RDF DATABASES MADE ARCGIS FRIENDLY

Michael D. Tsengouras
Lead Engineer - NAVTEQ

2010 ESRI User Conference

15 July 2010

Background

- Internal project initiated to develop a RDF Visualization tool using Commercially Available tools and databases.
- The goals of this project included:
  - RDF loaded into Oracle Spatial Database Server:
    - Capitalizing on the power and tools available with Oracle Spatial
  - The Oracle Spatial Database Server has been configured for use with ArcGIS Server Enterprise Basic (ArcSDE):
    - Capitalizing on the power and tools available with ArcSDE as it interacts with ArcGIS Desktop

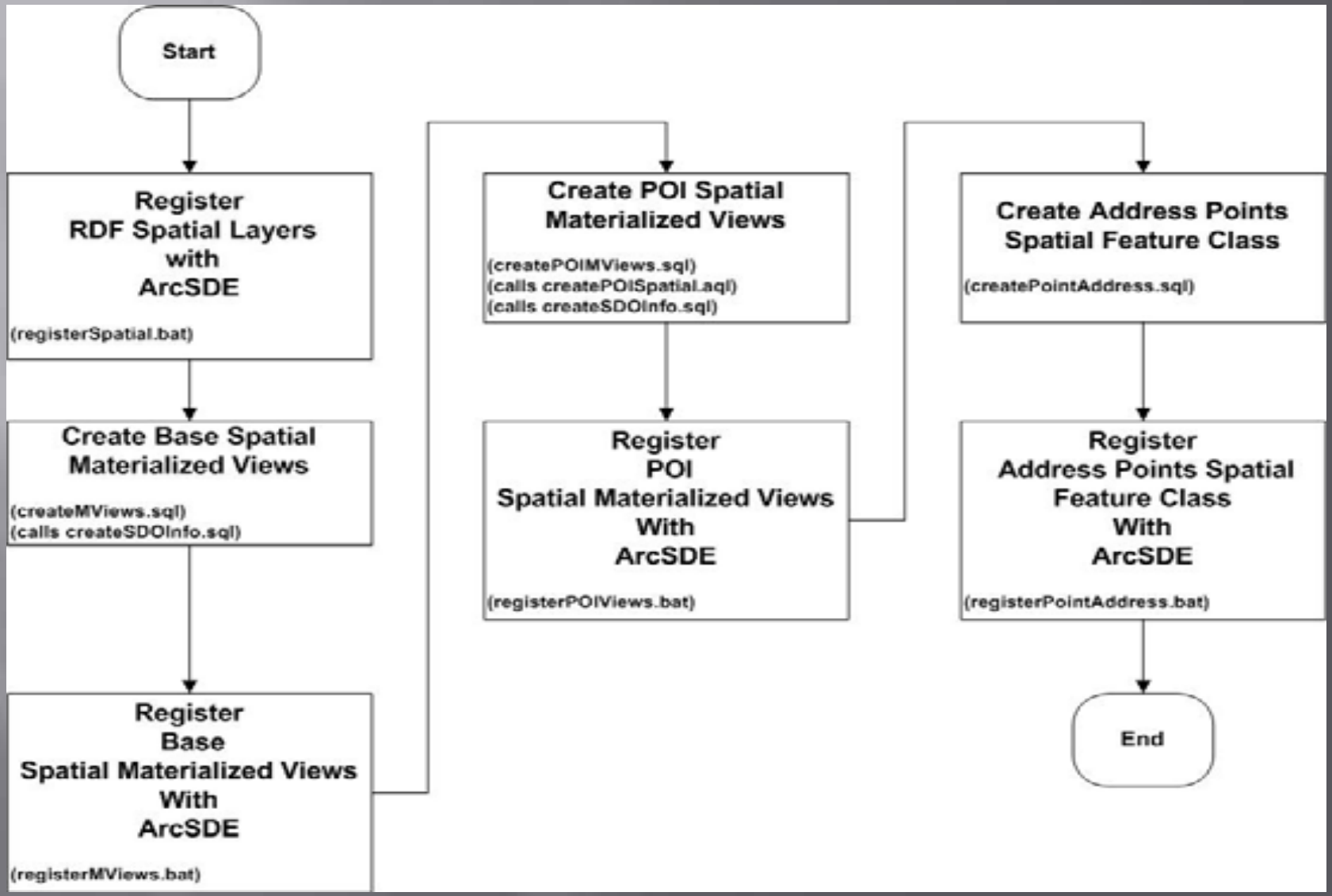
Background

- Providing a number of ways to access the ArcGIS Friendly data:
  - Accessible through ArcGIS Desktop directly from the Oracle Database (through ArcMAP)
  - Accessible through ArcGIS Desktop as a data feed served from ArcGIS Server Enterprise Advanced
  - Accessible through a Web based application served from ArcGIS Server Enterprise Advanced
- Researching the feasibility for providing this capability as:
  - Internal Development, Testing, and Quality Assurance Tool
  - RDF Product Enhancement for users of ESRI's ArcGIS tools

Data Source

- A standard RDF Oracle Customer Install is used to generate the content used for visualization within ArcGIS. For the purposes of this presentation, it is assumed that this installation has been completed.
- The following components of the standard RDF Oracle Customer Install are used:
  - Core
  - Spatial
  - Administrative
  - Address Points

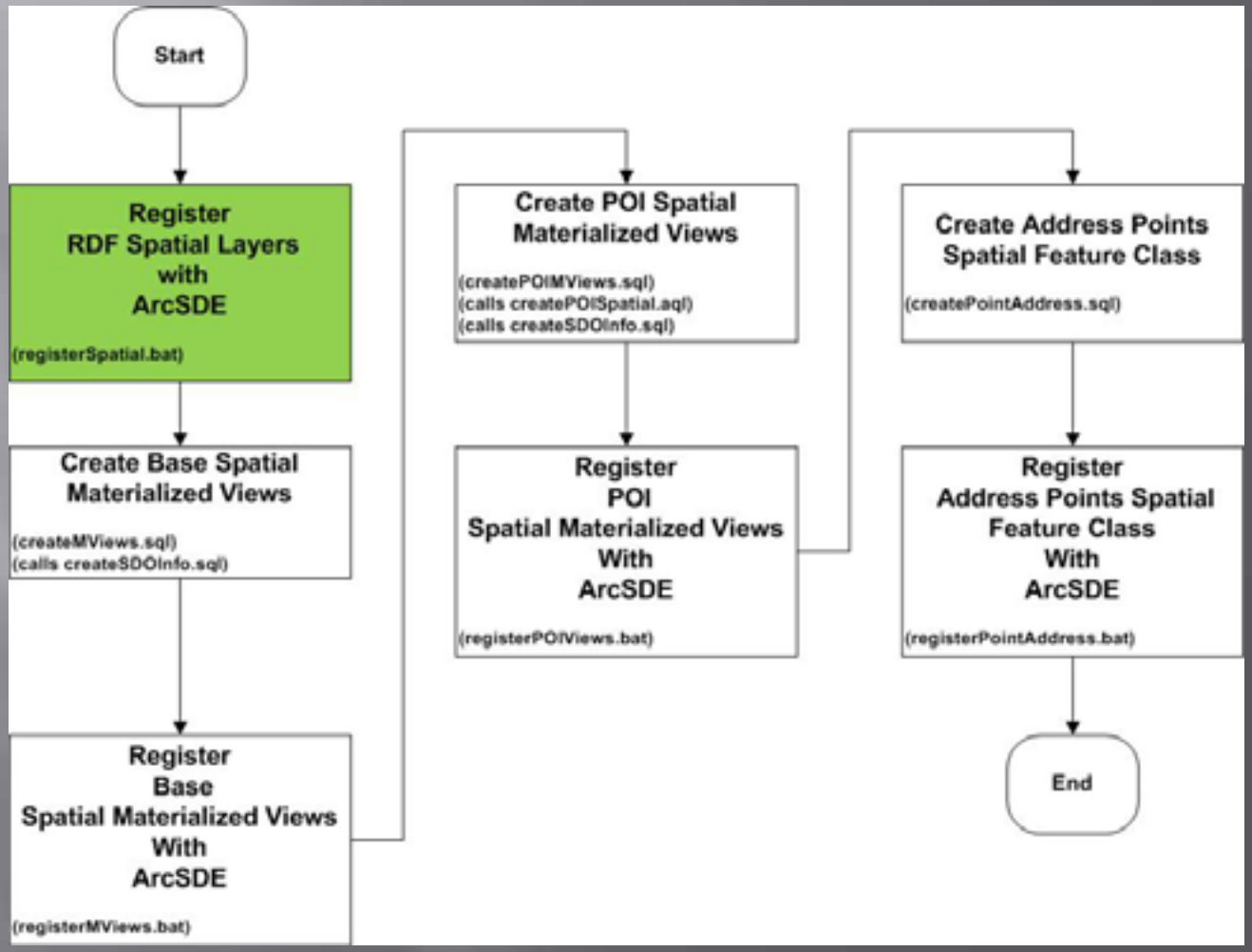
Data Preparation Flow



Data Preparation Flow

- This preparation flow can be totally automated
  - Currently implemented in Adobe ANT workflow Tool (Windows/DOS)
  - Uses standard out of the box Technology
    - ArcSDE Tools (sdelayer and sdetable)
    - SQL and PL/SQL (At the command line)
  - Can be implemented on UNIX/Linux command environments
- The data is flexible
  - RDF allows you to build your own, therefore the preparation flow is flexible.

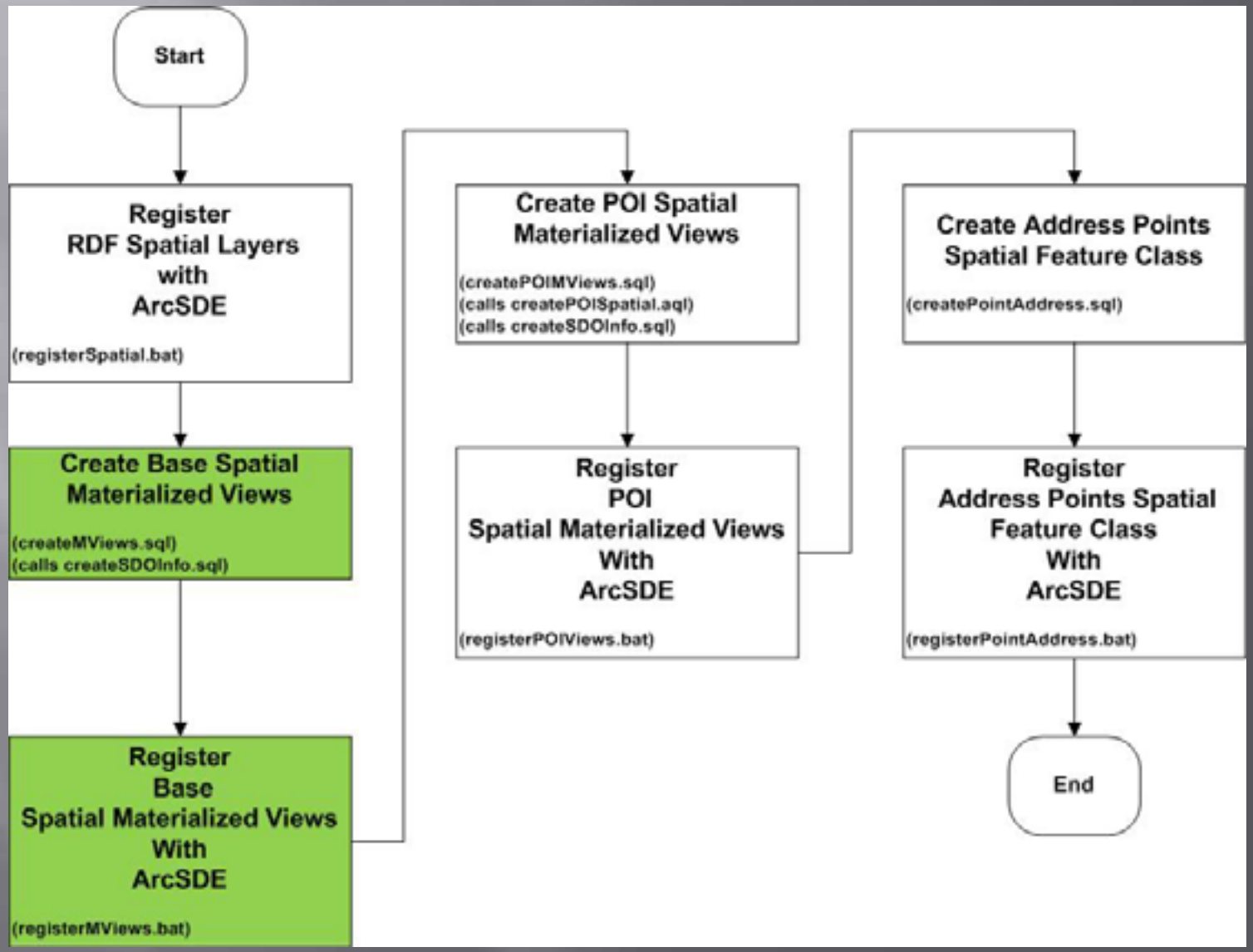
Step 1: Register RDF Spatial Layers



Register RDF Spatial Layers

- Provide the baseline information that ArcSDE needs to display this layer as an ArcGIS feature class.
- The "Spatial Component" installation of RDF provides the core tables to be used to develop other spatially enabled materialized views.
  - **SDO_CARTO – this spatial table contains the geometry for all polygonal features represented in the RDF dataset.** *Relationships to RDF Core tables are made through CARTO_ID.*
  - **SDO_LINK – this spatial table contains the geometry for all polyline (link) features represented in the RDF dataset. Relationships to RDF Core tables are made through LINK_ID.**
  - **SDO_LOCATION – this spatial table contains the geometry for all point features represented in the RDF dataset (with the exception of Address Points). Relationships to RDF Core tables are made through LOCATION_ID.**
  - **SDO_NODE – this spatial table contains the geometry for all node features represented in the RDF dataset. Relationships to RDF Core tables are made through NODE_ID.**

Step 2: Create Materialized Views



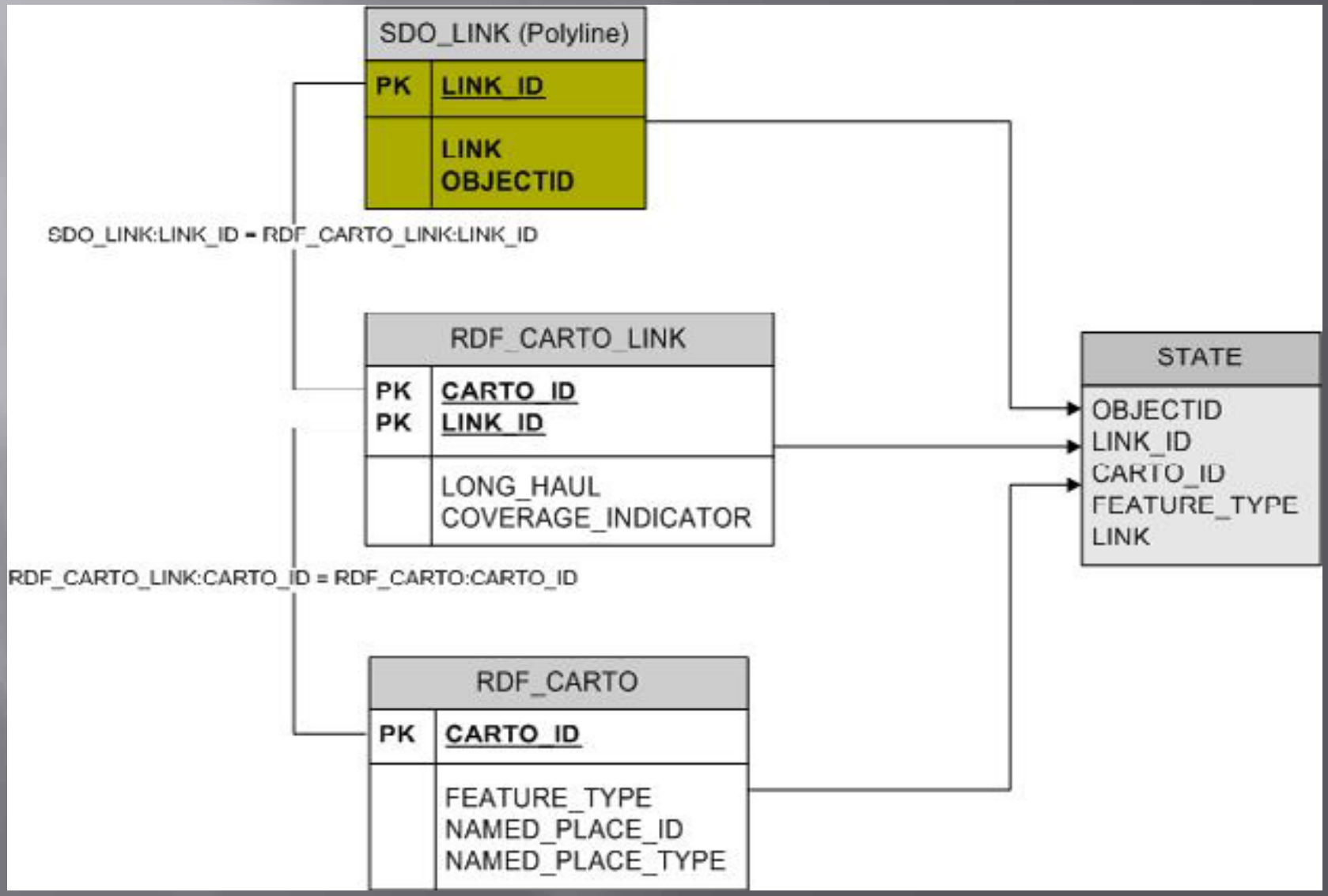
Why Create Materialized Views

- Primarily Performance as a Visualization Source within ArcGIS Desktop and Server
  - Where ArcSDE supports Views in Oracle, to create views required additional processing
    - Define and Register the view using ArcSDE Tools
    - Create the view using SQL
  - RDF Datasets used in the project were continental size
    - Oracle Views did not perform well in ArcGIS Desktop and Server because of their size
    - Small RDF Clips (At the Product Boundary level) performed well with Oracle View
  - Either View Methodology will work

Spatial Views Created

Materialized View	Feature Types represented
MAJOR_HWYS	9999999 Functional Class 1
SECONDARY_HWYS	9999999 Functional Class 2
MINOR_HWYS	9999999 Functional Class 3
STREETS	9999999
CARTO_FEATURES	All Named Cartographic Features
CARTO_BUILDING	Building Outlines (Unnamed) All feature Types in the range of 2005000 and 2005999
RAILROAD	1800201
COUNTRY	907196
STATE	909996
COUNTY	900170
CITY	900101
BUILT_UP_AREA	900156
LAKES	500421
RIVER_NAMED (Linear)	500412
RIVER_ALL (Linear)	500412 (unnamed)

Linear Features(Conceptual View)



Materialized View Script

```
-- Create STATE Polyline Materialized View
-- State in the US is a major identifying boundary.  The Feature Type
-- 909996 can be adjusted to represent other countries.
prompt ============ Create STATE Polyline Materialized View ==========
CREATE MATERIALIZED VIEW STATE
  BUILD IMMEDIATE
  REFRESH COMPLETE ON DEMAND
AS
SELECT
  SL.OBJECTID,
  SL.LINK_ID,
  RCL.CARTO_ID,
  RC.FEATURE_TYPE,
  SL.LINK
FROM
  SDO_LINK SL,
  RDF_CARTO_LINK RCL,
  RDF_CARTO RC
WHERE
  SL.LINK_ID = RCL.LINK_ID AND
  RCL.CARTO_ID = RC.CARTO_ID AND
  RC.FEATURE_TYPE = 909996
  ;

-- Add Oracle Spatial Information
@createSDOInfo STATE LINK STATE_SIN

-- Create Index on Primary Key(s)
CREATE INDEX STATES_LINK_ID_IDX ON STATE (LINK_ID);

commit;
```

Add Spatial Reference

```
-- Add Spatial Reference LAT/LONG WGS/84
declare
  diminfo mdsys.sdo_dim_array;
  srid number(10);
begin
  diminfo := mdsys.sdo_dim_array(
    mdsys.sdo_dim_element('XLONG', -180, 180, 0.5),
    mdsys.sdo_dim_element('YLAT', -90, 90, 0.5));
  srid := 8307;

  delete from user_sdo_geom_metadata
    where table_name = '&1';

  insert into user_sdo_geom_metadata
  values ('&1', '&2', diminfo, srid);

  commit;
end;
/

-- Create a Oracle Spatial Index
Create index &3 on  &1
      (&2) INDEXTYPE IS MDSYS.SPATIAL_INDEX
    PARAMETERS('SDO_INDX_DIMS=2
               SDO_MAX_MEMORY=2000000
               NEXT=2M
               MAXEXTENTS=UNLIMITED
               PCTINCREASE=0');
```

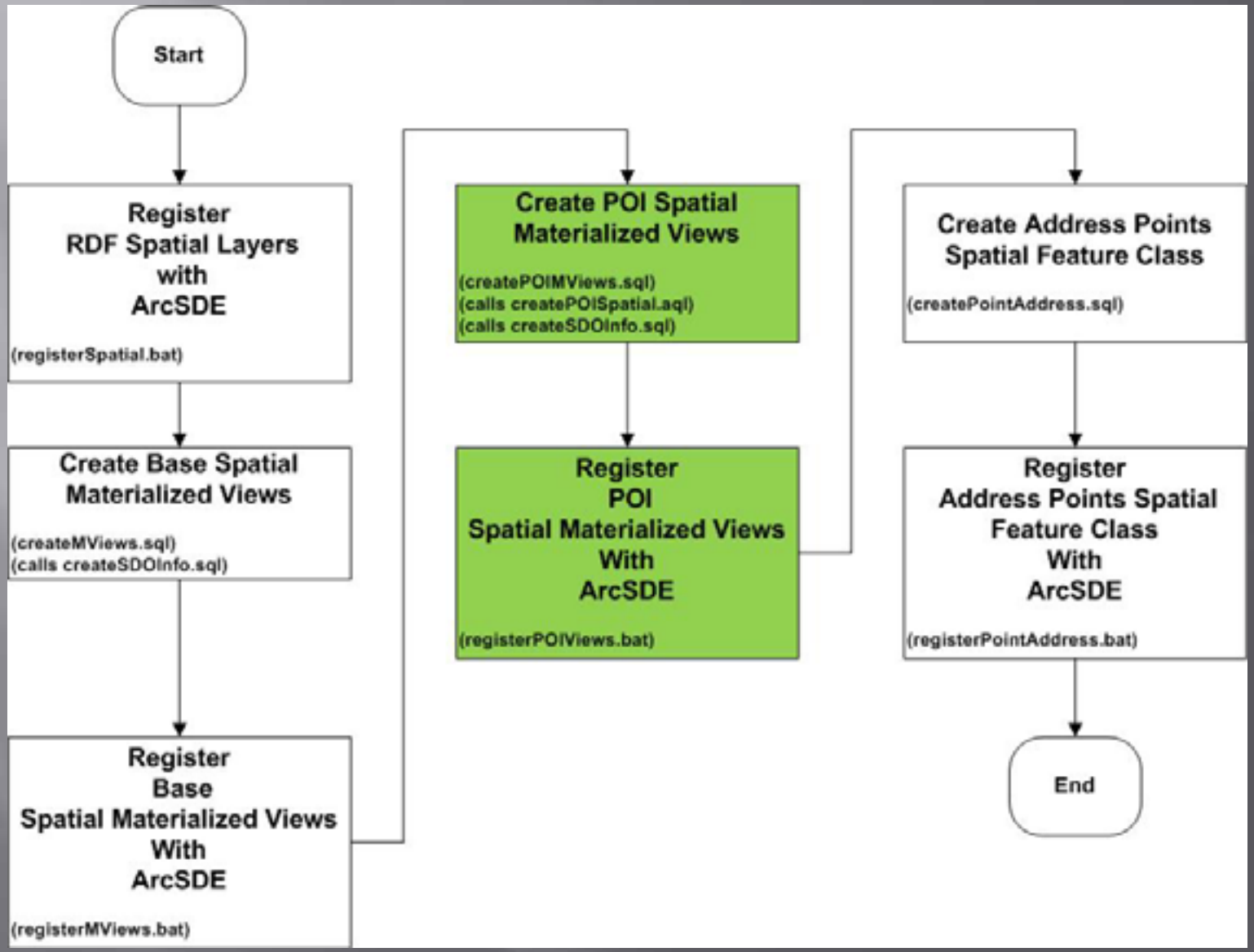
Linear (Polyline) Features

- Conceptually, all Linear features represented in the Materialized View followed the same methodology
- Attribution changed depending on the layers requirements
  - Navigable Road Links (STREETS)
    - 47 Attributes
    - Related 4 additional tables to SDO_LINK to create the desired layer.
  - State Polyline Feature
    - 5 Attributes
    - Related 2 additional tables to SDO_LINK to create the desired layer

Area (Polygonal) Features

- Conceptually, all Polygonal features represented in the Materialized View followed the same methodology
- Attribution changed depending on the layers requirements
  - Cartographic Features on some cases are unnamed
    - Separate Layers for unnamed features were created
    - Primarily for performance reasons
      - Named Cartographic Features are displayed differently in the visualization component

Step 3: Crate Points of Interest



Point Features (Conceptual View)

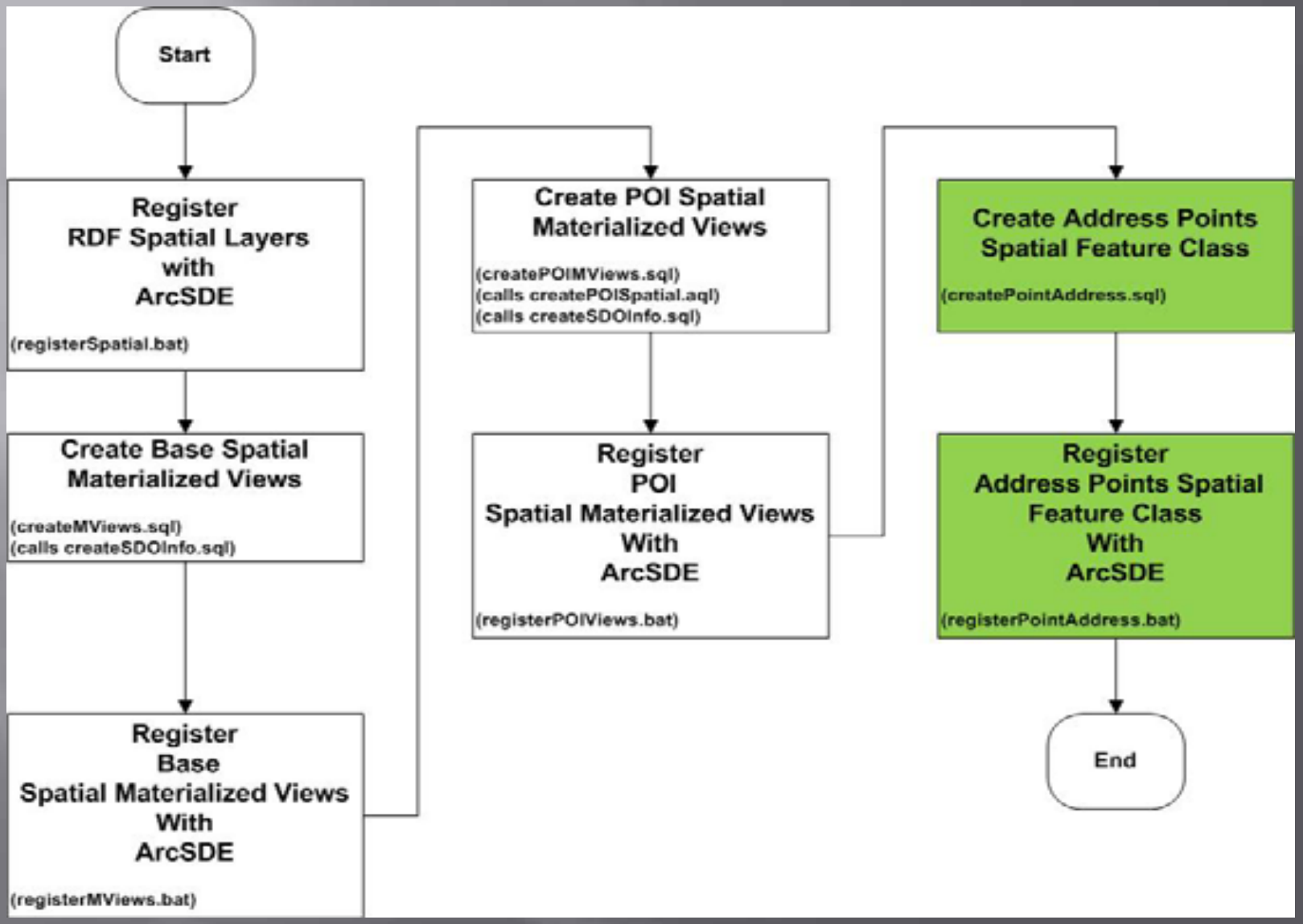
Point Features (Processing SQL Script)

```
-- Build Spatial Point Table
prompt ============ Create &1 POINT Table ==========
CREATE MATERIALIZED VIEW &1
  BUILD IMMEDIATE
  REFRESH COMPLETE ON DEMAND
AS
SELECT
  SLOC.OBJECTID,
  SLOC.LOCATION_ID,
  RPA.POI_ID,
  RDFPOI.CAT_ID,
  RFN.NAME,
  RFN.LANGUAGE_CODE,
  RPA.HOUSE_NUMBER,
  RPA.STREET_NAME,
  RPA.POSTAL_CODE,
  SLOC.LOCATION
FROM
  SDO_LOCATION SLOC,
  RDF_POI_ADDRESS RPA,
  RDF_POI RDFPOI,
  RDF_POI_NAMES RDFPN,
  RDF_POI_NAME RFN
WHERE
(SLOC.LOCATION_ID = RPA.LOCATION_ID AND RPA.POI_ID = RDFPOI.POI_ID) AND
(RPA.POI_ID = RDFPN.POI_ID AND RDFPN.NAME_ID = RFN.NAME_ID)
AND RDFPOI.CAT_ID = &2
;

-- Add Oracle Spatial Information
@createSDOInfo &1 LOCATION &3

-- Create Index on Primary Key(s)
CREATE INDEX &4 ON &1 (LOCATION_ID);
```

Point Features (SQL Script)

```
--
-- This script is a generic script for creating Points of Interest.
-- The createPOISpatial Script is called which:
--     creates the spatial table
--     adds Oracle Spatial Information
--     creates Indexes on Primary Keys
--

@createPOISpatial WINERY 2084 WINERY_LC_SIN WINERY_LID_IDX WINERY_PID_IDX
@createPOISpatial ATM 3578 ATM_LC_SIN ATM_LID_IDX ATM_PID_IDX
@createPOISpatial TRAIN_STATION 4013 TRAIN_LC_SIN TRAIN_LID_IDX TRAIN_PID_IDX
@createPOISpatial COMMUTER_RAIL_STATION 4100 COMMUTER_LC_SIN COMMUTER_LID_IDX COMMUTER_PID_IDX
@createPOISpatial BUS_STATION 4170 BUSSTATION_LC_SIN BUSSTATION_LID_IDX BUSSTATION_PID_IDX
@createPOISpatial FERRY_TERMINAL 4482 FERRY_LC_SIN FERRY_LID_IDX FERRY_PID_IDX
@createPOISpatial MARINA 4493 MARINA_LC_SIN MARINA_LID_IDX MARINA_PID_IDX
@createPOISpatial PUBLIC_SPORTS_AIRPORT 4580 PSAP_LC_SIN PSAP_LID_IDX PSAP_PID_IDX
@createPOISpatial AIRPORT 4581 AIRPORT_LC_SIN AIRPORT_LID_IDX AIRPORT_PID_IDX
@createPOISpatial BUSINESS_FACILITY 5000 BUSINESSF_LC_SIN BUSINESSF_LID_IDX BUSINESSFM_PID_IDX
@createPOISpatial GROCERY_STORE 5400 GROCERY_LC_SIN GROCERY_LID_IDX GROCERY_PID_IDX
@createPOISpatial AUTOMOBILE_DEALERSHIP 5511 AUTODL_LC_SIN AUTODL_LID_IDX AUTODL_PID_IDX
@createPOISpatial AUTOMOBILE_DEALERSHIP_USED 5512 AUTODU_LC_SIN AUTODU_LID_IDX AUTODU_PID_IDX
@createPOISpatial PETROL_GASOLINE_STATION 5540 PGASST_LC_SIN PGASST_LID_IDX PGASST_PID_IDX
@createPOISpatial MOTORCYCLE_DEALERSHIP 5571 MOTORCYCLE_LC_SIN MOTORCYCLE_LID_IDX MOTORCYCLE_PID_IDX
```

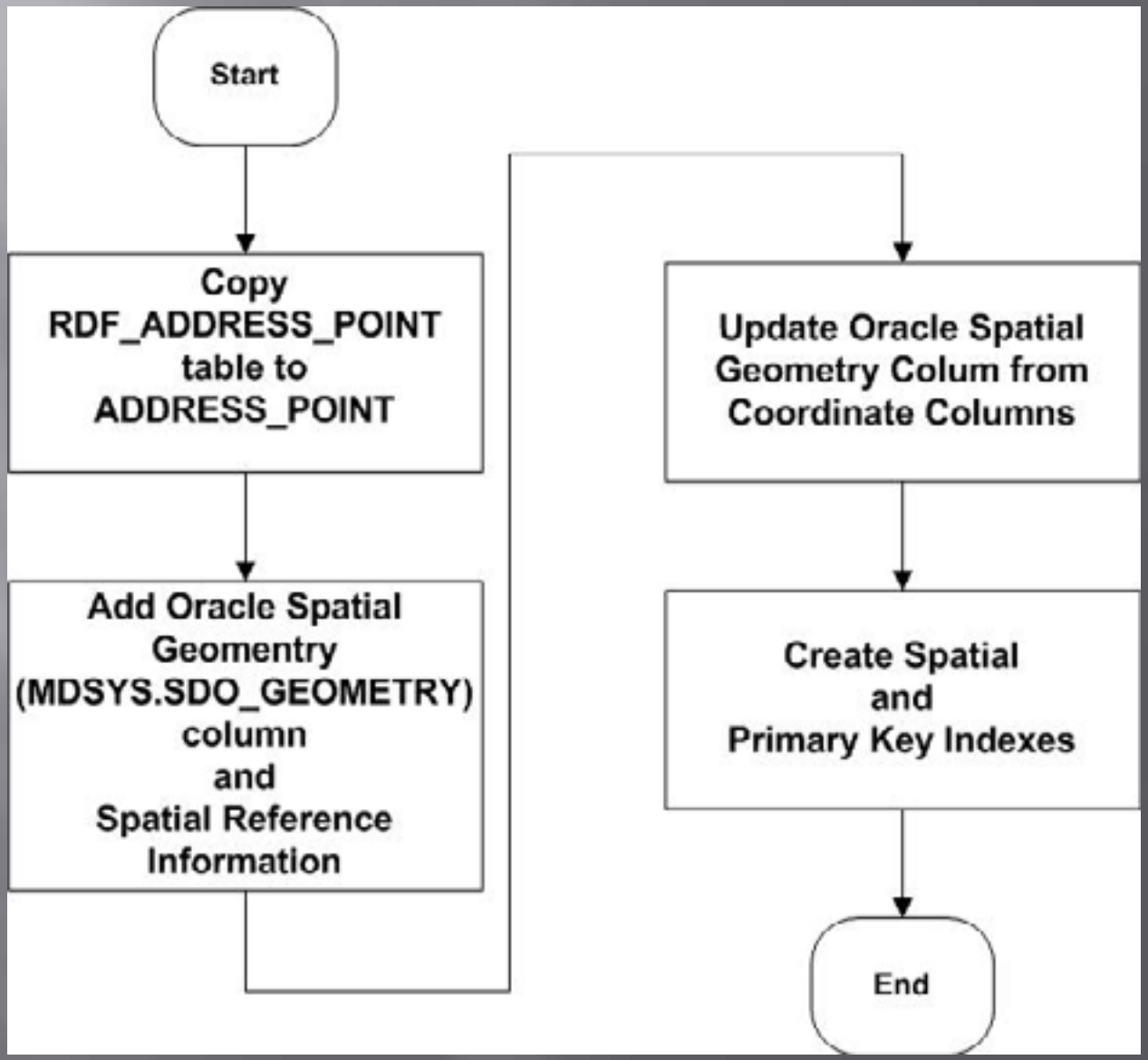
Step 4: Crate Point Addresses



Address Points (Special Case)

- This is a special case Core Table
  - RDF_ADDRESS_POINT table which contains Point Address features is currently not spatially enabled.
    - In order to display them, this table needs to be spatially enabled.
  - Two sets of coordinates are available stored in the Integer unit (With implied decimal point) which can be used to generate the components required for the Oracle Spatial Column.
    - Navigational Coordinate
    - Display Coordinate

Create Address Points (Conceptual View)



Visualization in ArcGIS Desktop

- Having completed the data preparation step outlined, the spatially enabled materialized views and feature classes are available for visualization within the ArcGIS Desktop.
- The generic ArcGIS Desktop can be used, and essentially, you are provided with a blank canvas to design or implement your own visualization.
- The sample scripts do not include an ArcGIS Desktop MAP document (.MXD) because there is no special functionality implemented.

Project Summary

- Able to successfully implement a RDF Visualization Tool using ArcGIS Technology
- Used Commercially Available Off The Shelf Software Applications
  - ArcGIS Server Enterprise Basic (ArcSDE)
  - ArcGIS Server Enterprise Advanced
  - Oracle Spatial
    - SQL and PL/SQL
- This project was viewed as a successful prototype
  - Most of the technology was re-used in other projects
  - Introduced methods for visualizing Continental Datasets
- Great learning experience for those directly involved.

Questions / Sample Scripts

- mike.tsengouras@navteq.com
- Send me an e-mail and I will forward you the ZIP file with the sample scripts