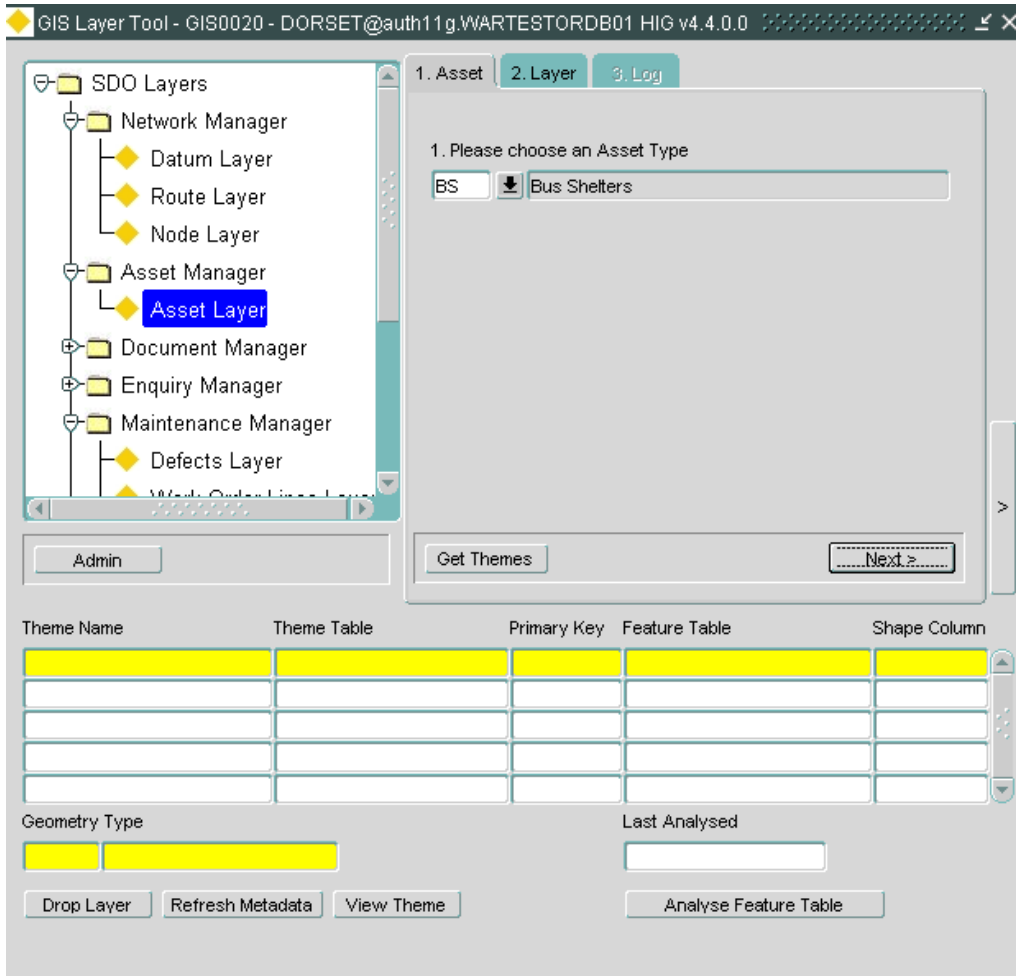


### Important Notes:

- The procedure will fail if the table or views already exist. To drop the spatial table and views see the Dropping SDO Layers section below.
- The \*\_DT view will only be created if the product option SDODATEVW is set to 'Y'. This **MUST** be the default value if the site uses SM as SM should be using only the date-tracked views.
- In order for the spatial tables and views to be registered with the SDE metadata the product option REGSDelay must be set to 'Y' (yes)
- SM users should ensure that the system options SDOSINGSHP is set to 'N' and SDOSURKEY is set to 'Y'

## Creating off-network assets

- 1) From the GIS0020 form select "Asset Manager" > "Asset Layer" from the Tree menu:
- 2) Choose Asset Type from the drop-down list. Press [Tab] on your keyboard to enable the [Next] button and press on [Next]



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01 HIG v4.4.0.0

SDO Layers

- Network Manager
  - Datum Layer
  - Route Layer
  - Node Layer
- Asset Manager
  - Asset Layer**
- Document Manager
- Enquiry Manager
- Maintenance Manager
  - Defects Layer

Admin

1. Asset 2. Layer 3. Log

1. Please choose an Asset Type

BS Bus Shelters

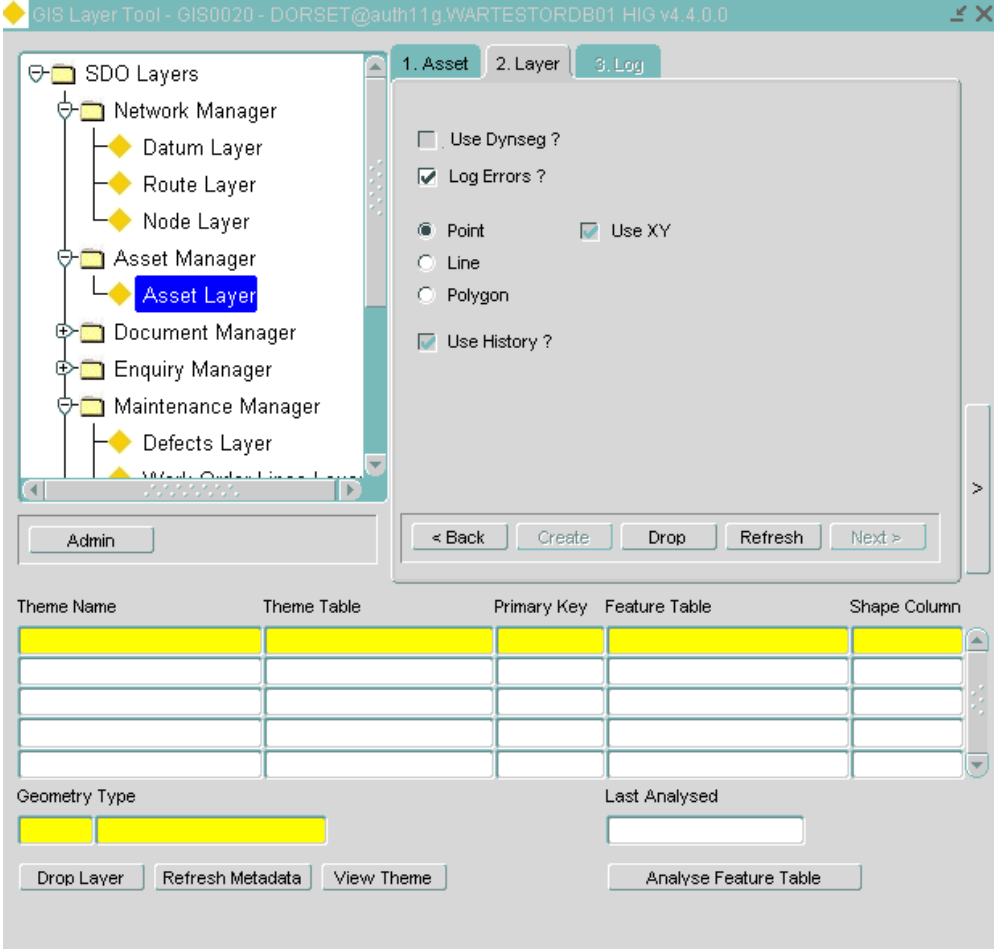
Get Themes Next >

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

Geometry Type Last Analysed

Drop Layer Refresh Metadata View Theme Analyse Feature Table

- 4) The system will default the geometry type to the type defined in the Asset Metamodel (Point or Contiguous). You can select the geometry type of the off-network asset you want to create (Point, Line or Polygon) .
- 5) Notice that the [Use History] check-box will be checked in order to create spatial views to allow storage of historic data.



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01 HIG v4.4.0.0

1. Asset 2. Layer 3. Log

☐ Use Dynseg ?

☒ Log Errors ?

☒ Point ☒ Use XY

☐ Line

☐ Polygon

☒ Use History ?

< Back Create Drop Refresh Next >

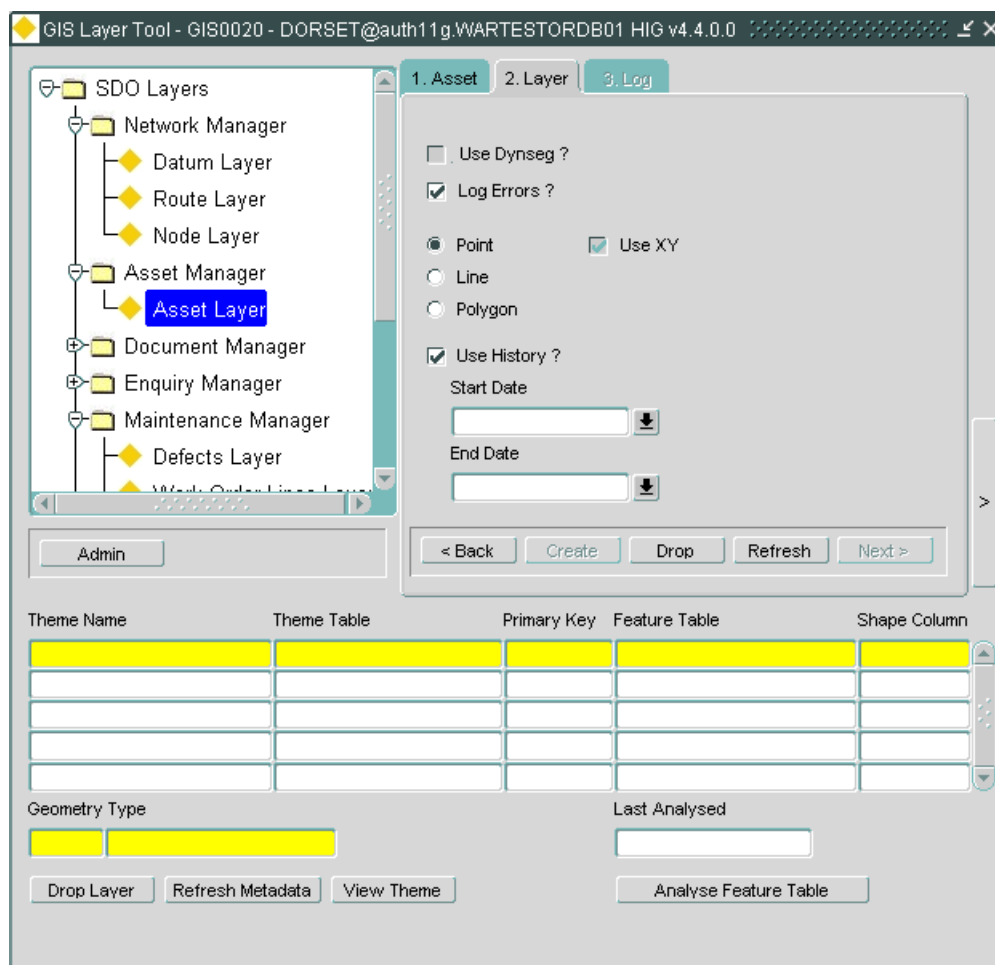
Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

Geometry Type

Last Analysed

Drop Layer Refresh Metadata View Theme Analyse Feature Table

- 7) If you use a Foreign table Asset type, selecting the [Use History] check-box will display two drop-down lists from where you can select the relevant Start and End date columns as shown in the next figure.

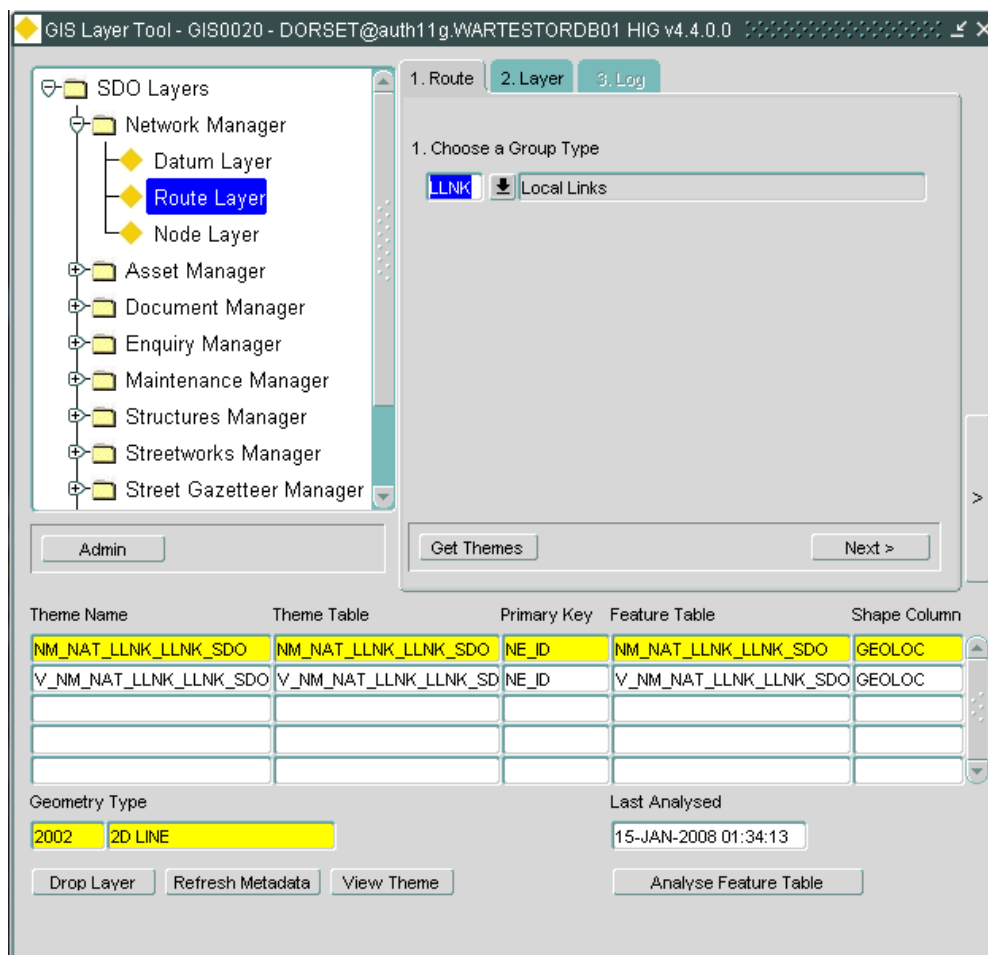


8) Click the [Create Layer] button to build the SDO Layer.

## Dropping SDO layers

You can either drop ALL or individual layers using the GIS0020 form. To drop all layers:

1. Select the type/product from the tree view. In this example Network→Route Layer
2. Select the type of layer. In this example “LLNK”
3. Click on the [Get Themes] button to display the available layers/themes



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01 HIG v4.4.0.0

1. Route 2. Layer 3. Log

1. Choose a Group Type

LLNK Local Links

Get Themes Next >

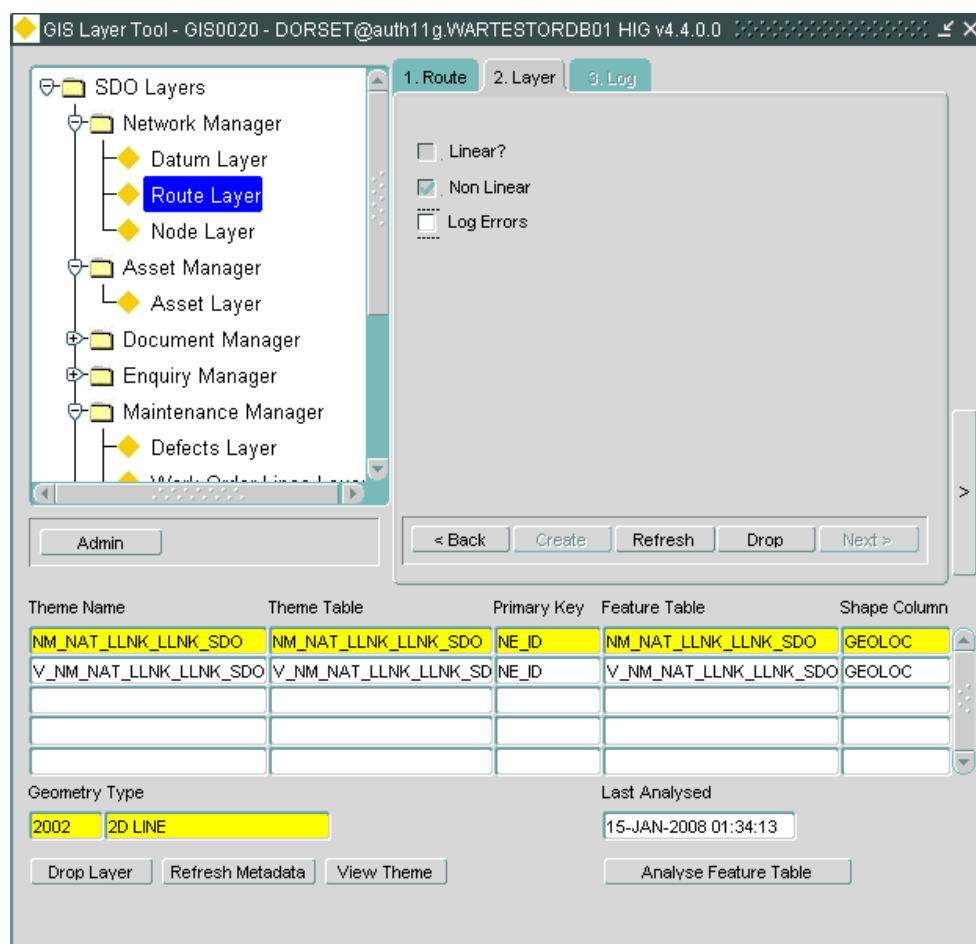
Theme Name	Theme Table	Primary Key	Feature Table	Shape Column
NM_NAT_LLNK_LLNK_SDO	NM_NAT_LLNK_LLNK_SDO	NE_ID	NM_NAT_LLNK_LLNK_SDO	GEOLOC
V_NM_NAT_LLNK_LLNK_SDO	V_NM_NAT_LLNK_LLNK_SD	NE_ID	V_NM_NAT_LLNK_LLNK_SDO	GEOLOC

Geometry Type: 2002 2D LINE

Last Analysed: 15-JAN-2008 01:34:13

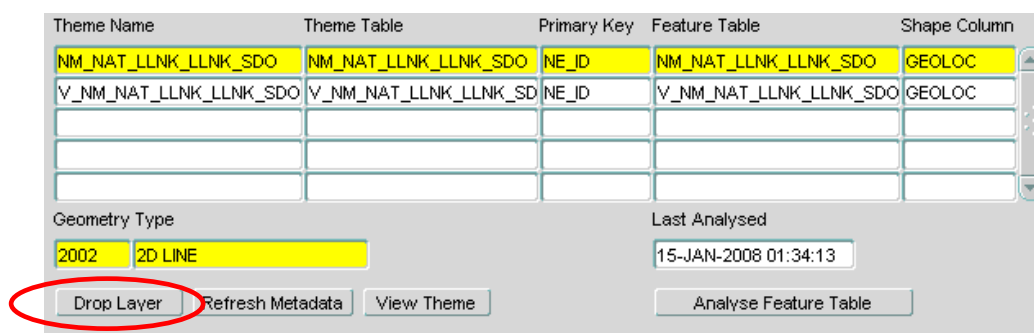
Drop Layer Refresh Metadata View Theme Analyse Feature Table

4. Click on [Next>]

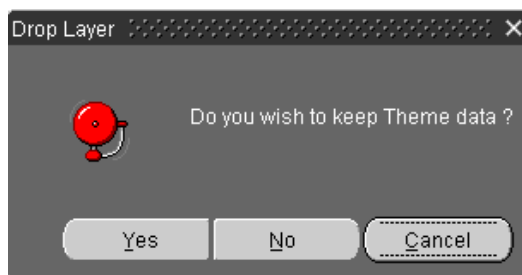


Click on the [Drop] button. This will drop ALL the displayed themes.

Alternatively you can select to drop an individual theme by pressing the [Drop Layer] button as indicated in the screenshot below:



The system will prompt you whether you also want to remove the relevant theme records.



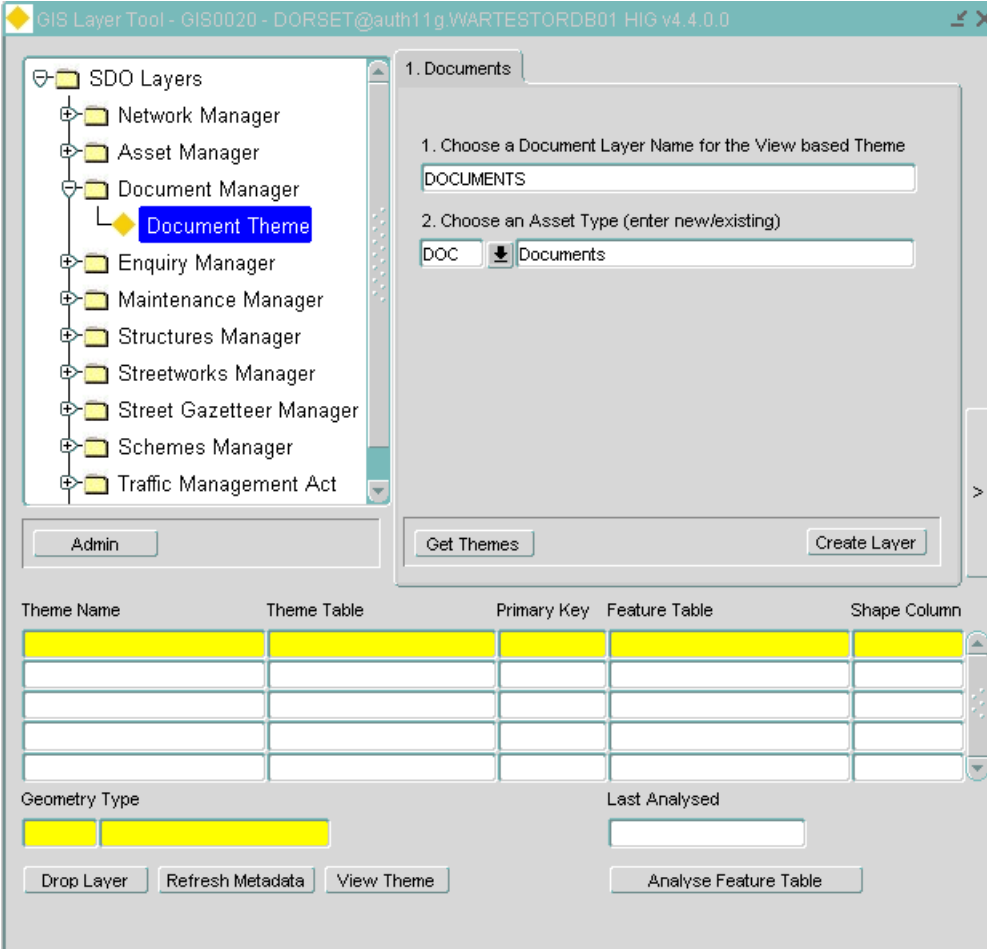
### Important Notes on dropping layers

**If the layer you are trying to drop is based on a foreign table the system will also drop the foreign table view. A message will prompt to confirm the dropping of the foreign table.**

## Creating Document Layers

To create a layer so you can display your documents on the map:

- 1) Using the GIS0020 form, choose “Document Manager” -> “Document Theme” from the Tree menu
- 2) Choose a suitable name and an Asset type to use in Locator. You can define a new asset type, which will create all the relevant metadata automatically or reuse an existing one.
- 3) Click on [Create Layer].



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01.HIG v4.4.0.0

SDO Layers

- Network Manager
- Asset Manager
- Document Manager
  - Document Theme**
- Enquiry Manager
- Maintenance Manager
- Structures Manager
- Streetworks Manager
- Street Gazetteer Manager
- Schemes Manager
- Traffic Management Act

Admin

1. Documents

1. Choose a Document Layer Name for the View based Theme  
DOCUMENTS

2. Choose an Asset Type (enter new/existing)  
DOC Documents

Get Themes Create Layer

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

Geometry Type Last Analysed

Drop Layer Refresh Metadata View Theme Analyse Feature Table

This will create the relevant spatial table and view as showing in the next screenshot:

GIS Layer Tool - HIG v4.3.0.0

SDO Layers

- Network Manager
- Asset Manager
- Enquiry Manager
- Maintenance Manager
- Streetworks Manager
- Street Gazetteer Manager
- Traffic Management Act
- Document Manager
  - Document Theme**
- Custom Layers

1. Documents

1. Choose a Document Layer Name for the View Base Theme  
DOCUMENTS

2. Choose an Asset Type (enter new/existing)  
DOC Documents

Get Themes Create Layer

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column
DOCUMENTS_TAB	DOCS	DOC_ID	DOC_DOCUMENTS_XY_SDO	GEOLOC
DOCUMENTS	V_DOC_DOCUMENTS	DOC_ID	V_DOCUMENTS_XY_SDO	GEOLOC

Geometry Type  
2001 2D POINT

Last Analysed  
26-OCT-2010 22:00:07

Drop Layer Refresh Metadata View Theme Analyse Feature Table



## Creating Accident Layers

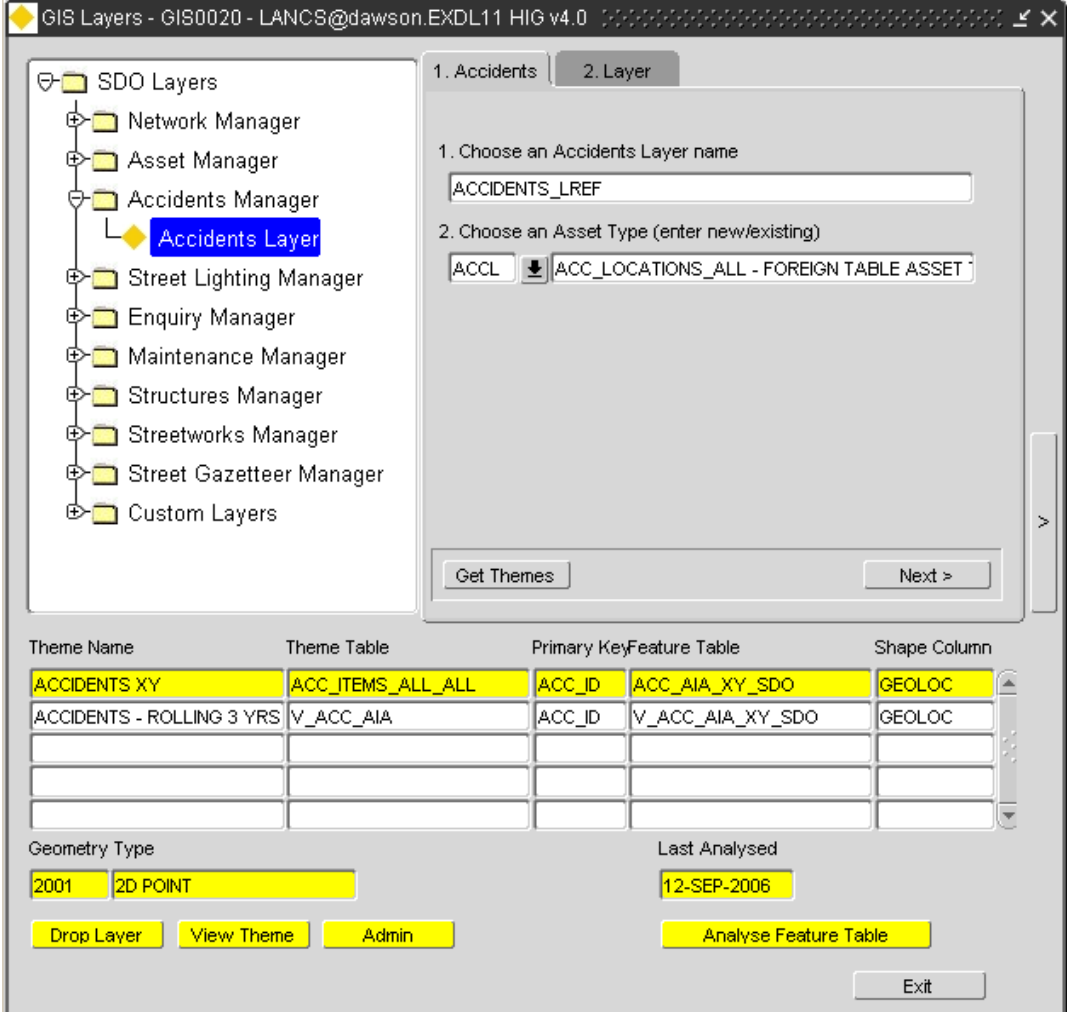
To create a layer for accident data:

- 1) Using the GIS0010 form, choose “Accidents Manager” -> “Accidents Layer” from the Tree menu

You can create an Accidents SDO layer for either

- i) data based on XY coordinates – based on ACC\_ITEMS\_ALL
- ii) data based on network reference (start chainage) – based on ACC\_LOCATIONS\_ALL

- 2) Choose a suitable name for a base table theme (i.e. ACCIDENTS\_LREF) in this case and a suitable Asset type name. You can define a new asset type, which will create all the relevant metadata automatically or reuse an existing one. Note that in this case the asset type must be defined as a Foreign table type. Click on [Next]



GIS Layers - GIS0020 - LANCBS@dawson.EXDL11 HIG v4.0

SDO Layers

- Network Manager
- Asset Manager
- Accidents Manager
  - Accidents Layer**
- Street Lighting Manager
- Enquiry Manager
- Maintenance Manager
- Structures Manager
- Streetworks Manager
- Street Gazetteer Manager
- Custom Layers

1. Accidents 2. Layer

1. Choose an Accidents Layer name  
ACCIDENTS\_LREF

2. Choose an Asset Type (enter new/existing)  
ACCL [Select] ACC\_LOCATIONS\_ALL - FOREIGN TABLE ASSET

Get Themes Next >

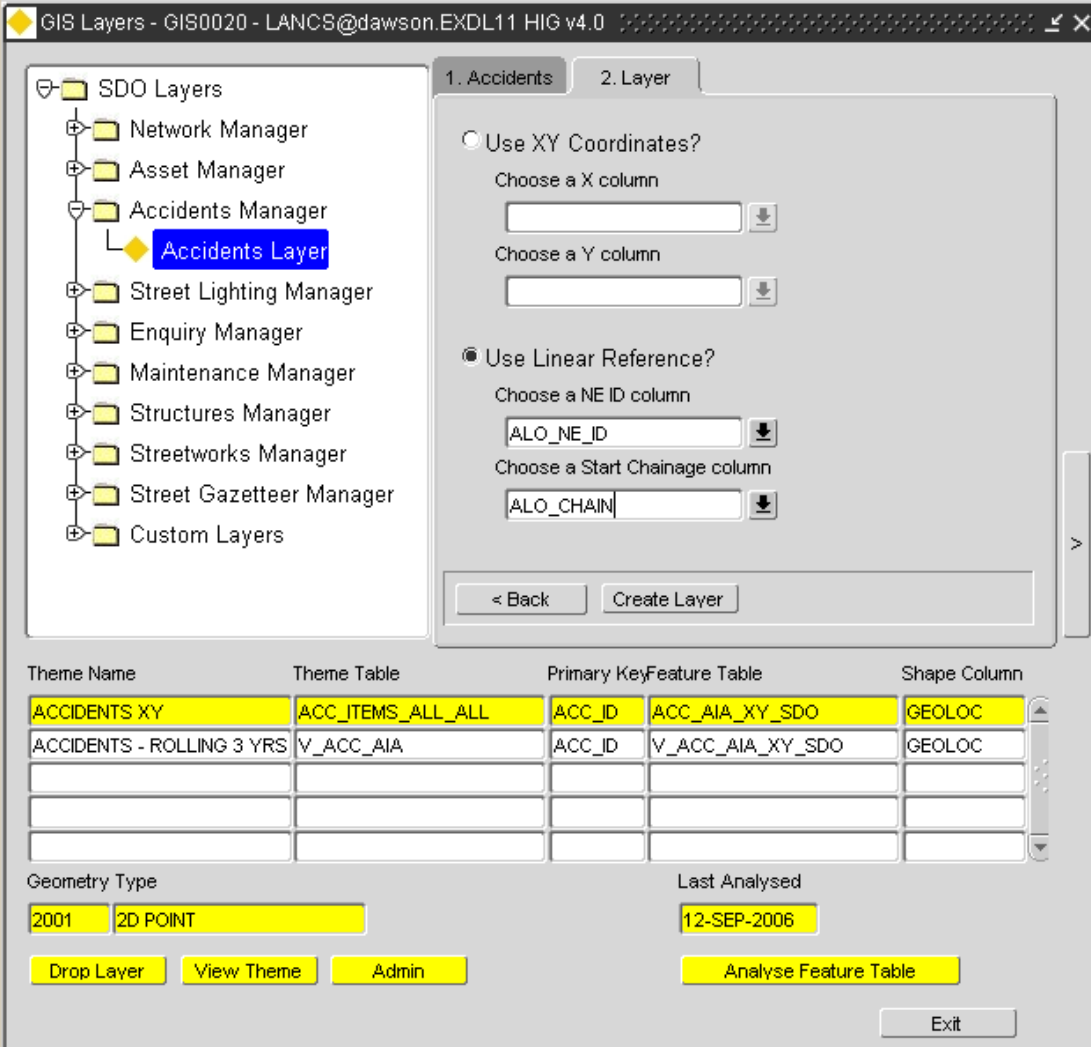
Theme Name	Theme Table	Primary Key	Feature Table	Shape Column
ACCIDENTS XY	ACC_ITEMS_ALL_ALL	ACC_ID	ACC_AIA_XY_SDO	GEOLOC
ACCIDENTS - ROLLING 3 YRS	V_ACC_AIA	ACC_ID	V_ACC_AIA_XY_SDO	GEOLOC

Geometry Type  
2001 2D POINT

Last Analysed  
12-SEP-2006

Drop Layer View Theme Admin Analyse Feature Table Exit

- 2) Select the relevant columns. If you choose to create you layer based on XY coordinates you must define the X and Y columns. If you choose to create the layer based on network reference, select the [NE\_ID] column (the column in your base table that contains the unique value for the network element) and the [Start Chainage] column (the column that contains the offset value from the start of the network element)



GIS Layers - GIS0020 - LANCS@dawson.EXDL11 HIG v4.0

1. Accidents 2. Layer

☐ Use XY Coordinates?

Choose a X column

Choose a Y column

☒ Use Linear Reference?

Choose a NE ID column

Choose a Start Chainage column

< Back Create Layer

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column
ACCIDENTS XY	ACC_ITEMS_ALL_ALL	ACC_ID	ACC_AIA_XY_SDO	GEOLOC
ACCIDENTS - ROLLING 3 YRS	V_ACC_AIA	ACC_ID	V_ACC_AIA_XY_SDO	GEOLOC

Geometry Type: 2001 2D POINT

Last Analysed: 12-SEP-2006

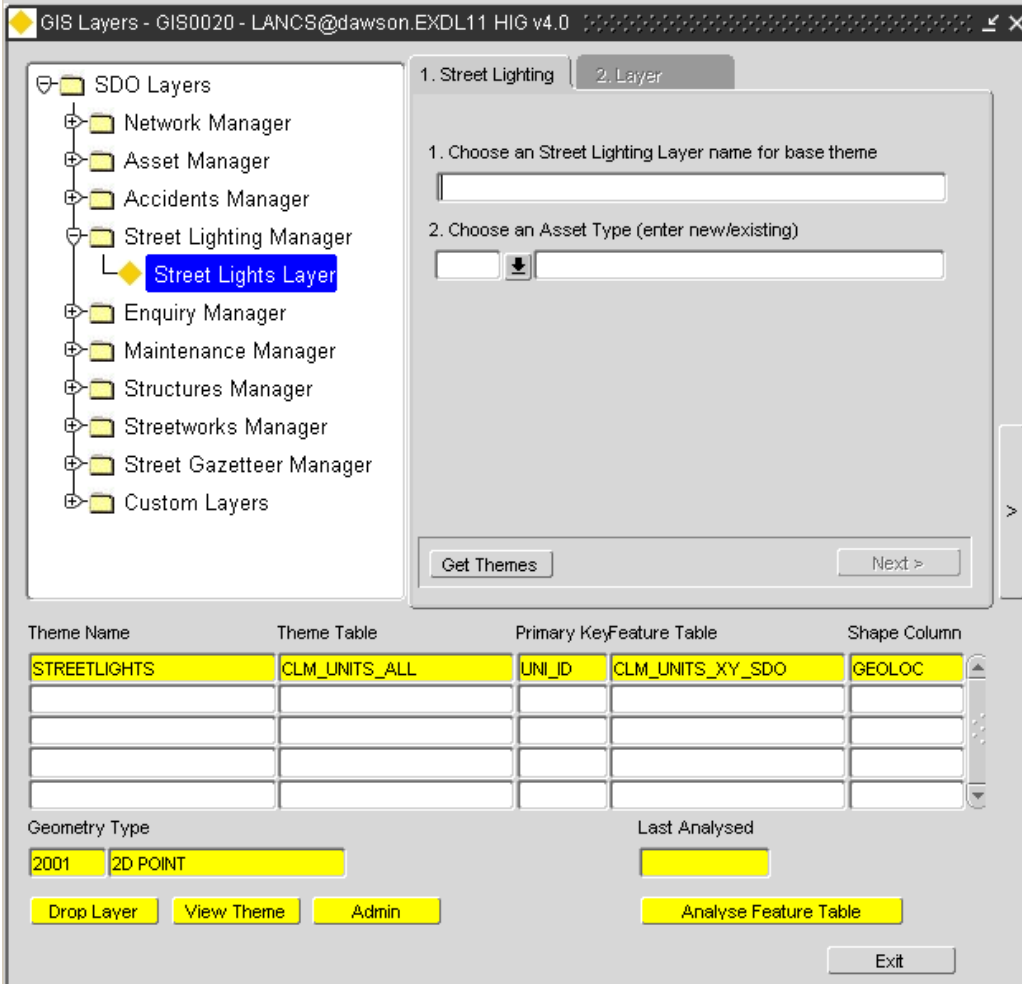
Drop Layer View Theme Admin Analyse Feature Table Exit

- 3) Press the [Create Layer] button to create the SDO layer. After the process completes the Details area of the form will be populated with details of the Base table theme with the name you have entered, plus another theme banding the last 3 yrs worth of Accidents. This would be the Theme suitable for use in Locator.

## Creating Street Lighting layers

To create a layer for street lighting data:

- 1) Choose “Street Lighting Manager” -> “Street Lights Layer” from the Tree menu:



GIS Layers - GIS0020 - LANCS@dawson.EXDL11 HIG v4.0

1. Street Lighting    2. Layer

1. Choose an Street Lighting Layer name for base theme

2. Choose an Asset Type (enter new/existing)

Get Themes    Next >

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column
STREETLIGHTS	CLM_UNITS_ALL	UNI_ID	CLM_UNITS_XY_SDO	GEOLOC

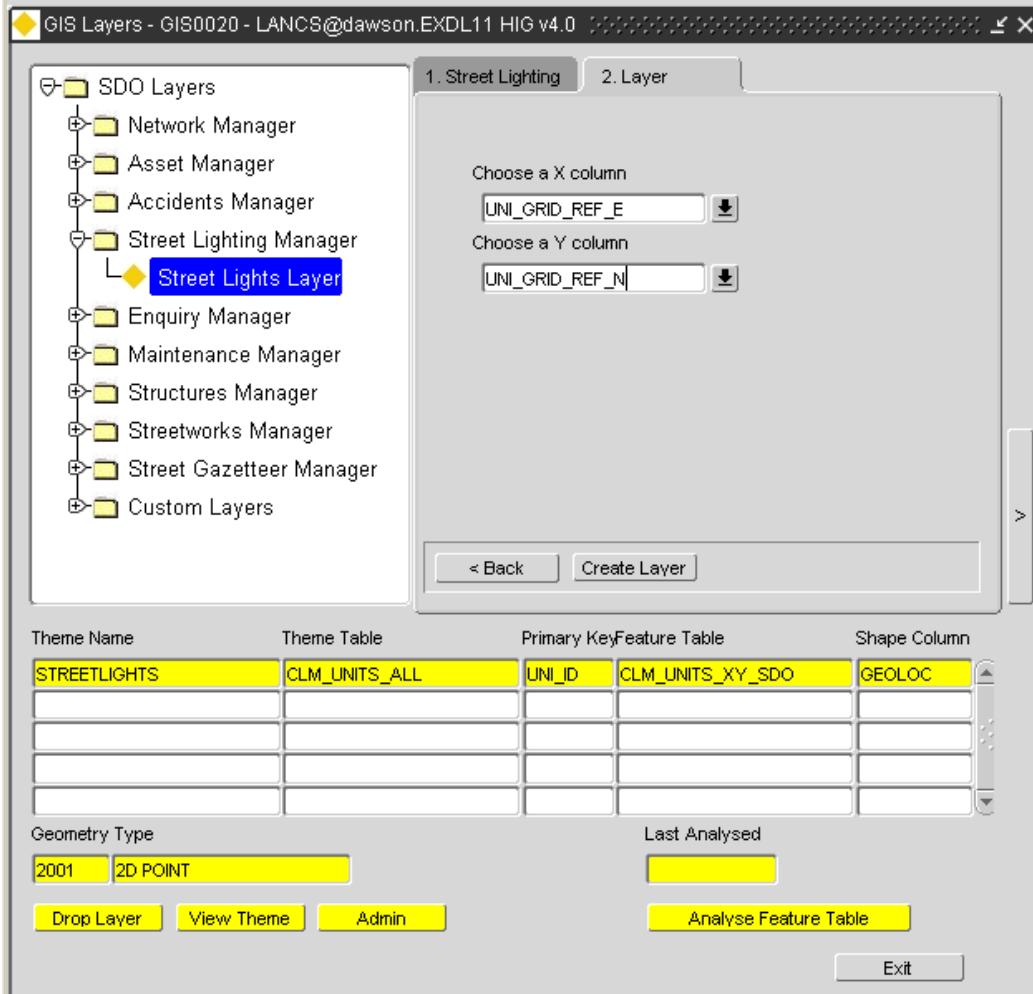
Geometry Type: 2001    2D POINT

Last Analysed:   

Drop Layer    View Theme    Admin    Analyse Feature Table    Exit

- 2) Choose a suitable name and an Asset type to use in Locator. You can define a new asset type, which will create all the relevant metadata automatically or reuse an existing one. Press on [Next]

3) Select the columns that hold the X and Y values from the drop-down list and press the [Create Layer] button.



The screenshot shows the 'GIS Layers - GIS0020 - LANCS@dawson.EXDL11 HIG v4.0' window. On the left, a tree view under 'SDO Layers' has 'Street Lights Layer' selected. The main panel has two tabs: '1. Street Lighting' and '2. Layer'. In the '2. Layer' tab, there are two dropdown menus: 'Choose a X column' with 'UNI\_GRID\_REF\_E' selected, and 'Choose a Y column' with 'UNI\_GRID\_REF\_N' selected. Below these are '< Back' and 'Create Layer' buttons. At the bottom, there is a table with columns: Theme Name, Theme Table, Primary Key, Feature Table, and Shape Column. The first row is filled with 'STREETLIGHTS', 'CLM\_UNITS\_ALL', 'UNI\_ID', 'CLM\_UNITS\_XY\_SDO', and 'GEOLOC'. Below the table are 'Geometry Type' (2001, 2D POINT) and 'Last Analysed' fields. At the bottom are buttons: 'Drop Layer', 'View Theme', 'Admin', 'Analyse Feature Table', and 'Exit'.

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column
STREETLIGHTS	CLM_UNITS_ALL	UNI_ID	CLM_UNITS_XY_SDO	GEOLOC

Geometry Type: 2001, 2D POINT

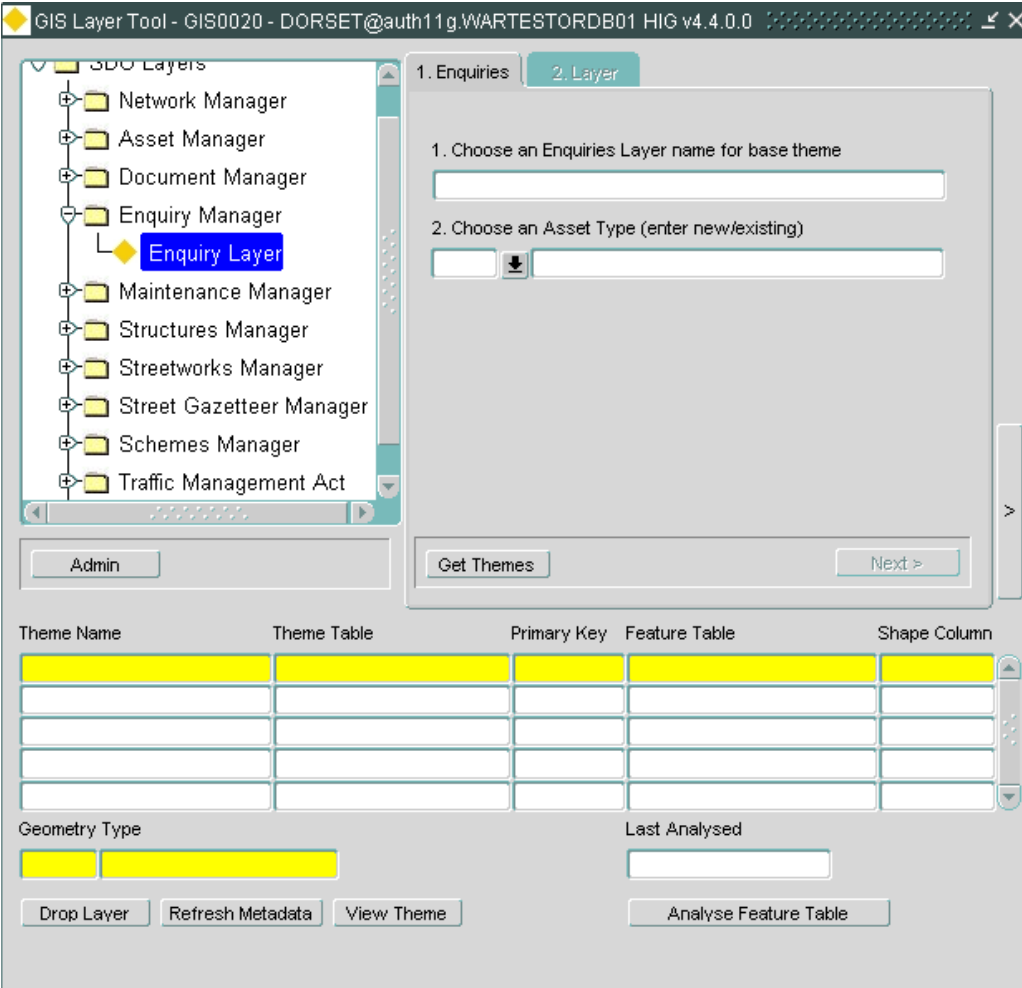
Last Analysed:

Buttons: Drop Layer, View Theme, Admin, Analyse Feature Table, Exit

## Creating Enquiries Layers

To create layers for Enquiries data:

1) Choose “Enquiries Manager” -> “Enquiries Layer” from the Tree menu:



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01 HIG v4.4.0.0

1. Enquiries 2. Layer

1. Choose an Enquiries Layer name for base theme

2. Choose an Asset Type (enter new/existing)

Admin

Get Themes Next >

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

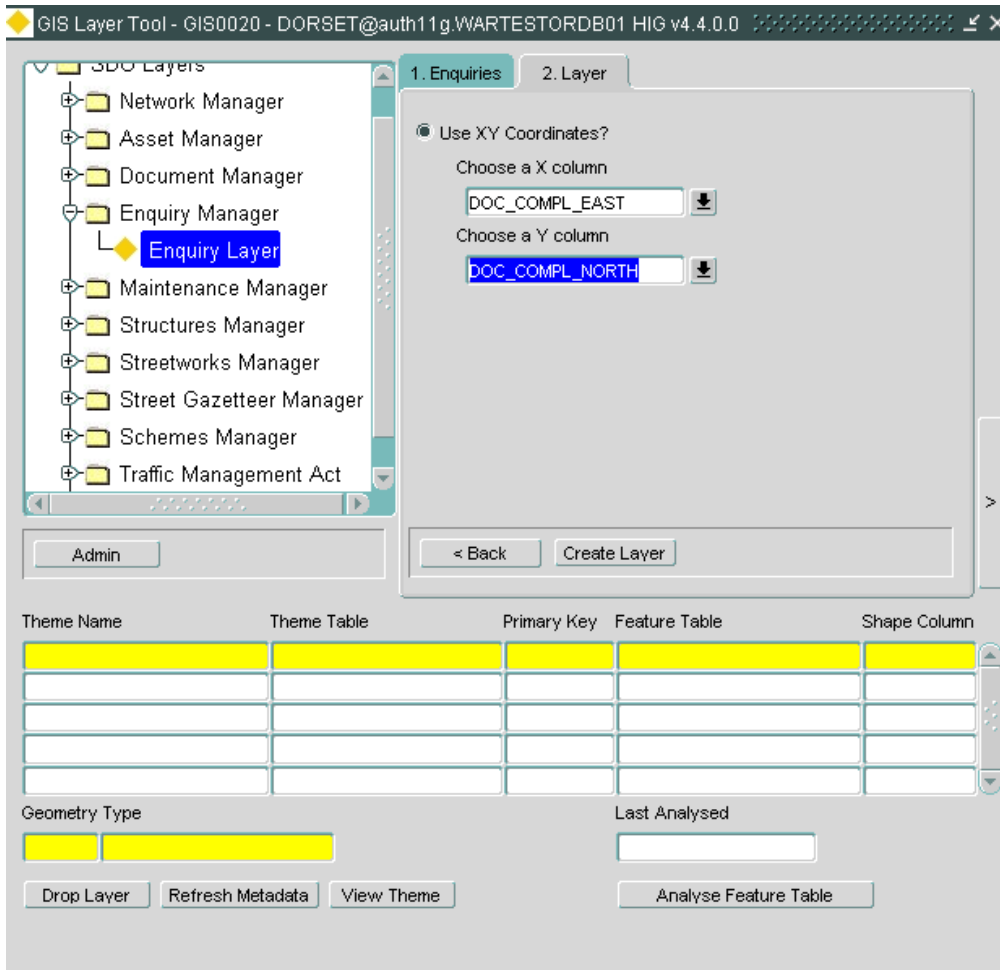
Geometry Type

Last Analysed

Drop Layer Refresh Metadata View Theme Analyse Feature Table

2) Choose a suitable name and an Asset type to use in Locator. You can define a new asset type, which will create all the relevant metadata automatically or reuse an existing one. Press on [Next]

- 3) Select the columns that hold the X and Y values from the drop-down list and press the [Create Layer] button.



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01 HIG v4.4.0.0

1. Enquiries 2. Layer

☒ Use XY Coordinates?

Choose a X column  
DOC\_COMPL\_EAST

Choose a Y column  
DOC\_COMPL\_NORTH

< Back Create Layer

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

Geometry Type  
Last Analysed

Drop Layer Refresh Metadata View Theme Analyse Feature Table

- 4) This will create a spatial table for the base table Theme created with the name you specified and the following views:

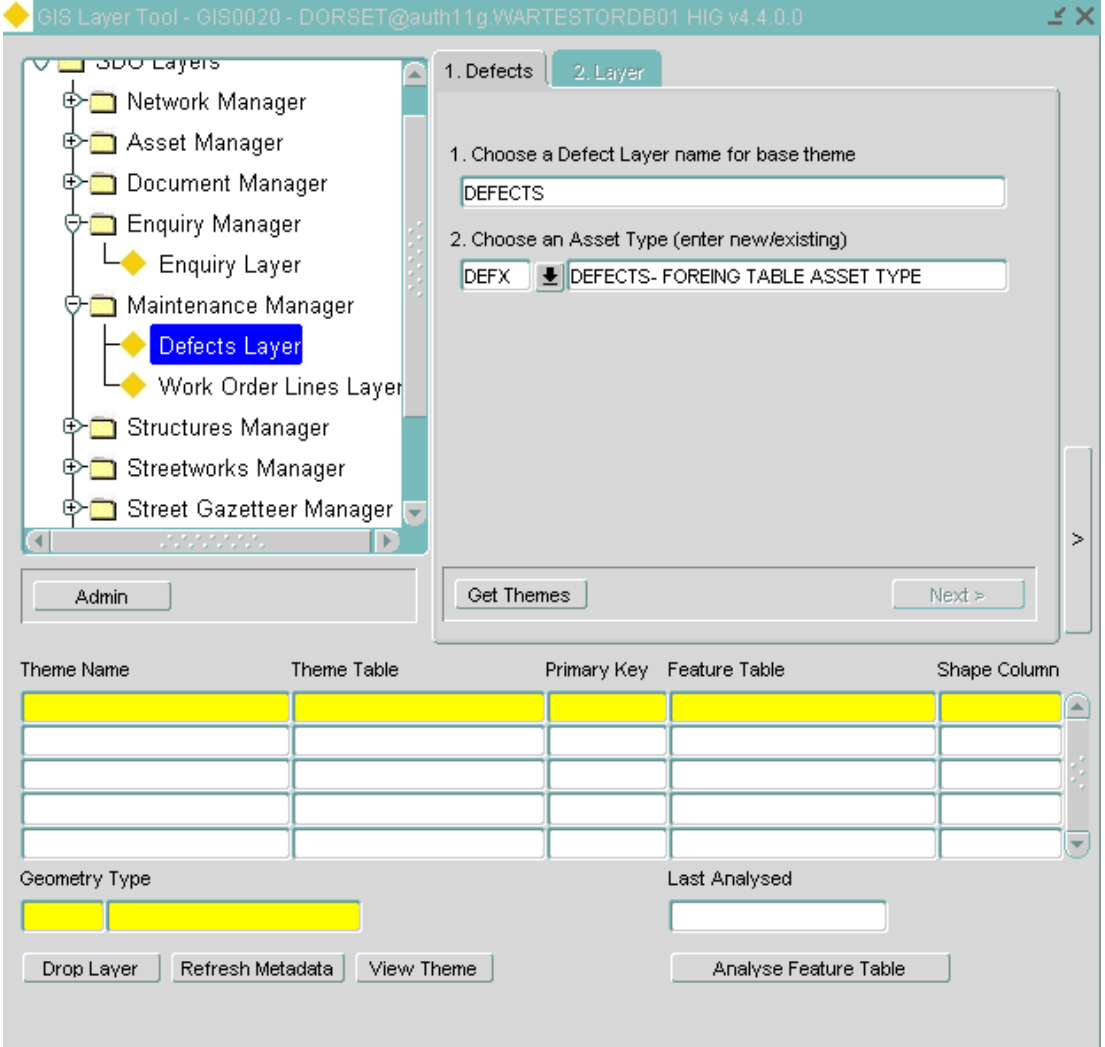
- i) Enquiries By Status
- ii) Enquiries By Source
- iii) Enquiries By Status & Source

These views can be used for banding in Locator. This will also set the SDOPEMNTNTH product option used in Locator, to the Theme ID of the base table Theme.

## Creating Defects Layers

To create layers for Defects data:

- 1) From the GIS0020 form choose “Maintenance Manager”-> “Defects Layer” from the tree menu:



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01 HIG v4.4.0.0

1. Defects 2. Layer

1. Choose a Defect Layer name for base theme  
DEFECTS

2. Choose an Asset Type (enter new/existing)  
DEFX DEFECTS- FOREIGN TABLE ASSET TYPE

Get Themes Next >

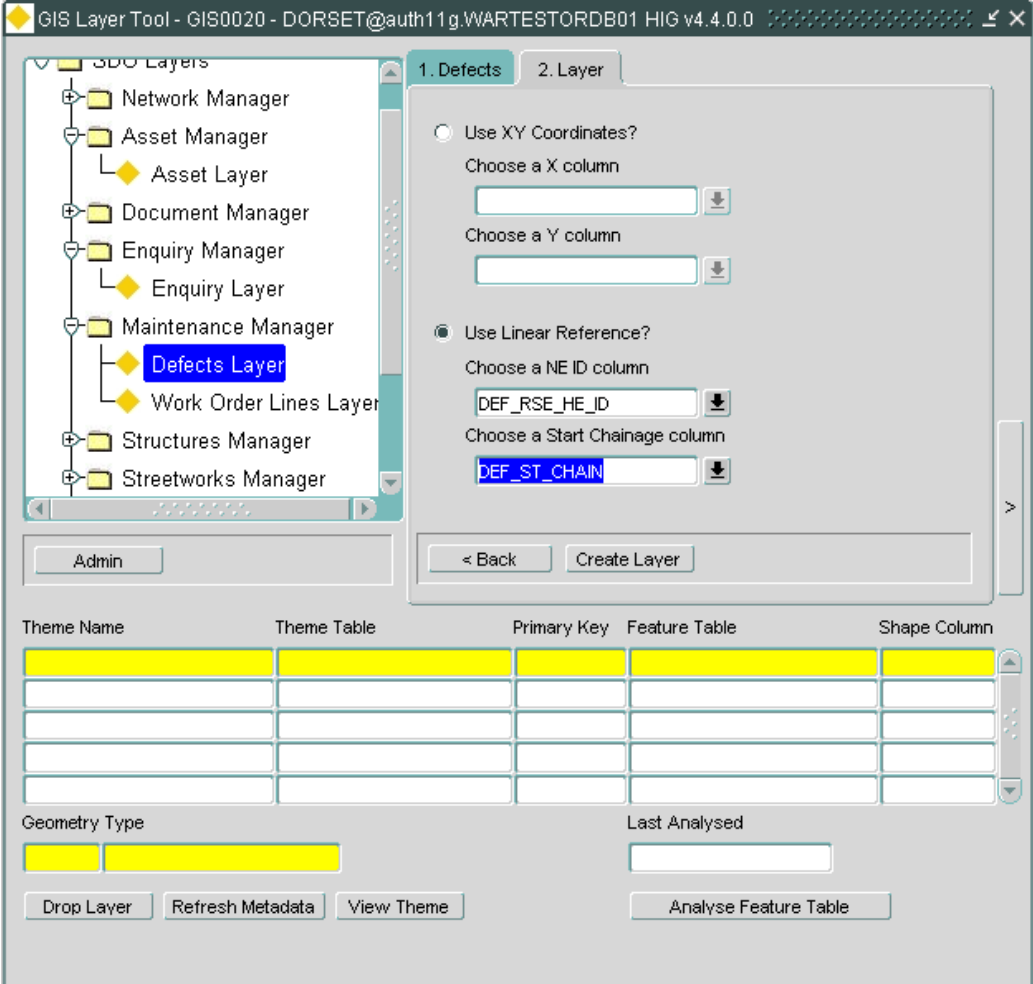
Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

Geometry Type Last Analysed

Drop Layer Refresh Metadata View Theme Analyse Feature Table

- 2) Choose a suitable name and an Asset type to use in Locator. You can define a new asset type, which will create all the relevant metadata automatically or reuse an existing one. Note that in this case the asset type must be defined as a Foreign table type. Click on [Next].

- 3) Select the relevant columns. If you choose to create you layer based on XY coordinates you must define the X and Y columns. If you choose to create the layer based on network reference, select the [NE\_ID] column (the column in your base table that contains the unique value for the network element) and the [Start Chainage] column (the column that contains the offset value from the start of the network element).



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01 HIG v4.4.0.0

**1. Defects**    2. Layer

☐ Use XY Coordinates?  
 Choose a X column  
  
 Choose a Y column

☒ Use Linear Reference?  
 Choose a NE ID column  
  
 Choose a Start Chainage column

< Back    Create Layer

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

Geometry Type:

Last Analysed:

Drop Layer    Refresh Metadata    View Theme    Analyse Feature Table

- 4) Press the [Create Layer] button. This process will create a spatial table for the the base table theme (with the name you have chosen) and the following spatial views:
- i) Defects By Status
  - ii) Defects By Activity
  - iii) Defects By Activity & Status

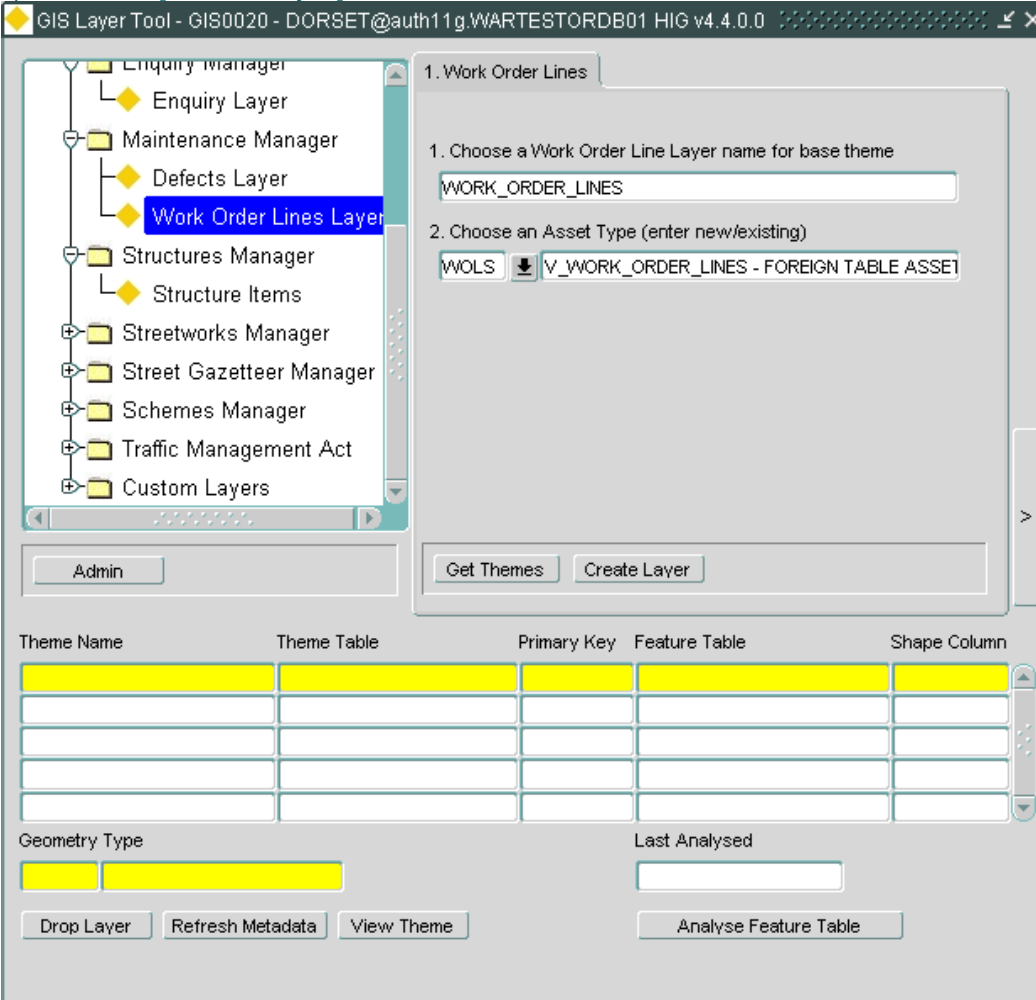
These views can then be used for banding in locator. This will also set the product option (SDODEFNTH) used in Locator to the Theme ID of the base table Theme.



## Creating Work Order Layers

To create layers for Work Orders data:

- 1) From the GIS0020 form choose “Maintenance Manager”-> Work Order Lines” from the tree menu:
- 2) Choose a suitable name and an Asset type to use in Locator. You can define a new asset type, which will create all the relevant metadata automatically or reuse an existing one.
- 3) Click on [Create Layer].



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01 HIG v4.4.0.0

1. Work Order Lines

1. Choose a Work Order Line Layer name for base theme

2. Choose an Asset Type (enter new/existing)

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

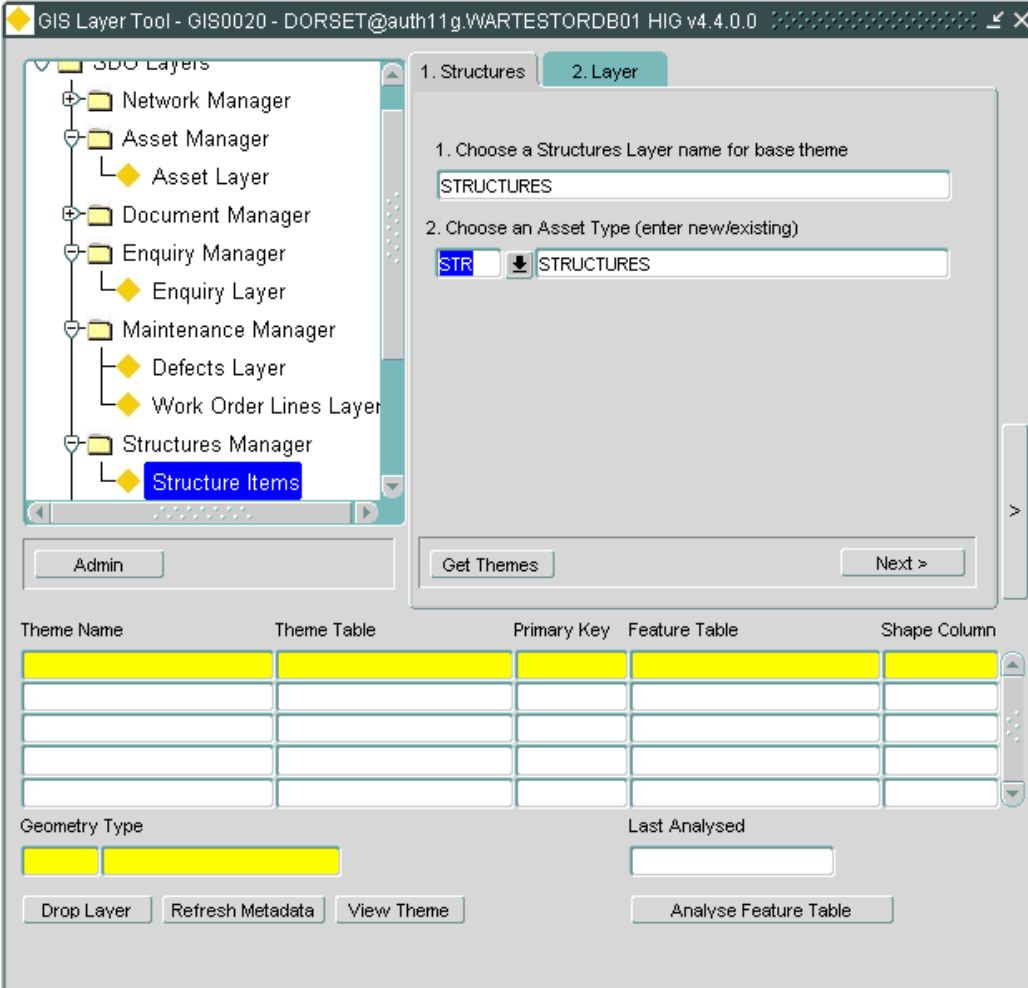
Geometry Type

Last Analysed

## Creating Structures Layers

To create layers for Structures data:

- 1) From the GIS0020 form choose “Structures Manager” -> “Structures Layer” from the Tree menu



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01 HIG v4.4.0.0

SDO Layers

- Network Manager
- Asset Manager
  - Asset Layer
- Document Manager
- Enquiry Manager
  - Enquiry Layer
- Maintenance Manager
  - Defects Layer
  - Work Order Lines Layer
- Structures Manager
  - Structure Items

1. Structures 2. Layer

1. Choose a Structures Layer name for base theme  
STRUCTURES

2. Choose an Asset Type (enter new/existing)  
STR STRUCTURES

Admin Get Themes Next >

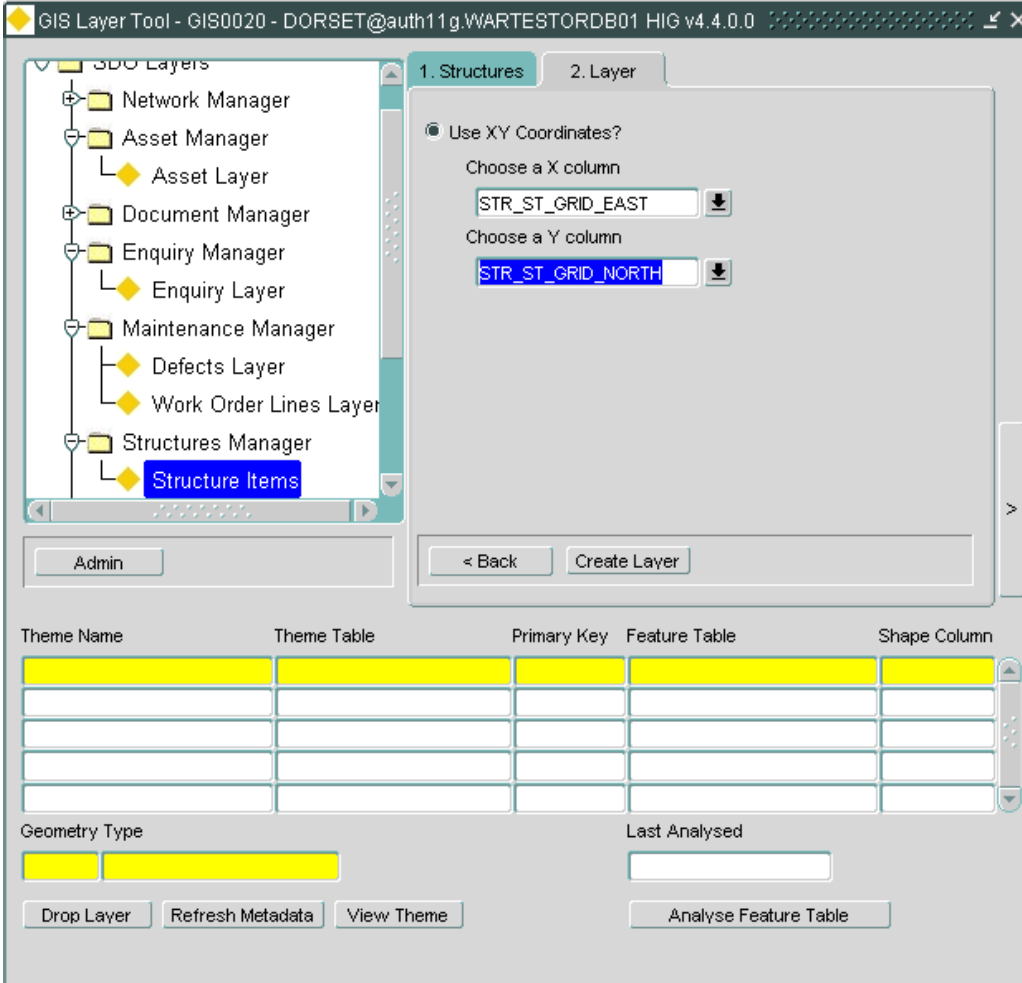
Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

Geometry Type Last Analysed

Drop Layer Refresh Metadata View Theme Analyse Feature Table

- 2) Choose a suitable name and an Asset type to use in Locator. You can define a new asset type, which will create all the relevant metadata automatically or reuse an existing one. Note that in this case the asset type must be defined as a Foreign table type.
- 3) Click on [Next].

3) Select the columns that store the XY coordinates from the drop down menu and press the [Create Layer] button.



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01 HIG v4.4.0.0

1. Structures 2. Layer

☒ Use XY Coordinates?

Choose a X column  
STR\_ST\_GRID\_EAST

Choose a Y column  
STR\_ST\_GRID\_NORTH

< Back Create Layer

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

Geometry Type

Last Analysed

Drop Layer Refresh Metadata View Theme Analyse Feature Table

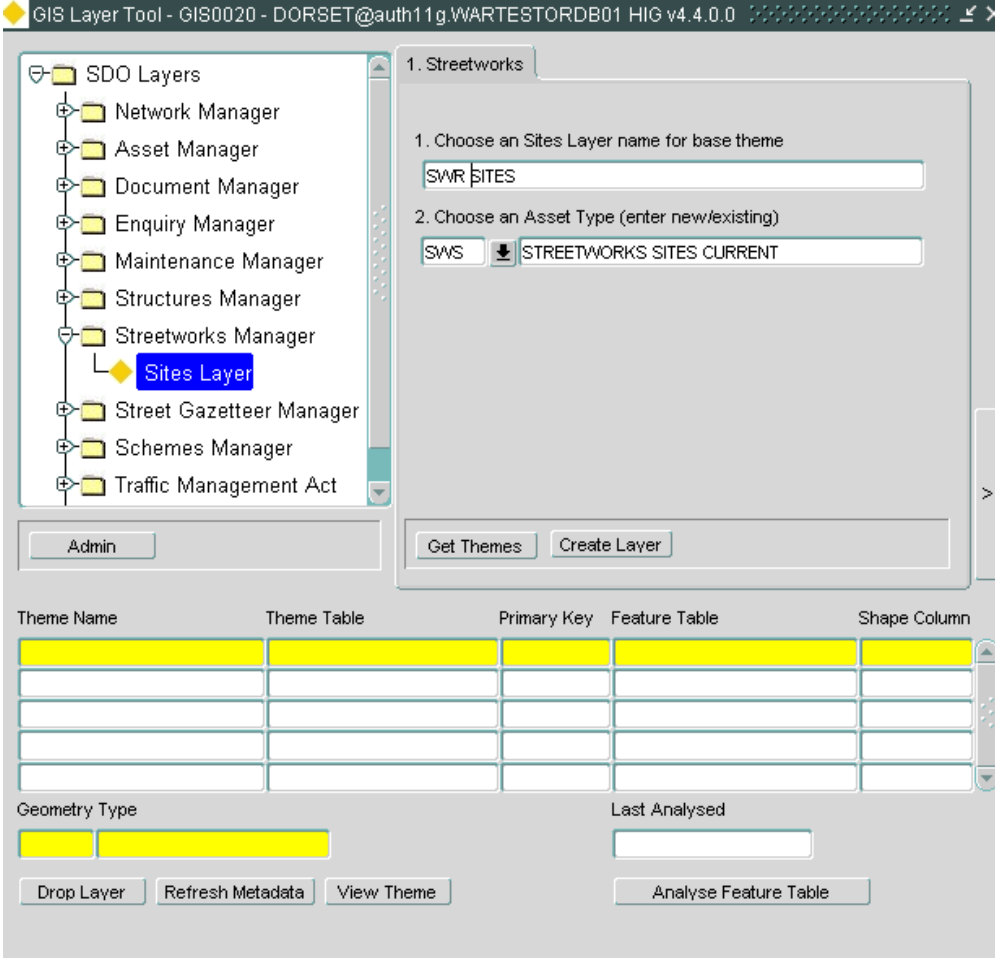
4) This process will create a spatial table for the base table theme (with the name you have chosen) and the following spatial views:

- i) Structures by Type
- ii) Structures by Condition BCI which can be used for banding in Locator.

## Creating Streetworks Sites Layer

To create layers for Streetworks data:

1) From the GIS0020 form select “Streetworks Manager” > “Sites Layer” from the Tree menu



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01 HIG v4.4.0.0

1. Streetworks

1. Choose an Sites Layer name for base theme  
SWR SITES

2. Choose an Asset Type (enter new/existing)  
SWS STREETWORKS SITES CURRENT

Get Themes Create Layer

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

Geometry Type Last Analysed

Drop Layer Refresh Metadata View Theme Analyse Feature Table

2) Choose a suitable name and an Asset type to use in Locator. You can define a new asset type, which will create all the relevant metadata automatically, or reuse an existing one. Note that in this case the asset type must be defined as a Foreign table type.

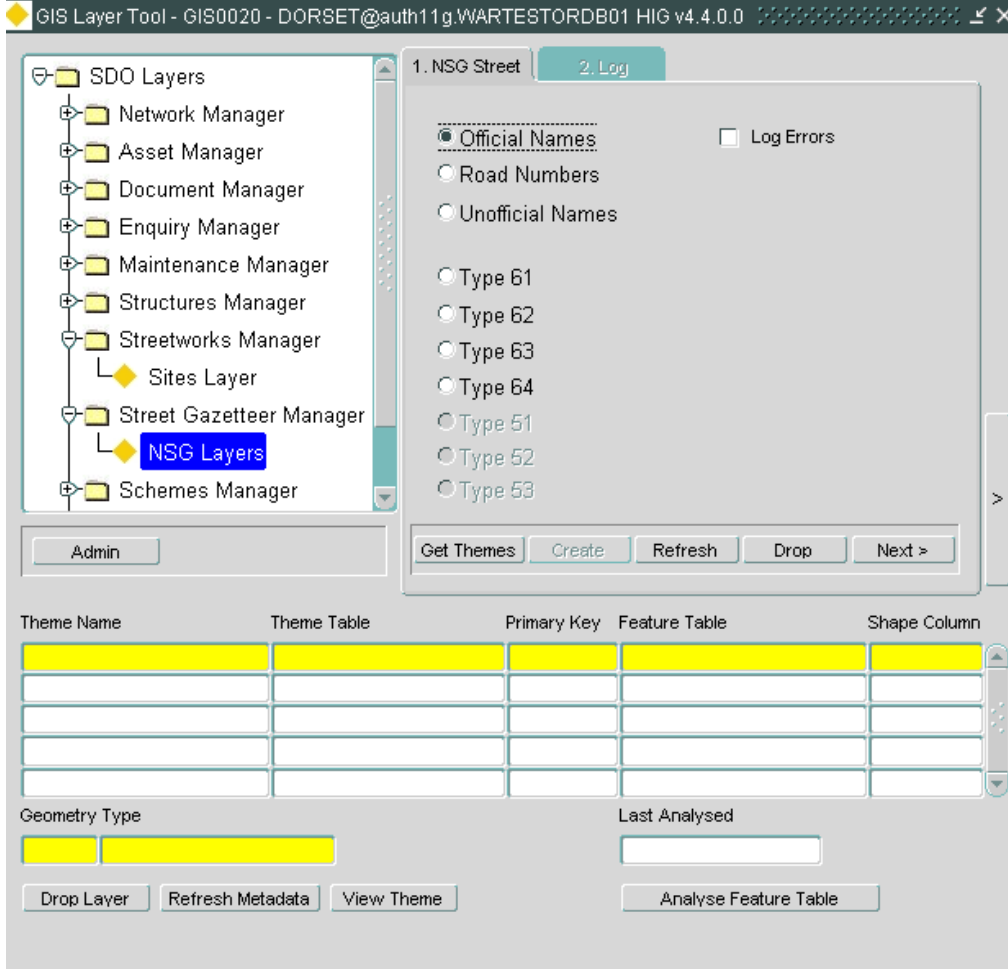
3) Press the [Create Layer] button.

4) The process will create a theme with the name you specified which uses a view called SWR\_SITE\_DETAILS. This view contains many attributes that SWM users should find useful for searching on in Locator.

## Creating Street Gazetteer Manager Layers

To create Street Gazetteer layers:

- 1) From the GIS0020 form select “Street Gazetteer Manager” -> “NSG Layers” from the Tree Menu:



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01 HIG v4.4.0.0

1. NSG Street 2. Log

☒ Official Names ☐ Log Errors

☐ Road Numbers

☐ Unofficial Names

☐ Type 61

☐ Type 62

☐ Type 63

☐ Type 64

☐ Type 51

☐ Type 52

☐ Type 53

Get Themes Create Refresh Drop Next >

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

Geometry Type Last Analysed

Drop Layer Refresh Metadata View Theme Analyse Feature Table

- 2) Using the wizard you can create the following spatial tables and views for the following data:

- i) Official Names
- ii) Road Numbers
- iii) Unofficial Names
- iv) ASD Type 61
- v) ASD Type 62
- vi) ASD Type 63

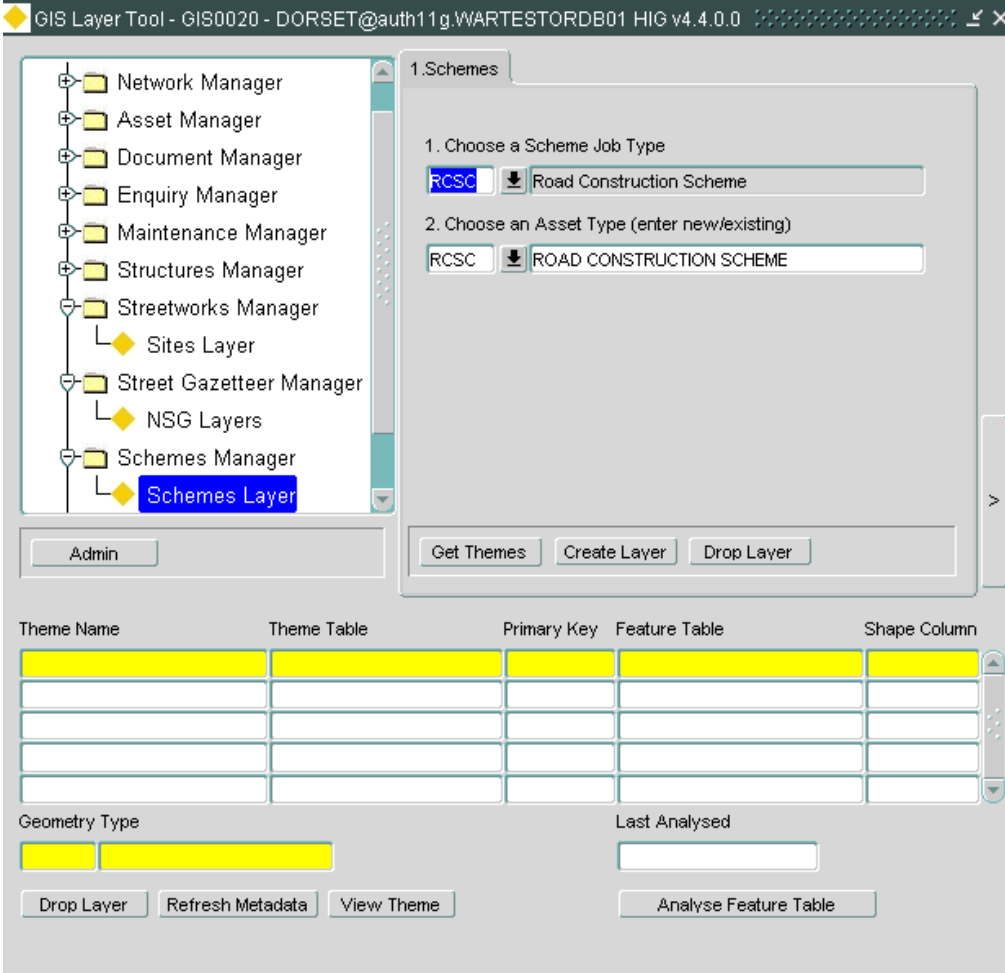
Select the data type you want to create layers for and press the [Create Layer] button.

You can also click on the [Refresh] button to truncate the underlying spatial table(s) and repopulate them. The [Refresh] button will not affect the Theme or SDE/SDO metadata.

## Creating Structural Schemes Layers

To create layers for TMA (Traffic Management Act) data:

1) Choose “Schemes Manager” -> “Schemes Layer” from the Tree menu:



The screenshot shows the 'GIS Layer Tool' window. On the left, a tree menu lists various managers, with 'Schemes Manager' expanded to show 'Schemes Layer' selected. The main panel, titled '1.Schemes', contains two dropdown menus: '1. Choose a Scheme Job Type' (set to 'RCSC') and '2. Choose an Asset Type (enter new/existing)' (set to 'ROAD CONSTRUCTION SCHEME'). Below these are buttons for 'Get Themes', 'Create Layer', and 'Drop Layer'. At the bottom, there is a table for themes and a section for geometry and analysis.

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

Geometry Type:  Last Analysed:

Buttons: Drop Layer, Refresh Metadata, View Theme, Analyse Feature Table

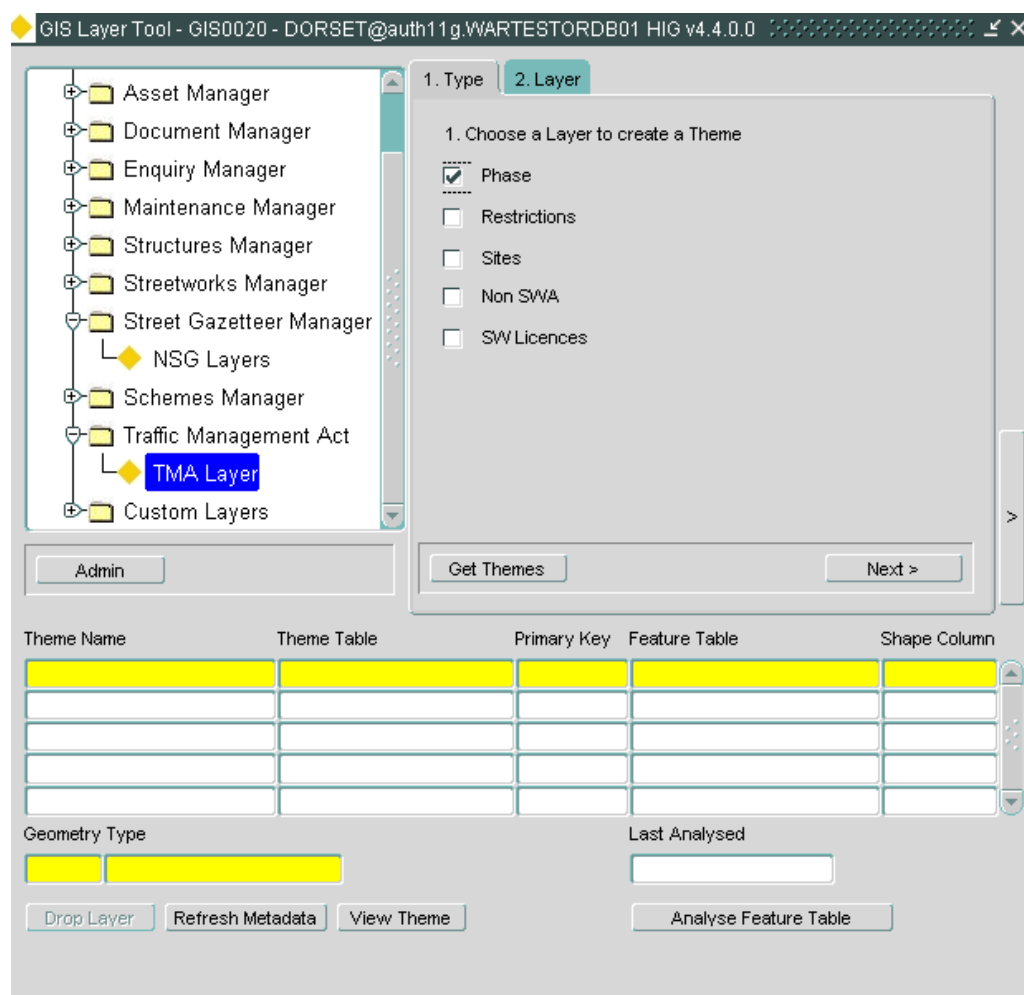
2) Choose a Scheme Job type and an Asset type. You can define a new asset type, which will create all the relevant metadata automatically, or reuse an existing one. Note that in this case the asset type must be defined as a Foreign table type.

3) Press the [Create Layer] button.

## Creating TMA Layers

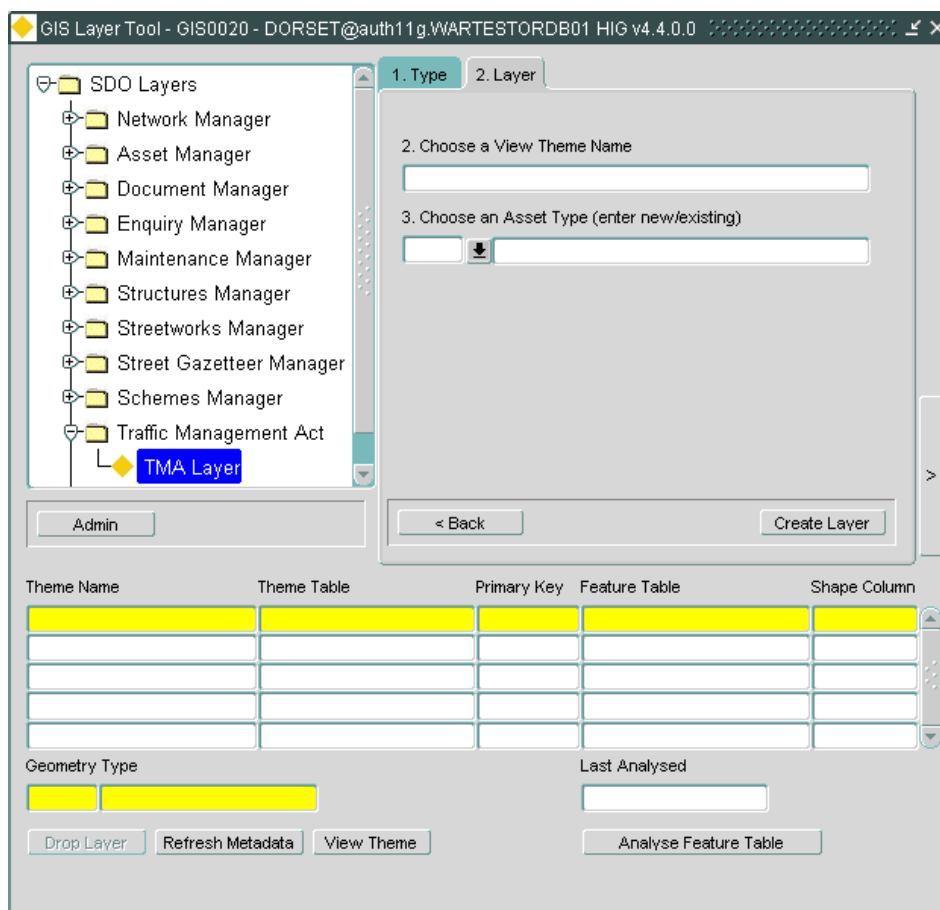
To create layers for TMA (Traffic Management Act) data:

2) Choose “Traffic Management Act” -> “TMA Layer” from the Tree menu:



- 3) Make sure the top check box (Phase is checked).
- 4) Click on the [Get Themes] button to make sure no TMA themes already exist.
- 5) Click on the [Next] button.





Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

- 6) Set the theme name to TMA\_ACTIVE\_PHASES and define a new asset type TMAP- TMA ACTIVE PHASES which will create all the relevant metadata automatically
- 7) Press on the [Create Layer] button. Note: If the TMA layer(s) already exist, the button will be labelled [Refresh Layer]
- 8) This may take some time but you will eventually get the following message.



- 9) You should now be able to see the new themes populated at the bottom of the form as shown in the following screenshot:

GIS Layer Tool - ATLAS@devonamh.GBEXOR730 HIG v4.1.0.0

SDO Layers

- Network Manager
- Asset Manager
- Streetworks Manager
- Street Gazetteer Manager
- Traffic Management Act
  - TMA Layer**
- Custom Layers

1. Type    2. Layer

2. Choose a View Theme Name  
TMA\_ACTIVE\_PHASES

3. Choose an Asset Type (enter new/existing)  
TMAP    TMA ACTIVE PHASES

< Back    Refresh Layer

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column
TMA_PHASES_TAB	TMA_PHASES	TPHS_ID	TMA_PHASES_SDO	TPHS_GEOME
TMA_ACTIVE_PHASES	V_TMA_WORKS_ACTIVE_PH	TPHS_PHASE	V_TMA_PHASES_SDO	TPHS_GEOME
TMA_PHASES_PT_TAB	V_TMA_WORKS_ACTIVE_PH	TPPS_TPHS_I	TMA_PHASES_POINT_SDO	TPPS_TPHS_C
TMA_PHASES_PT	V_TMA_WORKS_ACTIVE_PH	TPPS_TPHS_I	V_TMA_PHASES_PT_SDO	TPPS_TPHS_C

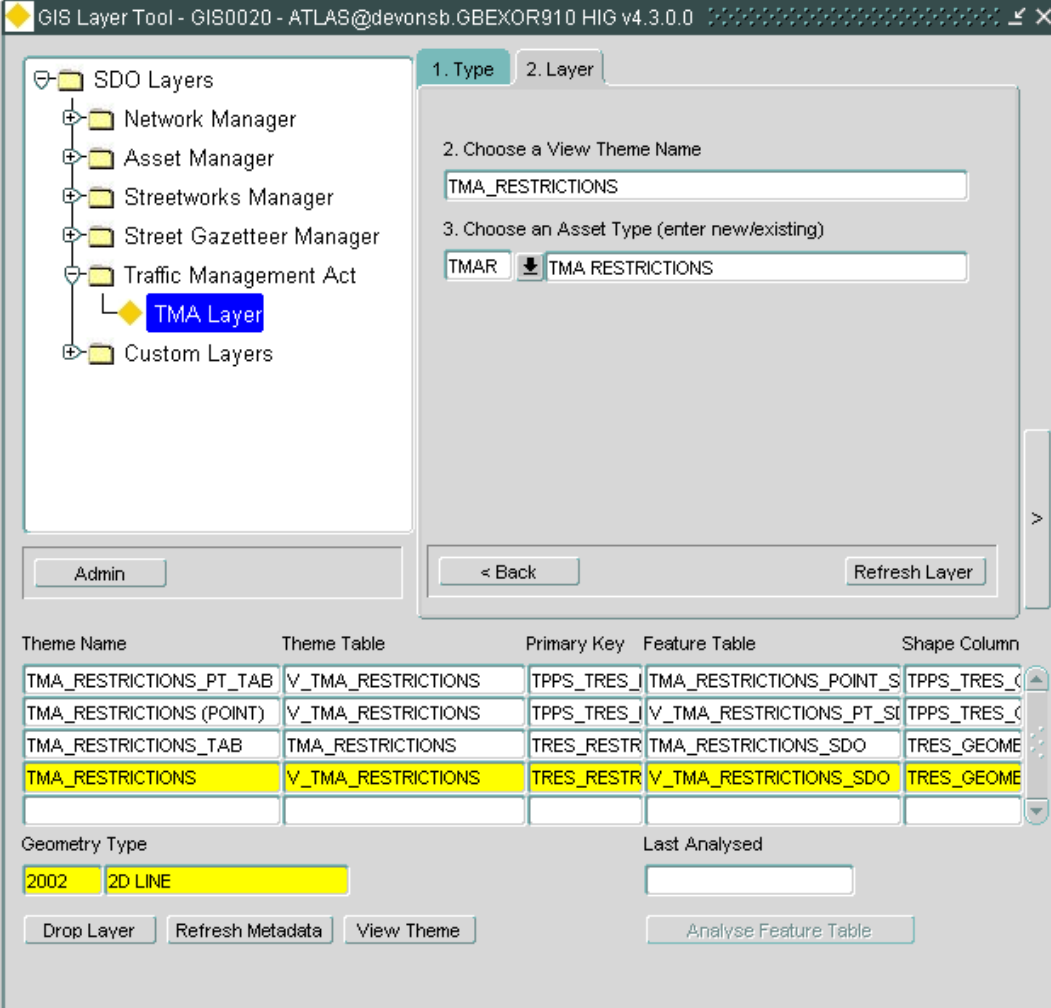
Geometry Type    Last Analysed

12-OCT-2009

Drop Layer    View Theme    Admin    Analyse Feature Table

- 10) Press the [Back] button and repeat the above steps for Restrictions
- 11) Set the theme name to TMA\_RESTRICTIONS and define a new asset type TMAR- TMA RESTRICTIONS

- 12) Press on the [Create Layer] button. Once completed you should be able to see the Restriction layers at the bottom of the form:



GIS Layer Tool - GIS0020 - ATLAS@devonsb.GBEXOR910 HIG v4.3.0.0

1. Type 2. Layer

2. Choose a View Theme Name  
TMA\_RESTRICTIONS

3. Choose an Asset Type (enter new/existing)  
TMA RESTRICTIONS

Admin < Back Refresh Layer

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column
TMA_RESTRICTIONS_PT_TAB	V_TMA_RESTRICTIONS	TPPS_TRES_	TMA_RESTRICTIONS_POINT_S	TPPS_TRES_C
TMA_RESTRICTIONS (POINT)	V_TMA_RESTRICTIONS	TPPS_TRES_	V_TMA_RESTRICTIONS_PT_SI	TPPS_TRES_C
TMA_RESTRICTIONS_TAB	TMA_RESTRICTIONS	TRES_RESTR	TMA_RESTRICTIONS_SDO	TRES_GEOME
TMA_RESTRICTIONS	V_TMA_RESTRICTIONS	TRES_RESTR	V_TMA_RESTRICTIONS_SDO	TRES_GEOME

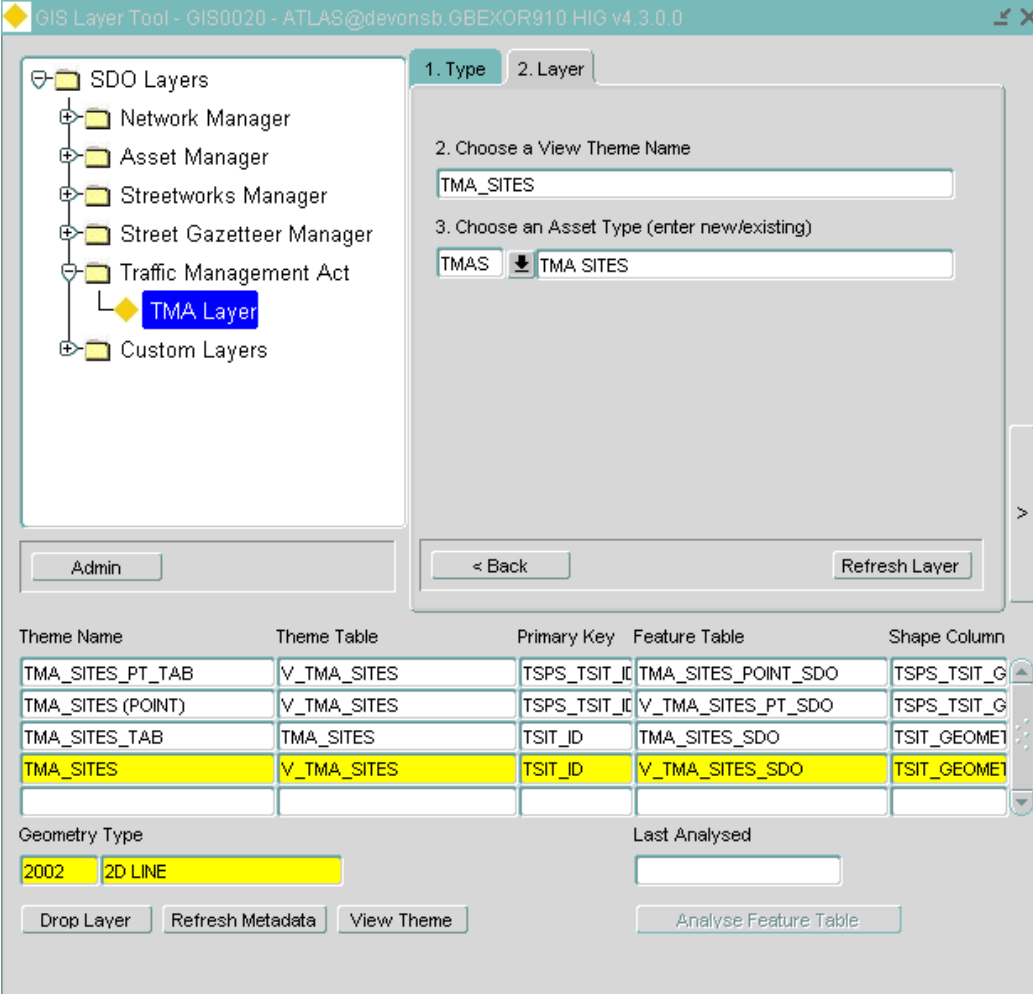
Geometry Type  
2002 2D LINE

Last Analysed

Drop Layer Refresh Metadata View Theme Analyse Feature Table

- 13) Press the [Back] button and repeat the above steps for Sites
- 14) Set the theme name to TMA\_SITES and define a new asset type TMA-SITES

15) Press on the [Create Layer] button. Once completed you should be able to see the Sites layers at the bottom of the form:



GIS Layer Tool - GIS0020 - ATLAS@devonsb.GBEXOR910 HIG v4.3.0.0

1. Type 2. Layer

2. Choose a View Theme Name  
TMA\_SITES

3. Choose an Asset Type (enter new/existing)  
TMA\_SITES

Admin < Back Refresh Layer

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column
TMA_SITES_PT_TAB	V_TMA_SITES	TSPS_TSIT_IC	TMA_SITES_POINT_SDO	TSPS_TSIT_G
TMA_SITES (POINT)	V_TMA_SITES	TSPS_TSIT_IC	V_TMA_SITES_PT_SDO	TSPS_TSIT_G
TMA_SITES_TAB	TMA_SITES	TSIT_ID	TMA_SITES_SDO	TSIT_GEOMET
<b>TMA_SITES</b>	<b>V_TMA_SITES</b>	<b>TSIT_ID</b>	<b>V_TMA_SITES_SDO</b>	<b>TSIT_GEOMET</b>

Geometry Type  
2002 2D LINE

Last Analysed

Drop Layer Refresh Metadata View Theme Analyse Feature Table

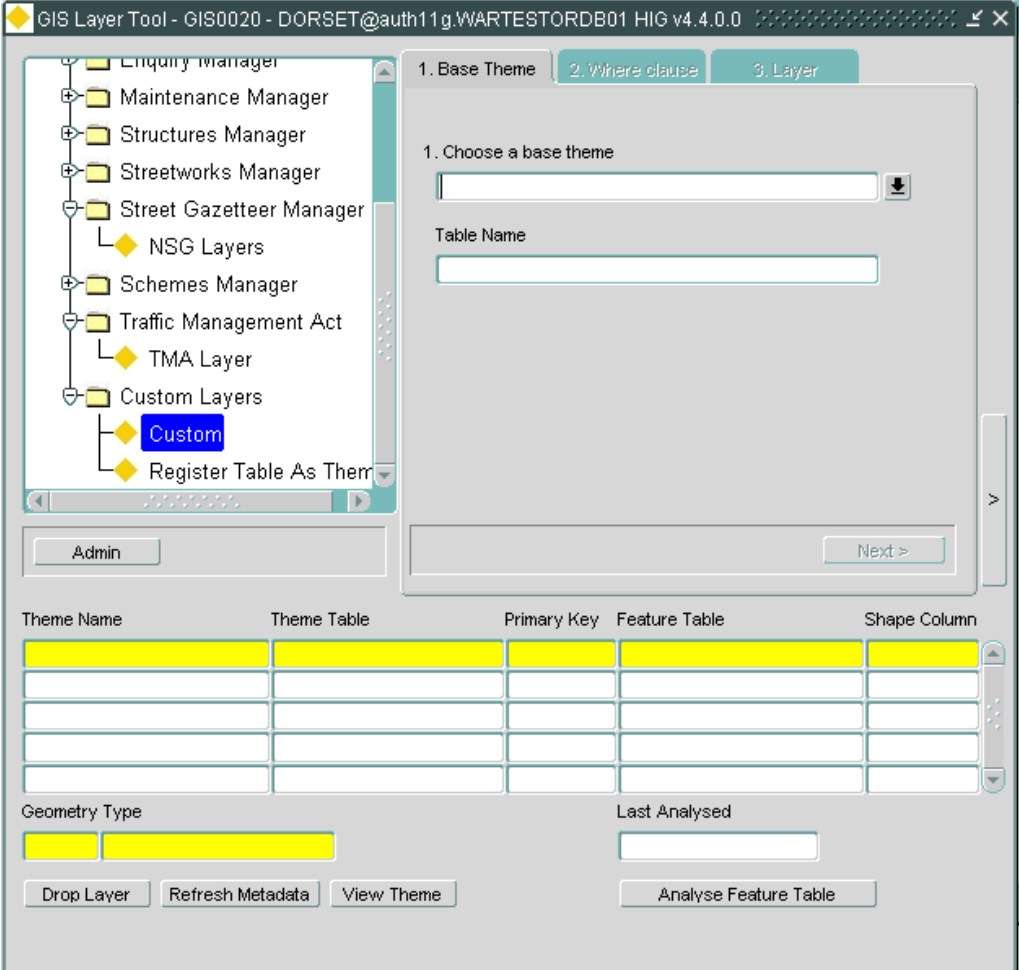
16) Ignore the rest of the TMA themes as they are not relevant at this release.

## Creating Custom Layers

Using the GIS0020 form you can also create custom layers based on existing base themes. This is particularly useful for example if you want to create a theme (and hence a layer) which will only contain features that satisfy some custom criteria. e.g. safety fences of a specific type.

To create a custom layer:

- 1) Choose “Custom Layers” > “Custom” from the Tree menu
- 2) Select an existing base table theme from the drop down list. The corresponding table will be displayed in the [Table Name] text box. Click on the [Next] button.



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01 HIG v4.4.0.0

1. Base Theme   2. Where clause   3. Layer

1. Choose a base theme

Table Name

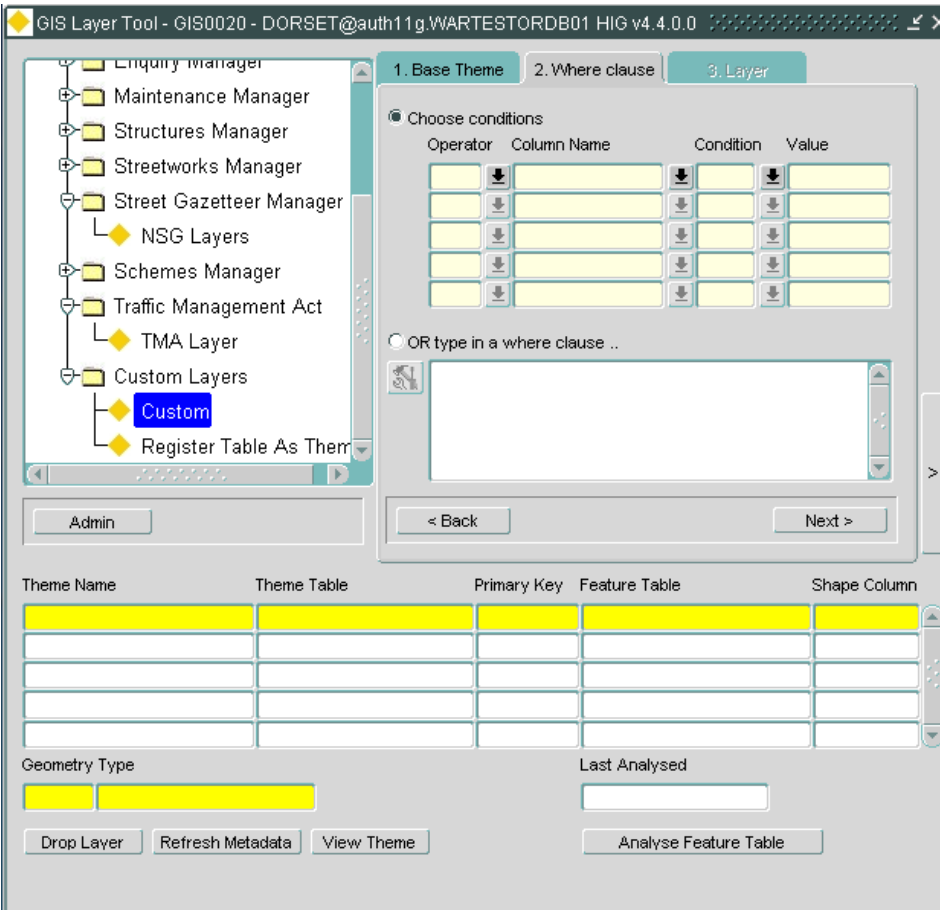
Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

Geometry Type

Last Analysed

Drop Layer   Refresh Metadata   View Theme   Analyse Feature Table

- 3) This will display a form which can help you add a where clause for the base table.



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01 HIG v4.4.0.0

1. Base Theme 2. Where clause 3. Layer

☒ Choose conditions

Operator	Column Name	Condition	Value

☐ OR type in a where clause ...

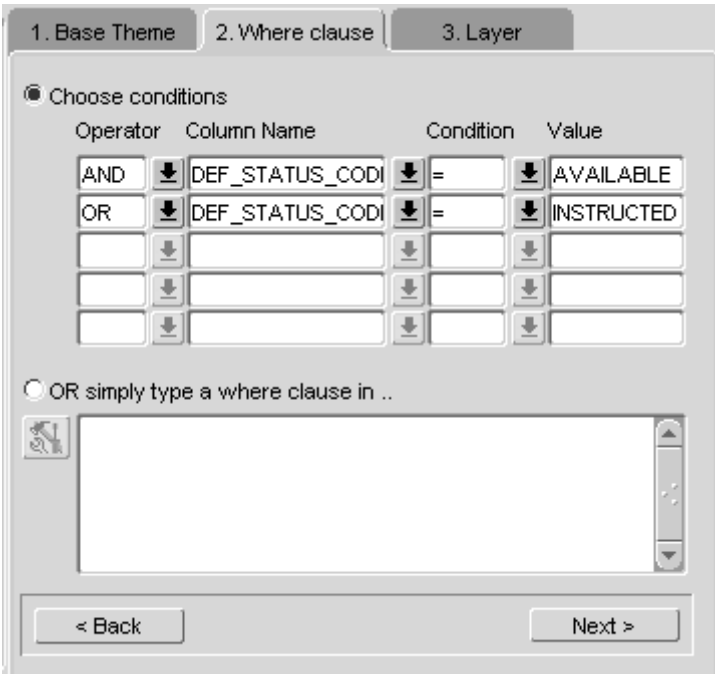
< Back Next >

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

Geometry Type Last Analysed

Drop Layer Refresh Metadata View Theme Analyse Feature Table

4) You can use the LOVs to enter column / value combinations,



1. Base Theme 2. Where clause 3. Layer

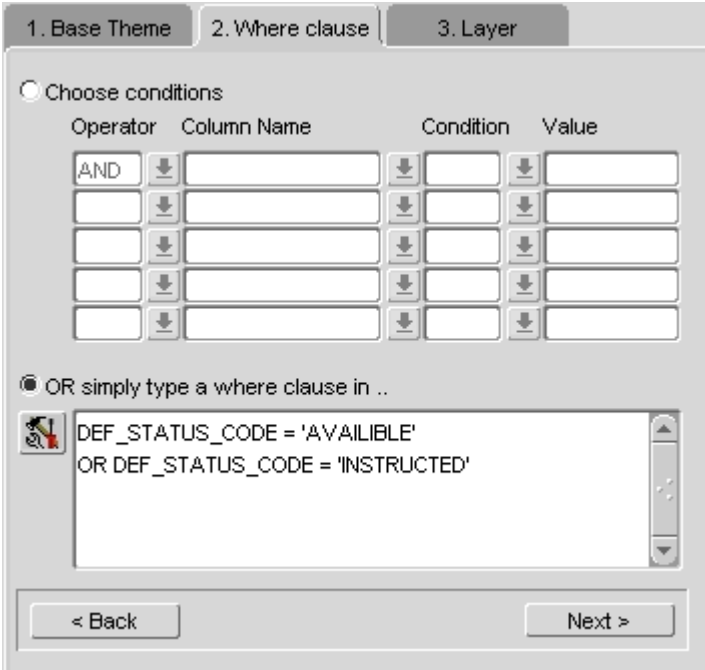
☒ Choose conditions

Operator	Column Name	Condition	Value
AND	DEF_STATUS_CODI	=	AVAILABLE
OR	DEF_STATUS_CODI	=	INSTRUCTED

☐ OR simply type a where clause in ..

< Back Next >

5) OR enter Where clause manually and Parse the query with the button:



1. Base Theme   2. Where clause   3. Layer

☐ Choose conditions

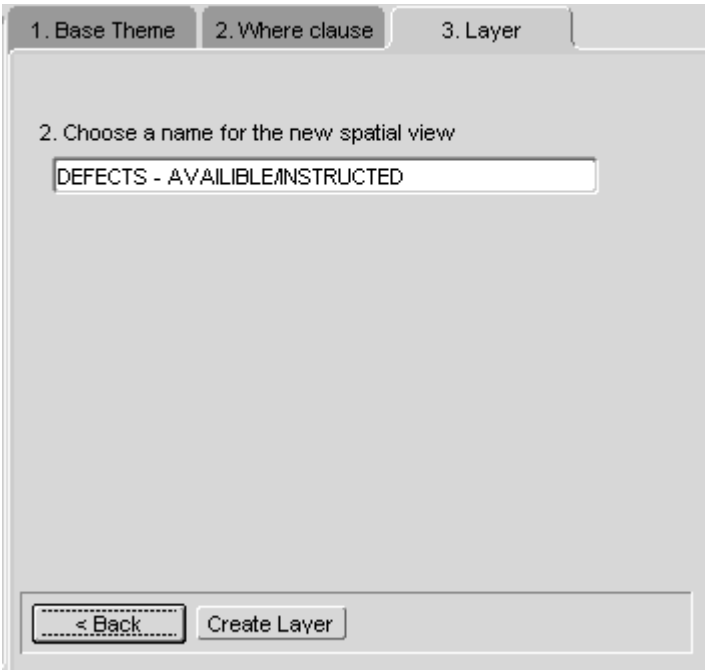
Operator	Column Name	Condition	Value
AND			

☒ OR simply type a where clause in ..

DEF\_STATUS\_CODE = 'AVAILABLE'  
OR DEF\_STATUS\_CODE = 'INSTRUCTED'

< Back   Next >

6) Once you are happy with the where clause you entered, press on [Next]. Enter a new Theme name and press the [Create Layer] button.



1. Base Theme   2. Where clause   3. Layer

2. Choose a name for the new spatial view

DEFECTS - AVAILABLE/INSTRUCTED

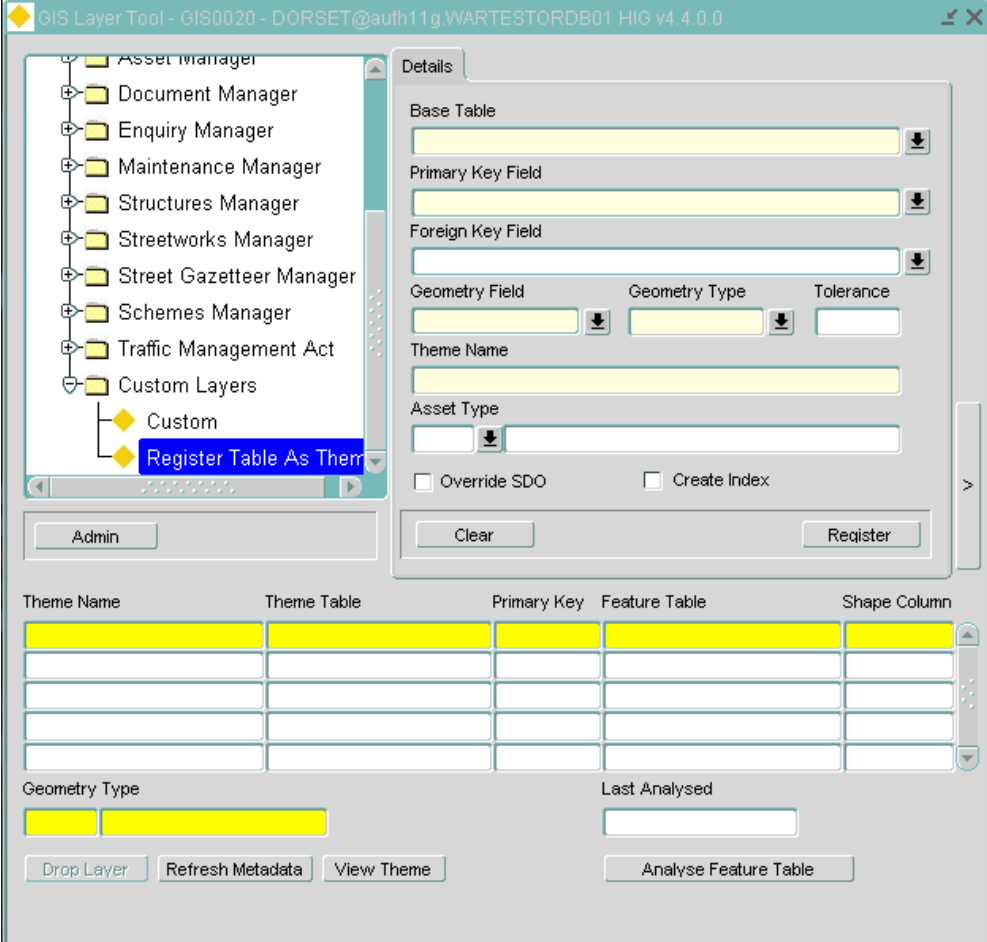
< Back   Create Layer

You will then get a new Theme based on a view with the Where clause you have entered.

## Register a spatial table as a theme

There may be cases where you have a spatial table that is not part of the exor system (e.g. a table containing property polygons). In order to be able to view it within the Exor application(s) such as Spatial Manager or Locator you can use the [Register Table As Theme] option in the GIS0010 form following the steps outlined below:

1. Choose “Custom Layers” > “Register Table As Theme” from the Tree menu



GIS Layer Tool - GIS0020 - DORSET@auth11g.WARTESTORDB01 HIG v4.4.0.0

Asset Manager  
 Document Manager  
 Enquiry Manager  
 Maintenance Manager  
 Structures Manager  
 Streetworks Manager  
 Street Gazetteer Manager  
 Schemes Manager  
 Traffic Management Act  
 Custom Layers  
   Custom  
   **Register Table As Theme**

Details

Base Table  
 Primary Key Field  
 Foreign Key Field  
 Geometry Field    Geometry Type    Tolerance  
 Theme Name  
 Asset Type  
☐ Override SDO    ☐ Create Index  
 Clear    Register

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column

Geometry Type    Last Analysed

Drop Layer    Refresh Metadata    View Theme    Analyse Feature Table

2. Select an existing spatial base table theme from the drop down list. Note that the base table MUST include a NUMBER(38) column to be used as the primary key. Selecting the base table will automatically populate the [Primary Key Field], [Geometry Field] and [Geometry Type] fields.
3. If the [Primary Key Field] is not the column you want to join back to the Themes table, set the Foreign Key field value to that column.
4. Enter a name for the new theme in the [Theme Name] field
5. If required select an asset type to be associated with this theme. This will allow you to be able to use the Search facilities in Locator.
6. If the spatial table already has an entry in the USER\_SDO\_GEOM\_METADATA table (which you don't want to use) check the [Override SDO] box.



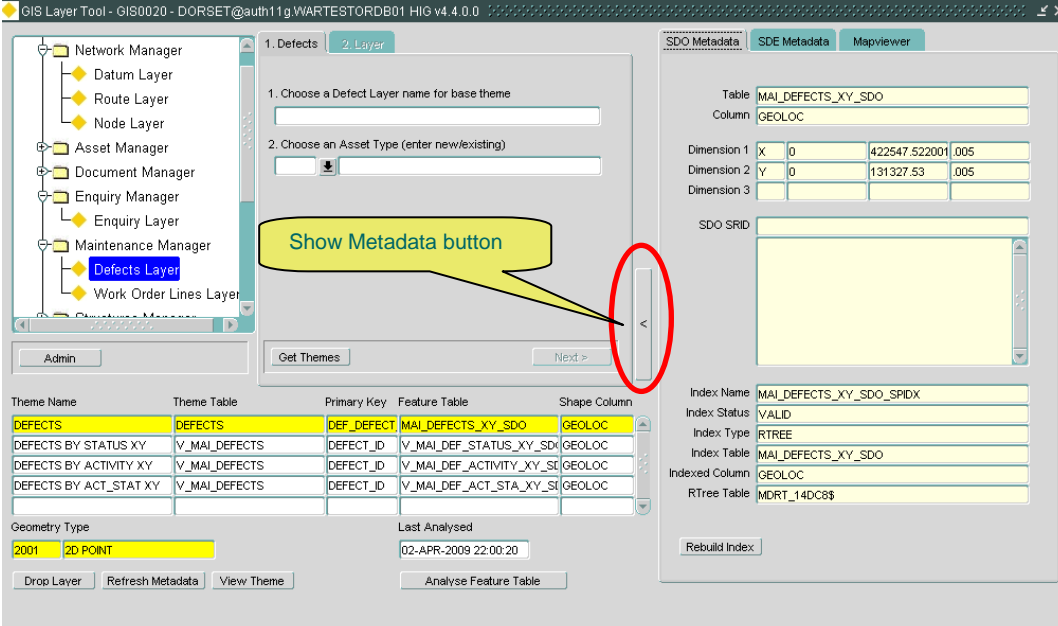
7. If the table does not have a spatial index, or you want to re-create the spatial index, check the [Create Index] checkbox.
8. Click the [Register] button to register the theme.

## GIS0020 form Utilities

The GIS0020 form contains a number of utilities that help you maintain and manage your spatial layers

### View Spatial Metadata

For any current Theme queried in the bottom block you can view the spatial metadata created.



1. Defects 2. Layer

1. Choose a Defect Layer name for base theme

2. Choose an Asset Type (enter new/existing)

Show Metadata button

Theme Name	Theme Table	Primary Key	Feature Table	Shape Column
DEFECTS	DEFECTS	DEF_DEFECT	MAI_DEFECTS_XY_SDO	GEOLOC
DEFECTS BY STATUS XY	V_MAI_DEFECTS	DEFECT_ID	V_MAI_DEF_STATUS_XY_SDO	GEOLOC
DEFECTS BY ACTIVITY XY	V_MAI_DEFECTS	DEFECT_ID	V_MAI_DEF_ACTIVITY_XY_SDO	GEOLOC
DEFECTS BY ACT_STAT XY	V_MAI_DEFECTS	DEFECT_ID	V_MAI_DEF_ACT_STA_XY_SDO	GEOLOC

Geometry Type: 2001 2D POINT

Last Analysed: 02-APR-2009 22:00:20

Drop Layer Refresh Metadata View Theme Analyse Feature Table

SDO Metadata SDE Metadata Mapviewer

Table: MAI\_DEFECTS\_XY\_SDO  
Column: GEOLOC

Dimension	X	Y	Z
Dimension 1	0	422547.522001	.005
Dimension 2	0	131327.53	.005
Dimension 3			

SDO SRID:

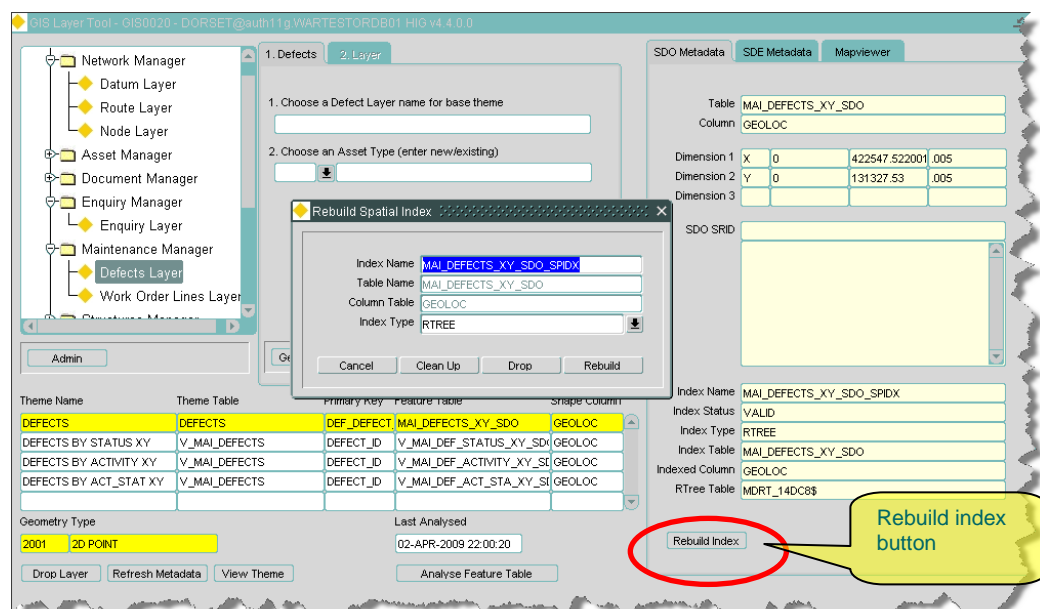
Index Name: MAI\_DEFECTS\_XY\_SDO\_SPIDX  
Index Status: VALID  
Index Type: RTREE  
Index Table: MAI\_DEFECTS\_XY\_SDO  
Indexed Column: GEOLOC  
RTree Table: MDRT\_14DC8\$

Rebuild Index

Press the 'Show Metadata' button and you can view the SDO, SDE and any Mapviewer metadata you may have defined.

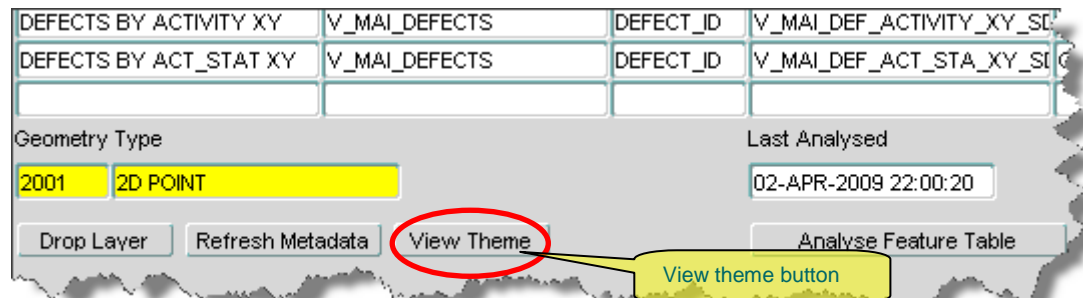
## Rebuild spatial index

For the current Theme, you can rebuild the spatial index if required by pressing the [Rebuild Index] button



## View current Theme

1) You can view any theme current in the Details area by pressing the [View Theme] button. This will call the GIS0010 form. For more details see the Using the GIS0010 form chapter.



DEFECTS BY ACTIVITY XY	V_MAI_DEFECTS	DEFECT_ID	V_MAI_DEF_ACTIVITY_XY_SD
DEFECTS BY ACT_STAT XY	V_MAI_DEFECTS	DEFECT_ID	V_MAI_DEF_ACT_STA_XY_SIG

Geometry Type: 2001 2D POINT

Last Analysed: 02-APR-2009 22:00:20

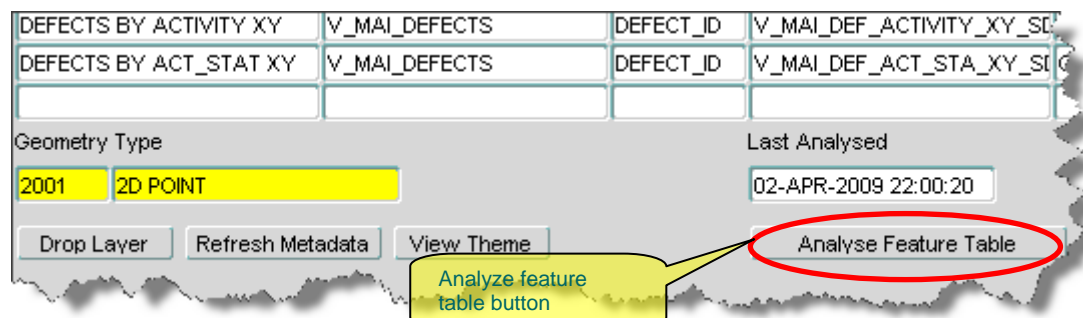
Buttons: Drop Layer, Refresh Metadata, **View Theme**, Analyse Feature Table

View theme button

## Analyze feature table

To enhance drawing performance in Spatial Manager and Locator it is sometimes necessary to analyze the feature table and associated indexes. You can use the [Analyze Feature Table] button to achieve this as shown in the following screenshot.

Note that this button will only be enabled when a base table (not view) is selected in the list of themes.



DEFECTS BY ACTIVITY XY	V_MAI_DEFECTS	DEFECT_ID	V_MAI_DEF_ACTIVITY_XY_SD
DEFECTS BY ACT_STAT XY	V_MAI_DEFECTS	DEFECT_ID	V_MAI_DEF_ACT_STA_XY_SIG

Geometry Type: 2001 2D POINT

Last Analysed: 02-APR-2009 22:00:20

Buttons: Drop Layer, Refresh Metadata, View Theme, **Analyse Feature Table**

Analyze feature table button

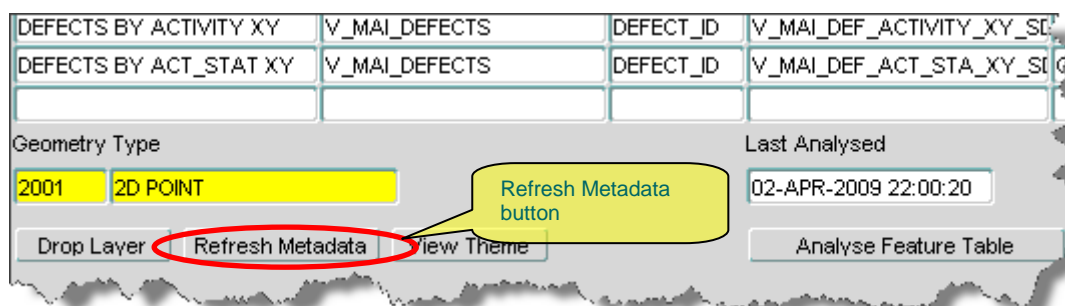
The last time you analyzed the table will be shown in the [Last Analysed] field.

## Refresh Metadata

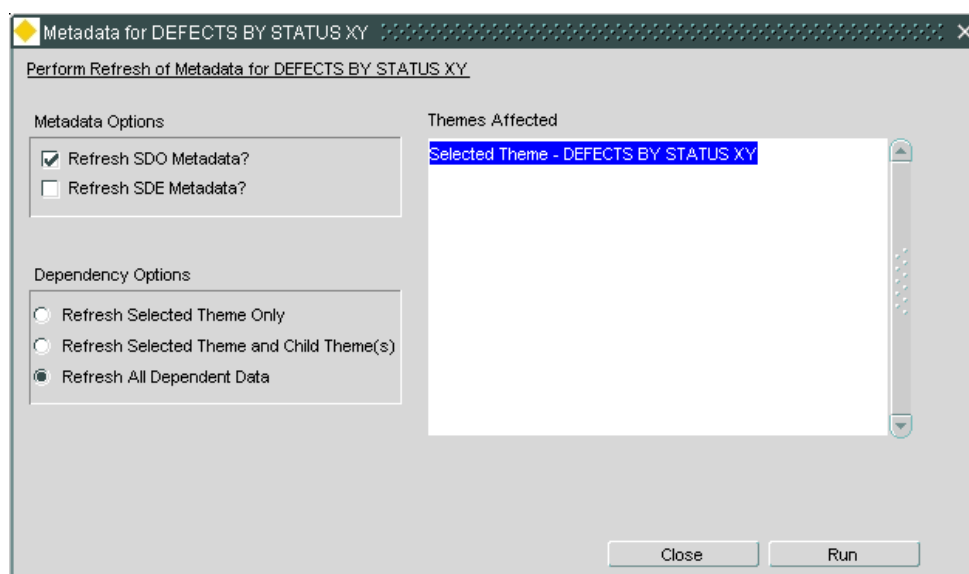
In order for a theme to be displayed in Spatial Manager or Locator a number of metadata tables need to be populated for the theme. The main metadata tables are:

- NM\_THEMES\_ALL (**Exor** Theme metadata)
- SDE.LAYERS, SDE.GEOMETRY\_COLUMNS, SDE.COLUMN\_REGISTRY, SDE.TABLE\_REGISTRY (**ArcSDE** metadata – this is how ArcMap/SM recognizes the spatial table)
- USER\_SDO\_GEOM\_METADATA (**Oracle** metadata tables)

The [Refresh Metadata] button will recreate the various theme metadata for the selected theme updating the above tables.



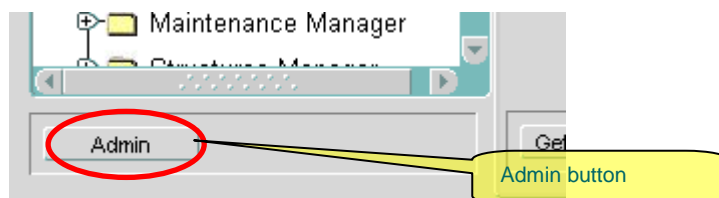
Clicking on this button will display the metadata refresh dialog:



From this dialog you can select the type of metadata you want to refresh as well as selecting to update only the selected theme or the dependant themes as well. Click on the [Run] button to execute the refresh.

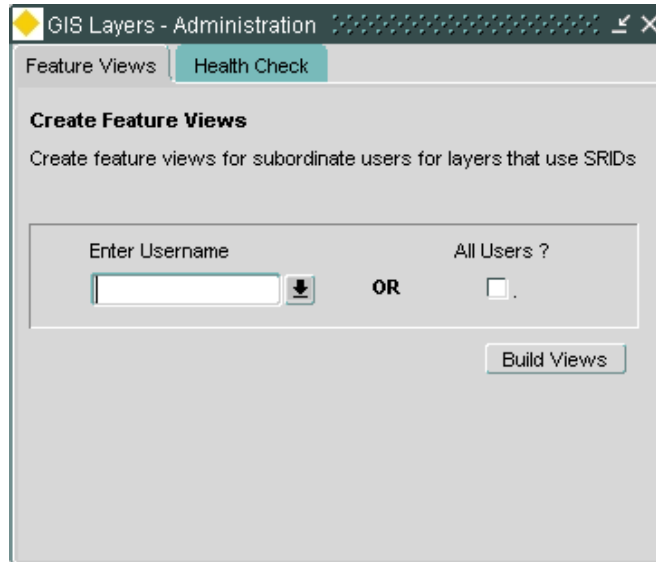
## Additional utilities

Some additional functionality is available from the [Admin] :



1. **Create Feature views for Subordinate users that use SRIDS.**

**Oracle mapviewer requires views (instead of synonyms) to assess which SRID to use, so for subordinate users who need access to feature tables owned by the Highways owner user, you can create such views [here](#).**

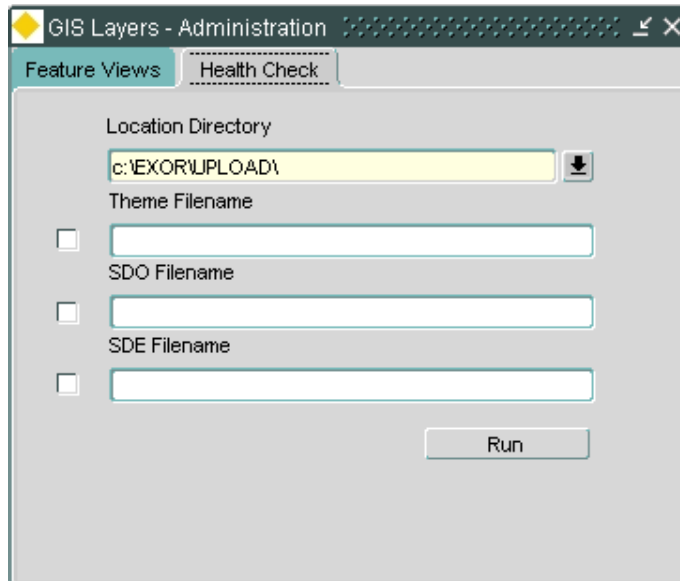


The screenshot shows a window titled "GIS Layers - Administration" with a tabbed interface. The "Health Check" tab is selected. Below the tabs, the text "Create Feature Views" is displayed, followed by the instruction "Create feature views for subordinate users for layers that use SRIDs". There are two input options: "Enter Username" with a text field and a download icon, and "All Users ?" with a checkbox. A "Build Views" button is located at the bottom right.

Clicking on the [Build Views] button, will loop through all feature tables using SRIDS and will creates a view for each the subordinate user schema.

## 2. Run a “Health Check” on your metadata.

Running the health check will provide listings of errors and warnings about the integrity of the spatial metadata. Some of these warnings have no impact on the operation of the product but form a useful test. Others are critical to the correct functionality of the spatial tools and need to be addressed. There are three check procedures. Each caters for one of the three forms of semi-dependent metadata namely the Exor Themes data, the Oracle metadata and the SDE metadata. Each report is delivered to a specified location. To run the Health Check select the [Health Check] tab in the GIS Layers- Administration form.



To run the health check fill in the following fields:

#### **Location Directory** (Required)

This is the path where the resulting report files will be generated. The path must be a file system directory on the database server or an Oracle Directory Name.

#### **Theme Filename**

This is the name of the file containing the results of the Exor Themes check. If you want to run this check, click on the checkbox on the left of this field. The checks performed are:

- Themes that are not based on SDO layers
- Themes that have a NULL Theme table
- Themes that have a NULL Feature table
- Themes that have an unsuitable PK/FK combination
- Themes that reference a non-existent RSE table
- Themes that reference a non-existent RSE FK column
- Themes that reference a non-existent Label Column
- Themes that reference a non-existent PK column
- Themes that reference a non-existent Start Chain column
- Themes that reference a non-existent End Chain column
- Themes that reference a non-existent X coordinate column
- Themes that reference a non-existent Y coordinate column
- Themes that reference a non-existent feature PK column
- Themes that reference a non-existent feature FK column
- Themes that reference a non-existent feature shape column
- Themes that reference a non-existent start date column
- Themes that reference a non-existent end date column
- Themes that reference a non-existent base theme
- Themes that reference a non-existent snapping themes
- Themes that incorrectly snap to network themes
- Themes that are immediate update on edit but have no base(s) theme set

- Themes that are immediate update on edit but do not reference Network themes
- Themes that are immediate update on edit but are View based themes
- Themes that have an invalid sequence name defined
- Theme sequences that exist but the Themes have been removed
- Triggers that have been used with a theme but the theme no longer exists
- Incorrectly set Base Themes

#### **SDO Filename**

---

This is the name of the file containing the results of the SDO Metadata check. If you want to run this check, click on the checkbox on the left of this field. The checks performed are:

- Missing USER\_SDO\_GEOM\_METADATA for Highways Owner themes
- Missing Spatial Indexes
- Missing USER\_SDO\_GEOM\_METADATA for Subordinate users based on Themes accessed via roles
- Missing feature views for Subordinate users based on Themes accessed via roles
- Unrecognised Geometry Types (Gtypes)

#### **SDO Filename**

#### **(Required)**

---

This is the name of the file containing the results of the SDO Metadata check. If you want to run this check, click on the checkbox on the left of this field. The checks performed are:

- SDE Layers that are missing ( \*\* UNRESTRICTED BY ROLE \*\* ) (only if running as Highways owner)
- SDE Layers that are missing
- SDE Layers that refer to missing table/views
- SDE Layers that refer to missing Themes
- SDE Layers that have missing Geometry Column metadata for feature columns
- SDE Layers that have missing Column Registry metadata
- SDE Layers that have Column Registry metadata for columns that do not exist on the table
- SDE Layers that have missing Table Registry metadata
- SDE Layers that have RowID Column registered, but the column is missing from the table
- SDE Layers that have RowID Column registered, but the column is not the first indexed column (or not indexed at all)
- SDE Layers that have incorrect EFlags metadata



## Configuring Maps with the Oracle MapBuilder Tool

The MapBuilder tool let system administrators create, modify, and delete styles, themes, and base maps. For example, they can enter the design information for a new line style, see a preview of the style, modify the design to taste, and then click Insert to insert their style definition in XML format into the database. The tool uses the information that you entered to generate the XML document for the style definition.

For each theme you would like to see in your map you will need to add that theme to the Map definition specified in WEBMAPNAME product option.

Run Mapbuilder, edit/add Themes and edit/add those themes to the appropriate map. Care should be taken when choosing which themes to add to the map.

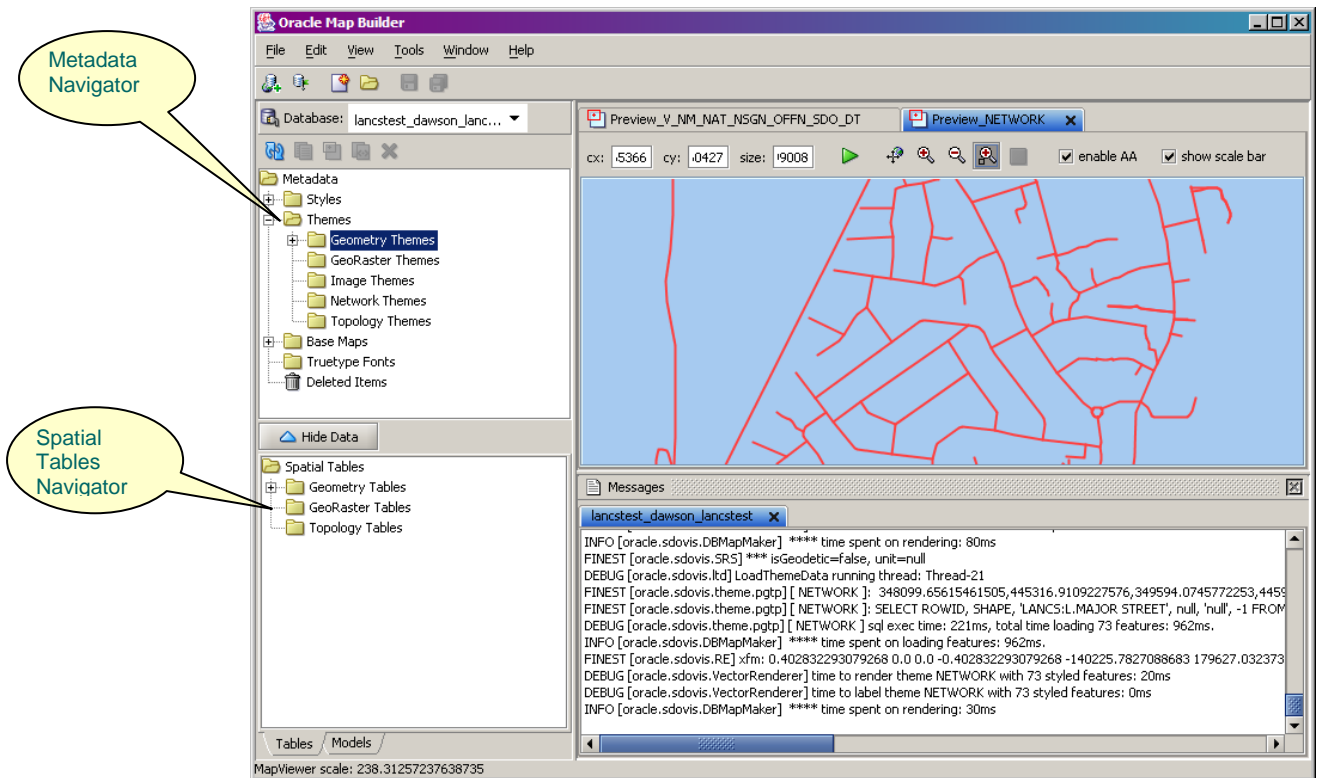
The styles, themes, and base maps for a user are maintained in the user's USER\_SDO\_STYLES, USER\_SDO\_THEMES, and USER\_SDO\_MAPS views, respectively. These views are created by MDSYS to provide access to mapping metadata. New mapping metadata can be created in these views. However, the styles created in your USER\_SDO\_STYLES view will be shared by all other database users.

System administrators should use MapBuilder instead of directly modifying MapViewer metadata views to create, modify, and delete information about styles, themes, and maps. The Map Definition Tool always checks and maintains the referential integrity between objects. If these operations are performed using SQL procedures or SQL\*Plus statements, the referential integrity of the mapping metadata may become corrupted. For example, if a user deletes a style using SQL\*Plus, a theme may still be referencing the name of that style.

The tool contains navigator trees that allow access to MapViewer metadata and to spatial tables and spatial models stored on the current database connection. The navigator trees are:

- Metadata Navigator: shows the MapViewer metadata
- Spatial Tables Navigator: shows the spatial tables for the different spatial data types
- Spatial Models Navigator: shows the network and topology spatial models (not used by exor)

The screenshot below shows the Mapviewer window and the Metadata and Spatial Tables navigator trees:



The tool consists of pages grouped under the following categories:

- Database Connection: a page for adding a connection to the database
- Styles: set of pages to add and edit styles
- Themes: set of pages to add and edit themes
- Base Maps: a page for base maps
- Truetype Fonts: a page for truetype fonts
- Spatial Tables: pages to access database spatial tables
- Spatial Models: pages to access database spatial network and topology models (not used by exor)
- Tools: pages to import external data

## Starting Mapbuilder

Run the MAPBUILDER.BAT file from the MAPBUILDER directory

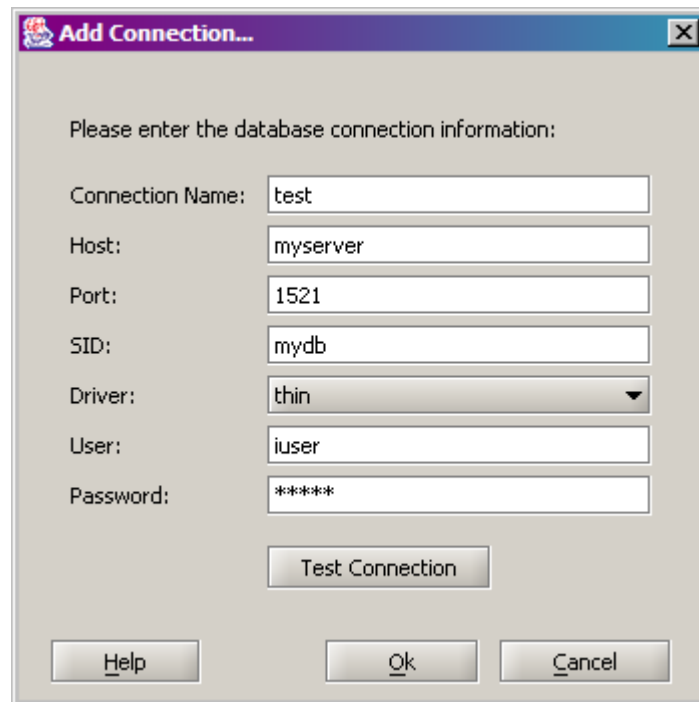
```
<exor_base>\msv\mapbuilder
```

The mapbuilder.bat will contain a line similar to (depending on where the Java Runtime Engine is installed):

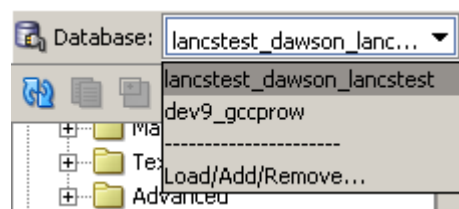
```
"C:\Program Files\Java\j2re1.4.0\bin\java" -Xmx512m -jar
mapbuilder.jar
pause
```

## Mapbuilder Connection Page

To connect to the database select File->New Connection. The figure below shows the Connection page for a new connection.



If there is an existing connection set up you can select it from the [Database:] drop-down list in the Mapbuilder window:



You can also create a new connection from here by selecting the [Load/Add/Remove...] option.

A style is a visual attribute that can be used to represent a spatial feature. The basic map symbols and labels for representing point, line, and area features are defined and stored as individual styles. Each style has a unique name and defines one or more graphical elements in an XML syntax.

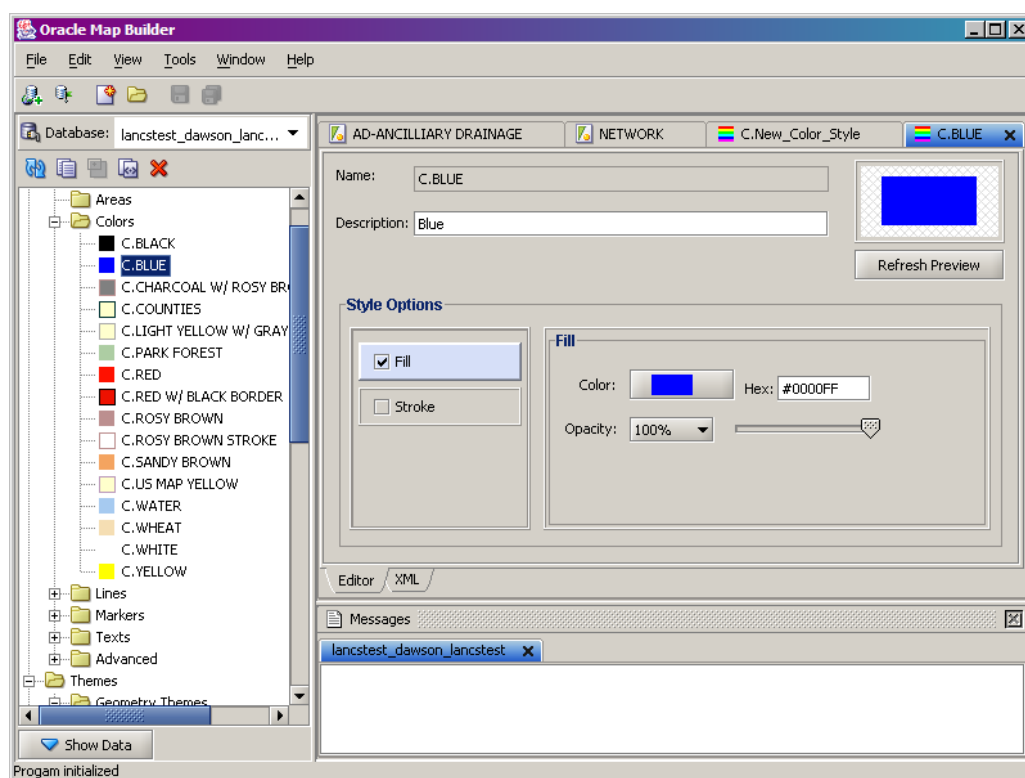
Each style is of one of the following types:

- **Color:** a color for the fill or the stroke (border), or both.
- **Marker:** a shape with a specified fill and stroke color, or an image. Markers are often icons for representing point features, such as airports, ski resorts, and historical attractions. When a marker style is specified for a line feature, the rendering engine selects a suitable point on the line and applies the marker style (for example, a shield marker for a U.S. interstate highway) to that point.
- **Line:** a line style (width, color, end style, join style) and optionally a center line, edges, and hash mark. Lines are often used for linear features such as highways, rivers, pipelines, and electrical transmission lines.

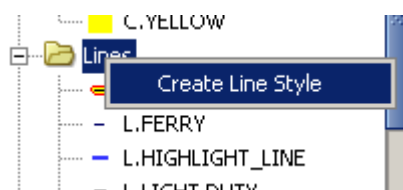
- **Area:** a color or texture, and optionally a stroke color. Areas are often used for polygonal features such as counties and census tracts.
- **Text:** a font specification (size and family) and optionally highlighting (bold, italic) and a foreground color. Text is often used for annotation and labeling (such as names of cities and rivers).
- **Advanced:** a composite used primarily for thematic mapping, which is described in Section 2.3.3. The core advanced style is RangeStyle, which defines the relationship between a set of simple styles and a set of ranges.

For Themes each feature to be plotted, a designated value from that feature is used to determine which range it falls into, and then the style associated with that range is used to plot the feature. The AdvancedStyle class is extended by RangeStyle, which is in turn extended by ColorSchemeStyle and VariableMarkerStyle.

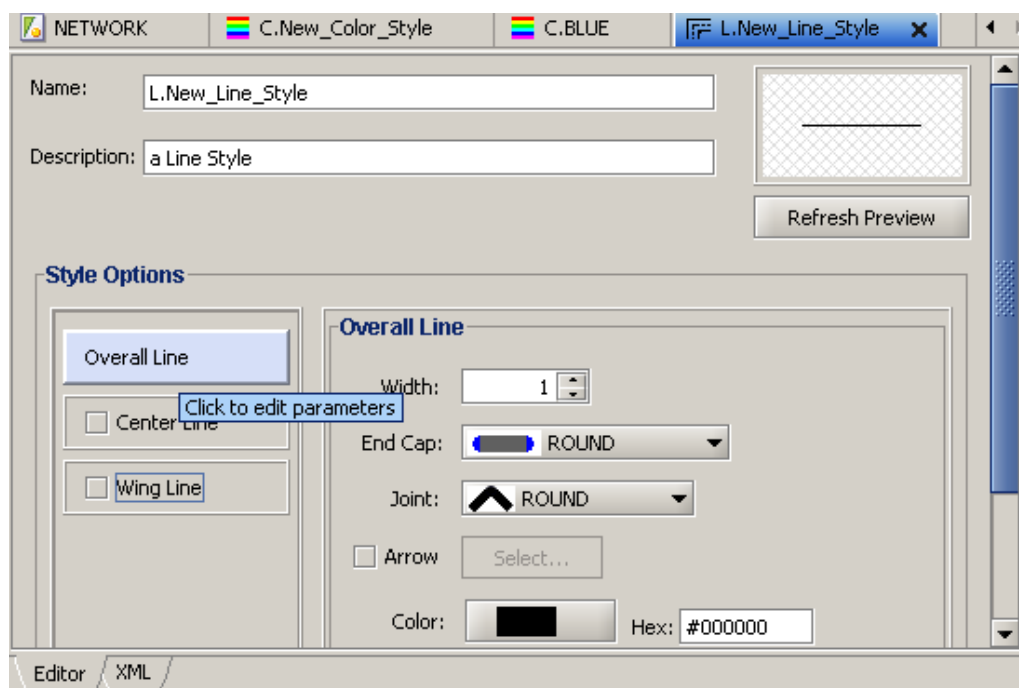
To edit styles, select them from metadata navigator and double-click on the style you want to change to open its properties. The following screenshot displays the properties for the color named 'C.BLUE':



You can also create new styles by selecting the Style Category from the Navigator tree, right-click and select the [Create <style\_name>] option. The following screenshot displays this menu for line styles:



The properties window will display where you can edit the properties for the style type:



## Map Builder Themes

A theme is a visual representation of a particular data layer. Typically, a theme is associated with a spatial geometry layer, that is, with a column of type SDO\_GEOMETRY in a table or view.

For example, a geometry theme named US\_STATES might be associated with a column named GEOMETRY in a STATES table.

Each theme is of one of the following types, each of which has a page for specifying properties for that type of theme:

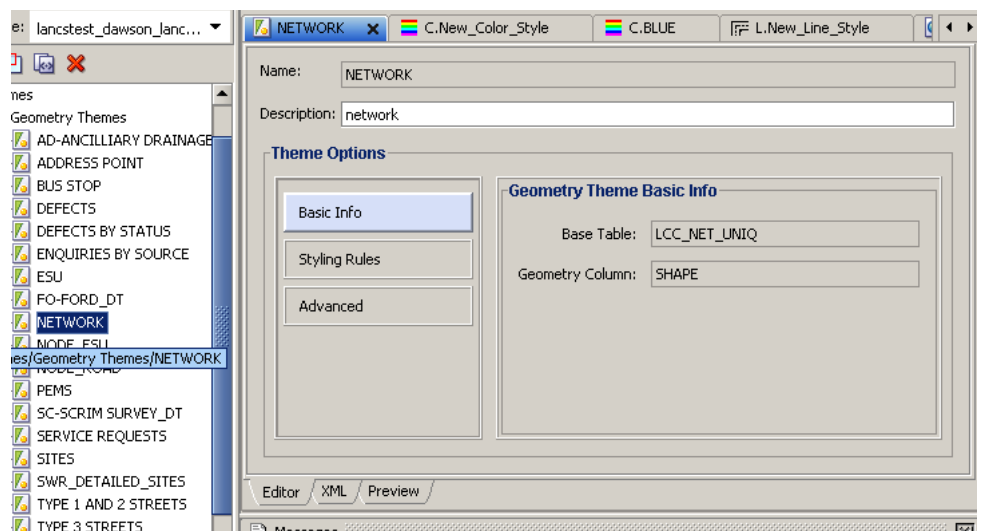
- Geometry themes are associated with Oracle Spatial geometry columns (type SDO\_GEOMETRY) in spatial tables.
- GeoRaster themes are associated with Oracle Spatial GeoRaster data (tables with SDO\_GEORASTER column). Image themes are associated with tables that have images (JPEG, GIF) stored on a BLOB column. The tables must also have a SDO\_GEOMETRY column to represent the image spatial extent.
- Network themes are associated with networks in the Oracle Spatial network data model (not used by exor)
- Topology themes are associated with topologies in the Oracle Spatial topology data model (feature tables with SDO\_TOPO\_GEOMETRY column). (not used by exor)

When you define a theme, you must specify a base table or view, a spatial data column in that table or view, and a set of additional parameters that depend on the theme type. For a predefined theme, the definition is permanently stored in the database. However, you can also dynamically define a theme (that is, create a JDBC theme) by supplying the definition within a map request. Oracle MapBuilder allows users to create predefined theme definitions.

In most cases you would be using and creating Geometry themes for use within the exor system.

Geometry Themes are associated with database tables that contain an attribute column of type SDO\_GEOMETRY. The database table must be registered on spatial USER\_SDO\_GEOM\_METADATA view. Each geometry theme can have the following properties: the Editor tab at the bottom displays properties in graphical mode; the XML tab displays the XML definition of the theme; the Preview tab displays how the theme might appear on a map.

The following screenshot displays the properties for the NETWORK theme (you can display the properties page by right-clicking on the theme and select the [Open] option):

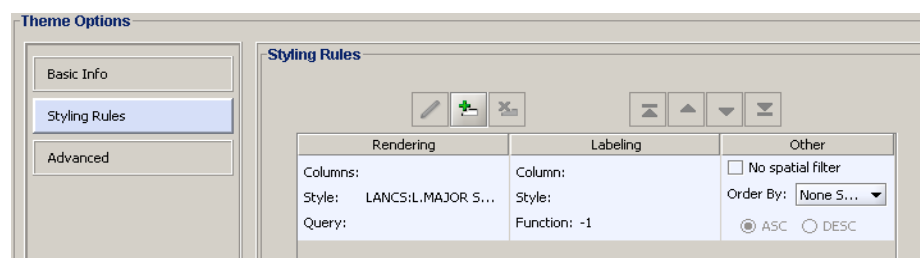


- **Name:** Name of the theme. Must be unique within a schema and **must be the same as the name specified in the GIS0020 form** for this theme.
- **Description:** Optional descriptive text about the theme.

**Basic Info:** select this item to view the base information associated with the theme.

- **Base Table:** Name of the table or view that has the spatial geometry column to be associated with this theme.
- **Geometry Column:** Name of the geometry column in the table or view to be associated with the theme.

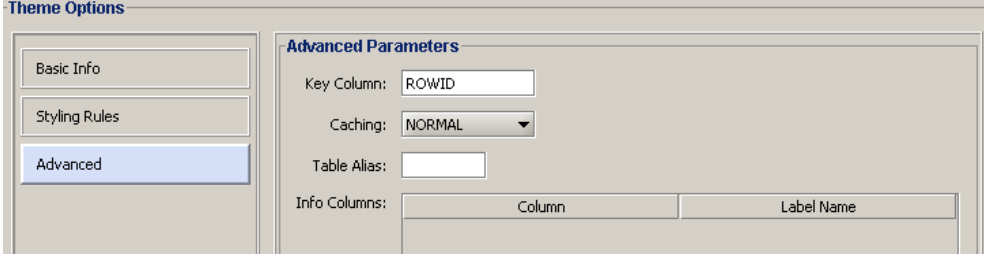
**Styling Rules:** Select this item to get a tabular visual representation of the XML styling rules to be used with the theme. The rendering, labeling, and other properties depend on the theme type.



The edit icons will be enabled by selecting a row on the table.

- **Icons :**
  - Edit the Current Row lets you edit the currently selected row. Add a New Row calls the dialogs to define the styling rule and adds it to table.
  - Delete the Current Row removes the selected row.
  - Move the Current Row to Top moves the selected row to the first row position.
  - Move the Current Row Up moves the selected row above the row that is currently above it.
  - Move the Current Row Down moves the selected row below the row that is currently below it.
  - Move the Current Row to Bottom moves the selected row to the last row position.

**Advanced:** Select this item to edit the advanced theme attributes:



The screenshot shows the 'Theme Options' dialog box with the 'Advanced Parameters' tab selected. The 'Key Column' is set to 'ROWID', 'Caching' is set to 'NORMAL', and 'Table Alias' is empty. Below these fields is a table for 'Info Columns' with two columns: 'Column' and 'Label Name'.

- **Key Column:** Name of the primary key column in the tables. This will usually default to ROWID. You will need to change this value when defining themes **for non-key preserved views**. For more information refer to the Non-Linear groups and Locator section
- **Caching:** NORMAL, NONE, or ALL.
  - **NORMAL** causes OracleAS MapViewer to try to cache the geometry data that was just viewed, to avoid repeating the costly unpickling process when it needs to reuse the geometries. Geometries are always fetched from the database, but they are not used if unpickled versions are already in the cache.
  - **NONE** means that no geometries from this theme will be cached. This value is useful when you are frequently editing the data for a theme and you need to display the data as you make edits.
  - **ALL** causes OracleAS MapViewer to pin all geometry data of this theme entirely in the cache before any viewing request. In contrast to the default value of NORMAL, a value of ALL caches all geometries from the base table the first time the theme is viewed, and the geometries are not subsequently fetched from the database.
- **Table Alias :** table alias name to be added on query. Must be defined if the geometry column is part of an embedded object.

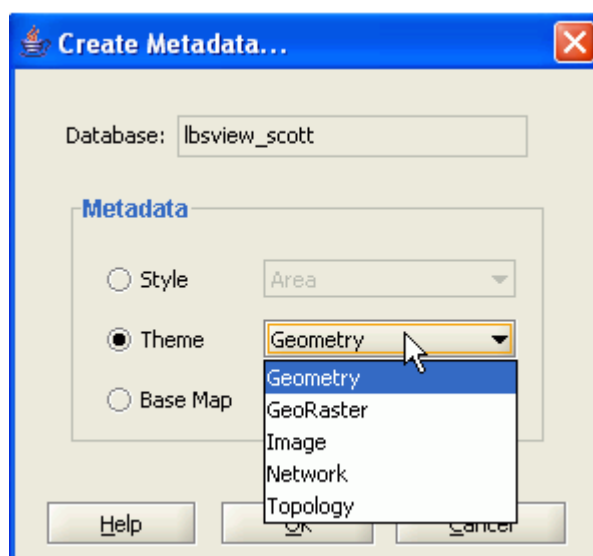
- **Info Columns** : One of more non-spatial columns (not of type SDO\_GEOMETRY) in the table. To add, delete, or edit informational columns, click pencil icon.

## Creating themes

You can create a theme using either of the following options:

### A. Using Menu or Tool Bar icon

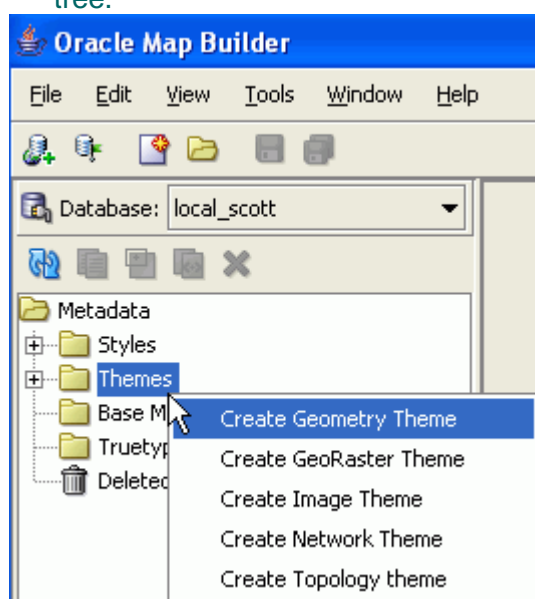
1. Navigate to File menu and select menu item New..., or click on tool bar icon . The create metadata dialog is shown



2. Select the Theme radio button, then select the theme type and press OK. The theme creation wizard will start.

### B. Using the Metadata Navigator

1. Right click the mouse over the Themes item on the metadata Navigator tree.

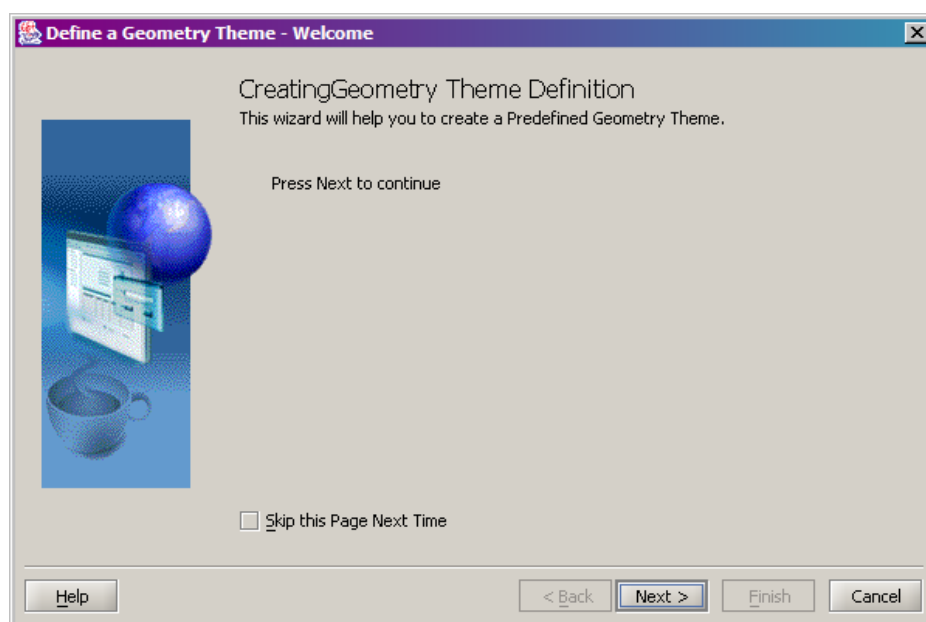


or

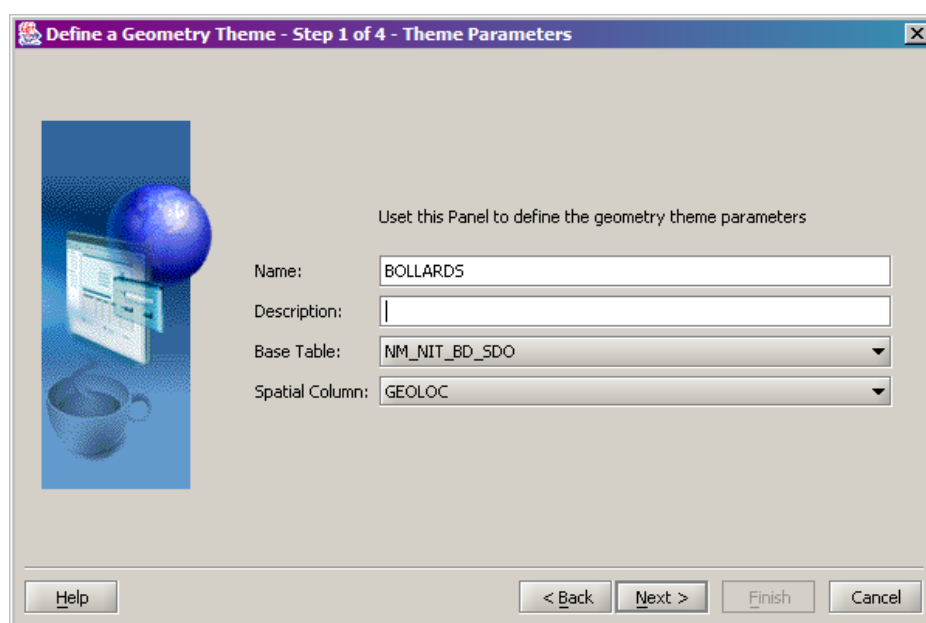


2. Right click the mouse over the theme type item on the metadata Navigator tree
3. Select the **[Create Geometry Theme]** menu option and the theme wizard will start.

### Theme Definition Wizard

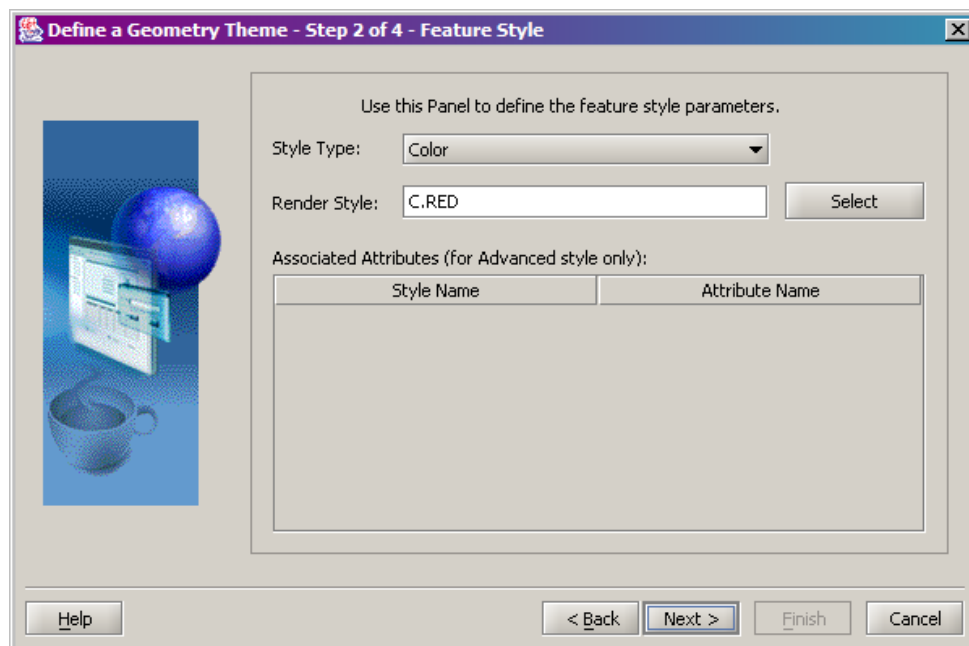


Click on [Next] after the initial page of the Theme Definition wizard displays



Enter the Theme Name, Theme Description (if needed) and select the base table for this theme from the drop-down list. Note that the Theme Name must be the same as the Name defined in the GIS0020 form for this theme.

Click on [Next]



Define a Geometry Theme - Step 2 of 4 - Feature Style

Use this Panel to define the feature style parameters.

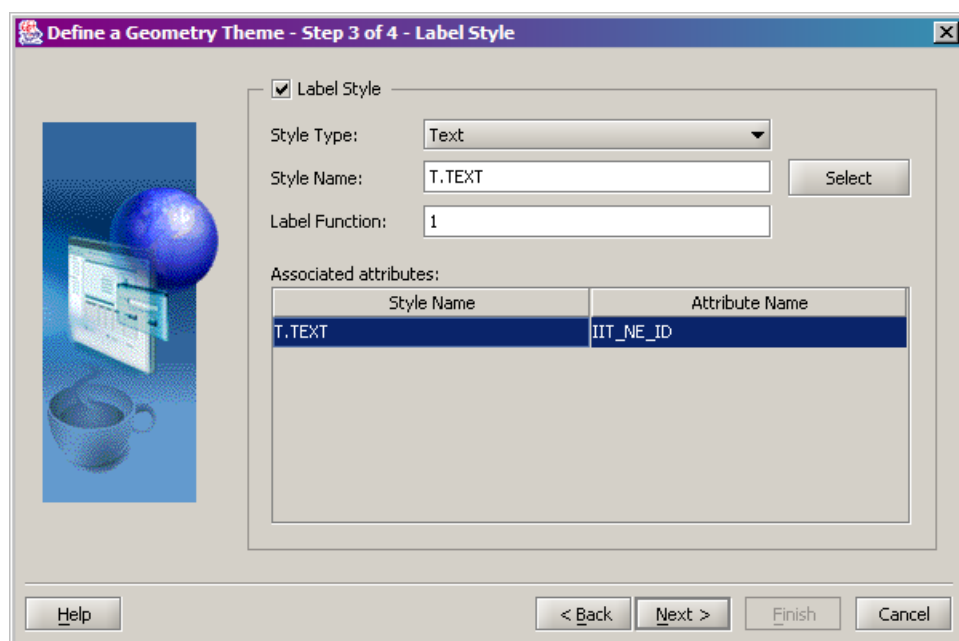
Style Type:

Render Style:

Associated Attributes (for Advanced style only):

Style Name	Attribute Name
------------	----------------

Define the styles for the theme in this form and click on [Next]



Define a Geometry Theme - Step 3 of 4 - Label Style

☒ Label Style

Style Type:

Style Name:

Label Function:

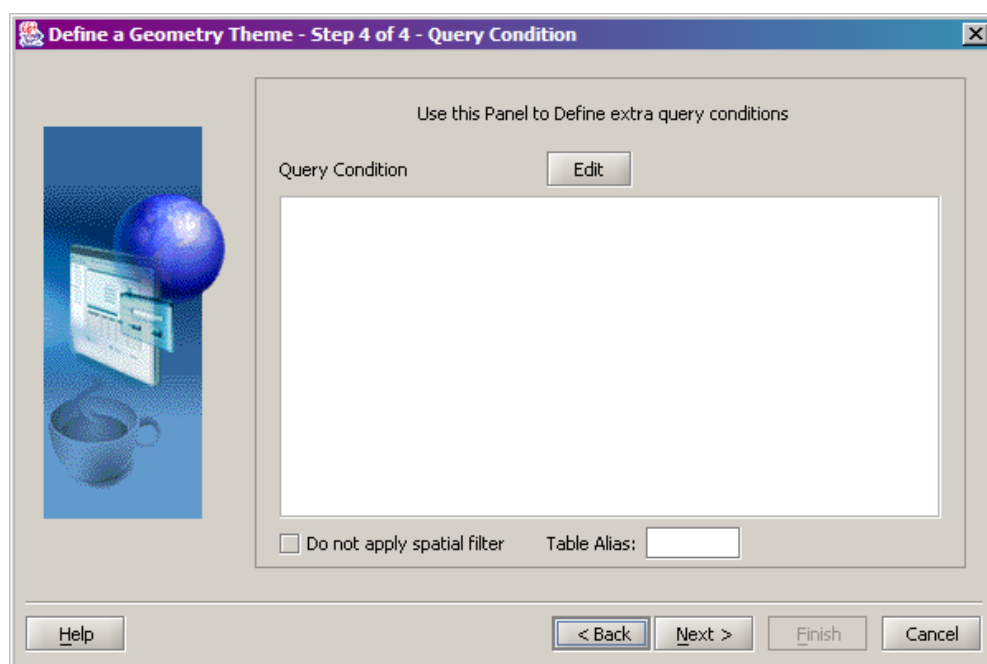
Associated attributes:

Style Name	Attribute Name
T.TEXT	IIT_NE_ID

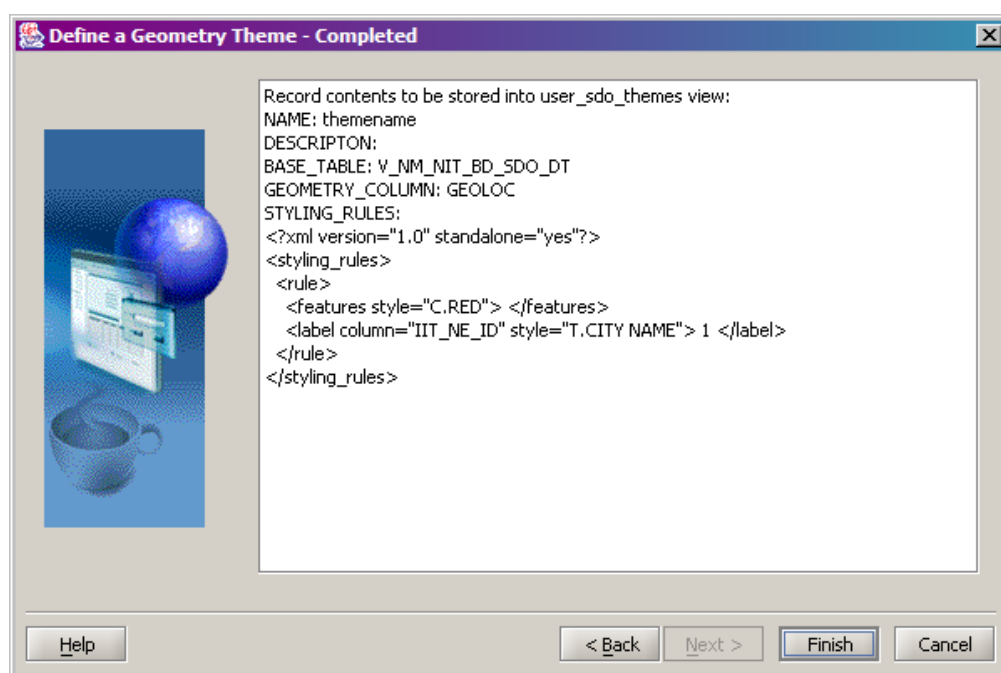
In this window you can define the theme label details:

- **Style Type:** The text style
- **Style Name:** Click on the [Select] button to select one of the predefined style names
- **Label function:** Will always default to 1
- **Associated attributes:** Defines the column name for the label. Click on the [Attribute Name] drop down list to select from the available columns in the spatial table

Click on [Next]



Define here any query conditions required to limit the number of features returned from the server and click on [Next]



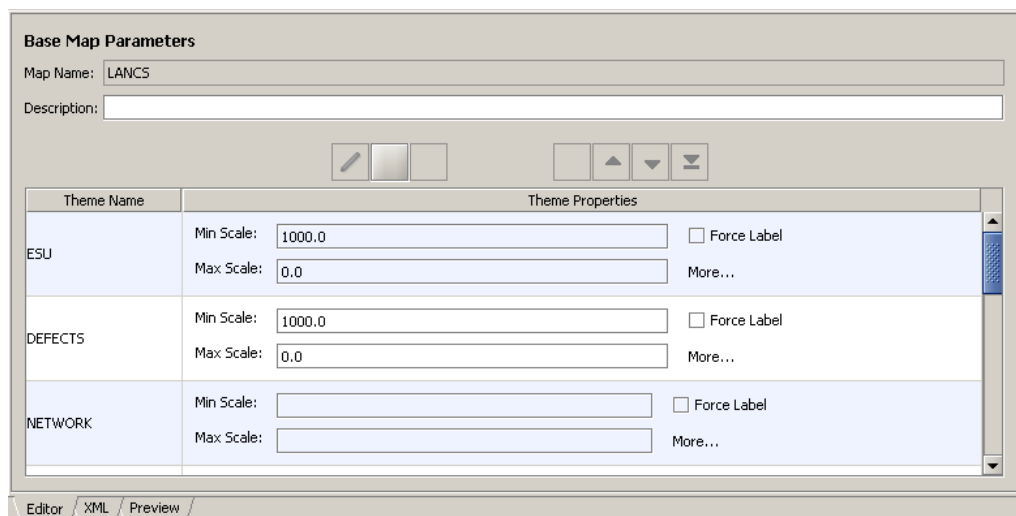
The last window will present an overview of the theme definition as defined in the previous steps. Press on [Finish] to create the new theme definition.

## Map Builder Base Maps

A base map consists of one or more themes to be used in rendering a map. If a base map is specified in a map request, the themes in the base map are rendered, one on top of each preceding one, in the order specified in the map definition. You can specify additional themes in a map request, to be rendered on top of the base map (if one is specified).

When OracleAS MapViewer receives a map request, it builds a theme list from all themes included in the base map (if a base map is specified), as well as any specified predefined or JDBC themes. All individual features in the request are grouped into a single temporary theme. In other words, after parsing the incoming request, all data that must be shown on the map is presented in a list of themes to the OracleAS MapViewer rendering engine.

For example, you may have two base maps, each displaying the same predefined theme named HIGHWAYS. In one base map, you might specify that the highways theme must always label its features, while in the other base map highway features gets labeled only if necessary and when there is no conflict. (The Force Labeling property lets you control this specific behavior.) The Base Map editor is shown on the following figure:



**Base Map Parameters**

Map Name: LANCS

Description:

Theme Name	Theme Properties	
ESU	Min Scale: 1000.0 Max Scale: 0.0	<input type="checkbox"/> Force Label More...
DEFECTS	Min Scale: 1000.0 Max Scale: 0.0	<input type="checkbox"/> Force Label More...
NETWORK	Min Scale: Max Scale: 	<input type="checkbox"/> Force Label More...

Editor XML Preview

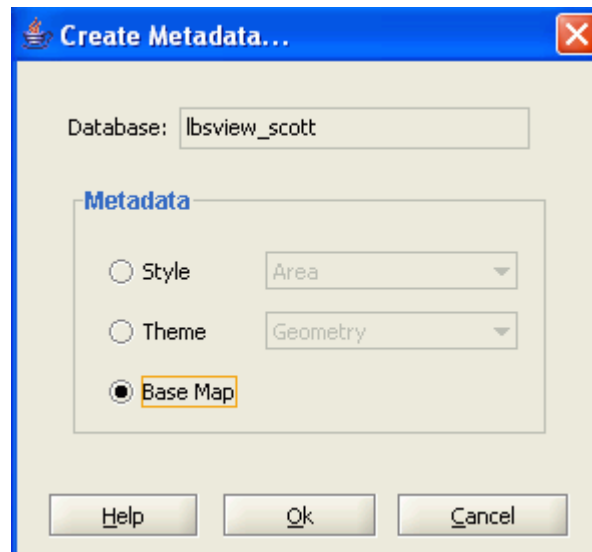
- **Map Name:** the base map name. This should reflect the **WEBMAPDSRC** product option
- **Description:** optional descriptive text about the base map.
- **Icons :**
  - **Edit Current Row** lets you edit the currently selected row.
  - **Add a New Row** opens a dialog to select a theme and add a row to the table.
  - **Delete the Current Row** removes the selected row.
  - **Move the Current Row to Top** moves the selected row to the first row position.
  - **Move the Current Row Up** moves the selected row above the row that is currently above it.
  - **Move the Current Row Down** moves the selected row below the row that is currently below it.
  - **Move the Current Row to Bottom** moves the selected row to the last row position.
- **Table:** contains the theme names and their properties. To edit the theme property, select the table row and click on the edit Icon above the table.

## Creating a Base Map

You can create a base map using either of the following options:

### A. Using Menu or Tool Bar icon

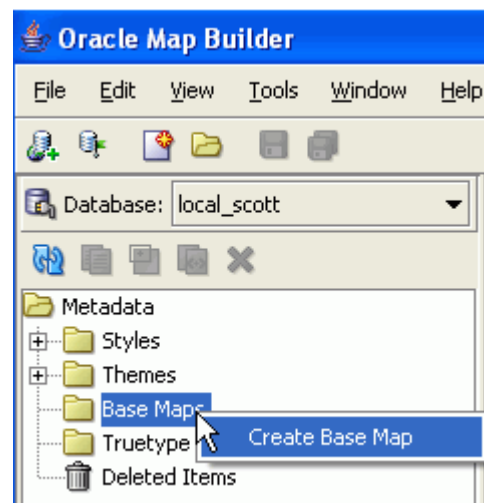
1. Navigate to File menu and select menu item New..., or click on tool bar icon . The create metadata dialog is shown



2. Select Base Map radio button and press OK. The base map creation wizard will start

### B. Using the Metadata Navigator

1. Right click the mouse over the Base Maps item on the metadata Navigator tree.



2. Select the menu option Create Base Map and the base map wizard will start.

### Base Map Wizard

This wizard allows users to create a base map definition to be stored into USER\_SDO\_MAPS view using the current themes.

#### Step 1: Base Map Basic Parameters

User defines the basic map information.

**Define a Base Map - Step 1 of 2 - Base Map Parameters**

Use this Panel to define the basemap basic parameters

Name:

Description:

Buttons: Help, < Back, Next >, Finish, Cancel

- **Name** : the new base map name. Must be different from the existing base maps. This should reflect the **WEBMAPSRC** product option
- **Description** : an optional text description about the theme.

## Step 2: Base Map Themes

User selects the themes to be part of the base map, and also assigns visualization scale values for themes.

**Define a Base Map - Step 2 of 2 - BaseMap Themes**

Use this Panel to select the base map themes

Theme Name	Min Scale	Max Scale
ADDRESS POINT		
BUS STOP		
DEFECTS		
DEFECTS BY STATUS		

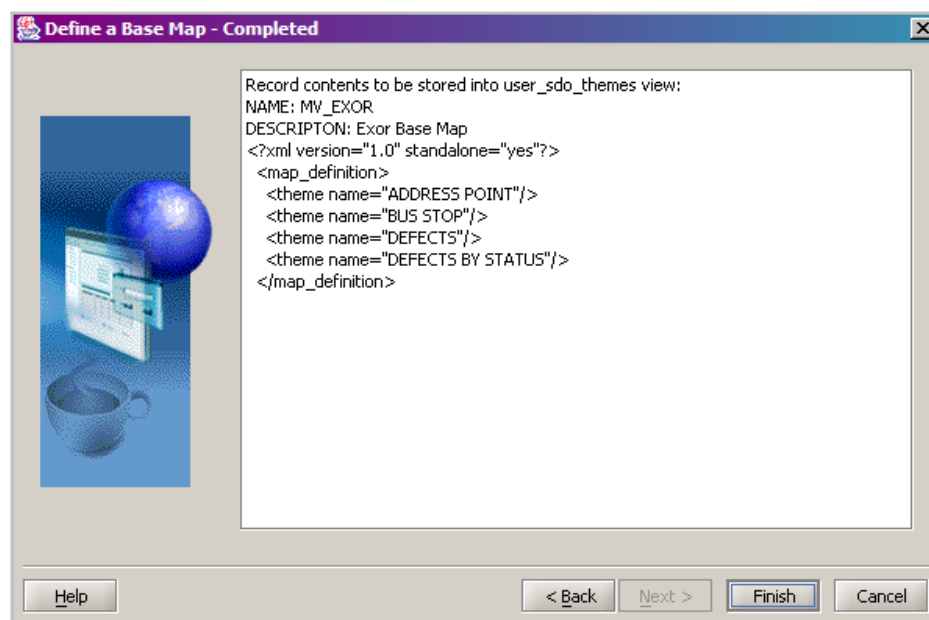
Buttons: Help, < Back, Next >, Finish, Cancel

These are the only two attributes that you can define for themes in this wizard. When editing an existing base map, the user will have the option to change other attributes. The interface has a list of themes on the left side, and a table with added themes on the right side. To add a theme, select it on

the theme list and click the Insert new theme icon. To edit a theme, select the table row and use the icon buttons.

### Step 3: Finish Press button

Finish to create base map.



The base map name will be added to the Metadata Navigator tree and the base map editor will be added to the editor page

## CHAPTER

## 2

## Importing Existing data

This chapter provides information on how to import spatial user schemas.

When you import an instance that contains spatial tables and views SDE will NOT automatically register them.

To check this run the following from a DOS prompt:

```
sdelay -o describe -u <user> -p <passwd>
```

If you get this message:

```
You do not have access to any layers in this database.
```

it means that no tables are registered.

According to SDE documentation this should have automatically registered any SDO tables but it doesn't. Sometimes, if you register the first table manually, SDE will trigger the automatic registration by then running the `sdelay -o describe`. Otherwise you will have to register all SDO tables manually.

If any of the spatial tables has a PK that is not of type NUMBER(38) you will have to change the column type before you register it in the SDE metadata.



**Example- Changing an integer column to a NUMBER(38)**

The following example uses the NM\_NSG\_ESU\_SHAPES table.

**You might have to disable these triggers first:**

- \* \_A\_ROW and
- \* \_B\_ROW

Now run this in SQL:

```
alter table nm_nsg_esu_shapes
add ne_id2 number(38);

update nm_nsg_esu_shapes
set ne_id2 = ne_id;

alter table nm_nsg_esu_shapes drop constraint esu_pk;

alter table nm_nsg_esu_shapes
drop column ne_id;

alter table nm_nsg_esu_shapes
rename column ne_id2 to ne_id;
```

## Registering the spatial tables and views

You will first need to register the nodes layer. To do this start a SQL session as the highways owner and type:

```
BEGIN
  nm3sdm.refresh_node_layers;
END;
```

This will create a theme for the NM\_POINT\_LOCATIONS and register all the spatial views for each node type.

You will then need to register all the spatial views in the MDSYS metadata by running this script:

```
DECLARE
  CURSOR tabs
  IS
    SELECT table_name, column_name, diminfo, srid
      FROM user_sdo_geom_metadata
     WHERE table_name LIKE '%NM_NIT%'
        OR table_name LIKE '%NM_NAT%'
        OR table_name LIKE '%NM_NLT%'
        OR table_name LIKE '%NM_ONA%';
BEGIN
  FOR i IN tabs
  LOOP
    --
    BEGIN
      INSERT INTO user_sdo_geom_metadata
        (table_name, column_name, diminfo, srid
         )
        VALUES ('V_' || i.table_name, i.column_name,
                i.diminfo, i.srid
                );
    EXCEPTION
      WHEN OTHERS THEN NULL;
    END;
    --
    BEGIN
      INSERT INTO user_sdo_geom_metadata
        (table_name, column_name, diminfo,
         srid
         )
        VALUES ('V_' || i.table_name || '_DT',
                i.column_name, i.diminfo,
                i.srid
                );
    EXCEPTION
      WHEN OTHERS THEN NULL;
    END;
```

```
--  
    END LOOP;  
END;
```

Finally you will have to register all the tables and view in the SDE metadata using this procedure:

```
BEGIN  
    nm3sde.regenerate_sde_from_themes;  
END;
```

Note that you must have completed the previous steps and register the tables and views in MDSYS, before you run this script otherwise the script will fail

## Addressing problems with individual layers

If for any reason registration for any of the above layers fails you can perform the following steps manually for each layer

### Step 1- registering the datum layer

```
D:\>sdelayer -o register -l <sdo_table_name>,shape -e lm
-c NE_ID -C USER -k SDO_GEOMETRY -u <user> -p <passwd>
```

### Step 2- registering the points spatial table

To register the points table (*NM\_POINT\_LOCATIONS is a standard NM3 spatial table*)

```
D:\>sdelayer -o register -l
nm_point_locations,npl_location -e p -c NPL_ID -C USER -k
SDO_GEOMETRY -u <user> -p <passwd>
```

### Step 3- registering the node spatial views

When importing data, any spatial views do not have an entry into the USER\_SDO\_GEOM\_METADATA (a synonym for MDSYS.SDO\_GEOM\_METADATA\_TABLE so you will have to add these entries manually.

The following example is for the nodes view of type 'ROAD'

```
insert into mdsys.sdo_geom_metadata_table select
sdo_owner, 'V_NM_NO_ROAD_SDO', 'GEOLOC', sdo_diminfo,
sdo_srid from mdsys.sdo_geom_metadata_table where
sdo_table_name='NM_POINT_LOCATIONS' and sdo_owner='NM3CC'
```

You then must register the view in ArcSDE e.g.:

```
C:\>sdelayer -o register -l V_NM_NO_ROAD_SDO,GEOLOC -e p -
c NO_NODE_ID -C USER -k SDO_GEOMETRY -u nm3ncc -p nm3ncc
```

```
ArcSDE 9.1 Oracle9i Build 1269 Thu Mar 3 18:54:58 PST
2005
Layer Administration Utility
-----
Successfully Created Layer.
```

## Step 4 – Setting the extents of the layers in the sde metadata

When registering a layer in the SDE metadata the extents are not set. To change them to the correct extents you will first need to find the min and max values for the layer extents. You can retrieve this information from the MDSYS metadata by typing the following in a SQL window:

```
SQL> select diminfo from user_sdo_geom_metadata where
table_name='SDO_NETWORK';

DIMINFO(SDO_DIMNAME, SDO_LB, SDO_UB, SDO_TOLERANCE)
-----
-
SDO_DIM_ARRAY(SDO_DIM_ELEMENT('X', 3812068.92, 5994598.99, .0000005),
SDO_DIM_ELEMENT('Y', 3349586.96, 4302916.57, .0000005),
SDO_DIM_ELEMENT('M', 0, 214748.365, 0))
```

Note down the first two values for the X and Y coordinates. You can refer in the DIMINFO Description topic for more information on the Diminfo object type

*(Note: Instead of 'X' and 'Y' your settings might be NULL or 'EASTING' and 'NORTHING' but that should not make any difference. The first and second set of values will always denote the X and Y dimensions respectively.)*

You will now need to change the layers envelope using the `sdelay -o alter` command

```
sdelay -o alter -l <table,column>
[-E xmin,ymin,xmax,ymax]
-u <DB_User_name> -p <DB_User_password>]
```

Example (using the datum layer from the previous paragraph):

```
sdelay -o alter -l sdo_network,shape -E 3812068.92,3349586.96,
5994598.99, 4302916.57 -u nm3 -p nm3
```

**Note: You have to repeat this command for EVERY layer you manually registered in ArcSDE**

## Step 5 – Group and Asset layers

You can use the same steps as described above to re-register Groups and asset layers that may have not been registered properly

*CHAPTER*

**4**

## Using the GIS0010 form

This chapter describes the processes and forms used to manage theme-specific metadata. The form covered in this chapter is:

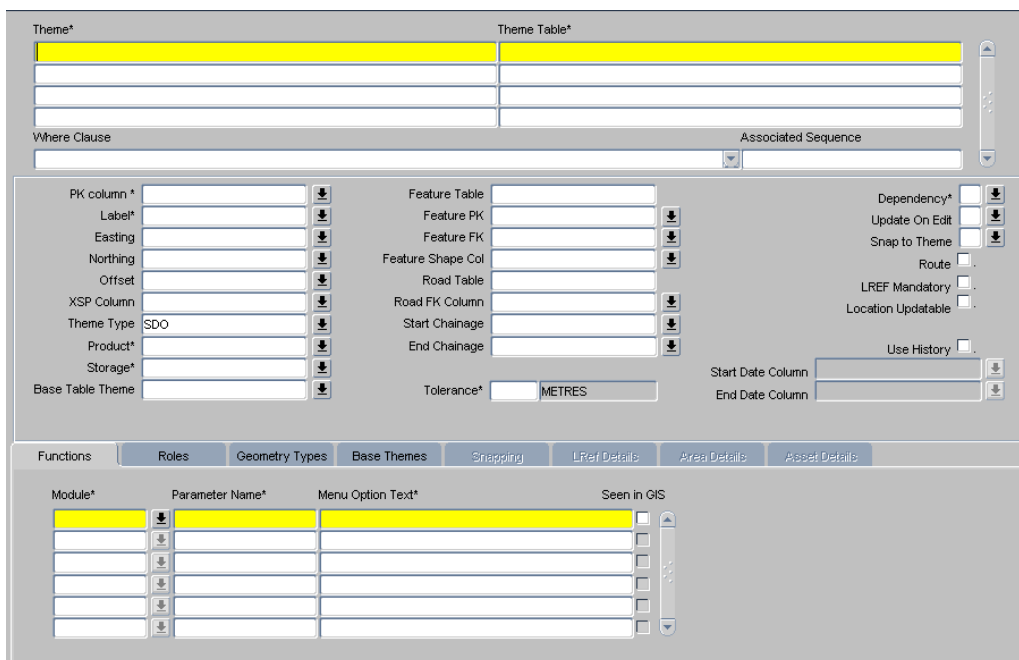
- GIS Themes – GIS0010

## Introduction

The GIS Themes—GIS0010 module provides an environment to manage information about spatial data used by any Exor product. Each different view of each spatial data set must be assigned a GIS theme (also called Exor theme) record using this module.

**Important Note:** The GIS0010 form will be automatically populated after creating a theme using the GIS0020 form as described in the previous paragraphs. You shouldn't really have to update the themes using this form apart for error correction purposes or layer fine-tuning such as adding roles or functions for each theme.

**Figure 1-1**  
GIS Themes Form



The screenshot displays the GIS Themes Form (GIS0010) with the following sections:

- Theme\*** and **Theme Table\*** fields at the top.
- Where Clause** and **Associated Sequence** dropdown menus.
- PK column \*** and **Label\*** fields.
- Easting**, **Northing**, and **Offset** fields.
- XSP Column** and **Theme Type** (set to SDO) fields.
- Product\*** and **Storage\*** fields.
- Base Table Theme** field.
- Feature Table** and **Feature PK** fields.
- Feature FK** and **Feature Shape Col** fields.
- Road Table** and **Road FK Column** fields.
- Start Chainage** and **End Chainage** fields.
- Tolerance\*** field (set to METRES).
- Dependency\***, **Update On Edit**, **Snap to Theme**, **Route**, **LREF Mandatory**, **Location Updatable**, and **Use History** checkboxes.
- Start Date Column** and **End Date Column** fields.
- Functions** tab selected, showing a table with columns: **Module\***, **Parameter Name\***, **Menu Option Text\***, and **Seen in GIS**.

## General Information

GIS Themes is an environment to manage information about spatial data representing Assets and other record types needed for them to be kept up to date in the Exor database by **Spatial Services (SSV)** and displayed and edited using Exor spatial application products: **Spatial Manager (SM)**, and **Map Services (MSV)**. Theme information is stored in the Exor database in the NM\_THEMES\_ALL table and a number of related tables.

Asset themes use the Inventory Type Views created from the **Asset Metamodel—NM0410** form and the **Group Types—NM0004** form. Theme records for these views are generated from the Asset type metadata **Metamodel—NM0410** forms.

This form must contain an entry for every spatial layer used by any Exor Product.

This includes linear and non linear group types, asset types or other record type stored in the database as Oracle Spatial (SDO) tables for use with Exor products. It may also include Oracle Spatial tables holding background data

(such as coastline, rivers or other topographic data) so that this data is easily accessible in SM and MSV.

**Please note that this form is populated automatically by the system when new asset or linear group types that will be stored as SDO layers are created. Changes to these entries should only be made by the Database Administrator.**

You can access the GIS0010 form from the Themes Option in the GIS Data menu in Launchpad as shown in Figure 2.2

**Fig. 1.2**  
**GIS Themes Launchpad Icon**



## GIS Themes

When this form is opened the cursor sits in the 'Theme' field waiting for the name of a new exor theme to be entered or existing Themes to be queried back. To query back one or more existing Exor Themes press the [**Enter Query**] button on the menu toolbar (or press **F7**), enter the selection criteria for the themes to be returned, then press the [**Execute Query**] button (or press **F8**).

To create a new theme record fill in the following fields:

### Theme (Required)

This is a unique descriptive key field for the Theme record. Enter a unique theme name. This name will be displayed in the SM legend when the theme is displayed. (MSV will display whatever is in the Oracle Map Viewer map definition tables)

This updates the NTH\_THEME\_NAME field in the NM\_THEMES\_ALL table.

### Theme Table (Required)

The name of the table or view where the source attribute data for the Theme resides.

Caution should be used before entering the network view name (the V\_NM\_<????>\_NW 'View Name' field of the **Asset Metamodel—NM0410** form) in this field.

This view was formally used for client side dynamic segmentation but is no longer used. It will be slow and only reason to use this view is to have datum measures as attributes of the item.



This updates the NTH\_THEME\_TABLE field in the NM\_THEMES\_ALL table.

---

**Where Clause** (Optional)

Enter the where clause for the layer if you wish to use to limit the number of inventory items returned.

This updates the NTH\_WHERE field in the NM\_THEMES\_ALL table.

---

**Associated Sequence** (Optional)

This is used to assign values to the OBJECTID column on Spatial Services layers in the database.

If the system is set up to use surrogate key, then each Layer generated using the non-dynseg method from Asset Metamodel should populate the NTH\_SEQUENCE attribute on the Theme.

This updates the NTH\_SEQUENCE\_NAME field in the NM\_THEMES\_ALL table.

---

**PK Column** (Required)

The primary key column for the asset view (displayed in the Asset Metadata-NM0410 form).

For standard inventory views (i.e. not external tables) this value must be set to **NE\_ID**.

This updates the NTH\_PK\_COLUMN field in the NM\_THEMES\_ALL table.

---

**Label** (Required)

The value in the label column indicates the field in the Theme entity table to be used by SM and MSV for labeling functionality.

This updates the NTH\_LABEL\_COLUMN field in the NM\_THEMES\_ALL table.

---

**Easting** (Optional)

The column in the view that contains the X-coordinate for the location of each record in the Theme. Note that only locations for point assets, Accidents, Structures or Enquiries can be defined by XY coordinates

This updates the NTH\_X\_COLUMN field in the NM\_THEMES\_ALL table.

---

**Northing** (Optional)

The column in the view that contains the Y-coordinate for the location of each record in the Theme. Note that only locations for point assets, Accidents, Structures or Enquires can be defined by XY coordinates

This updates the NTH\_Y\_COLUMN field in the NM\_THEMES\_ALL table.

---

**Offset** (Optional)

This field can either hold the name of the column in the view that contains the horizontal offset distance for each inventory record from the centerline.

**The Offset value can be any number.** If it is a number all inventory items will be offset from the road centerline by that value.

**The offset only applies to UKPMS layers used in SM.** It does not apply to SDE or SDO layers. A negative number is used to offset inventory items to the left.

This updates the NTH\_OFFSET\_COLUMN field in the NM\_THEMES\_ALL table.

---

#### **XSP Column (Not Used)**

This field is not used in the current release. It is reserved for an optional column within the Theme entity table to indicate the cross-sectional position.

This would update the NTH\_XSP\_COLUMN field in the NM\_THEMES\_ALL table.

---

#### **Theme Type (Optional)**

This field can take three possible values:

- LOCL
- SDO
- SDE

The values are used by the spatial server procedures and indicate the format of the layer. Themes of type LOCL (local) will be assumed that are created using dynamic segmentation in the client GIS and will be ignored by the server procedures. Themes of type SDO (Oracle Spatial) will be handled by server procedures. Themes of type SDE (ESRI SDE format BLOB) will be ignored by server procedures and will be added in SM as 'standard' ArcGIS themes.

This updates the NTH\_THEME\_TYPE field in the NM\_THEMES\_ALL table.

---

#### **Product (Required)**

This field will contain one of the product codes (NET, ACC, STR, etc) for the licensed Exor product that will be used to maintain the record type of the theme.

A list of example product values for different record types is provided below:

- | • <b>Record type</b>  | <b>Product value</b> |
|-----------------------|----------------------|
| • Assets              | NET                  |
| • Network             | NET                  |
| • Accidents           | ACC                  |
| • Structures          | STR                  |
| • Enquiries           | ENQ                  |
| • Defects             | MAI                  |
| • Structural Projects | PMS                  |

This updates the NTH\_HPR\_PRODUCT field in the NM\_THEMES\_ALL table.

---

#### **Storage (Required)**

The values are used by the spatial server procedures and indicate whether the spatial layer will be stored persistently or derived by the client GIS using dynamic segmentation. In most cases the location will be stored and the Storage field should be set to "S".

This field can take two possible values:

- D (Derived)
- S (Stored)

If the layer is derived, server procedures will ignore it.

This updates the NTH\_STORAGE field in the NM\_THEMES\_ALL table.

---

#### **Base Table Theme (Optional)**

If the theme table is based on a database view this field will contain the name of the table with the true spatial features that the theme view is based on.

You only need to fill this out for LOCL themes which you wish to convert to SDO layers. You do not need to edit this for asset layers generated by the SSV server.

For example if the theme table (not the SDO table) that contains all the defects is called DEFECTS and the theme name for this table is ALL\_DEFECTS and you have a view of defects which is based on all Priority 1 defects called V\_DEFECT\_PR1 with a theme name of DEFECTS\_PR1, you would set the base table theme to 'ALL\_DEFECTS'. So for V\_NM\_NIT\_PVIN\_SDO the Base Table would be the NM\_NIT\_PVIN\_SDO table theme.

This updates the NTH\_BASE\_TABLE\_THEME field in the NM\_THEMES\_ALL table.

---

#### **Feature table (Optional for Assets – Required for SDO or SDE type Themes)**

This field contains the name of the Oracle spatial table or view that contains the geometries of the items to be displayed in this theme.

This updates the NTH\_FEATURE\_TABLE field in the NM\_THEMES\_ALL table.

---

#### **Feature PK (Optional but Required for SDO or SDE type Themes)**

This field is used only for the network layer to specify the primary key column for the spatial table or if an SDE inventory view is used to define the inventory items.

This updates the NTH\_FEATURE\_PK\_COLUMN field in the NM\_THEMES\_ALL table.

---

#### **Feature FK (Optional but Required for SDO or SDE type Themes)**

It will be populated automatically by server procedures for Asset or Group layers

This updates the NTH\_FEATURE\_FK\_COLUMN field in the NM\_THEMES\_ALL table.

### **Feature Shape Col** (Optional but Required for SDO or SDE type Themes)

This field defines the column in the spatial table (Feature Table value) which holds the geometries. It can hold the actual shape as a geometry or a foreign key to a long-rawout-of-line shape.

It is populated automatically by server procedures for Asset or Group layers

This updates the NTH\_FEATURE\_SHAPE\_COLUMN field in the NM\_THEMES\_ALL table.

### **Road Table** (Required)

This should always be set to NM\_ELEMENTS, preceded with the table owner (i.e. NM31.NM\_ELEMENTS)

This updates the NTH\_RSE\_TABLE\_NAME field in the NM\_THEMES\_ALL table.

### **Road FK Column** (Optional)

The route name column in the entity table on which the Theme is linearly referenced. The foreign key column in the view that links back to the NM\_ELEMENTS table. Only applies to inventory data created using dynamic segmentation.

For standard inventory views (i.e. not external tables) this value will need to be set to **NE\_ID\_OF**.

This updates the NTH\_FEATURE\_FK\_COLUMN field in the NM\_THEMES\_ALL table.

### **Start Chainage** (Optional)

The column in the inventory view that contains the value for the start measure for each inventory record. This only applies to **point or continuous** features created using dynamic segmentation.

For standard inventory views i.e. not external tables this value will need to be set to **NM\_BEGIN\_MP**.

This updates the NTH\_ST\_CHAIN\_COLUMN field in the NM\_THEMES\_ALL table.

### **End Chainage** (Optional)

The column in the inventory view that contains the value for the end measure for each inventory. Only applies to **linear** inventory data created using dynamic segmentation. If it is null then the entity is assume to be a point representation.

For standard inventory views i.e. not external tables this value will need to be set to **NM\_END\_MP**.

This updates the NTH\_END\_CHAIN\_COLUMN field in the NM\_THEMES\_ALL table.

### **Tolerance (Required and Defaulted)**

This field defines the search tolerance for snapping functions. It is only used for snapping Defects and Enquiries associated with a network location or asset type. These records will not be snapped to Theme elements further away than this tolerance value.

This updates the NTH\_TOLERANCE field in the NM\_THEMES\_ALL table.

### **Tolerance Units (Display only)**

The Tolerance Units field is to the right of the Tolerance field and displays the units of the Tolerance value. This defaults to metres.

For projected spatial feature layers where x and y coordinates increment in another unit the system may apply this unit instead of metres.

This displays the NTH\_TOL\_UNITS field in the NM\_THEMES\_ALL table.

### **Dependency (Required)**

This value indicates to the SSV procedures whether the spatial layer is dependent on another layer or not. Typically the datum layer will be flagged as 'I' whereas server-created asset and route layers which depend on the datum layer will be flagged as 'D'. For external SDE or dynamic segmentation layers this value should be set to 'I'.

This field can take two possible values:

- D (Dependent)
- I (Independent)

This updates the NTH\_DEPENDENCY field in the NM\_THEMES\_ALL table.

### **Update on Edit (Required)**

The values are used by the spatial server procedures and indicate whether any edits in the logical data will be reflected dynamically in the layer or whether they should be deferred and be updated at later stage using a DBMS job.

This field can take three possible values:

- I (Immediate)
- D (Deferred)
- N (None)

Note that for the base tables of route layers (NM\_NLT\_<NWTYPE>\_<GRPTYPE>\_SDO) this option should be set to 'I' whereas for the resulting views (V\_NM\_NLT\_<NWTYPE>\_<GRPTYPE>\_SDO and V\_NM\_NLT\_<NWTYPE>\_<GRPTYPE>\_SDO\_DT) it should be set to 'D' due to the complexity of the process and for performance reasons.

For external SDE or dynseg layers set this value to 'N'

This updates the NTH\_UPDATE\_ON\_EDIT field in the NM\_THEMES\_ALL table.

### **Route** (Display only)

Route theme is a display only flag. It indicates whether the theme represents the datum network. It must be unchecked or set to null for other themes. Used in older versions and is only maintained for backward compatibility reasons.

This displays the GT\_ROUTE\_THEME field which is derived from the NM\_NW\_THEMES table (based on Linear types).

### **LREF Mandatory** (Flag field)

Flag to denote if features in the Theme are to be snapped to linear layer. It only applies to off-network assets. It is unchecked by default

This updates the NTH\_LREF\_MANDATORY field in the NM\_THEMES\_ALL table.

### **Snap to Theme** (Flag field)

Flag to denote that the spatial feature must be snapped to the centerline of a route (or located along a route using reverse dynamic segmentation). This only applies to off-network assets and non-asset layers (such as enquiries or defects).

This field can take two possible values:

- N (No)
- S (Snap)

It is set to “N” by default.

If the Flag is set to “S” then the Snapping Tab will be activated and must be used to specify the Theme that records are to be snapped to.

This updates the NTH\_SNAP\_TO\_THEME field in the NM\_THEMES\_ALL table.

### **Location Updateable** (Flag field)

This is a flag to indicate if the spatial representation of the object is updatable. If this box is not checked, users would not be able to relocate the feature in MSV or SM.

This updates the NTH\_LOCATION\_UPDATABLE field in the NM\_THEMES\_ALL table.

### **Use History** (Flag field)

Flag to indicate if the spatial feature table supports history. It should be checked for dynamic segmentation layers only (where the Dependency flag is set to D).

If the 'Use History' flag is set for Asset Type layers, then the Start Date and End Date Column fields MUST be 'START\_DATE' and 'END\_DATE' (to be consistent with the rest of SSV Themes and procedures to create layer tables).

The Use History flag - this has been set on 3.2.0.0 to 3.2.1.0 upgrade script so all Asset and Groups SSV layers have this field updated.

If a new Theme is created using the **Create Layer – Reuse Existing Theme form** the 'Use History' flag can be used to generate another Theme, based on a view of the table theme. If the history flag is not set, then only one Theme will be created based on the table. If a layer is created with History, then Roles are applied to the View theme, therefore the view theme should be used.

For FT layers, you use any column that exists on the FT at time of creation.

This updates the NTH\_USE\_END\_DATE field in the NM\_THEMES\_ALL table.

### Start Date Column

Column that holds the start date column name if history is used.

This updates the NTH\_START\_DATE field in the NM\_THEMES\_ALL table.

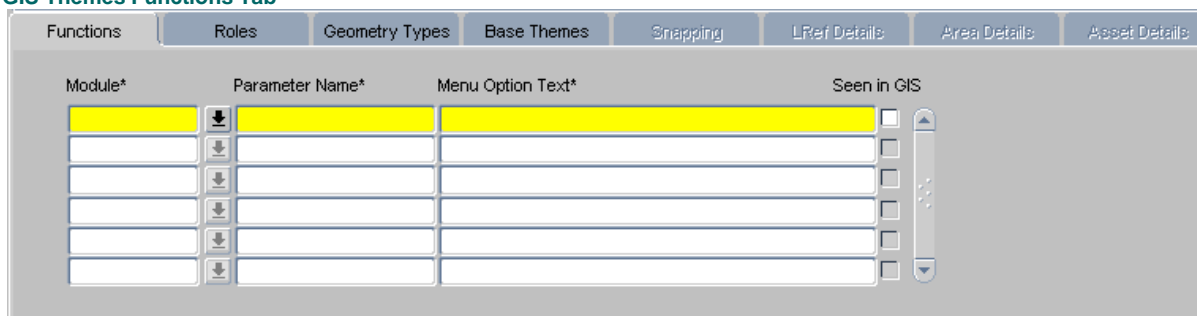
### End Date Column

Column that holds the end date column name if history is used.

This updates the NTH\_END\_DATE field in the NM\_THEMES\_ALL table.

## Functions Tab

Figure 2-GIS Themes Functions Tab



Module*	Parameter Name*	Menu Option Text*	Seen in GIS
			<input type="checkbox"/>
			<input type="checkbox"/>
			<input type="checkbox"/>
			<input type="checkbox"/>
			<input type="checkbox"/>
			<input type="checkbox"/>

The functions tab determines the forms (modules) to be used for records represented by an 'Exor Theme' when called from the **Locator—NM0572** module "Select to" function.

The form that is specified will be called from Locator if users have selected features in the relevant theme and the user selects the Module's name under the [Select to] function. Any selected records are written to a database table called GIS\_DATA\_OBJECTS, which is queried by the form and used to call the objects selected in Locator. A form listed against the Theme can also be used to zoom to objects selected in the form by subsequently clicking on the 'Show Map' button. Again the GIS\_DATA\_OBJECTS table is utilized, this time the form writes to the table and Locator queries the table to pick up the objects to select and zoom to.

For a Module to be available in the **Locator—NM0572** form “Select to” menu, it must:

- Be defined in theme functions
- Have 'Seen in GIS' set to Y
- Have a role matching one currently allocated to the connected user

#### **Module (Required)**

Enter the name of the module as per the table above

This updates the NTF\_HMO\_MODULE field in the NM\_THEME\_FUNCTIONS\_ALL table.

#### **Parameter Name (Required)**

This value will always default to GIS\_SESSION\_ID

This updates the NTF\_PARAMETER field in the NM\_THEME\_FUNCTIONS\_ALL table.

#### **Menu Option Text (Required)**

Maintained only used for historic purposes. The Locator function does not use this value.

This updates the NTF\_MENU\_OPTION field in the NM\_THEME\_FUNCTIONS\_ALL table.

#### **Seen in GIS (Flag Field)**

This flag is used to indicate if the module can be called directly from the GIS client application. There are occurrences where the a module needs a lot more information than is available inside the GIS, and in these cases only a one-way interface can be supported.

This will be set to “Y” for any themes generated by the server. It must be set to yes This flag will be set to “Y” for any themes generated by the server.

It must be set to yes for the module to be available in the **Locator—NM0572** form “Select to” function.

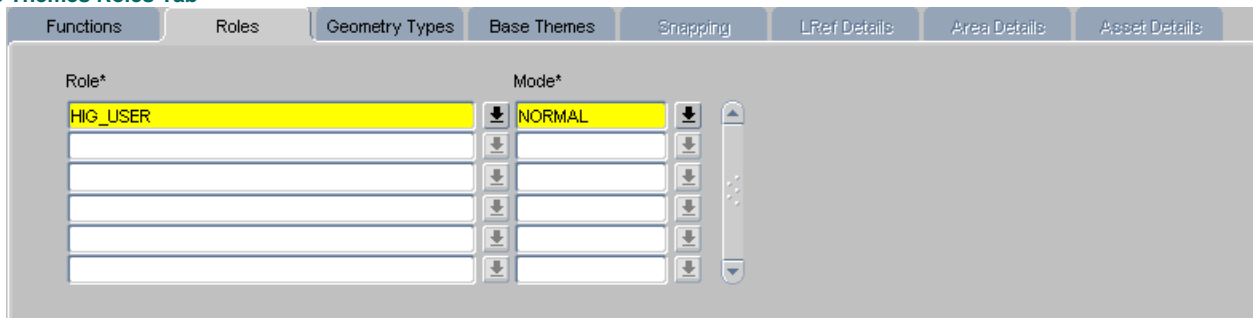
A module such as that which contains results of a query tool such as merge or pbi can easily navigate to the GIS - each result record is identifiable as a particular theme and this allows the GIS to query the records. This means the module is configured against many themes (ie every theme which relates to a potential result). However, in the current structure of the forms, it makes no sense for the GIS to attempt to call the module - how would the GIS call the merge results form? Any modules that have this flag set to false should not be present in the LOV.

This updates the NTF\_SEEN\_IN\_GIS field in the NM\_THEME\_FUNCTIONS\_ALL table.



## Roles Tab

Figure 1-4  
GIS Themes Roles Tab



The Roles tab associates GIS Themes and User roles and controls the way a theme is accessed via SD, SM, and MSV. If a user does not possess a role that is listed under a theme, the theme will not be displayed in the tree view in SM.

User Roles must have been set up using the **Roles—Hig1836** form before they can be selected here.

### Role (Required)

Enter the Role a user must have to access spatial features of this type.

This updates the NTHR\_ROLE field in the NM\_THEME\_ROLES table.

### Mode (Required)

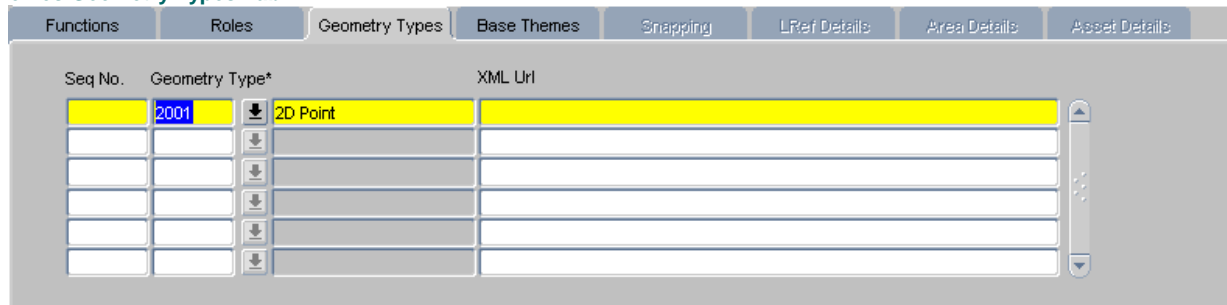
Enter the Mode of access the user may have. Valid entries are:

- NORMAL (read and write permissions to the database)
- READONLY (no write permission to the database)

This updates the NTHR\_ROLE the NM\_THEME\_ROLES table.  
This updates the NTHR\_MODE field in the NM\_THEME\_ROLES table.

## Geometry Types Tab

Figure 1-5  
GIS Themes Geometry Types Tab



The Geometry Type Tab indicates the spatial feature types used in the Theme.

### Seq No. (Required)

Enter a unique sequence number for the geometry type.

This will be important for future-releases where a one to many relationship between Theme and Geometry type will be supported - so that users can specify a mixture of Geometry types in a Layer. At present only one Geometry Type should be specified for each theme.

### **Geometry Type (Required)**

Enter the Oracle Spatial code for the Geometry Type used in the theme layer.

Valid values are:

- 2001 2D Point
- 2002 2D Line
- 2003 2D Polygon
- 2004 2D Collection
- 2005 2D Multi-point
- 2006 2D Multi-line
- 2007 2D Multi-polygon
- 3001 3D Point
- 3002 3D Line
- 3003 3D Polygon
- 3004 3D Collection
- 3005 3D Multi-point
- 3006 3D Multi-line
- 3007 3D Multi-polygon

For layers generated by the server, only three Geometry Type values are used:

- 2001 - 2D Point
- 2002 - 2D Line
- 3002 - 3D Line

Layers registered by users for foreign table assets or background layers can be any valid Geometry type, as long as the Theme matches the Layer, but there are geometry types that **may not be rendered by the ESRI ArcView technology used for SM.**

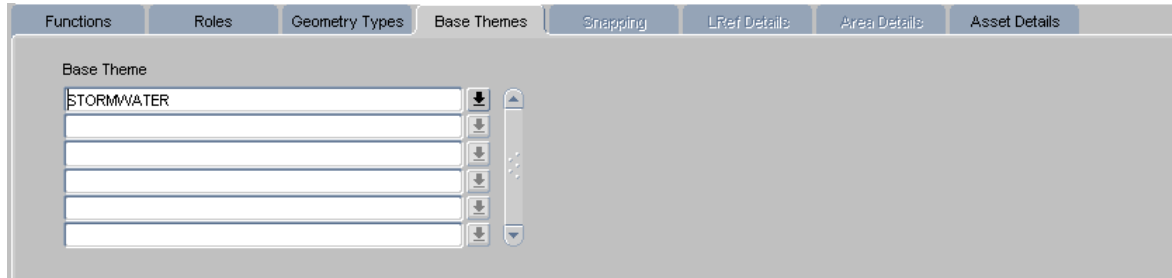
ESRI tools should have no problem with geometry types 2001, 2002, 2003, 3001 3002, and 3003. For types 3006 and 3007, it will depend on the ESRI metadata and whether this matches the geometry type.

### **XML Url (Not Used)**

This field is not used in this release. It will be the URL of the XML schema of the geometry which will be used to convert geometries from and to GML clobs.

## Base Themes Tab

Figure 3- GIS Themes Base Themes Tab



The Base Themes tab displays the Base Theme(s) for the current them

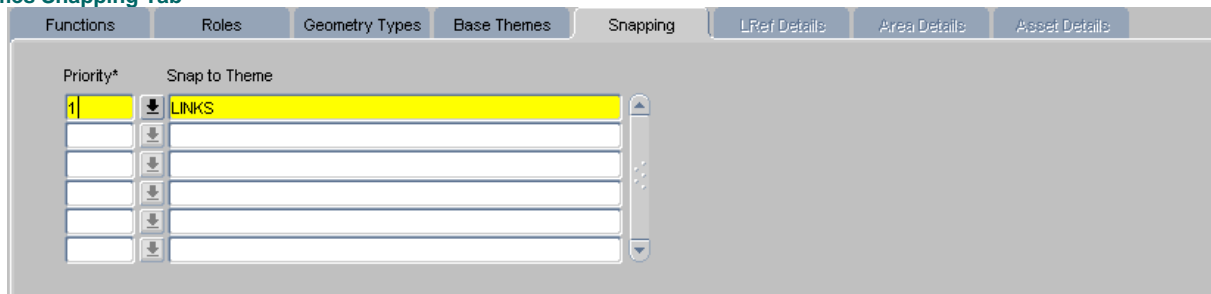
### Base Theme (Read Only)

Shows layers (themes) from which the theme has been derived.

This updates the NM\_BASE\_THEMES table.

## Snapping Tab

Figure 1-6  
GIS Themes Snapping Tab



The Snapping tab is only enabled when the Theme is flagged as Snapping. (Snap to Theme is set to S)

Users then must enter the Linear Network layers to be snapped to - in a priority order (there can be more than one). This is used in MSV when creating a new feature.

### Priority (Required)

Enter the order in which features in the theme are to be snapped to the target.

This updates the NTS\_PRIORITY field in the NM\_THEME\_SNAPS table.


### Snap to Theme (Required)

Enter the name of the target theme that features in the theme are to be snapped to. This is limited to Linear groups and datum layers.

This updates the NTS\_SNAP\_TO the NM\_THEME\_SNAPS table.

## LRef Details Tab

Figure 1-7  
GIS Themes LRef Details Tab



Functions	Roles	Geometry Types	Base Themes	Snapping	LRef Details	Area Details	Asset Details
<div>Seq: 1</div> <div>Description: CS1</div> <div>Network Type: CS1</div> <div>Group Type: </div> <div>Units: METRES</div> <div>Admin Type: NET</div> <div>Start Date: 23-FEB-2006</div> <div>End Date: </div>							

The LREF tab provides information about the Linear Network Type (Linear flag checked in **Network Types – NM0002** form) details of the linear referencing layer being used for dynamic segmentation for the Theme.

It is only active when the Theme is associated with a Linear Network Type.

The LRef tab is read only and fields cannot be updated.

### **Seq (Read Only)**

Displays the sequence number for each Linear Reference type layer

### **Description (Read Only)**

Displays the description for the layer

### **Network Type (Read Only)**

Displays the network type for this layer

### **Group Type (Read Only)**

Displays the group type for this layer

### **Units (Read Only)**

Displays the units for the group type

### **Admin Type (Read Only)**

Displays the Admin Type for the specified network type

### **Start Date (Read Only)**

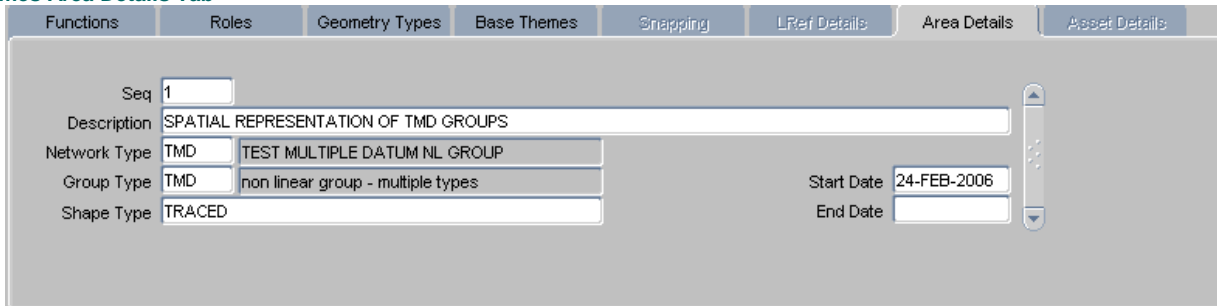
Displays the start data of the Linear Network Type

### **End Date (Read Only)**

Displays the start data of the Linear Network Type

## Area Details Tab

**Figure 1-8**  
**GIS Themes Area Details Tab**



Functions	Roles	Geometry Types	Base Themes	Snapping	LRef Details	Area Details	Asset Details
<div>Seq <input type="text" value="1"/></div> <div>Description <input type="text" value="SPATIAL REPRESENTATION OF TMD GROUPS"/></div> <div> <div>Network Type <input type="text" value="TMD"/></div> <div>Group Type <input type="text" value="TMD"/></div> <div>Shape Type <input type="text" value="TRACED"/></div> </div> <div> <div>Start Date <input type="text" value="24-FEB-2006"/></div> <div>End Date <input type="text"/></div> </div>							

The Area Details tab provides information about the Non-Linear Network Type (Linear flag unchecked in **Network Types – NM0002** form). information from nm\_area\_types - the non-linear Network type details, and whether it's Traced etc.

It is only active when the Theme is associated with a Non-Linear Network Type.

The Area Details tab is read only and fields cannot be updated.

### **Seq (Read Only)**

Displays the sequence number for each Non-linear group type

### **Description (Read Only)**

Displays the Description for each Non-linear group type

### **Network Type (Read Only)**

Displays the network type for this layer

### **Group Type (Read Only)**

Displays the Group type for this layer

### **Shape Type (Read Only)**

Displays the Shape Type for the layer

### **Start Date (Read Only)**

Displays the start date for the layer

### **End Date (Read Only)**

Displays the End data for the layer

## Asset Details Tab

**Figure 1-9**  
GIS Themes Asset Details Tab

Functions	Roles	Geometry Types	Base Themes	Snapping	LRef Details	Area Details	Asset Details
<div> <div>Asset Type</div> <div>ABS</div> </div>							
<div> <div>Description</div> <div>ABANDONED STRUCTURE</div> </div>							
<div> <div>View Name</div> <div>V_NM_ABS</div> </div>							
<div> <div>Point or Cont.</div> <div>P</div> </div>							
<div> <div>Use XY</div> <div>N</div> </div>							
<div> <div>Linear</div> <div>Y</div> </div>							
<div> <div>Foreign Table Name</div> <div></div> </div>							
<div> <div>Start Chain Column</div> <div></div> </div>							
<div> <div>End Chain Column</div> <div></div> </div>							
<div> <div>Network Foreign Key Column</div> <div></div> </div>							

The Asset Details tab provides information about the Asset Type (from **Asset Metamodel – NM0410** form).

It is only active when the Theme is associated with an Asset Type.

The Asset Details tab is read only and fields cannot be updated.

### **Asset Type (Read Only)**

Displays the Asset type

### **Description (Read Only)**

Displays the Asset type description

### **View Name (Read Only)**

Displays the view name for the asset type

### **Point or Cont (Read Only)**

Denotes whether the asset type is Point (P) or Continuous (C)

### **Use XY (Read Only)**

Denotes whether the asset type uses XY coordinates

### **Linear (Read Only)**

Displays whether the asset type is Linear or not

### **Foreign Table Name (Read Only)**

Displays the foreign table name the asset is derived from (if one exists)

### **Start Chain Column (Read Only)**

Displays the start chainage column for the foreign table

### **End Chain Column (Read Only)**

Displays the start chainage column for the foreign table

### **Network Foreign Key Column (Read Only)**

Displays the foreign key column for the foreign table