FINAL REPORT

to

THE FLORIDA DEPARTMENT OF TRANSPORTATION

STATISTICS OFFICE

on Project

"FDOT Central Data Warehouse Enhancements"

FDOT Contract BDK-75-934-05 (UF Project 00087302)

June 2011

University of Florida

Transportation Research Center

Department of Civil and Coastal Engineering

Disclaimer

SI* (MODERN METRIC) CONVERSION FACTORS

APPROXIMATE CONVERSIONS TO SI UNITS

Symbol	When You Know	Multiply By	To Find	Symbol
		LENGTH		
in	inches	25.4	millimeters	mm
ft	feet	0.305	meters	m
yd	yards	0.914	meters	m
mi	miles	1.61	kilometers	km
		AREA		
in^2	square inches	645.2	square millimeters	mm^2
ft^2	square feet	0.093	square meters	m^2
yd^2	square yard	0.836	square meters	m^2
ac	acres	0.405	hectares	ha
mi^2	square miles	2.59	square kilometers	km^2
		VOLUME		
fl oz	fluid ounces	29.57	milliliters	mL
gal	gallons	3.785	liters	L
ft^3	cubic feet	0.028	cubic meters	m^3
yd^3	cubic yards	0.765	cubic meters	m^3
		NOTE: volumes greater than 1000 L shall be shown in m^3		
		MASS		
oz	ounces	28.35	grams	g
lb	pounds	0.454	kilograms	kg
T	short tons (2000 lb)	0.907	megagrams (or "metric ton")	Mg (or "t")
		TEMPERATURE (exact degrees)		
°F	Fahrenheit	5 (F-32)/9 or (F-32)/1.8	Celsius	°C
		ILLUMINATION		
fc	foot-candles	10.76	lux	lx
fl	foot-Lamberts	3.426	candela/m^2	cd/m^2
		FORCE and PRESSURE or STRESS		
lbf	poundforce	4.45	newtons	N
lbf/in^2	poundforce per square inch	6.89	kilopascals	kPa

APPROXIMATE CONVERSIONS FROM SI UNITS

Symbol	When You Know	Multiply By	To Find	Symbol
		LENGTH		
mm	millimeters	0.039	inches	in
m	meters	3.28	feet	ft
m	meters	1.09	yards	yd
km	kilometers	0.621	miles	mi
		AREA		
mm^2	square millimeters	0.0016	square inches	in^2
m^2	square meters	10.764	square feet	ft^2
m^2	square meters	1.195	square yards	yd^2
ha	hectares	2.47	acres	ac
km^2	square kilometers	0.386	square miles	mi^2
		VOLUME		
mL	milliliters	0.034	fluid ounces	fl oz
L	liters	0.264	gallons	gal
m^3	cubic meters	35.314	cubic feet	ft^3
m^3	cubic meters	1.307	cubic yards	yd^3
		MASS		
g	grams	0.035	ounces	oz
kg	kilograms	2.202	pounds	lb
Mg (or "t")	megagrams (or "metric ton")	1.103	short tons (2000 lb)	T
		TEMPERATURE (exact degrees)		
°C	Celsius	1.8C+32	Fahrenheit	°F
		ILLUMINATION		
lx	lux	0.0929	foot-candles	fc
cd/m^2	candela/m^2	0.2919	foot-Lamberts	fl
		FORCE and PRESSURE or STRESS		
N	newtons	0.225	poundforce	lbf
kPa	kilopascals	0.145	poundforce per square inch	lbf/in^2

*SI is the symbol for the International System of Units. Appropriate rounding should be made to comply with Section 4 of ASTM E380.
(Revised March 2003)

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle FDOT Central Data Warehouse Enhancements		5. Report Date June 2011
		6. Performing Organization Code UF-TRC
7. Author(s) Scott S. Washburn, Vipul Modi, and Arun Purushan		8. Performing Organization Report No. TRC-FDOT-87302-2011
9. Performing Organization Name and Address Transportation Research Center University of Florida 512 Weil Hall / P.O. Box 116580 Gainesville, FL 32611-6580		10. Work Unit No. (TRAIS)
		11. Contract or Grant No. FDOT Contract BDK-75-934-05
12. Sponsoring Agency Name and Address Florida Department of Transportation 605 Suwannee St. MS 30 Tallahassee, Florida 32399 (850) 414 – 4615		13. Type of Report and Period Covered Final Report
		14. Sponsoring Agency Code
15. Supplementary Notes		

16. Abstract

The Statewide Transportation Engineering Warehouse Archived Regional Data (STEWARD) database was developed to organize the raw traffic data collected by the traffic management centers (TMC's) in the metropolitan areas of Florida with the help of the SunGuide system. The database processes the raw data and converts it into useful reports for various users of the system. However, as the TMC's continue to expand their network detector coverage, the amount of data transferred to STEWARD continues to increase. This led to a significant decrease in the performance of the database system, and required the system to be taken offline for extended periods to update the system. This project was undertaken to enhance the software and hardware aspects of the STEWARD system in an attempt to improve the system availability, data availability, and the system performance through the following actions: migrating the database server to a very high performance computer, developing automatic utilities to replace the manual activities used for daily maintenance tasks, reviewing the database schemas/queries for increased system performance, and improving the lane-configurations and interactive maps of the STEWARD system. The documentation for the installation of the STEWARD system was also updated and improved.

17. Key Words	18. Distribution Statement
STEWARD, Traffic database, Central Data Warehouse	No restrictions. This document is available to the public through the National Technical Information Service, Springfield, VA, 22161

19 Security Classif. (of this report)	20.Security Classif. (of this page)	21.No. of Pages	22 Price
Unclassified	Unclassified	156	

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

Acknowledgments

The authors would like to express their sincere appreciation to Mr. Richard Reel of the Florida Department of Transportation (Central Office) for the support and guidance he provided on this project.

Table of Contents

Table of Figures

1 Introduction

The traffic management centers (TMC's) in the metropolitan areas of Florida collect data from their roadway detectors through the SunGuide system that is designed specifically for the Florida Department of Transportation (FDOT). The SunGuide system creates a daily text file containing the basic data produced by each of its detectors at a specific time interval. In a preceding FDOT-sponsored project[1], the Statewide Transportation Engineering Warehouse for Archived Regional Data (STEWARD) was developed, which processes the raw data obtained from the TMC's, organizes the processed data in an Oracle database system, creates a number of useful traffic information and traffic data diagnostic (based on traffic flow principles) reports, and provides a web interface for data users to access and retrieve the traffic data and reports. More specifically, the different processes that are performed under STEWARD are: the daily retrieval and assimilation of the raw data from the SunGuide centers to STEWARD, the conversion of the raw data into various formats, the application of a quality assurance method to identify invalid data, and posting the processed data on the STEWARD web site for general access.

The STEWARD system provides a mechanism for researchers, practitioners, and others to obtain high-resolution traffic data from Florida's busiest roadways for a variety of purposes. Additionally, customized reports are available on the web site for each of the SunGuide TMCs. These reports are available upon selection of several stations, time periods, and data time intervals. The different reports that can be created are: Section level reports (performance measures, travel time reliability), Station level reports (all data fields, traffic counts, maximum flow rates, effective vehicle length), and Facility level reports (all data fields, volume map and I/O balance, traffic counts). These reports can be downloaded to a personal computer as well as be viewed on screen.

Apart from the traffic data and reports, the system's web site also provides several desktop utility programs that process and create special analysis reports from data and reports that summarize the analysis activities. To facilitate efficient identification of detector locations of interest, the web site displays the detector locations in a geographic format with a satellite photo overlay. Additionally, the STEWARD provides the following functions:

- Identification of detector malfunctions
- Calibration guidance for detectors
- Quality assessment and data reliability tests
- Daily performance measures
- Support for periodic reporting requirements

[1] FDOT BDK545-37, "Development of a Central Data Warehouse for Statewide ITS and Transportation Data Phase II: Proof of Concept"

- Extraction of traffic counts as an input to the FDOT central and district office traffic counting programs
- Support for the analysis of traffic volume trends

The diagnostic reports that were furnished to the TMC managers in the past have proven to be valuable in the maintenance of their detector systems. These reports and the quality assurance procedures have provided the TMC managers the information they need to maintain their detector systems such that the completeness and validity of their data is on a par with, and sometimes exceeds, the corresponding measures in other systems throughout the nation. STEWARD has also created an important resource for a wide variety of traffic data users in Florida, including both practitioners and researchers. STEWARD also provides data for research and special studies. Areas in which the data have already been used by researchers include: analyzing speed-flow-density relationships, analyzing the effects of an incident, analyzing managed lanes and travel time reliability. There are several projects and activities that have already benefited from the available data. The remainder of this chapter discusses the general architecture of the STEWARD system, the emerging issues with the previous STEWARD system, and the proposed actions that are performed to enhance the system.

1.1 General Architecture of STEWARD

In general, the STEWARD system consists of three main elements, the ETL (Extraction, Transfer and Loading) process, the database (DB), and the web user interface which is developed using a variety of tools to design, deploy, and maintain the system efficiently. Prior to the start of this project, the Windows Server 2003 and Microsoft Internet Information Services were selected as the operating system and web server, respectively and Oracle database platform was selected per FDOT's recommendation. The Oracle Warehouse Builder 10g2, Oracle Enterprise Manager, and ASP/JavaScript were selected for the integrated ETL processes, the database management, and the web development.

The STEWARD configuration has the FTP (File Transfer Protocol) server in its front end. The system receives compressed (zipped) archive data from each SunGuide TMC every day and the archived files are loaded onto the FTP server. Different connections are created for different FDOT districts. The ETL utility is used to process the raw data. The STEWARD DB server retrieves and loads these data files into the STEWARD database, and then creates the custom reports.

1.2 Emerging Issues with Current STEWARD

The primary issues facing the ongoing operation and maintenance of STEWARD are related to the quantity of archived data, the processing time of the data, and the number of users of the data. The STEWARD system receives approximately 1 GB of data every day. As the TMC's continue to expand their network detector coverage, the amount of data that must be transferred

to STEWARD and then processed and loaded into the database will continue to increase. This increases the processing time of the data and the time it takes to update the database. This also increases the data storage needs of the system. Adding to the data storage need is of course the fact that with each passing day, one more day of data storage is added to the system. If other FDOT districts eventually start contributing data to the system, this will also add to the data retrieval, processing, and storage needs of the system. Prior to the start of this project, STEWARD hardware setup was nearing its practical limit on data storage capacity.

Also prior to the start of this project, the six steps involved from receiving the data to uploading the data on the web server were performed in a mostly manual manner. As the amount of daily data received increases, the overall time it takes to ultimately get this data on the web server increases. The system was just about to the point that the amount of processing time (including integrating the data into the Oracle database) is equal to the time period of the data. In other words, putting 24 hours of new data into the system takes nearly 24 hours. During this process, the web server must be taken off-line. This is obviously a highly undesirable situation, as it essentially precludes the ability have the system on-line for a large percentage of the day and have up-to-date data content. One of the major causes of that situation was due to the database server and the web server being run from the same computer.

The STEWARD web site has thus far had a modest amount of user activity, but continues to increase at a steady rate. It is anticipated that activity levels will increase even more as more data become available and awareness of STEWARD increases as it transitions from the research and development phase to the operational phase. As the user activity increases, more demands on placed on the STEWARD system. Additionally, as the user activity increases, it will be harder “to get away with” having the web server down for large periods of time while data are being added to the system.

1.3 Research Approach

The objective of this project was to enhance both the hardware and software aspects of the STEWARD system in order to improve system availability, data availability, and system performance. The tasks that were undertaken to accomplish this objective are described briefly as follows:

- Purchase and setup new hardware for the STEWARD system
- Critical database review of the STEWARD system for performance improvement
- Develop utilities to automate the daily operations of STEWARD
- Improve the documentation for the lane configurations and detector information
- Update the facility information and consequently add new districts to the database
- Update the documentation of hardware and software installation for STEWARD

1.4 Organization of Report

Chapter 1 provides an overview of the previous STEWARD system, the bottlenecks encountered in the STEWARD system, and the research approach from the STEWARD operators. Chapter 2 provides a detailed description of the various activities that were undertaken during this project. Chapter 3 provides a brief summary of the activities performed and the potential activities to be undertaken in future. Appendix A provides detailed information on the software and hardware aspects of STEWARD and its daily operations and appendix B provides an example of one of the scripts used in STEWARD's daily operations.

2 Activities Performed

This chapter describes the activities that were undertaken during this project. Some of the activities associated with these tasks are also described in further detail in Appendix A.

2.1 Purchase and Setup New Hardware for STEWARD

The major performance bottleneck for the previous STEWARD[1] system, hereafter referred to as 'previous server', was due to the presence of the web/database server on a desktop computer and the FTP (File Transfer Protocol) server on a slower desktop computer. As the previous server was installed on slower desktop computers, it appeared that the system performance decreased significantly as more data was added onto the database, and it took longer time to finish the daily operations of STEWARD. To address this issue, the University of Florida Transportation Research Center (TRC) bought several hardware components to replace the previous server and re-developed the previous server to the current STEWARD system (hereafter, referred as just 'new server'). The objective behind the re-development of the STEWARD system was to make it faster, more stable, and sustainable.

The next section of this chapter gives an overview of the previous server, the new hardware purchases, the new server setup, a summary of the daily STEWARD operations, and the Web-interface features of the new server. A more detailed description of the STEWARD daily operations is presented in Appendix A of this report.

2.1.1 Previous STEWARD System Overview

The previous server was developed from primarily three components: the FTP server, the database server, and the web server that hosts the web-interface. The FTP server and the database server were built on Windows XP desktop computers with 32-bit processor operating systems. To support and host the web server, Microsoft Internet Information Services 5.0 was installed on the database server. The Oracle database 10gR2 version was chosen as the platform for the database management whereas the ASP/JavaScript was used for developing the web-interface. The previous server had the FTP server in its front end where the raw traffic data were received from each district, and were archived. The archived data were then retrieved, processed and loaded onto the database server, and the customary reports were created. The processed data and the reports were then made available to the outside users via a web-based interface.

As the STEWARD system expanded its database network across several FDOT districts, the number of users accessing the web-interface increased, and the performance of the system was decreased considerably. With the slower desktop computers, the previous server was unable to meet the expectations of the Internet users and this motivated the STEWARD operators to transfer the STEWARD system to a higher performance computer. To configure the STEWARD system on a new computer, several hardware purchases were made. The description of the new hardware purchases are given below.

2.1.2 Hardware for New STEWARD System

In an attempt to make STEWARD a stable system and to meet the expectations of its users, several new computer hardware components were purchased. This included a high professional-grade server, Dell T610 – Windows Server 2008 with Intel Xeon Processor (64-bit) and 8 GB RAM (Random Access Memory) for hosting the web/database server of the new STEWARD system. The motivation behind purchasing a high-end server was to improve the STEWARD system's performance and for performing multiple tasks on it simultaneously while the daily operations were executed. This meant that the web-users could download the traffic data and the customized reports much faster as compared to the previous server and could also help the STEWARD operators to avoid the shutting off the system temporarily for data loading purposes and enable them to keep the web-server online for a longer period of time.

The computer that previously hosted the FTP server was also upgraded with a higher performance processor. The operating system of this computer was chosen as the Windows XP desktop computer (32-bit) operating system with a 3 GB RAM memory. The internal hard drive's storage capacity of the old computer was also replaced by two new 2 TB (terabytes) SATA hard-drives. The motivation behind this purchase was to store the historical raw data and processed data (from 2007), and to accommodate the daily incoming raw traffic data and the processed data. As the previous server's hard-disk was running out of space, this purchase was necessary. After the new computer and other hardware purchases were made[2], the new STEWARD system was installed and configured.

2.1.3 New STEWARD Setup

The first step in setting up the STEWARD system was to install the Oracle Database 10g Release 2 version (Source: Oracle Corporation, available for free over Internet) on the professional-grade server. The Oracle 10g version for the Windows Server 2008 (64-bit) operating system was downloaded and installed. The Oracle database companion software and the Oracle database client software were also installed on the new server. The Oracle workflow, Oracle Warehouse Builder (OWB), Oracle Database Configuration Assistant, Oracle Net Configuration Assistant and the Oracle Enterprise Manager were also configured on the new server. The installation process and the configuration of all these components and packages are explained in details in Appendix A of this report. The instructions are presented as a step-by-step process with screen captures displayed to describe each step.

The next step in configuring the STEWARD system was to design and create a new database through the OWB. The OWB includes the pre-defined rules that are generally required in the warehouse designing. The database manager of the OWB was then used to create several types of tables and functions like: the external tables, the dimension tables, the cubes or the fact tables

[2]This project money was not used for these purchases.

and the materialized views. These tables in general store the daily traffic data and other reports that are created in STEWARD. The detail description on how the tables and functions were created and deployed is also provided in Appendix A of this report.

After the Oracle related components were configured in the system, the Microsoft Internet Information Services 7.0 was installed on the new computer for establishing a connection between the Oracle database and the web users. The steps involved in this procedure are also included in the Appendix A of this report. After these components were installed, the daily operations of the STEWARD were configured. A brief overview of the daily STEWARD operations and the functions of the different components are provided in the next portion of this section.

2.1.4 STEWARD Operations

The architecture of the new STEWARD system was designed in a similar fashion as the previous server. The new STEWARD system is operated by integrating three components which are: the FTP server, the external hard disk, and the web/database server. To ensure smooth operations of the STEWARD system, a systematic procedure is planned to bind all the three components and all the sub-tasks within each component. The different functions that are performed by each of these three components and the channels through which they are connected with each other are explained in details as below:

FTP Server

The first two steps in the daily operations of the STEWARD system starts with the ETL procedure, i.e., the extraction and the transformation of the raw data. First, the SunGuide TMC's transfer the compressed (zipped) archived data files to the FTP server in its raw format. This communication between the TMC's and the FTP server is configured through a software "File Zilla Server" (available for free). This software is installed and configured on the FTP server to receive the daily archived data files from different districts through their unique login credentials. The raw data that every district transfers are stored in their respective local folders on the FTP server.

Second, the raw data files are then accessed by an automatic utility (refer Section 2.3 of this chapter) that transforms/convert the raw data into the processed data format, i.e., into aggregated 5-, 15-, and 60-minute interval station and lane data. This utility then organizes the processed data in a local folder on the FTP server on the basis of the data's aggregation level. For example, all the station level 5-minute aggregated data files are transferred to a folder named as 'Station 5-minute' and all the lane level 5-minute aggregated data files to a folder named as 'Lane 5-minute'. Similarly, all the other aggregated data files are transferred and organized accordingly. This local folder is referred as "master folder" in this report, hereafter. This automatic utility runs on the FTP server with the help of the Windows Scheduler. During the data

transformation/conversion, a backup of all the raw data files and the processed data are taken on the FTP server. After the data backup, the master folder (that contains the organized data) is copied to the external hard disk. This process is executed through the "SynBack" software that is installed on the FTP server.

External Hard Disk

An external hard disk is directly plugged into the FTP server and is primarily used to transfer the master folder from the FTP server to the web/database server. The external hard disk is connected remotely to the database server and transfers the master folder through the "SynBack" software scheduler.

Web/Database Server

The web/database server is the computer that hosts the Oracle database and the STEWARD's web-interface. As this is a high performance computer, the operations of data uploading and web-hosting are performed simultaneously unlike the previous server. This computer is also used for transferring the master folder data files from the external hard disk to the database server using the "SynBack" software.

After the master folder is transferred, the last step in the ETL procedure, i.e. the data uploading onto the database is completed. The master folder is kept at a local directory location on the database server and the aggregated data files within its sub-folders are uploaded to the database through the Oracle's OWB. The data uploading is performed individually for each data file type like the, 5-, 15-, and 60-minutes station data and the 5-, 15-, and 60-minutes lane data. The data uploading are scheduled to run at the database server at different time periods in order to avoid overlapping of jobs timeline. After each data file type are uploaded to the database, the materialized views or the customary reports are created through the OWB. After the reports are created, the STEWARD system is updated with the most recent data and is available to be used by users via the web-based server. These operations are briefly described in Section 2.3.4 and in details in Appendix A of this report. The Appendix A is developed for personnel who install, operate and maintain the system. A schematic diagram of the operations of these components is provided in Figure 1.

A summary of the daily STEWARD operations as shown in Figure 1 are provided below:

- Data transfer from TMC's SunGuide
- Data backup and transformation on the FTP server
- Data transfer from FTP server to Database/Web server through the External Hard Disk
- Data uploading to the database in the Database/Web server
- Creation of customary reports in the Database/Web server
- STEWARD's web-based interface access in the Database/Web server

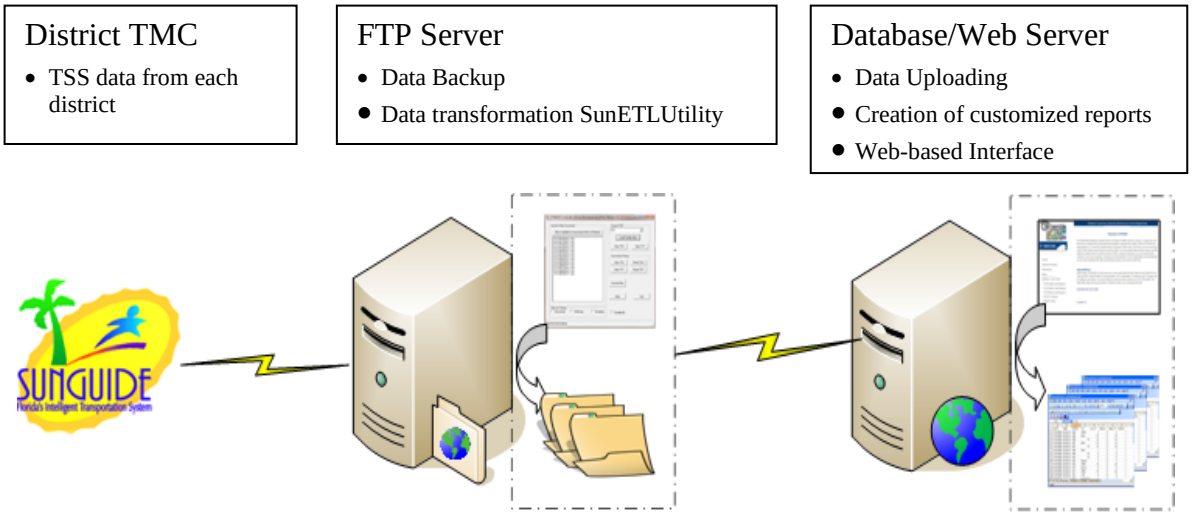


Figure 1. Schematic diagram of STEWARD's daily operations

2.1.5 Web Features of new STEWARD

The web architecture of STEWARD was not changed from the previous server. However, a few sections from the previous server were modified during the installation of the new server. A summary of all these modifications and the general features of the web-interface are provided next in this chapter. This web-interface is developed for an audience that wishes to view the user-specific data on screen and download them on their personal computer in a comma-delimited (CSV) format. The web address where the new STEWARD system could be accessed is:

http://cce-trc-cdwserv.ce.ufl.edu/steward/index.html

To host the web-interface of the STEWARD system on Internet, the Microsoft Internet Information Services (IIS 7.0) was installed on the database server. The Oracle's net assistant configuration tool was used to establish the communication between the ASP/Javascript of the web-interface with the Oracle database. These configurations along with the IIS setup are described in details in Appendix A of this report.

STEWARD in general, has four main categories on the homepage of its website. These are: the overview, resources, maps, and reports sections. During the configuration of the STEWARD's website on the new server, all these categories were edited with the latest information and were modified accordingly. A brief description of all these edits is provided next.

Overview: This section gives a brief introduction on STEWARD's development and other details. The content under this section of the webpage was edited and modified as required.

Resources: This section provides the web-users an access to STEWARD's user manuals, desktop utilities and facility files for the FDOT districts. The facility files located on this section are replaced with the most recent updated files. The SunETLUtility[3] utility (used for converting the raw data into aggregated/processed format) is also replaced with the most recent updated version. All the other sub-sections including the utilities and presentation materials were not modified.

Maps: An interactive map is provided for STEWARD users to access the information on the detector locations in every district. All the detector locations are superimposed on the Google Maps (Source: Google). These maps were updated with the most recent information from the respective facility files. Detailed information on the modified maps is provided in Section 2.4 of this chapter.

Reports: This section is the most important part of the STEWARD system. This provides the users an access to retrieve the daily traffic data in the form of customary reports. This includes the facility-, section-, and station-level reports. The users can retrieve these reports after specifying certain selection criterions like the district, the facility or the roadway within that district, roadway direction, date and time range, day or week or combination of days, and the desired aggregation level (5-, 15- and 60-minutes). After these reports are created, the users can download them into their personal computers in a comma-delimited (CSV) format or they can view it directly on screen. Based on the requirements from the FDOT Statistic Office, only few of these reports are generated and created at present on the new server. The status of the availability of these reports is:

TSS-Station Level Reports: *(available)*

- All Data Fields
- Traffic Counts
- Maximum Flow Rates*
- Effective Vehicle Lengths*

(*not created on a regular basis)

TSS-Facility Level Reports: *(not available)*

- All Data Fields
- Volume Map and I/O Balance
- Traffic Counts

TSS-Section Level Reports: *(not available)*

- Performance Measures
- Travel Time Reliability

[3] FDOT BDK545-37, "Development of a Central Data Warehouse for Statewide ITS and Transportation Data Phase II: Proof of Concept"

It is expected that with the support from FDOT in future, other reports will also be created and provided periodically.

2.2 Critical Review of STEWARD Database Structure

As the new server's architecture and database instruments were kept the same as present in the previous server, a critical review of the database structure, data organization, and query scripting was performed to improve the STEWARD's database performance and response time to queries. The enhancements to these aspects of the database were tested on the previous server and were implemented on the new server. The following section describes the changes that enhanced the STEWARD system's performance.

2.2.1 Database Schema Review

The database schemas in the previous server were reviewed in details for optimizing the performance of the system. The indexing of the database tables for optimum and effective performance of the queries were also checked. These tables were found to be indexed as required and in a normalized form. The partitioning of the database tables was also extensively investigated. It was found that all the database tables that stored large data were partitioned appropriately in the current schema. For all the tables that had only a limited amount of data were found to be un-partitioned. It was further investigated that the partitioning of the smaller tables would not provide any significant performance increase and hence, the partitioning of the smaller tables was not done and are kept un-partitioned in the new server.

The queries associated with the STEWARD website for data retrieval/extraction were also reviewed. These queries were investigated, analyzed, and tested for performance improvement. It was found that in some of the queries, a stored procedure, PL/SQL (Procedural Language/Structured Query Language) function, was used to fetch the respective data that used redundant loops for the data extraction. To avoid the use of redundant loops, these procedures were modified by replacing the slower 'switch-case' constructs with faster decoding options wherever possible. With the faster decode option, an improvement in the system procedure for extracting data was observed.

Another area for improvement was identified in the STEWARD database that addressed the issue of size and space of the database tables. It is evident that with the increasing amount of data in the STEWARD database, there is a definite need to monitor the increasing size of the database tables. It was observed that the tables used to store the database would run out of space every two to three weeks. Due to this shortage of space, the data uploading procedures on the system were halted on a regular basis. This was resolved through the Oracle Enterprise Manager tool by spurring the automatic increment of the space/size of the database tables. In the new server, these tables automatically accommodate the additional space requirements and hence, do not affect the system performance. However, after further investigation, it came to the knowledge of the STEWARD operators that there is still a limit at which the automated

increment of the database tables stops. This happens when the size of the database tables reach 33 GigaBytes (GB). Since these tables are a critical part of the STEWARD system, it is necessary to monitor them on a regular (approximately monthly) basis. The monitoring of such tasks will be a part of the maintenance of the STEWARD database in the future. It is expected that further research will be performed by the STEWARD operators that will look into this matter more extensively.

2.2.2 Materialized Views Review

The database schema in the STEWARD uses materialized views (customized reports) to retrieve the reports requested by the users at the web-server end. These materialized views in the database system are the virtual tables that represent the results of a database query (according to the user selected criterions). In general, a materialized view takes a different approach where the query result is cached as a concrete table that may be updated from the original base tables from time to time. This makes the data retrieving an efficient process in the data warehousing scenarios, which on the other hand is an extremely time consuming process with frequent queries from the actual base tables.

In STEWARD, the reports (such as "Traffic Counts", "Effective Detector Length", "Performance Measures", etc.) are associated with these materialized views in order to minimize the report creation time. It also means that the time taken to retrieve/fetch these reports from the web-interface decreases as it improves the query response time. To make these materialized views or the customized reports available on the database, the tables associated with it are required to be updated after the daily data is added to the database. It was found that these materialized views or data tables could be updated with two inbuilt features of Oracle database: "refresh-on-demand" and "refresh-on-commit". The "refresh-on-demand" option is used to update customized reports when users wish to create such reports, whereas the "refresh-on-commit" option updates the customized reports automatically after data is added without the users' wish. In the previous server, all the materialized views were updated using the refresh-on-demand method. Both the options were explored and tested by the researchers or STEWARD operators in order to achieve optimal system performance. However, the researchers selected the refresh-on-method for updating the materialized views on the basis of the following two reasons.

First, in the refresh-on-commit option, the materialized views are updated whenever their base tables or daily basic tables are updated. This meant that the data loading of the base tables and the materialized views are updated at the same time. This process was found to take more time as compared to the refresh-on-demand method. Also, with the refresh-on-commit method, it was difficult to estimate the execution time of the data loading and updating the materialized views processes. Second, with the refresh-on-commit option, the materialized views were not refreshed or updated if the code included the aggregated functions like sum or average. As these functions required the reparative calculations for each data insertion, it took longer time for this

process to get executed. For example, one of the materialized views, "Performance Measures" uses sum functions and hence, the refresh-on-commit could not be used for updating the views.

The next step in configuring the updating process of these materialized views was to specify the type of "refresh" method used for each particular materialized view. The different refresh methods that are available to be used in the OWB Design Center are FAST, COMPLETE, FORCE or NEVER. The details of these options as per the Oracle guidelines are described below:

- FAST: It is used to enable the incremental refresh method. It means that the materialized views will be updated only for the newly added data to the base tables. This method requires creation of log files for all the materialized views and is the fastest method. However, this method cannot be used to update materialized views with aggregated functions.
- COMPLETE: Used to enable the complete refresh method. It means that the materialized views will be updated for the entire data set (including all the previous day's data) on the base tables. This method is not an efficient way to refresh a materialized view as it takes longer time to execute but this method can be used for updating materialized views that includes aggregate functions.
- FORCE: It is used to indicate that when a refresh occurs, the Oracle database will try to perform a FAST refresh first if possible, or else a COMPLETE refresh if FAST is not possible. FORCE is a default option that Oracle sets for any materialize view.
- NEVER: Used to prevent the materialized view from being refreshed with any Oracle database refresh mechanism or packaged procedure.

The previous server used the FORCE option to update all the materialized views. Since, the materialized view logs were not created (required for FAST option), the incremental refresh of the materialized views did not occur in the previous server. As mentioned above, the complete refresh method was performed all the time, which meant that whenever there were changes in the base tables like the addition of new data, the corresponding materialized views were re-built entirely from scratch. This refresh method decreased the system's performance and the database had to be kept off-line during this process. However, the researchers found a scope of significant improvement for the database performance by using and implementing the "FAST" refresh option that updated the respective materialized views with the incremental approach.

To achieve and implement the FAST refresh option, the materialized view logs were created for all the materialized views. The materialized view logs helped in refreshing the views on an incremental basis. These logs maintain the details of the last update, and the historical reports of the views. Also, the structure of the materialized views was modified in order to meet the requirements for FAST refreshing option. This included the insertion of additional "ROWID" columns in all the materialized views tables and to each of their base tables. These columns were exclusively added for the incremental refreshing of the materialized views. The additional

columns definitely increased the size of the materialized views but it provided a significant improvement in the refreshing time of these tables and the reports. The customized reports that were initially created/refreshed in about 30 to 60 minutes were found to be completed in less than two-minutes after the changes were implemented.

The FAST option is implemented in two ways for a materialized view. The first method is by changing the parameters of a respective materialized view during its deployment in the Control Center of the OWB, or by writing and implementing a PL/SQL code that is executed through the SQL plus or any other data query software. The STEWARD operators implemented the FAST option through the second method by writing a PL/SQL code in the new server. An example of this code is provided in Appendix B of this report. As the 'FAST' refresh option is implemented for all the materialized views through the PL/SQL code, the refresh option during the parameter settings of the materialized views were/are kept as 'FORCE' only.

After further review of the materialized views, it was found that the materialized views were partially partitioned. In the new server, the tables for the materialized views are properly partitioned with date and district. The years/dates are also categorized by each quarter from 2007 Q3 to 2013 Q4 and the districts are categorized by District 1 to 7 and the Turnpike. Also the materialized views are indexed with station ID, hour and date to improve the performance. A bitmap index is also used for this indexing.

2.3 Develop Automated Processes for the Database Update Functions

Another major bottleneck for the daily previous server's operations was that several activities for the data processing and the data uploading were executed manually. Four activities were identified that were performed manually: first, transferring the raw data from the FTP server to the SunETLUtility application for data processing; second, transferring the processed data from the FTP server to the database server; third, uploading the 5-, 15-, and 60-min station/lane data files to the Oracle database; and fourth, updating the customized reports or materialized views in the Oracle database manually.

During the period when these activities were executed manually, the web-server was temporarily shut down. The manual activities took a considerable amount of time and the STEWARD operators were unable to provide the data to the users on a regular basis. To make the data available to the users on a regular basis, the manual activities were performed by STEWARD operators frequently. This was obviously a highly undesirable situation, as it limits the ability of the STEWARD web-server to remain on-line for a large percentage of the day and meet the users demand. Thus, with an aim to keep the web-server online throughout the day and to provide the data to the users on a regular basis, it was necessary to develop an application/utility or plan a systematic procedure that would substitute the manual work. It was also felt that the creation and development of this utility will decrease the lag time of data availability to STEWARD users.

2.3.1 Development of Automated Data Uploading Utility

To address the issues mentioned above and to automate the STEWARD's daily operations, a systematic and timely procedure was planned that substituted for the manual work. This led to the development of an automated utility that transformed the raw data from the FTP server and uploaded the data into the database server directly. Other settings were also configured in the Oracle database and the FTP server to make STEWARD self-sustainable. All the components that are involved in the automated process of data uploading are described briefly in the next section. The technical details of these procedures are described in Appendix A of this report.

2.3.2 Daily Processing of TSS data

The first step to automate the STEWARD's operations was to build a utility program that could extract the raw data and transform the raw data into the aggregated/processed data format automatically without the need of having an operator monitoring them. This utility (referred as the automatic utility in this report) program was written in the C# programming language in the .NET environment such that it is compatible with the FTP server's 32-bit operating system. This utility executes the first two steps of the ETL procedure, i.e., the data extraction and transformation of the raw data. This utility is run on the same computer that hosts the FTP server for it to access the archived files easily.

In the previous server, the raw data for a district was processed with the help of the SunETLUtility utility or application. This utility transformed the raw data into the aggregated 5-, 15-, and 60-minutes station/lane data in a pre-defined format. This utility required the operators to enter specific district related inputs for executing data transformation for every district separately. In a long run, this approach became tedious for the operators as the data transformation had to be monitored manually for every district. With this in mind, the operators created the automatic utility in such a way that it enters the district specific inputs for the SunETLUtility on its own and transform the data for all the districts at one go. The SunETLUtility, from the previous server, was embedded into this automatic utility code. This enabled the data transformation of the raw data into the processed data format for all the districts at one go without the need of any operator to monitor the progress. The final code of this utility was optimized after multiple trials of data transformation. This utility was named as the "Steward Data Import" and it is run on the FTP server. A snapshot of this utility is provided in Figure 2. There are two main features of this utility and these are represented in the form of two tabs as shown in its GUI (Graphical User Interface).

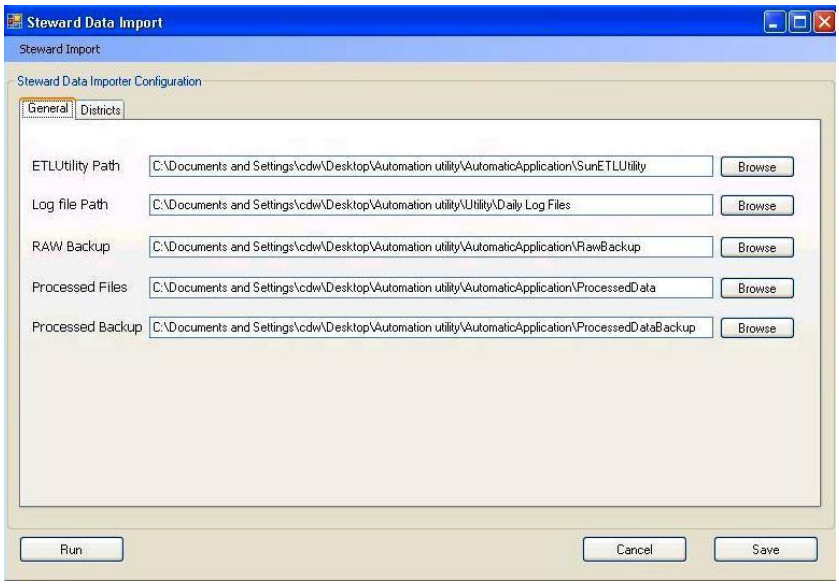


Figure 2. GUI for first tab of the automatic utility

First, the 'General' tab that asks the users to specify the paths for different folders on the FTP server where the utility is running. The following fields are expected to be entered to run this utility. These fields are:

ETLUtility Path: The folder location where the SunETLUtility is placed on the local directory or the FTP server. It is specified so that the automatic utility can call the SunETLUtility for data transformation.

Log File Path: During the data transformation process, a text (.txt) file is created that summarizes the data processing activities of each district. The folder location where the daily files are expected to be stored is specified here. These files are created every day and it helps the STEWARD operators in monitoring the status of data processing. For e.g., if a data file is not sent to STEWARD from a particular TMC, this gets reported in this log file. This also helps the STEWARD operators to keep a check on any missing data.

RAW Backup: This path specifies the location where the raw data, required for the data processing, are backed up.

Processed Files: After the data are processed by the automatic utility, the transformed/aggregated data are organized into the master folder (Section 2.1). The location where the master folder is kept is specified at this area. Also, the folder that is located here is copied to the external hard disk during the data uploading process.

Processed Files Backup: This path specifies the location where the processed data are expected to be backed up.

Second, the 'Districts' tab that ask the users to specify the paths of the folders where the raw data for all districts are located on the FTP server. In general, these folder locations are same where the raw data are received from the TMC's during the archived data transfer process. These paths enable the automatic utility to read and extract the raw data during the data transformation for each district. Since, the raw/archived data are received in a zipped format, it is essential that this utility also unzip the raw data. To accommodate this, the "7 Zip" software program is embedded with the new utility and is included in the utility package. A snapshot of this tab is provided in Figure 3. After all the fields are provided, the utility is executed by clicking the "Run" button.

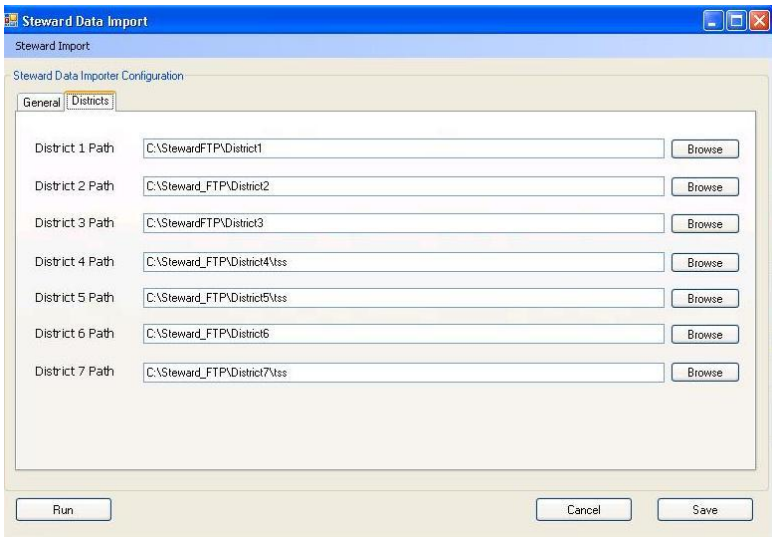


Figure 3. GUI for second tab of the automatic utility

For this utility to be a part of the automated process, it is run after the raw data from all the districts are transferred to the FTP server. In the new STEWARD system, the last file is received at 12 noon from District 5 every day. So this utility is scheduled to run every day at 2:00 PM with the help of the Windows Scheduler. It should be noted that since this utility runs on the FTP server during the day time, it does not affect the STEWARD system or the database server's performance in anyway. This utility, on an average runs for three to four hours every day for data processing and to complete the data transformation. As a result, the next activity of the automated process is scheduled to run only at 7:00 PM, which is the data transfer from FTP server to the external hard disk. The next portion of the automated process is described as follows.

2.3.3 Daily data uploading to STEWARD database

After the data are processed on the FTP server through the automatic utility, the processed data files are organized under the master folder. This master folder is then copied to an external hard disk with the help of the "SynBack" software. This operation is scheduled to occur at the FTP server machine at 7:00 PM every day. The next step in the automated process is the transfer of

the master folder from the external hard disk to the database server computer. This is performed through a remote connection at 7:30 PM every day with the help of the "SynBack" software. The location where the master folder is copied on the database server is the same as provided by the STEWARD operators during the creation of 'Mappings' in the database (refer Appendix A). It should be noted that the external hard disk is only used to establish a connection between the FTP server and the database server for the transfer of master folder. The external hard disk does not play any other role in the daily operations of STEWARD. The next step of the automated process is the data uploading of the processed data, i.e. the station/lane data onto the STEWARD database. This is performed by configuring the Oracle OWB as follows:

First, a data loading algorithm (refer Section A3.7 of Appendix A) is written in the OWB using the tools provided in the graphical interface of the OWB Control Center for all the aggregated time intervals data files, i.e. the 5-, 15-, and 60-minutes station/lane data. Each algorithm is associated with the respective data folder inside the master folder. After these algorithms are written and tested, they become the base for the data loading procedure in Oracle. These loading procedures are then started for data uploading. The categorical data inside the master folder, i.e., 5-, 15-, and 60-minutes station/lane data are then uploaded to the respective 5-, 15-, and 60-minutes station or lane tables in the database.

To incorporate the data loading procedures with the automated process, all the procedures are scheduled to start at a particular time of the day through the OWB Design Center. For each procedure, a job is created to run at a particular time. An example of creating a schedule for these jobs is given in Figure 4. As shown in this figure, the jobs can be scheduled after defining the start date, end date and the time run for these procedures.

In the new server, these jobs are scheduled to run after the master folder is transferred from the external hard disk to the database server. As the last process of the automated process ended just shortly after 7:30 PM, the first loading procedure is scheduled to run at 8:10 PM. The timeline for all the loading procedures for all categorical data inside the master folder are given below:

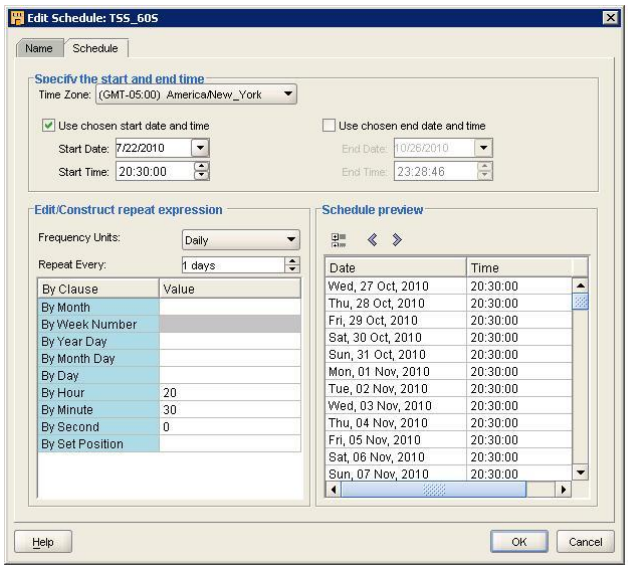


Figure 4. Schedule a job in Oracle OWB

At 8:10 PM: TSS_60S, 60 minute station data

At 9:00 PM: TSS_60L, 60 minute lane data

At 10:00 PM: TSS_15S, 15 minute station data

At 11:00 PM: TSS_15L, 15 minute lane data

At 00:05 AM*: TSS_5S, 5 minute station data

At 01:00 AM*: TSS_5L, 5 minute lane data

(*-following next day)

As the processed data files from all the districts are uploaded at the same time for a respective loading procedure, a substantial time gap is given between two consecutive jobs to avoid an overlap. The jobs are then started in the Control Center of the OWB after the scheduling is completed. Once started, the data loading procedures will automatically begin for a particular data file at its respective time. It should be noted that the loading procedures and the job schedules are created manually only for the first time during the STEWARD implementation. Thereafter, only the job schedules are monitored.

After all the jobs are finished, the station and the lane data are available at the STEWARD database. At this point, only the daily traffic data or the "All Data Fields" reports are available to the users. The next section describes the generation of the customized reports or the materialized views and its integration with the automated process.

2.3.4 Daily updating of Reports on STEWARD

The last step in the automated process of STEWARD's data loading is updating the materialized views automatically. As explained earlier, the materialized views are primarily the alternative

name for the reports that are generated by the STEWARD system in the Oracle database environment. For the generation of these reports and to make them available on the STEWARD website, it is necessary to update these materialized views. As new traffic data gets uploaded to the database on a daily basis, the customized reports are also expected to be updated on a daily basis. To include the updating of the materialized views as a part of the automated process, two components of the STEWARD system were configured.

First, the parameters associated with the respective materialized view are configured as discussed in Section 2.2.2 of this report. As shown in Figure 5, the "refresh-on-demand" method and the "refresh-FORCE" are selected.

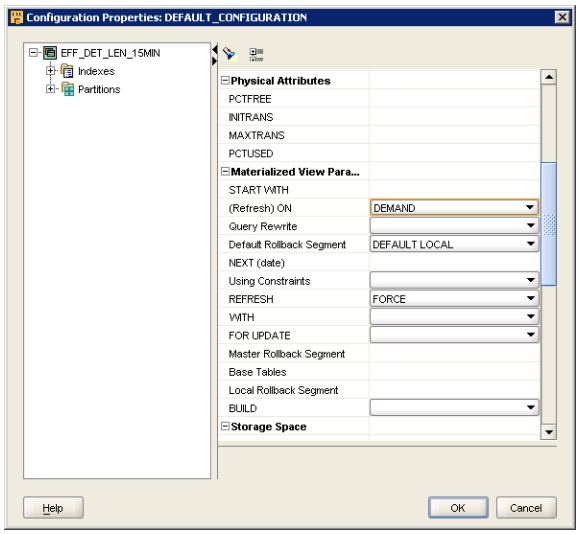


Figure 5. Refresh settings for materialized views

After the parameter settings are configured, a PL/SQL code is run through the SQL software for automatically updating these materialized views. A sample code for the "Traffic Counts 5mins" is provided in the Appendix B of this report. Within this code, the refresh method and the scheduler are configured. It was observed that with the FAST option within the PL/SQL statement, the materialized views are updated incrementally according to the schedule.

In the current STEWARD system, the "Traffic Counts" reports or materialized views are updated with the help of these PL/SQL statements and these statements are run at 3:00 AM after the station and lane traffic data are loaded into the system. At this stage only the Traffic Counts reports are provided to the users as they are updated daily. However, in future as the demand for other reports increases, the automatic updating of all the materialized views will be enabled except for the "Performance Measures" (refer to Section 2.2.2 of this chapter). This forms the last part of the automated process of the STEWARD operations. After the data is uploaded and

the reports are generated, the users are able to access these reports through Internet on the next day.

2.4 Improved Documentation for Lane Configuration and Detector Information

From the previous server, it was felt that some additional information is required for the users to understand the lane configurations and detector information and there is a need to present these details in a more descriptive manner. Also, it was found out that some of the basic information is missing from the lane configurations of the deployed detectors. To address this issue, the basic information related to the lane-mapping of the detectors at a given station level were updated in the new server. All the respective enhancements that are implemented in the new server are described next.

2.4.1 Inclusion of Auxiliary Lanes Data

To ensure that all the detectors collect and report the most accurate per lane traffic counts data at a given station level, the traffic data from the auxiliary lanes or the acceleration lanes were included in the new server along with the general purpose lanes. In the previous server, the daily traffic data from these lanes were not included and thus, it became a concern for the FDOT Statistics Office for their comparisons of the traffic counts from the FDOT District ITS detectors (STEWARD detectors) with the permanent detectors maintained by the FDOT Central Office. As the FDOT Statistics Office includes the traffic counts from these detectors in their annual report, it was necessary to include these additional lane(s).

This task was performed on a detector by detector basis by comparing the facility information received from the respective TMC's with the satellite images (Source: Google Earth) of the facilities in all the districts. With the help of the satellite images and the lane-mappings of detectors at each station level, all the auxiliary lanes and the general purpose lanes were identified. The lane mappings of all the detectors in the new server were extensively reviewed and the changes to the detectors were implemented accordingly. All the stations that includes the data from the additional auxiliary lane are marked with an asterisk (*) sign on the list of the detectors on the STEWARD web-server to clearly distinguish them with the other detectors. It is assumed that all the other detectors without any markings contain data only from the general purpose lanes.

2.4.2 Inclusion of HOT Lanes and Ramp Data

With the increasing number of users for STEWARD data, the daily traffic data are currently used for a wide range of research projects. A few of these projects required the data from the ramps and the newly built HOT (High Occupancy Toll) lanes in District 6 which led the STEWARD operators to include the data from all the on/off-ramps and the HOT lanes in the STEWARD database. The STEWARD operators had two available options through which they could have

accomplished this objective. First, to include the ramp/HOT lane data in the current list of lane mappings of the detectors and make the changes accordingly in the database. Second, to create a new set of detectors that only provides the ramp data and the HOT lanes data at a given location along with the existing detectors. After reviewing the database structure, it was observed that the first option would lead to a lot of complexity within the database structure and was not considered as a feasible option from the STEWARD operators' point of view. Therefore, it was decided that an entire new list of detectors would be prepared to accommodate the HOT lanes and the ramp data in the new server.

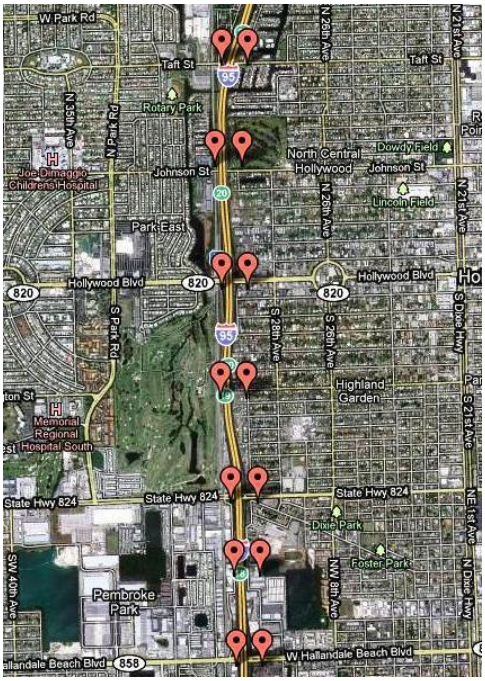
To include the HOT lanes, a numbering scheme was added to the District 6 facility file and an additional numbering scheme was included to accommodate the ramp detectors for all the districts that had active ramp detectors. The format of all the new detectors and their numbering scheme were kept consistent with the format of the other available detectors. The numbering scheme is further explained in details in Section 2.5 of this report with all the necessary information. With the inclusion of these detectors, the daily traffic HOT lane data and the ramp data are easily accessible to the users. It is also expected that a similar numbering scheme would also be created for the HOV (High Occupancy Vehicles) lanes in District 4. This task is expected to be accomplished in a future project.

2.4.3 Changes to the Interactive Maps and Detector Information

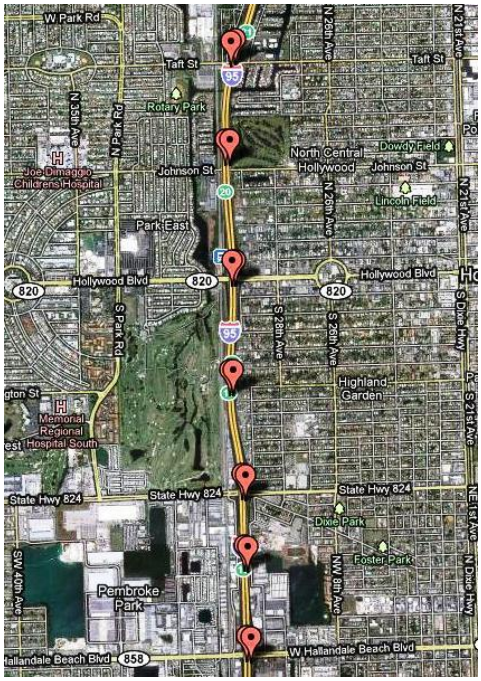
All the detectors in the STEWARD database are located geographically in the form of an interactive map which uses the Google Maps application. These detectors are mapped from their latitude and longitude values provided in the facility file from the respective district's TMC. After the detectors were placed on these maps, it was found that some of the detectors were dislocated and were not properly placed on the facilities/interstates. For example, some detectors were found to be in the middle of a lake or on top of a building, or some of them were found to be present on the arterials instead of an interstate. After further investigation, two reasons were attributed to these problems. First, the detector's latitude/longitude values provided by the TMC's were not completely correct, or second, the Google Maps are not updated with the latest satellite imagery.

To address this issue of dislocated detectors on the interactive maps, the latitude/longitude values of the entire network were verified with the respective TMC's. After verifying these details, only few detectors were found to be placed at their appropriate locations whereas all the other detectors were still found to be dislocated. As the location of the detectors becomes critical for some of the research projects and for the users reference, the STEWARD operators decided to tweak the detector's latitude and longitude values so that there are no obvious visual discrepancy in locating these detectors on the map. All the detectors were then reviewed one by one and their latitude/longitude values were modified accordingly. The changes made to the latitude/longitude were provided to the respective TMC's for their approval of the newly located detectors. Figure

6 shows that the differences in the locations of the detector before and after the latitude/longitude were changed.



A) Previous server Interactive Maps



B) New server Interactive Maps

Figure 6. Comparisons of new and previous interactive maps

After further reviewing the detector locations, it was observed that at locations where the detectors collect the data on both directions of a roadway, only one direction of the roadway was displayed on the maps. This discrepancy occurred because the place-marks for both the directions had the same latitude/longitude values. To remove this discrepancy, only one place-mark was displayed on the interactive maps which included the details of the detectors from both sides of the roadway.

Another area of improvement that was identified in the previous server was in describing the detectors on the interactive maps. In these interactive maps, the detector details are provided in the form of a tooltip which is unique for all the detectors in the database. To make the tool-tip more informative for the users, the milepost values for each detector location were added in the tool-tip. The inclusion of the milepost values has proved to be a useful tip for the FDOT and other researchers. Last, an example of the tooltip was provided on every interactive map that helps the users to understand them clearly. It is expected that the tooltip information for the interactive maps will be made more informative in the future and will be made even more user-friendly. An example of such tool-tip is shown in Figure 7 that will enable the users to easily see the detector station details in a balloon format.

Figure 7. An example of an interactive map with detector details in balloon format

2.5 Facility Information Updating

As TMC's continue to add detectors to their network, it is necessary to update the detector location information (latitude/longitude), the lane-mapping (lane by lane configuration) and other useful information. This work was performed for all the districts that were added to the new server with assistance from the FDOT districts and the TMC's that manage the ITS detectors. Additionally, the STEWARD station numbering scheme were reviewed and revised as necessary.

2.5.1 Introduction

The ETL (Extraction Transformation Loading) utility, i.e., the SunETLUtility, is used by STEWARD to prepare the data for the database. This utility converts the raw data into the processed data that are loaded to the STEWARD database. For this process of converting the raw data into the processed data format, the utility requires a facility file. This facility file is different and unique for all the districts and is prepared by the STEWARD operators. This file includes the detailed information on every district's detector network with their unique detector-id's, description, locations and other important details and thus, becomes the most important part of the data conversion from the raw data to the processed data format. It is necessary for the STEWARD system to have the most accurate details of the detectors that are deployed on the field and the facility files should be prepared with the most recent/available information from the FDOT districts and the TMC's. Therefore, these files were prepared very carefully by the STEWARD operators as they become the base for all the other processes in the STEWARD system. All these facility files are unique for each district and the numbering scheme applied in these files are also unique and specific to a respective district. Some of the benefits that are provided by these unique files are described below:

- To represent each detector and their records unique globally
- To support and help the TMC's for managing these detectors for system-based measures and quality assessment
- To relate the measures obtained from a specific detector to other databases like RCI (Road Characteristics Inventory), FDOT Statistics Office counts, crash database, etc.

During the migration of the previous server to the new server, the detector information for all the previously existing facility files were reviewed, revised and prepared according to the latest information. While the reviewing was done, it was observed that new detectors were added by the TMC's in their respective districts. All the new detectors were included in the new facility files for the new server and the other necessary information were revised according to the new details provided by the TMC's. While these files were prepared, it was found that not all the information was provided by the TMC's. All the other necessary details that were important to process the data from the raw data were added by the STEWARD operators. The information in the facility files are represented in an excel sheet with two spread sheets namely: Station Data spreadsheet and Lane Data Spreadsheet. The following section gives a general idea of all the changes that were made in the new facility files for all the districts.

2.5.2 Station Data Spreadsheet

The following section discusses the fields for each station in a particular facility file in the Station Data spreadsheet and the changes that were made during the migration of the previous server.

- *Station_Index:* This is a number assigned sequentially to all the stations for a given district and is mainly required for all internal processing purposes and does not appear in the database or in the reports. These are marked numerically from 1 to the value that represents the number of detectors or stations in a particular district.

- *Stationcdw_Num*: This is a 6 digit unique number that is given to all detectors in a district. The format of these numbers are in the form of "dfnnns" where,
  d: represents the district number
  f: represents the facility number within the district (0-9)
  nnn: represents the station number or detector number within the facility (0-999)
  s: represents the direction at which the detector is located (0 for NB/EB, 1 for SB/WB)

  An example of a station number or detector ID in District 6 would be "611231" where 6 represents the District 6, 1 representing the facility (i.e., SR-826), 123 representing the station number on the facility 1 and 1 representing the NB direction. Similarly, all the other detectors were also numbered in a similar manner. In addition to the detector network in the previous server, District 5 and District 7 detectors were added in the new server. Other

roadways or facilities were also added in every district. The HOT lanes and ramp data were also added in the station list as described in the last section. For all the HOT lanes stations, the facility number 9 was given. Similarly, for all the newly added ramp detectors, the facility number was given a value 8. The inclusion of these numbers uniquely distinguished them from the other detectors and thus, it helps the users to identify them clearly.

- *Description:* This gives a description of the detector station. For example, in District 2, the description for the detector 210391 is given as "210391, I-95 NB South of Bowden Road". These descriptions were either contained in the raw data of the respective detector or they were named after reviewing the locations of the detectors through the satellite imagery. All the detectors in each district were reviewed one by one and their descriptions were changed accordingly. Also, to distinguish a particular detector with another detector with similar name, the station id and the mile post are added in the description.

- *Status:* This indicates the status of a detector. The status is indicated as 0 for a normal working detector, 1 is indicated for an offline detector and 2 is indicated for an undetected detector. After consulting with the TMC's, all the stations that were active were included in the facility file and all their respective values were changed to 0. For some detectors that were notified as offline stations in the facility file are currently not reported in the database.

- *Road:* This is the name that is given to a particular facility/roadway in a given district. All the newly added facilities names were changed accordingly in the facility file.

- *Longitude and Latitude:* This indicates the locations of a particular station or a detector and these are expressed in degrees and decimal degrees. These values were verified with the TMC's and were revised accordingly.

- *State_Milepost:* This number gives the state mile post of a particular station. For all the detectors in the database network, this value was provided by the TMC's and for all the detectors that did not have any values, their state mile posts were calculated by using the latitude/longitude values. This is mainly required for sequential ordering of stations.

- *Roadway_Id:* This number specifies the roadway id of the facility where the detector is located. This is required for correlating the database network with RCI and crash data. This also helps in identifying the county number for generating traffic count files. The entire roadway id's were reviewed and added to the facility file in the new server. For example, 72020000, specifies that the county number is 72.

- *Roadway_Milepost:* This value gives the roadway milepost of a station on a particular roadway. This is also required for correlating the database network with RCI and crash data.

In the new server, all the roadway mileposts were added from the information available from the respective TMC's. For the cases, where the roadway mileposts were not given, they were calculated by using the state milepost. For example, 2.7.

- *Max_Speed:* This value is the posted speed limit of the roadway segment where the detector is located. This information is required for travel time reliability reporting because the notion of on time arrival is based on the speed limit. These values were obtained from the each district's TMC and were revised accordingly. The detectors that did not have their posted speed limits were verified with the respective TMC's and were included in the new server.

- *Num_of_Lanes:* This value gives the number of lanes that a detector covers across the roadway segment on which the detector is located. These values were verified from the lane-mapping information from the TMC's. The facility files in the previous server did not include the additional auxiliary lane in the number of lanes value. This was included in the new facility files. For all other detectors, where the lane count was not provided, these were obtained from Google Earth (Source: Google).

- *Upnode:* This value indicates the station number "*Stationcdw_Num"* that is upstream of the current detector location. As new detectors were added, the upnode values were verified and included in the new facility files.

- *Lane_Capacity:* A test capacity value which is only used by the SunETLUtility while the data is transformed from raw data to processed data format. This value does not play any role in any calculations.

- *Detector_Type:* This gives the information on the type of detector at its location. For example, RTMS, Wavetronix, or BiTrans Loop. This is not used in any analysis at present, but is provided for future use.

- *County:* This value gives the county number value assigned by the Department of Transportation to the counties within the state of Florida. These were added in all the facility files as this information was missing in the old facility files.

- *Count_Station:* This is a number assigned by the FDOT Statistics Office or District Planning Office for comparing the traffic counts from the FDOT permanent count stations with the SunGuide detectors. The count stations were received from the FDOT statistics office and were included in the new facility files.

2.5.3 Lane Data Spreadsheet

Another section of the facility file was reviewed and revised in details. The details of the items included in the lane data spreadsheet are described next and the respective changes during the migration of the server are clearly stated.

- *CDWStation:* This is the same 6 digit number as present in the station data spreadsheet, i.e. *Stationcdw_Num*. This is used as a key to relate the station data spreadsheets with the lane data spreadsheets.

- *Lane:* The lane number is a unique number given to each lane at a station level. The format of this number is dfnnnslm, where dfnnns is similar to the station data format and

  l: represents the type of lane,

  1 - Left entrance ramp
  2 - Left exit ramp
  3 - Freeway main lane
  4 - Right entrance ramp
  5 - Right exit ramp, and

  m: represents the lane number from the median, i.e., the lane closest to the median is 1 and increases as the lanes increases from the median. The assignment of this number is different for every district and their facilities, as the lane closest to the shoulder could also be given as number 1. Thus, the assignment of this number becomes the most critical part of the data transformation and has to be dealt with utmost care. At the same time, this part of updating the facility file was the most time consuming job during the creation of the new facility files.

- *Tmc_Id:* This is the lane ID that is assigned by the TMC to each lane at a station. This ID is obtained from the archived zipped files from each TMC. This field also becomes very critical in data transformation process, as the ID's provided in the facility files must match the ID's in the archived zipped files. If this ID is not found in the archived data records, the corresponding lane is reported as an orphan lane. Else, if the ID's in the archived data records are not matched with the ID's in lane data spreadsheet; the lanes are reported as Null lanes.

- *Det_Type:* Similar to the one in the station spreadsheet, but these are provided for all lanes.

- *Direction:* It gives the direction of the traffic detected on this lane. For example, 1 for NB/EB directions and 2 for SB/WB directions.

- *Status:* Similar to *Status* in station data spreadsheet.

- *Roadway_Id:* Similar to *Roadway_Id* in station data spreadsheet.

- *Roadway_Milepost:* Similar to *Roadway_Milepost* in station data spreadsheet. They are required here because stations that detect traffic in both directions may have different roadways assigned.

- *Max_Speed:* Similar to *Max_Speed* in station data spreadsheet.

- *TSS_Speed:* This speed value is only a test parameter used during the data transformation. It is assigned as 20 mi/h for all the lanes.

- *Occupancy:* This is also a test parameter used during the data transformation. The value is assigned as 0 for all cases.

- *Count_Station:* Similar to *Count_Station* in station data spreadsheet.

After the two sections of the facility files were reviewed, the revised and modified versions of the facility files were included in the SunETLUtility for data compilation and transformation. As mentioned earlier, the task to update the facility files became the most time-consuming part of enhancing the STEWARD system because different lane-mapping structure existed for different districts. This meant that a different approach was applied to finalize the mapping structure for each facility file. It is expected that in future, if additional districts/facilities are added to the STEWARD network, the number of hours in associated with the respective task would be mentioned accordingly. The detailed information on how these facility files were configured and prepared are provided in Chapter 5 of Appendix A of this report.

2.6 Updated Documentation of STEWARD

The documentation on the setup of STEWARD from the software and hardware aspects was created. This documentation also includes the detailed information on the daily operations of STEWARD and how the different components are configured. It should be noted that this document was updated from an earlier version that as developed for the FDOT project "Statewide Transportation Engineering Warehouse for Regional Archived Data" (FDOT project BDK545-37). Several new sections were also added. This documentation is contained in Appendix A of this report.

3 Summary and Future Work

3.1 Summary

The software and hardware aspects of STEWARD were enhanced in order to improve the system availability, data availability, and the system performance. Several hardware purchases were made during the project completion. The FTP (File Transfer Protocol) server of the STEWARD system was upgraded to a faster computer and additional storage capacity (4 TB) was included. This storage capacity is in the form of internal SATA drives, replacing what was a daisy-chain of several USB-interfaced external hard drives, resulting in significant performance gains. A professional-grade server computer was also bought to replace the desktop-based computer that hosted the database and web-interface. The database server of the STEWARD system was upgraded with Windows Server 2008 (previously Windows Server 2003) as the operating system and Microsoft Internet Information Services as the web server.

A utility application was developed that reads the daily incoming data feeds from the FTP server, then processes the raw data using the ETL (Extraction, Transfer and Loading) utility, and uploads the data directly to the database server. The application is run on the FTP server during the day time after all the raw data is received and the data is uploaded during the late-night hours. the creation of this automatic utility application allowed the system to take better advantage of the improved hardware performance, as well as make the entire process of getting processed data onto the server and available to user much more efficient. The end result was that the lag time for the data availability from TMC's to the users was reduced to two days. To further enhance the system performance, the customized reports are also now updated automatically.

While this enhancements project was a good start to making the STEWARD system more useful in a production mode, there are still several improvements that can be made to the system. Furthermore, there is the ongoing need for system maintenance. The following section suggests the objectives and tasks for a follow-on project that could provide these improvements and continued maintenance.

3.2 Future Work

With the implementation of the automatic utility application on the STEWARD system, STEWARD has reached to a self-sustainable stage and is up-to-date with every district's detector network. However, the system is not yet self-expandable and there could be other issues and concerns that may arise while operating and maintaining the system in the long-term. These issues, briefly, are:

- It is anticipated that STEWARD user activity levels will increase as more data become available and awareness of STEWARD increases as it transitions from the research and development phase to the operational phase. With the increase in the number of users of

the STEWARD system, other FDOT districts' data should be included in the future since it is not self-expandable. And as the districts currently included in the system continue to update their facility information and expand their network on a regular basis, it is necessary that these changes are incorporated accordingly in a timely manner.

- The STEWARD system receives approximately 1 GB of data every day. As the TMC's continue to expand their network detector coverage, the amount of data that must be transferred to STEWARD and then processed and loaded into the database will continue to increase. It is necessary that this operation is regularly monitored with proper maintenance of the hardware and timely review of the database and web-scripts.

- As STEWARD gains increased user activity, there will be increased demand for additional pre-configured reports to be provided on the STEWARD website. It is also necessary that other updates, such as data availability and tools/resources availability should be provided in a more user-friendly manner.

APPENDIX – A

Table of Contents for Appendix A

Table of Figures for Appendix A

A1 Introduction

This is a user manual that describes a series of steps and procedures to install the "Statewide Transportation Warehouse for Regionally Archived Data" (STEWARD) database and software required for its functionality. The document is prepared as an appendix for the "FDOT CDW Enhancements" project, by modifying and editing the previous user manual prepared during the course of "FDOT BDK545-37, "Development of a Central Data Warehouse for Statewide ITS and Transportation Data Phase II: Proof of Concept". The motivation behind creating this manual is to guide FDOT and ITS personnel in setting up STEWARD in the Traffic Engineering Research Laboratory (TERL) facility. The topics include Oracle database program installation, STEWARD deployment and the STEWARD web site installation. This material also includes the daily operations of STEWARD from data receiving at Traffic Management Centers (TMCs) to the data uploading on the STEWARD website. It is assumed that the concerned ITS personnel has a working knowledge of Oracle databases and internet site management.

A2 Oracle Database Program Installation

To install the Oracle database on a system, the Oracle Database 10g Release 2 version is required. Along with this, the Oracle database companion and the database client software are also required. This is available from the Oracle Corporation website. After all the related software is downloaded, STEWARD is setup on the chosen system. The installation steps are as follows:

A2.1 Install Oracle 10*g* Release 2

For the current system, the Oracle 10*g* Release 2 is installed on the Microsoft Windows Server 2008 (64-bit). It should be noted that multiple versions of the software are available at the Oracle website, but it is necessary to download and install the compatible version for the chosen system. Therefore, the 64-bit software should be downloaded from the Oracle website. The 32-bit software may also work, but the installation may run into problems at a later stage.

A2.1.1 Run the Setup.exe File

Locate the setup.exe file and run the Oracle database software just like a standard Windows procedure.

A2.1.2 Select Advanced Installation

As shown in Figure A1, select *Advanced Installation* for the installation method and click 'Next'.

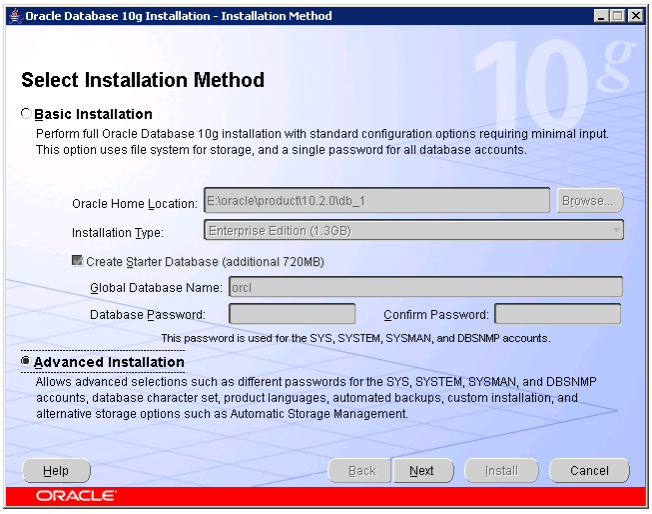


Figure A1. Oracle DB Installation - Installation Method

A2.1.3 Select Enterprise Edition

As shown in Figure A2, select Enterprise Edition for the installation type and click 'Next'.

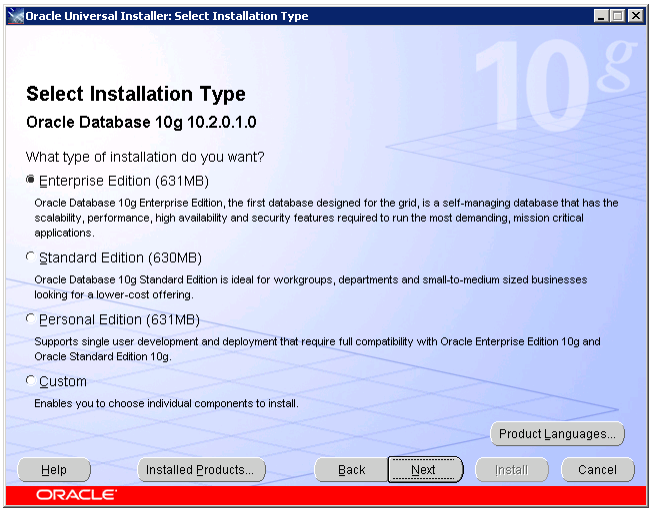


Figure A2. Oracle DB Installation - Installation Type

A2.1.4 Select Default Location

As shown in Figure A3, select the home name and path. This location should be chosen where the user wishes to keep the home of Oracle. The D: directory was chosen in this case because of the availability of space. Select the name as OraDb10g_home1 and the path as D:\oracle\product\10.2.0\db_1 and click 'Next'.

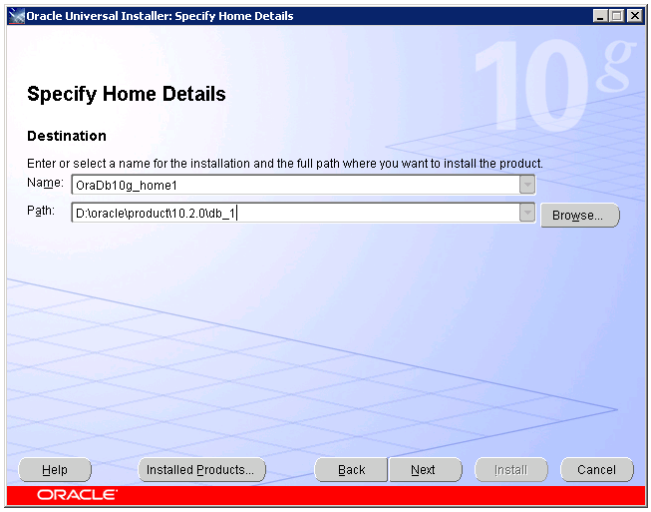


Figure A3. Oracle DB Installation - Installation Home Details

A2.1.5 Run the Prerequisite Checks

As shown in Figure A4, a *prerequisite* check will be performed and click 'Next'.

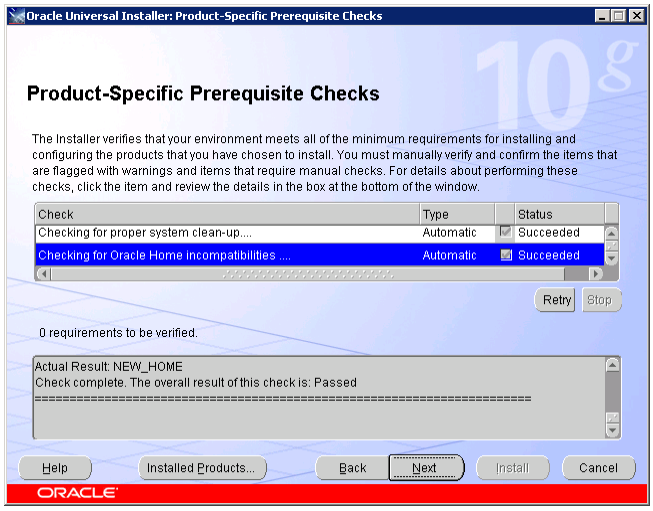


Figure A4. Oracle DB Installation - Prerequisite Check

A2.1.6 Select Database Software Only Option for Configuration Option

As shown in Figure A5, select the *Database Software Only* option and click 'Next'.

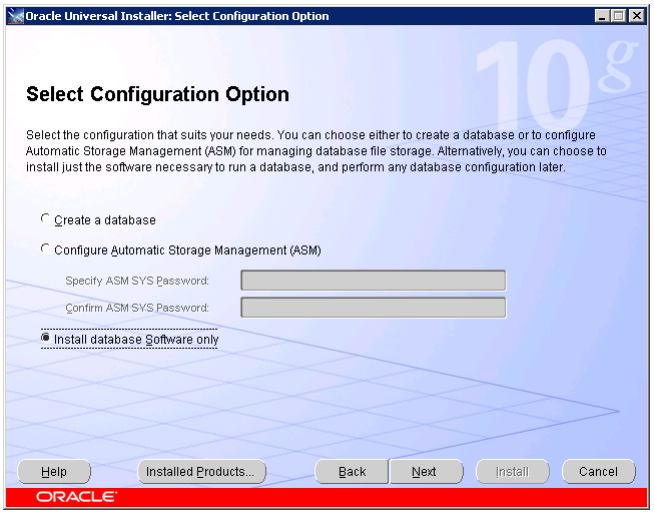


Figure A5. Oracle DB Installation - Configuration Option

A2.1.7 Oracle Database Installation

As shown in Figure A6, the Oracle Database installation summary is displayed before the installation is completed. Click 'Install' to start the installation.

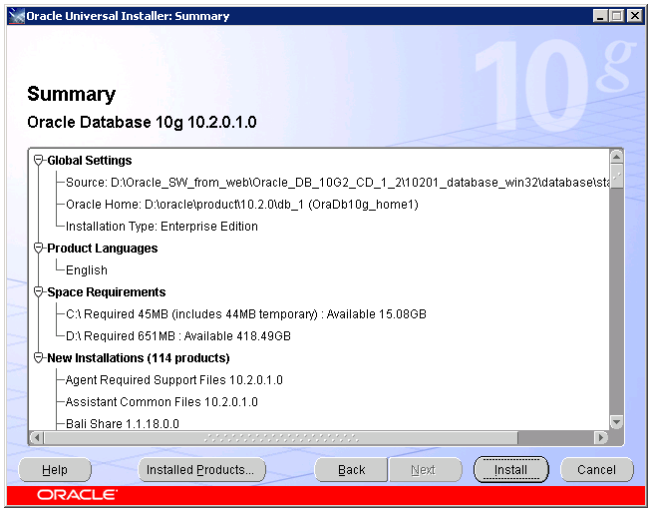


Figure A6. Oracle DB Installation - Installation Summary

After the summary of the installation is displayed, click 'Install' to begin the installation process. As shown in Figure A7, the Oracle database is getting installed.

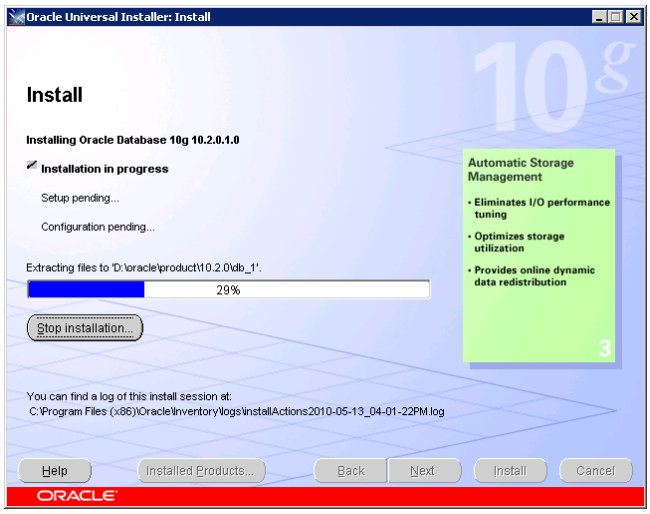


Figure A7. Installation Summary - Installation

As shown in Figure A8, the Oracle database installation is completed. Click 'Exit'.

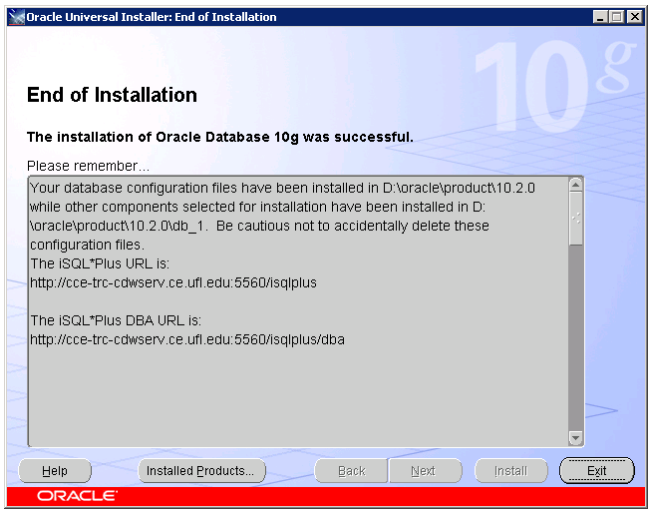


Figure A8. Oracle DB installation - End of Installation

After the Oracle database is installed, the oracle work flow is installed. This is described in the following section.

A2.2 Install Oracle Workflow

The Oracle Workflow is installed through the companion CD of the Oracle 10*g* Release 2 version for Microsoft Windows Server 2008, with a 64-bit operating system. The following steps are performed to install this component of the database.

A2.2.1 Run the Setup.exe File

Within the companion CD, run the setup.exe file. As shown in Figure A9, Oracle Workflow is installed with Oracle Universal Installer.

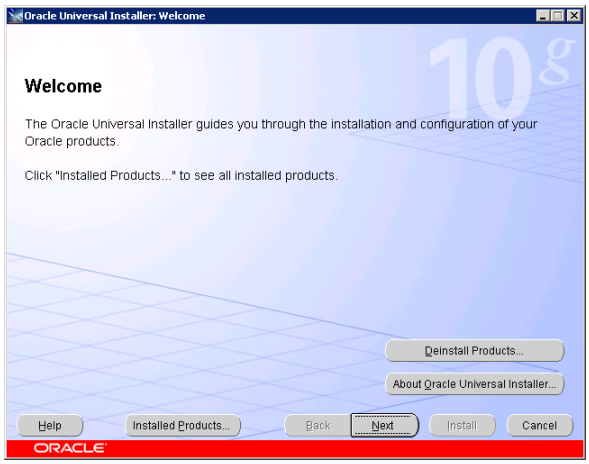


Figure A9. Oracle Workflow Installation

A2.2.2 Select the Oracle Database 10g Product

As shown in Figure A10, select the Oracle Database 10g Products for Oracle Workflow installation and click 'Next'.

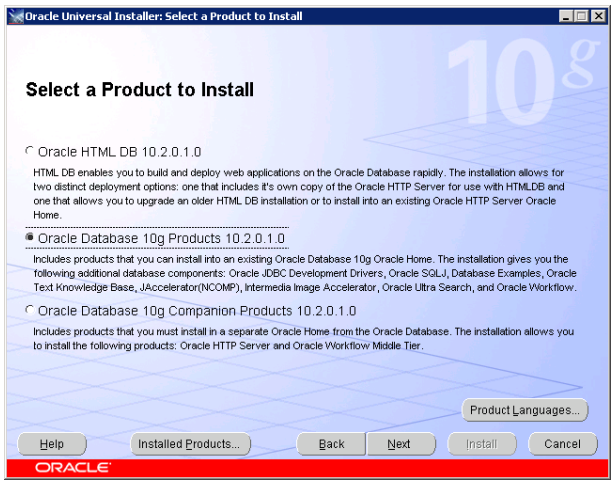


Figure A10. Oracle Workflow Installation – Product Selection

A2.2.3 Select the Home Name and Address

As shown in Figure A11, provide the home name and path name as OraDb10g_home1 and "D:\oracle\product\10.2.0\db_1" respectively and click 'Next'.

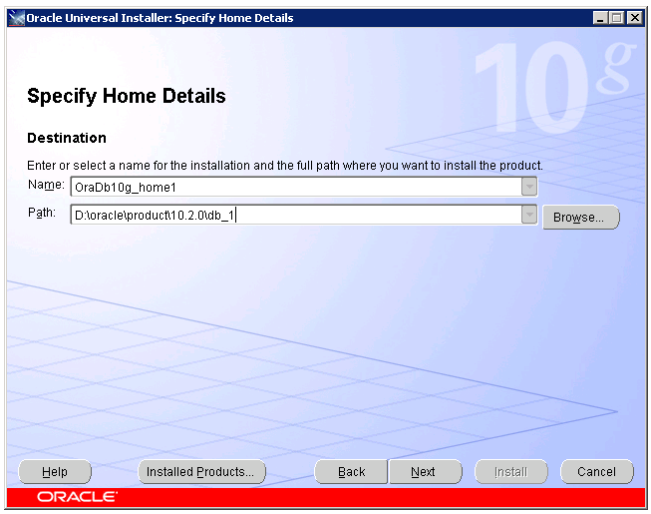


Figure A11. Oracle Workflow Installation - Installation Home Details

A2.2.4 Run the Prerequisite Checks

As shown in Figure A12, a prerequisite check will be performed and click 'Next'.

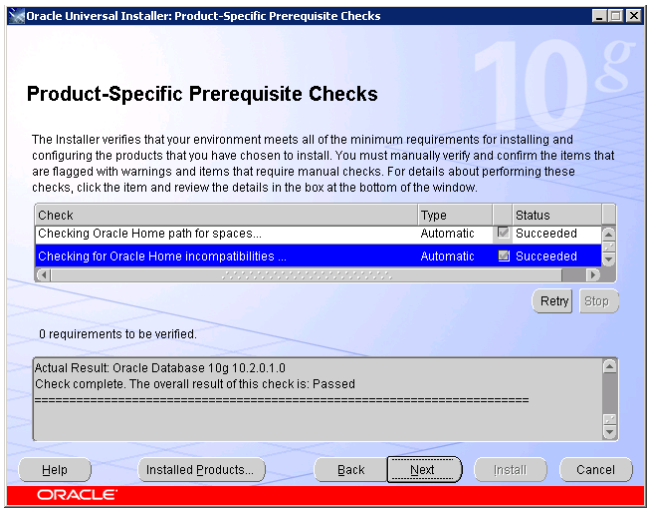


Figure A12. Oracle Workflow Installation - Prerequisite Checks

A2.2.5 Oracle Workflow Installation

As shown in Figure A13, the Oracle Workflow installation summary is displayed before installation. Click 'Install' to start the installation.

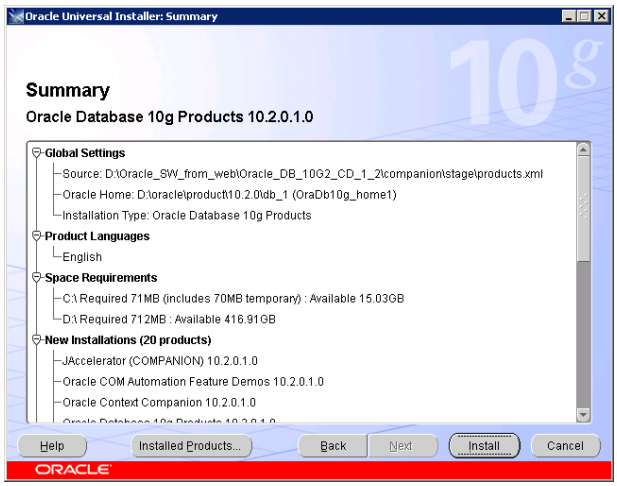


Figure A13. Oracle Workflow Installation - Installation Summary

A2.2.6 Oracle Workflow Installation Completion

As shown in Figure A14, the Oracle Workflow installation is completed. Click 'Exit' to exit from the application.

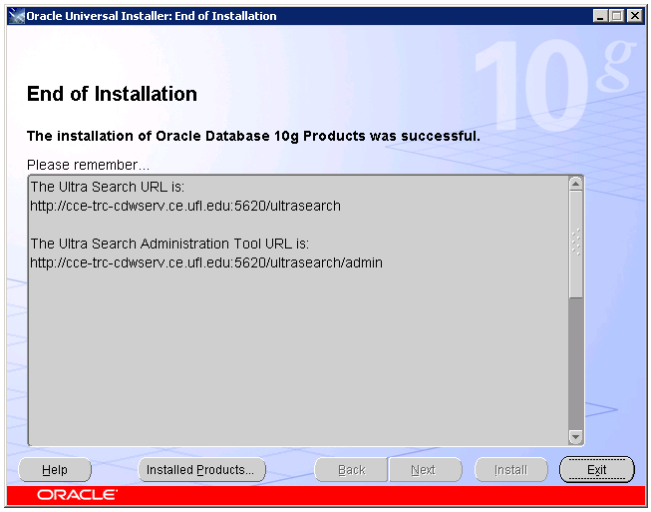


Figure A14. Oracle Workflow Installation - End of Installation

The next step in setting up STEWARD is to install the Oracle Warehouse Builder. This is described next in the following section.

A2.3 Installing the Oracle Warehouse Builder (OWB)

The OWB is installed from the Oracle Warehouse Builder 10*g* Release 2 for Microsoft Windows Server 2008 with a 64-bit operating system. The following steps are performed to complete this installation.

A2.3.1 Run the Setup.exe File

As shown in Figure A15, the OWB is installed with the Oracle Universal Installer available within the downloaded software. Click 'Next' to continue.

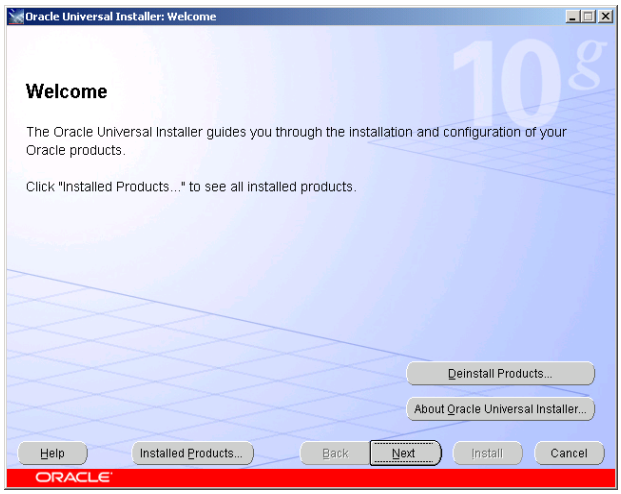


Figure A15. Oracle Warehouse Builder (OWB) Installation - Welcome

A2.3.2 Specify Home Details

As shown in Figure A16, the home name and the path where the OWB will be installed is provided. These are given as OUIHome1 and "D:\OraHome_1" respectively. Click 'Next' to continue.

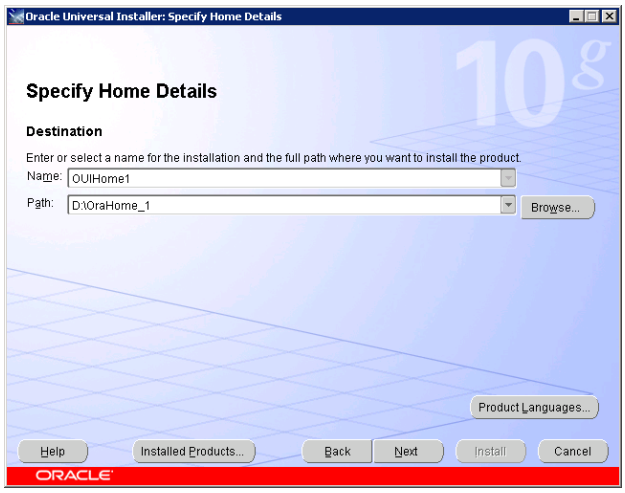


Figure A16. OWB Installation - Specify Home Details

A2.3.3 Preparation to install the OWB

As shown in Figure A17, the OWB installation summary is displayed before installation. Click 'Install'.

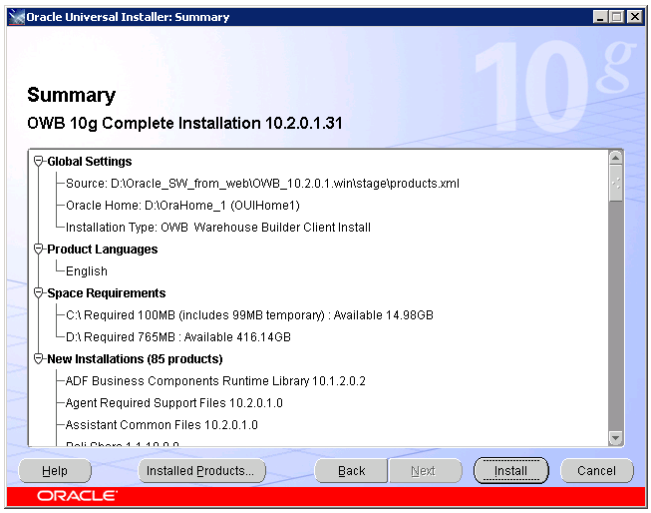


Figure A17. OWB Installation - Summary Information for Installation

A2.3.4 Install the OWB

As shown in Figure A18, Oracle Warehouse Builder is currently getting installed.

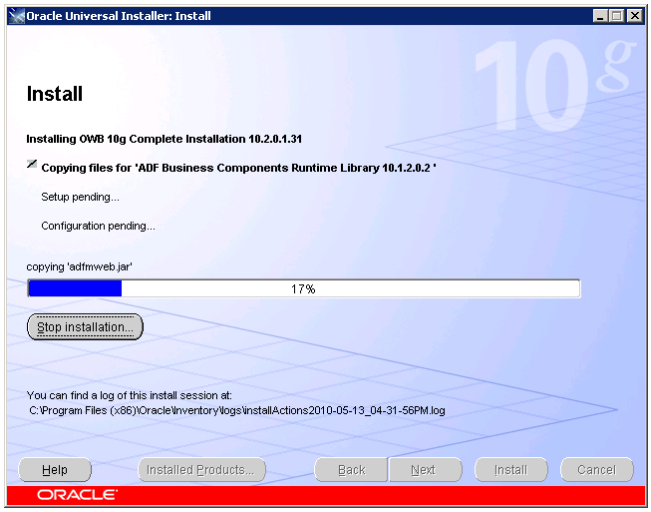


Figure A18. OWB Installation

A2.3.5 Installation Completed (OWB)

As shown in Figure A19, installation of Oracle Warehouse Builder has been completed. Click 'Exit' to exit from the screen.

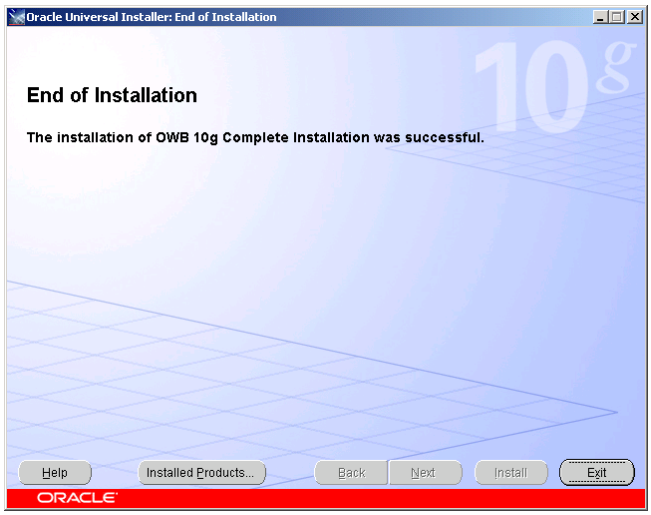


Figure A19. OWB Installation - End of Installation

After the OWB in installed, the OWB needs to be configured using the Design Center of the Oracle OWB. These set of procedures and steps are described next in this manual.

A2.4 Oracle Database Configuration Assistant

Before setting up the Oracle Warehouse Builder using the Design Center and Oracle Net Configuration (see 2.5), the Database Configuration Assistant should be run. This is by navigating to the Oracle Database Configuration Assistant from the already installed applications. This is available in Windows machine at:

Windows: Start\All Programs\Oracle-OraDb10g_home1\Configuration and Migration Tools\Database Configuration Assistant as shown in Figure A20.

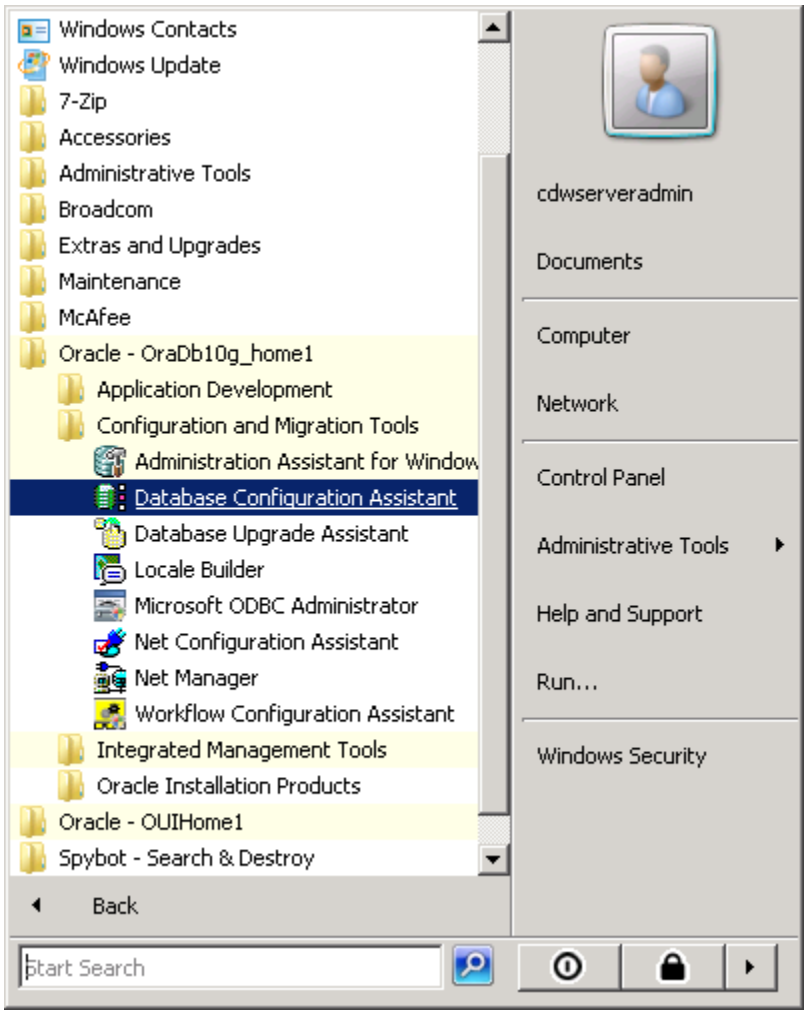


Figure A20. Running the Oracle Database Configuration Assistant

A2.4.1 Oracle Database Configuration Assistant -Welcome

As shown in Figure A21, the Database Configuration Assistant shows the welcome message.

Click 'Next' to continue.

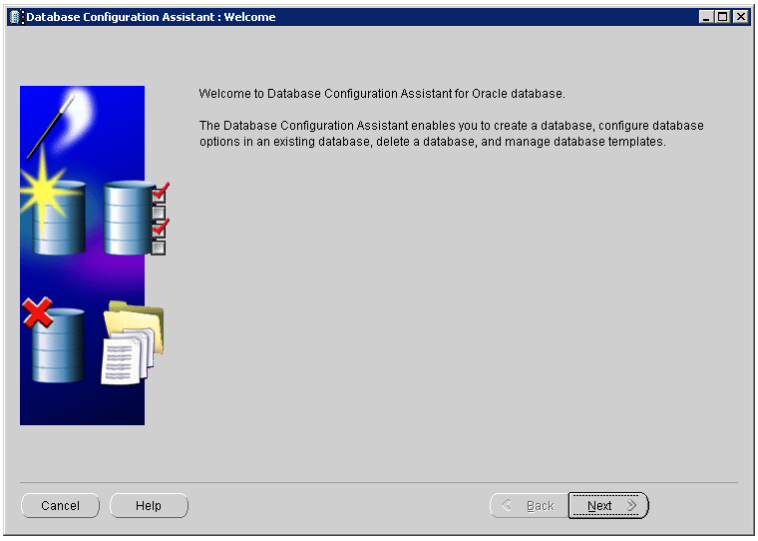


Figure A21. Oracle Database Configuration Assistant - Welcome

A2.4.2 Oracle Database Configuration Assistant - Operations

As shown in Figure A22, select 'Create a Database' and click 'Next' to continue.

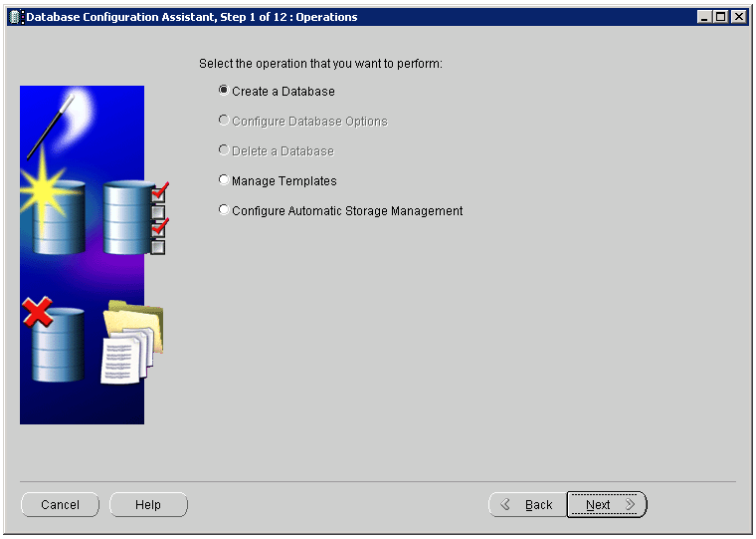


Figure A22. Oracle Database Configuration Assistant - Operations

A2.4.3 Oracle Database Configuration Assistant -Database Templates

As shown in Figure A23, select ‘Data Warehouse’ and click ‘Next’ to continue.

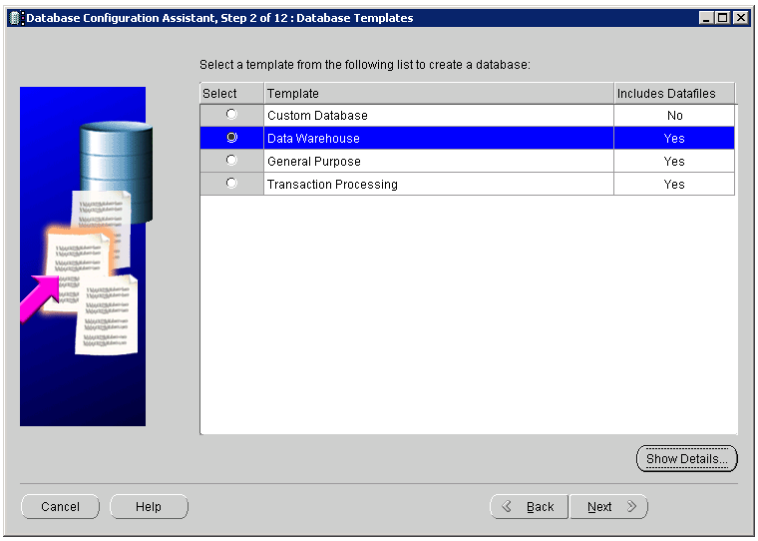


Figure A23. Oracle Database Configuration Assistant - Database Templates

A2.4.4 Oracle Database Configuration Assistant - Database Identification

As shown in Figure A24, enter ‘STEWARD’ as the global database name and the SID will be filled with ‘STEWARD’ automatically. Click ‘Next’ to continue.

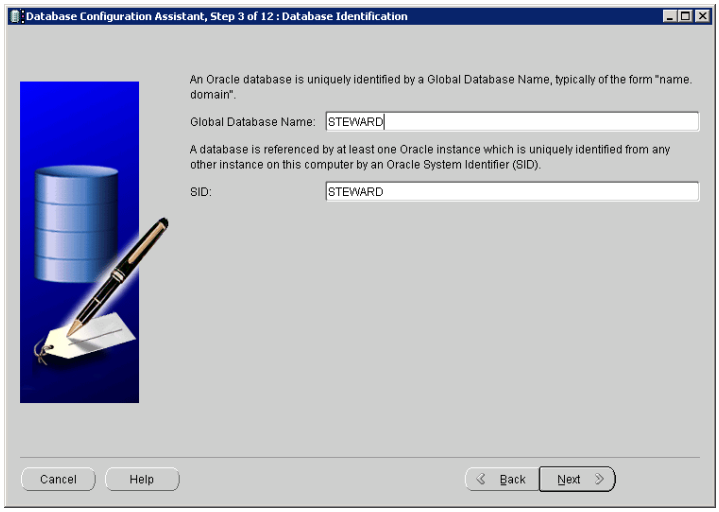


Figure A24. Oracle Database Configuration Assistant - Database Identification

A2.4.5 Oracle Database Configuration Assistant -Management Options

As shown in Figure A25, toggle the "Configure the Database with Enterprise Manager" option and keep the default selected option and click 'Next' to continue.

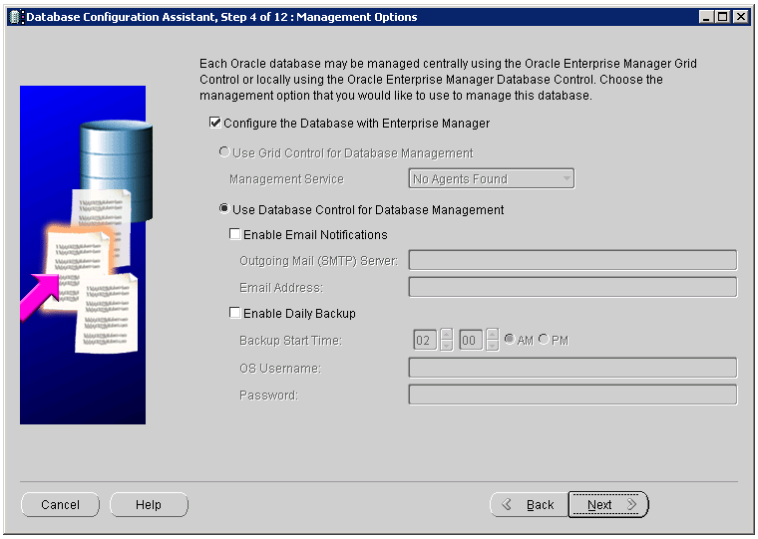


Figure A25. Oracle Database Configuration Assistant - Management Options

A2.4.6 Oracle Database Configuration Assistant - Database Credentials

As shown in Figure A26, enter your password. For the current system "trc513" is used as the password. Click 'Next' to continue. It should be noted that the password throughout the installation and configuration steps should be kept same to avoid complexity of the installation process.

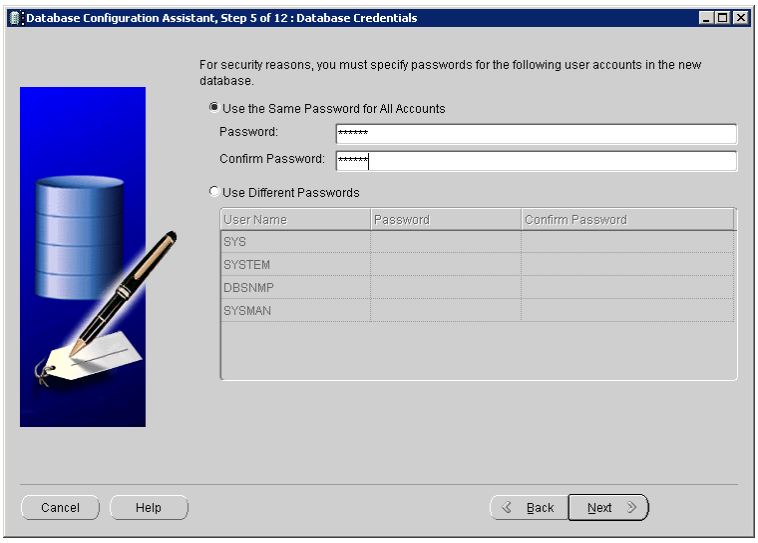


Figure A26. Oracle Database Configuration Assistant - Database Credentials

A2.4.7 Oracle Database Configuration Assistant -Storage Options

As shown in Figure A27, select ‘File System’ which is default option and click ‘Next’.

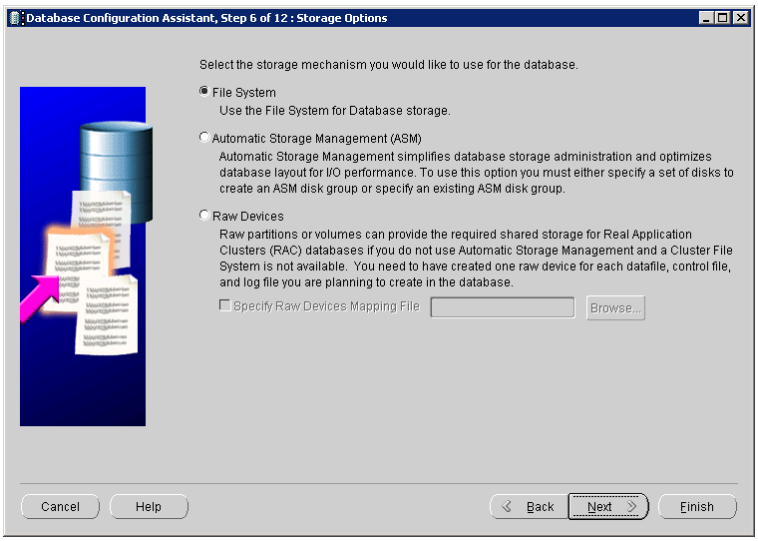


Figure A27. Oracle Database Configuration Assistant - Storage Options

A2.4.8 Oracle Database Configuration Assistant - Database File Locations

In this step, you may select the default option ‘Use Database File Location from Template’ or you may select another location as shown in Figure A28, where you may select ‘Use Common Location for All Database Files’. After selecting ‘Use Common Location for All Database Files,’ enter ‘K:\ORADATA’ into the Database Files Locations and click ‘Next’. This location becomes the place where the database is stored, so it must be chosen where space requirements are met.

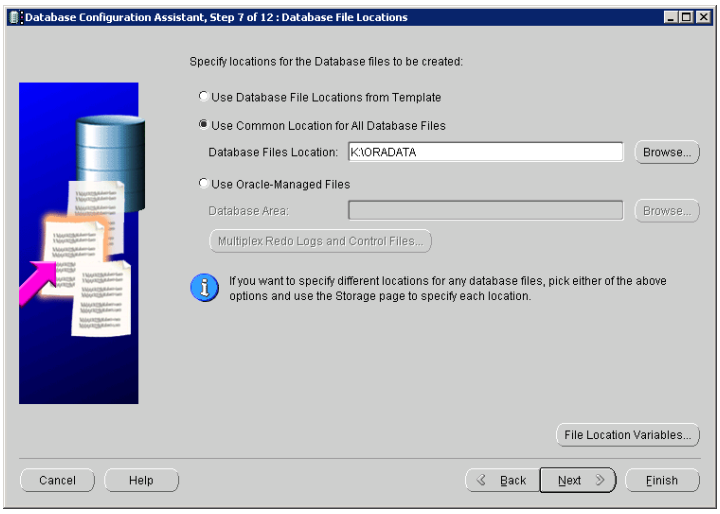


Figure A28. Oracle Database Configuration Assistant - Database File Locations

A2.4.9 Oracle Database Configuration Assistant -Recovery Configuration

As shown in Figure A29, select the default options and click 'Next'.

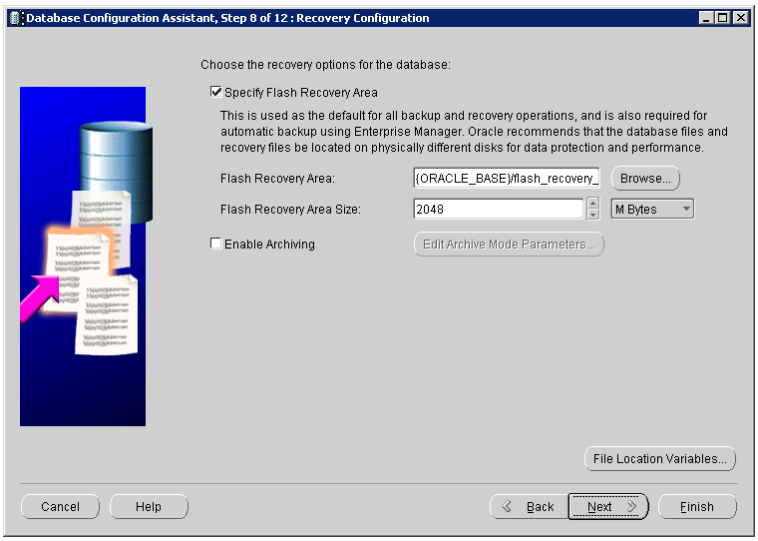


Figure A29. Oracle Database Configuration Assistant - Recovery Configuration

A2.4.10 Oracle Database Configuration Assistant - Database Contents

As shown in Figure A30, select the default options and click 'Next'.

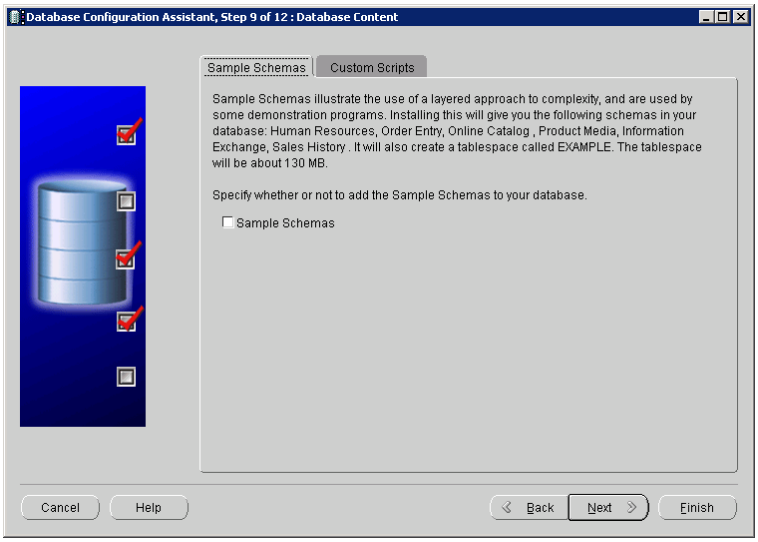


Figure A30. Oracle Database Configuration Assistant - Database Contents

A2.4.11 Oracle Database Configuration Assistant -Initialization Parameters

As shown in Figure A31, default options are kept unchanged and click 'Next'.

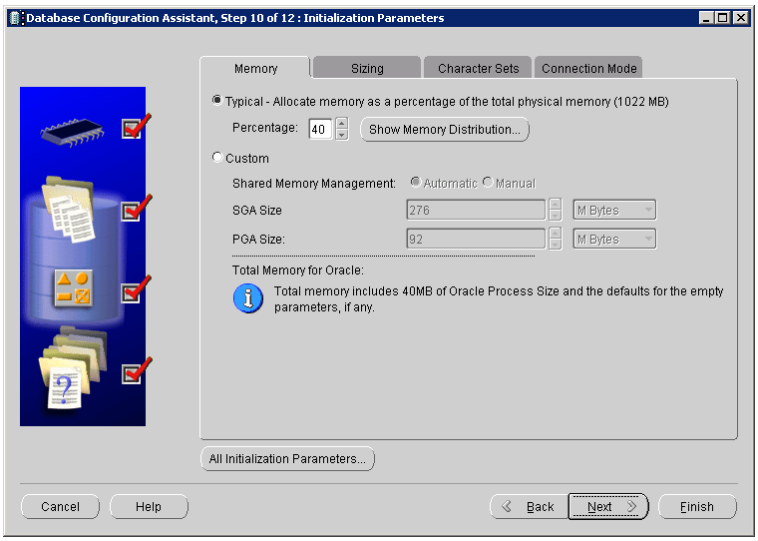


Figure A31. Oracle Database Configuration Assistant - Initialization Parameters

A2.4.12 Oracle Database Configuration Assistant - Database Storage

As shown in Figure A32, the 'Datafiles' subfolder on the left should be set by the 'K:\ORADATA\{DB_NAME}\'. After verifying it, click 'Next'.

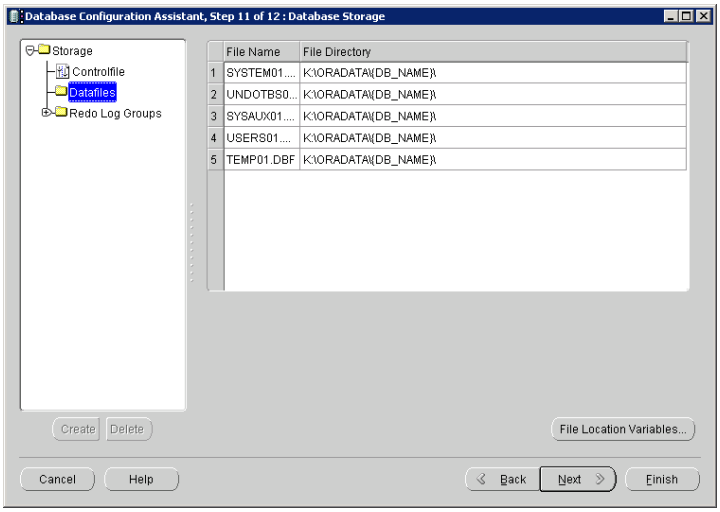


Figure A32. Oracle Database Configuration Assistant - Database Storage

A2.4.13 Oracle Database Configuration Assistant -Creation Options

As shown in Figure A33, nothing needs to be changed. Click 'Finish'.

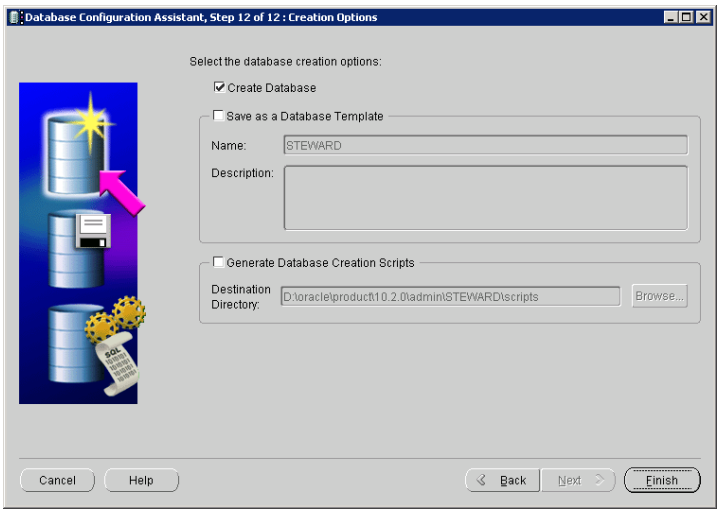


Figure A33. Oracle Database Configuration Assistant - Creation Options

A2.4.14 Oracle Database Configuration Assistant - Confirmation

As shown in Figure A34, the summary of details of the warehouse are checked. Click 'Ok' to continue.

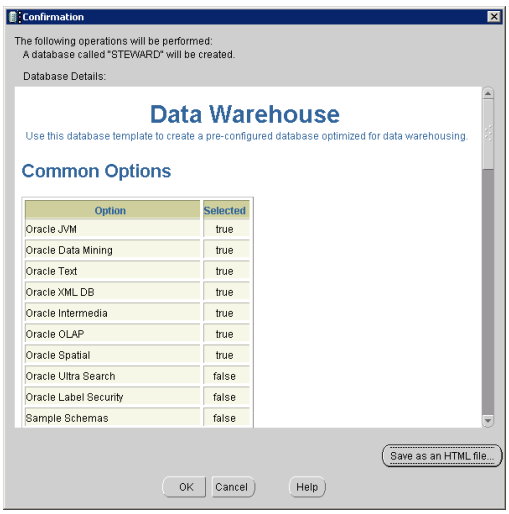


Figure A34. Oracle Database Configuration Assistant - Confirmation

A2.4.15 Oracle Database Configuration Assistant -Creating Database

As shown in Figure A35, the database is created in this step. Wait till the copying, creating and completing of the database creation is performed.

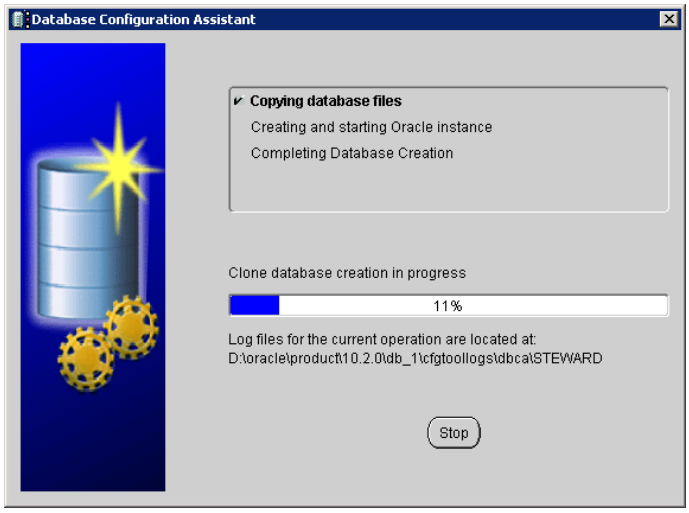


Figure A35. Oracle Database Configuration Assistant - Creating the Database

A2.4.16 Oracle Database Configuration Assistant - End of Creating Database

As shown in Figure A36, the end of creating the database is displayed. Wait till the entire setup is completed.

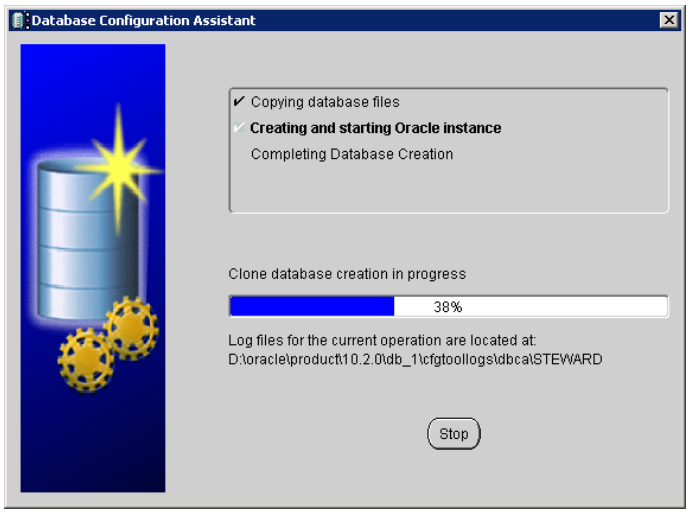


Figure A36. Oracle Database Configuration Assistant - End of Creating Database

A2.4.17 Oracle Database Configuration Assistant -Password Management

After the database is created, the following error as shown in Figure A37 may or may not appear depending on the system configurations. If the error is displayed, the following code has to be run through the command prompt.

D:\oracle\product\10.2.0\db_1\bin\emctl.bat config emkey – remove_from_repos –sysman_pwd trc513

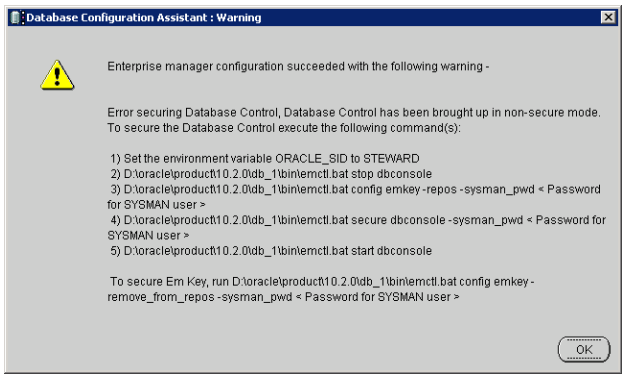


Figure A37. Oracle Database Configuration Assistant – Error while Creating Database

After the code is run at command prompt, the following message screen appears as shown in Figure A38. (Note: the screen capture is from a test STEWARD database installation and not the current STEWARD)

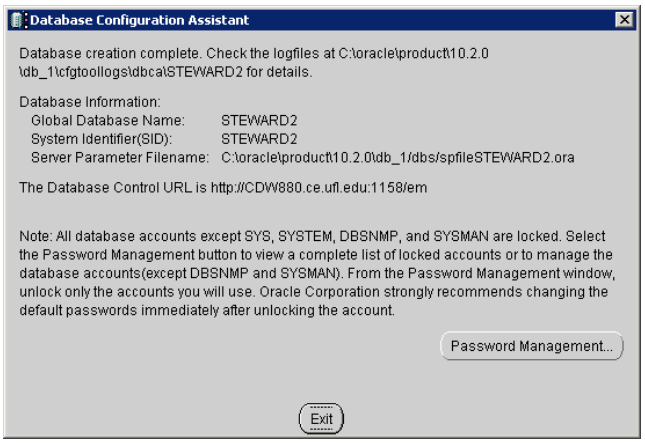


Figure A38. Oracle Database Configuration Assistant - Password Management

The Database Control URL is ‘http://cce-trc-cdwserv.ce.ufl.edu:1158/em’.

When you go to the above URL, the username ‘sys’ and password ‘trc513’ will be used. At this stage, the database has been created. The next step is to configure the Internet connections in the database. These steps are described in the following section.

A2.5 Oracle Net Configuration Assistant

Before setting up the Oracle Warehouse Builder using the Design Center, the Oracle Net Configuration should be checked and verified if it is working. This is done by navigating to the Oracle Net Configuration Assistant from the already installed applications. This is available in Windows machine at:

Windows: Start\All Programs\Oracle-OraDb10g_home1\Configuration and Migration Tools\Net Configuration Assistant as shown in Figure A39.

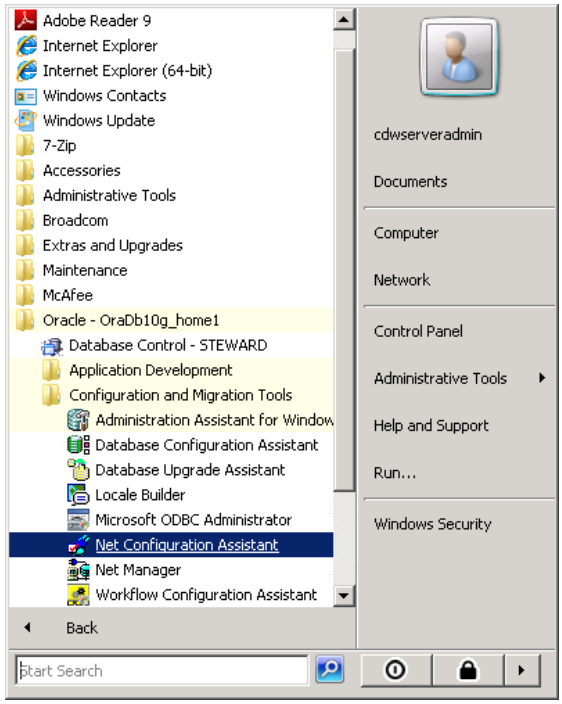


Figure A39. Oracle Net Configuration Assistant - Run the 'Net Configuration Assistant'

A2.5.1 Oracle Net Configuration Assistant - Listener Configuration

As shown in Figure A40, select the 'Listener Configuration' and click 'Next'. This configuration could already have been setup, but it should be checked again.

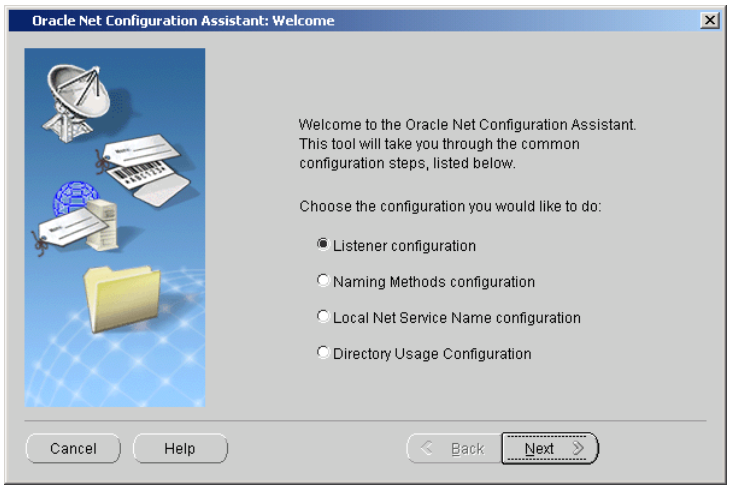


Figure A40. Oracle Net Configuration Assistant - Listener Configuration

As shown in Figure A41, the 'Add' option is the only available option to be selected, which means that there is no other running listener on this server. Click 'Next'.

Figure A41. Oracle Net Configuration Assistant - Listener Configuration

As shown in Figure A42, give a listener name for remote connection to your Oracle database. It is advisable that you provide the default name 'STEWARD' to avoid any problems with the database operations but you may enter the name of any listener you want. After setting the name, click 'Next'.

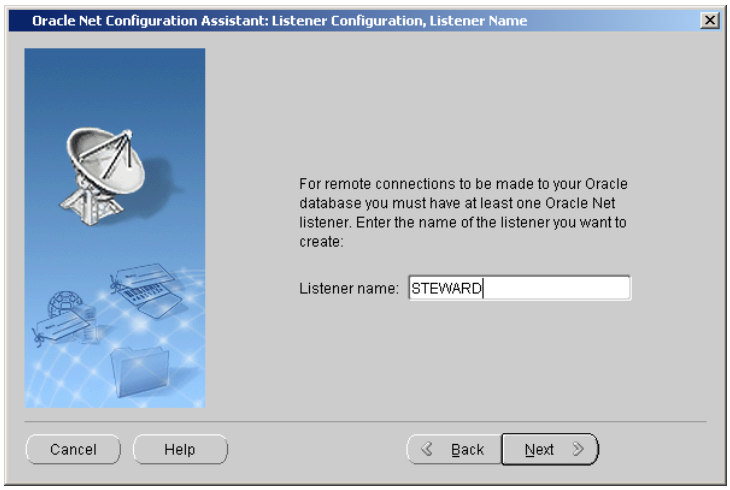


Figure A42. Oracle Net Configuration Assistant - Listener Name

As shown in Figure A43, select the 'TCP' protocol from the list of available protocols. You may also select multiple protocols but it is advisable to use only 'TCP'. After setting the protocols, click 'Next'.

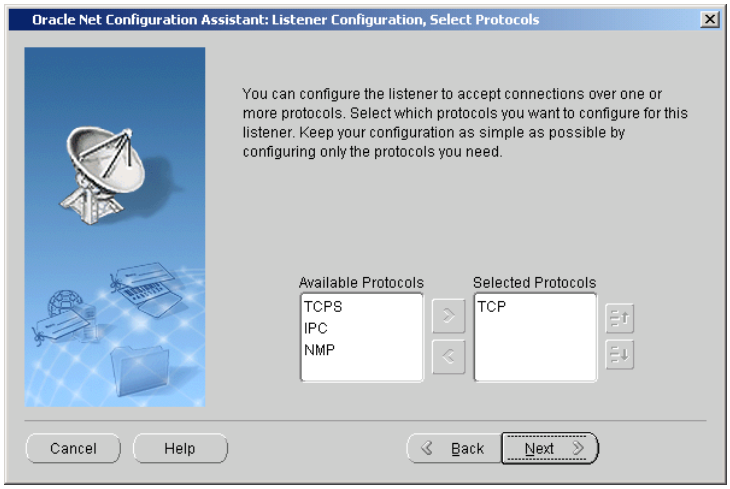


Figure A43. Oracle Net Configuration Assistant - Select Protocols

Under the assumption that the TCP protocol has been selected as shown in Figure A43, the next step is to select the port number for connection. As shown in Figure A44, you can choose the default port number '1521' or you can change the port number to whatever you want. After setting the port number, click 'Next'.

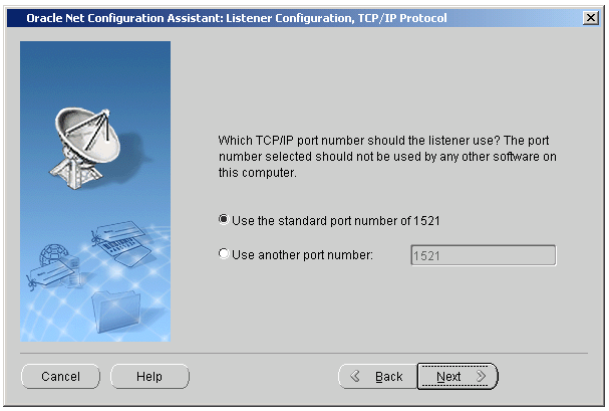


Figure A44. Oracle Net Configuration Assistant - TCP/IP Protocols and Port Number

As shown in Figure A45, the Install-Shield asks if there is another listener to be set up. In the current configuration, you don't need to add another listener. Select 'No' and click 'Next'.

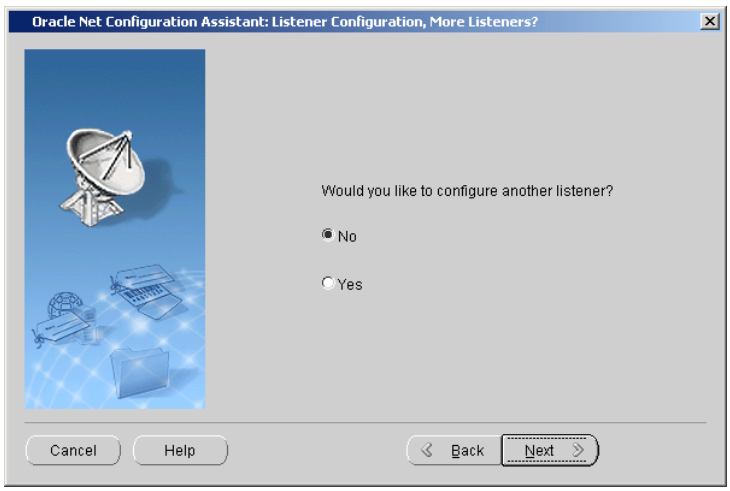


Figure A45. Oracle Net Configuration Assistant - Listener Configuration

As shown in Figure A46, the listener configuration has been completed. Click 'Next' for the next step, i.e. the Oracle Net Configuration Assistant – Local Net Service Name Configuration.

Figure A46. Oracle Net Configuration Assistant - Listener Configuration Done

A2.5.2 Oracle Net Configuration Assistant -Local Net Service Name Configuration

As shown in Figure A47, the local net service name configuration should be configured after completing the setup of the listener configuration. Select the 'Local Net Service Name Configuration' option and click 'Next'.

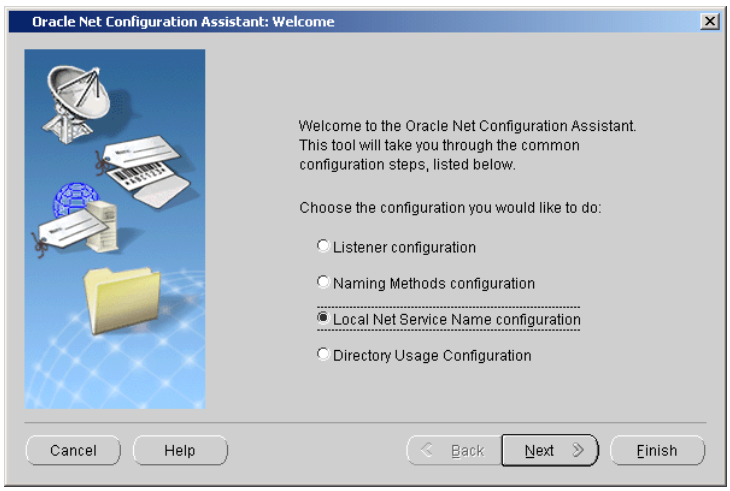


Figure A47. Oracle Net Configuration Assistant -Local Net Service Name Configuration

As shown in Figure A48, select the ‘Add’ menu and click ‘Next’. It is assumed that there is no other service running in the system.

Figure A48. Oracle Net Configuration Assistant -Local Net Service Name Configuration (Add)

As shown in Figure A49, enter ‘STEWARD’ as the service name and click ‘Next’.

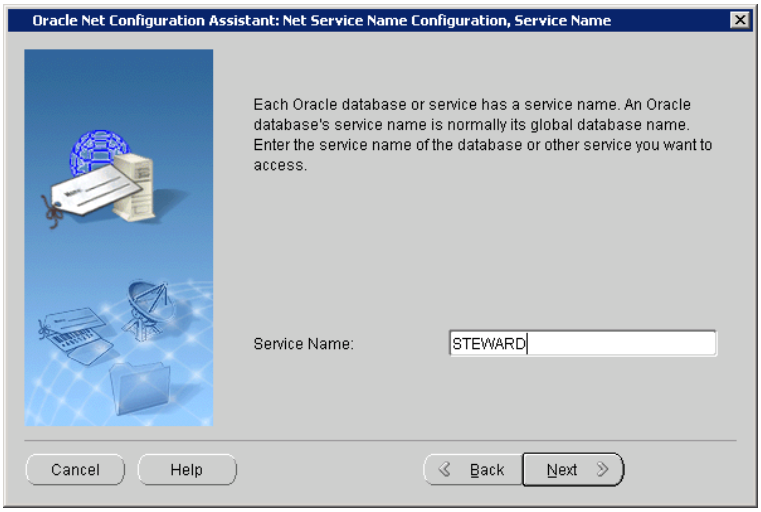


Figure A49. Oracle Net Configuration Assistant - Local Net Service Name Configuration (Service Name)

As shown in Figure A50, select the protocols for communication. Select the 'TCP' and click 'Next'.

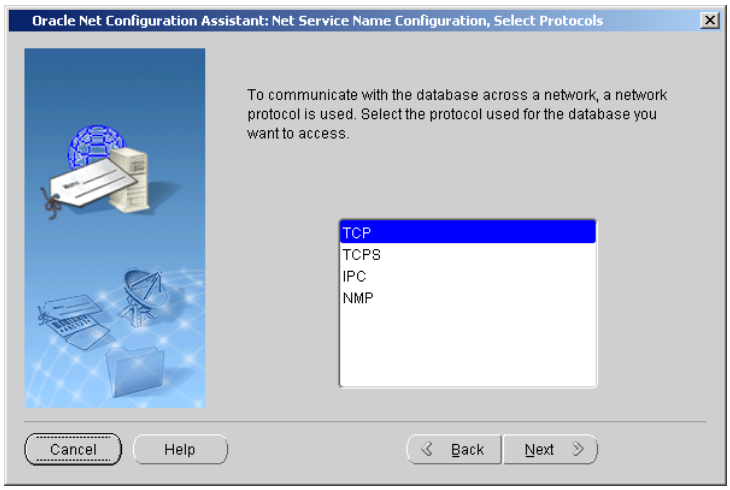


Figure A50. Oracle Net Configuration Assistant - Local Net Service Name Configuration (Select Protocols)

As shown in Figure A51, enter the name of the machine that hosts the STEWARD. In this case, it is 'cce-trc-cdwserv.ce.ufl.edu' and select the port number 1521. Click 'Next'.

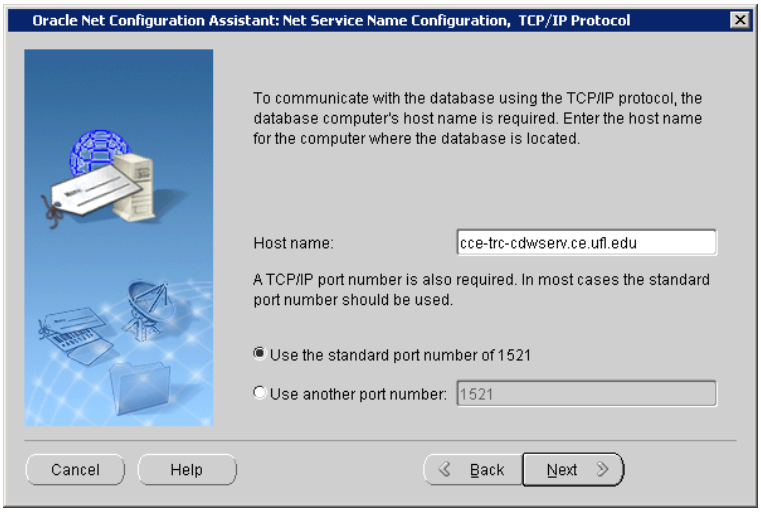


Figure A51. Oracle Net Configuration Assistant - Local Net Service Name Configuration (TCP/IP Protocols)

As shown in Figure A52, select the 'Yes, perform a test' option to test the connection. Click 'Next'.

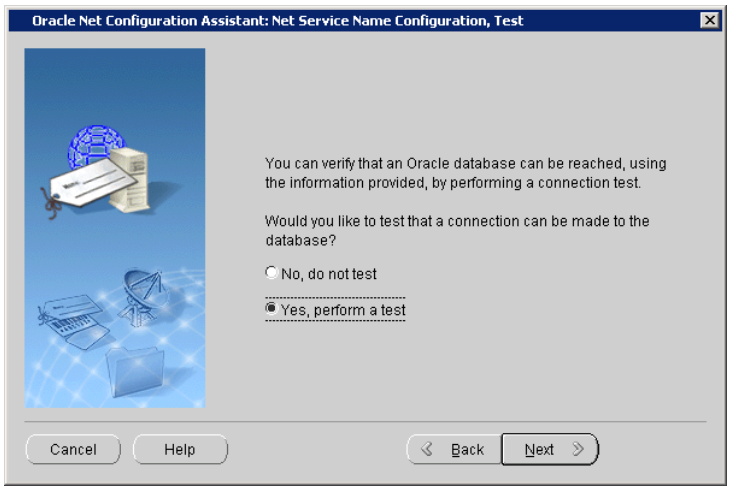


Figure A52. Oracle Net Configuration Assistant - Local Net Service Name Configuration (Connection Test)

As shown in Figure A53, if the test result is failure, select 'Change Login' to test the connection.

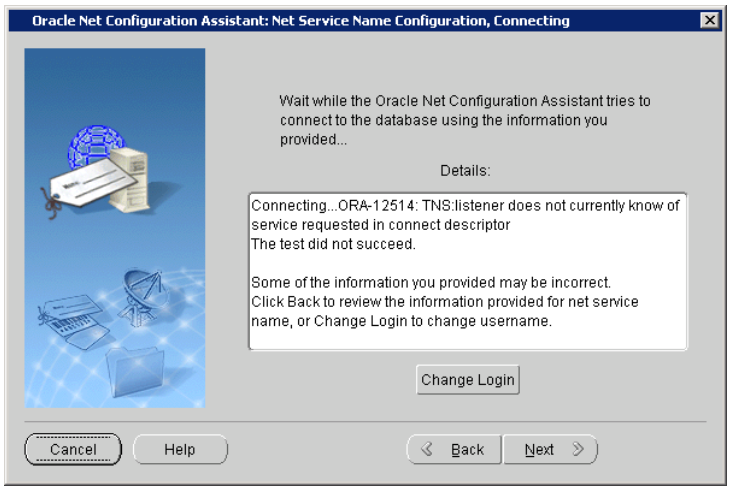


Figure A53. Oracle Net Configuration Assistant - Local Net Service Name Configuration (Failure)

If the test is a failure, as shown in Figure A54, enter ‘sys’ or ‘system’ as ‘Username’ and ‘trc513’ as the password. Click ‘OK’.

Figure A54. Oracle Net Configuration Assistant – Change Login

Check the connection status, as shown in Figure A55. If this test is successful, click ‘Next’.

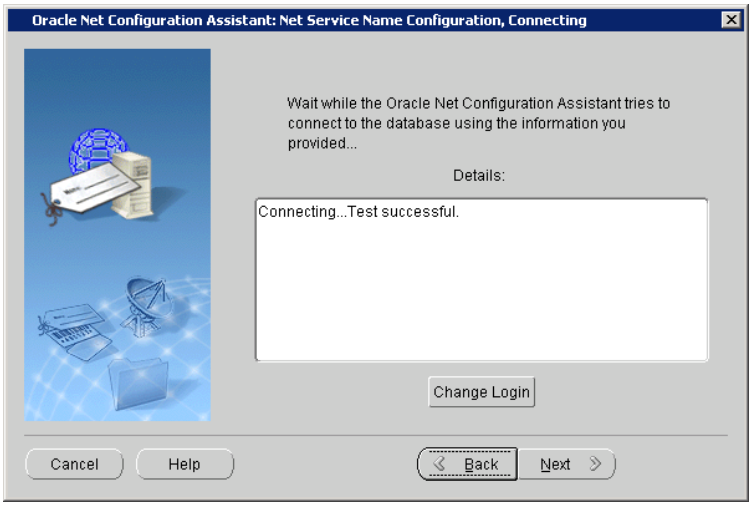


Figure A55. Oracle Net Configuration Assistant - Connection Succeeded

As shown in Figure A56, enter ‘STEWARD’ as ‘Net Service Name’ and click ‘Next’.

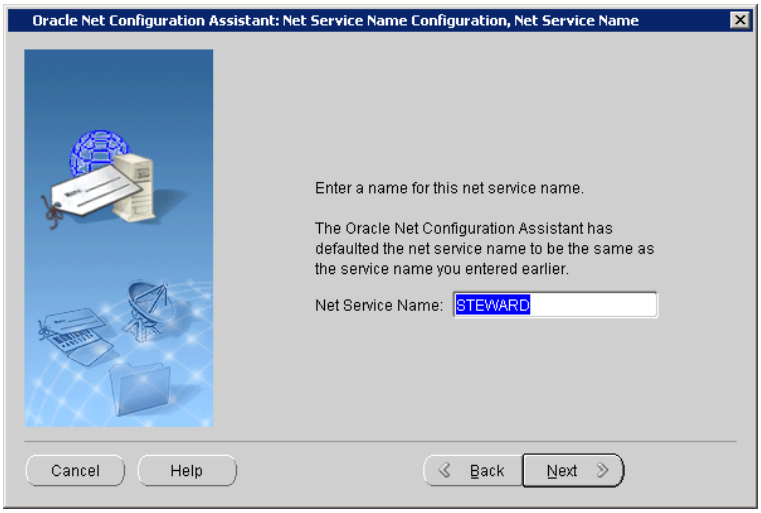


Figure A56. Oracle Net Configuration Assistant - Net Service Name

As shown in Figure A57, select ‘No’ and click ‘Next’.

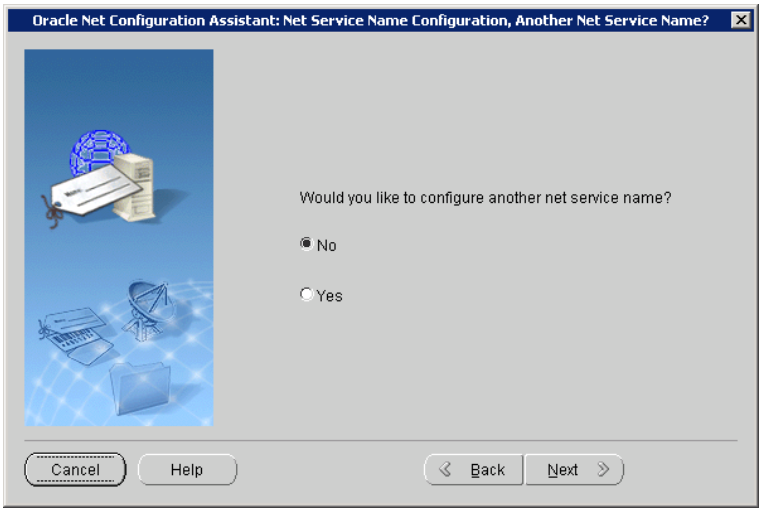


Figure A57. Oracle Net Configuration Assistant - Another Net Service Name?

As shown in Figure A58, click ‘Next’ and then just click ‘Finish’.

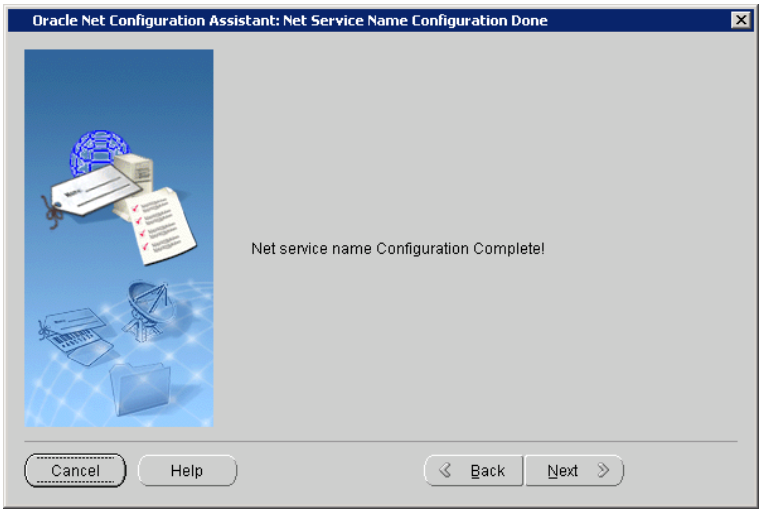


Figure A58. Oracle Net Configuration Assistant - Net Service Name (Configuration Done)

A2.5.3 System Variables Setting

In the Windows system, ORACLE_HOME needs to be defined as a system variable. If this is not performed, the following error message might be generated in next steps of creation of STEWARD.

“ORA-12154: TNS: could not resolve the connect identifier specified.”

This error should be removed by adding ORACLE_HOME as a system variable. This is done by going to the Start menu of Windows to the following location:

Start → Control Panel → System properties

System properties: Advanced → Environmental Variables → System variables

In this window, the system variables are set manually as follows:

Variable: ORACLE_HOME

Value: D:\oracle\product\10.2.0\db_1

A snapshot of this step is provided in Figure A59. After this is completed, the system needs to be restarted.

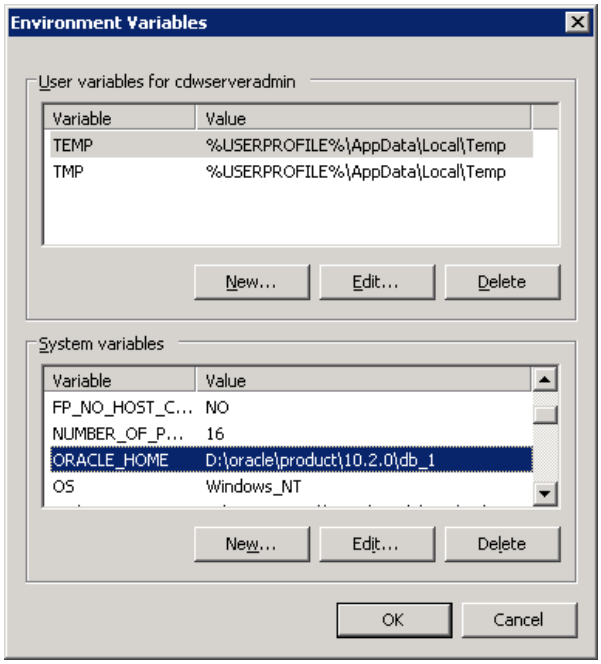


Figure A59. System Variables Setting

A2.6 Oracle Warehouse Builder Setup

To install the OWB, the Design Center needs to be configured first. This is performed by executing the Design Center application under the Warehouse Builder. This is shown in Figure A60.

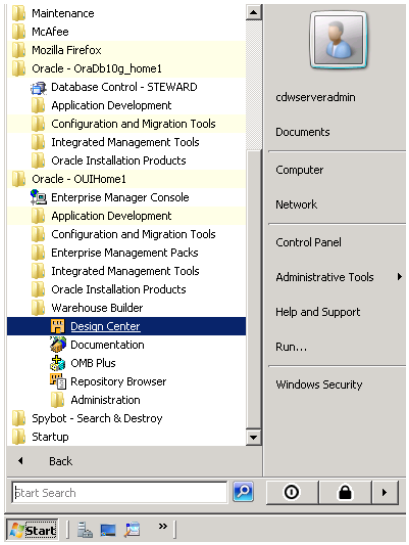


Figure A60. Oracle Warehouse Builder Configuration

A2.6.1 Create the Design Center User and its Repository

The first step is to run the Design Center and to create a "Warehouse Builder User" and "Warehouse Builder Repository." These will be used to archive information for the new data warehouse. As shown in Figure A61, click 'Get Started'. If the *'Get Started'* button is not visible, click 'Show Details' button.

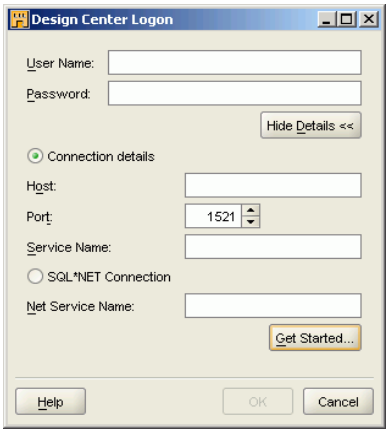


Figure A61. Oracle Warehouse Builder Login Window

As shown in Figure A62, select 'Basic Install' in the *Install Type* window.

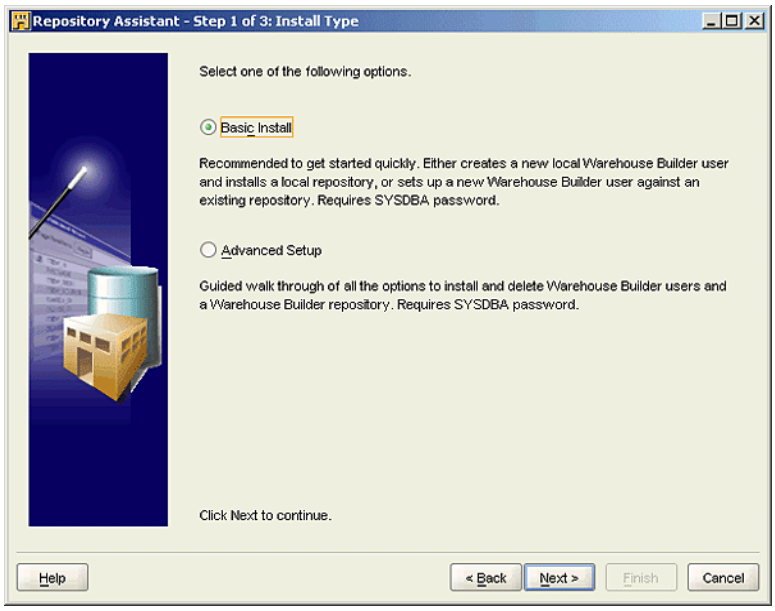


Figure A62. Oracle Warehouse Builder- Installation Type

As shown in Figure A63, create a user named *'steward_user'* and fill the other items as follows in the *Repository User and Connection Information* window.

Repository User Name: **steward_user**

Repository User Password: **trc513**

SYSDBA User Name: **SYS**

SYSDBA Password: **trc513**

Host Name: **LOCALHOST**

Port Number: **1521**

Oracle Service Name: **steward**

After filling these fields, click 'Next'.

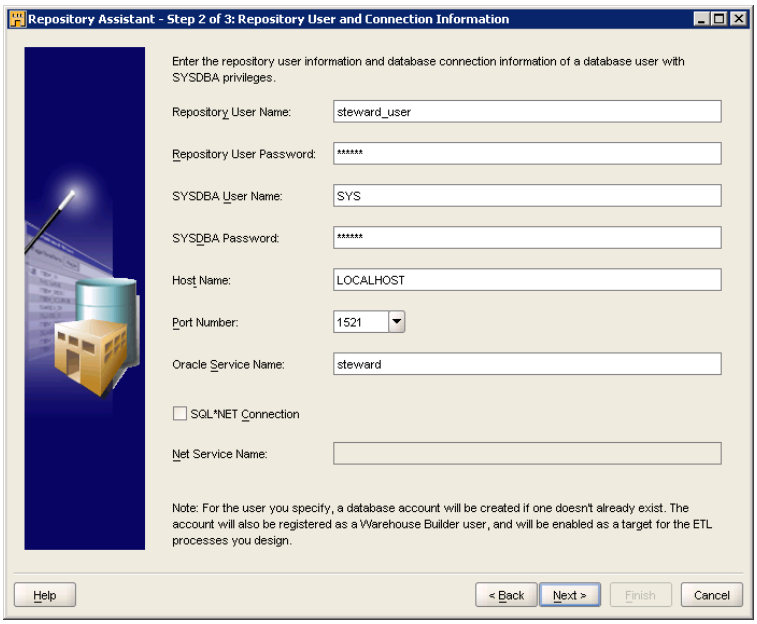


Figure A63. Oracle Warehouse Builder- Repository User and Connection Information

As shown in Figure A64, re-enter the password, **trc513**, for *'steward_user'* in the *Password Confirmation* window.

Figure A64. Oracle Warehouse Builder- Password Confirmation

As shown in Figure A65, type the owner name in the *Repository Owner Information* window. The repository owner is a highly privileged Warehouse Builder user with access to additional security features. Enter '**steward_owner**' as the username and password '**trc513'** and click 'Next'.

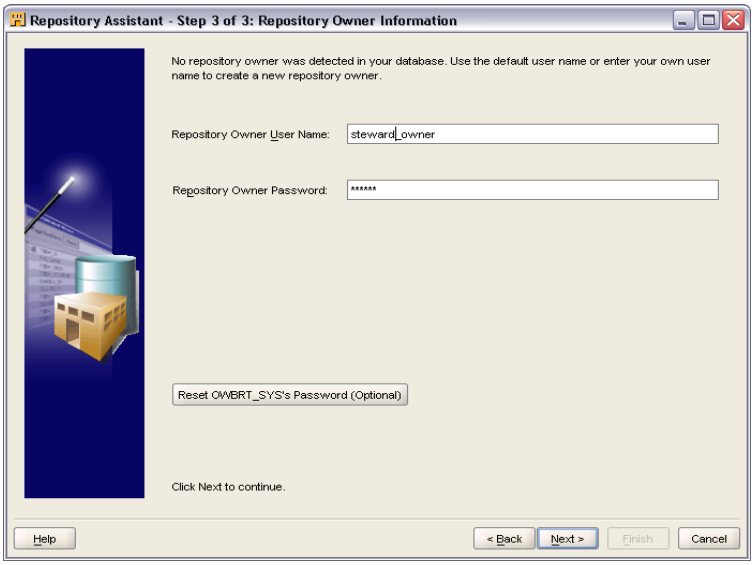


Figure A65. Oracle Warehouse Builder- Repository Owner Information

As shown in Figure A66, an installation summary will appear. Click 'Finish' to start the installation.

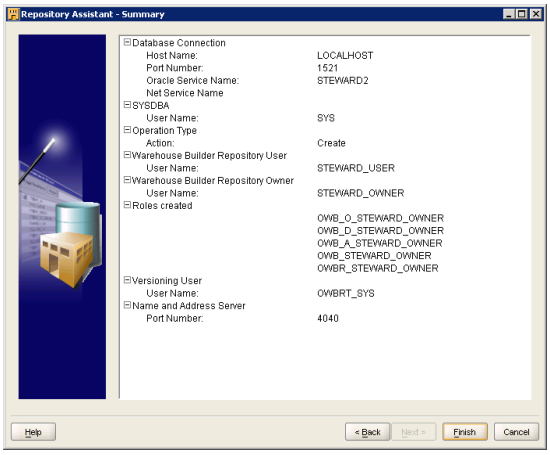


Figure A66. Oracle Warehouse Builder- Installation Summary

The repository installation will take several minutes before it gets completed. A window will appear as shown in Figure A67.

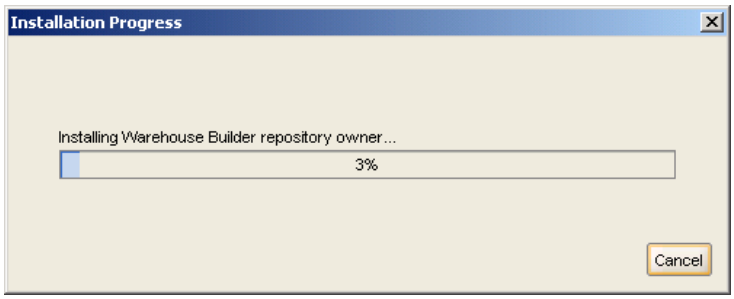


Figure A67. Oracle Warehouse Builder- Installation Progress

An *Installation Successful* message will appear as shown in Figure A68. Click 'Ok' to exit this installation.

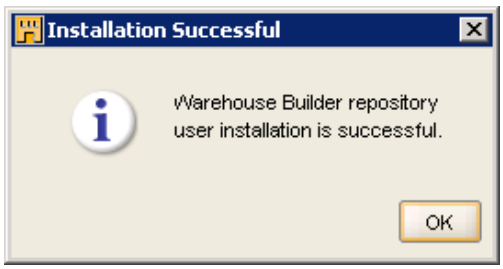


Figure A68. Oracle Warehouse Builder- installation progress

A2.7 Oracle Workflow Configuration

A2.7.1 Create the owf_mgr Workflow Schema using the Workflow Configuration Assistant

The next step is to create the *'owf_mgr'* Workflow Schema using the Workflow Configuration Assistant. This is performed by running the Oracle Workflow Configuration Assistant program as shown in Figure A69 by navigating through the following:

Windows: Start\All Programs\Oracle_Home\Configuration and Migration tools\Workflow Configuration Assistant.

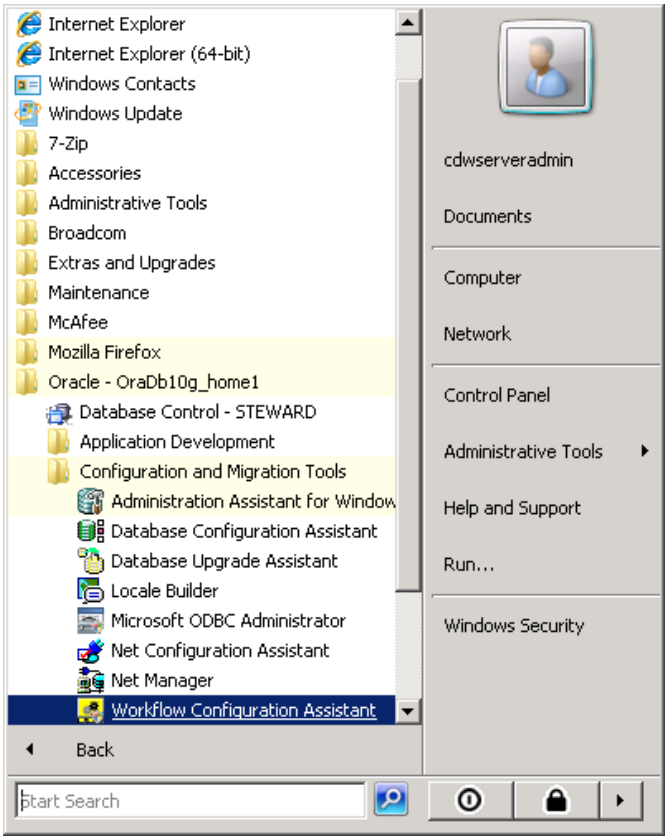


Figure A69. Oracle Workflow Configuration

A2.7.2 Set the Workflow Configuration

To configure the workflow account, enter the information as below in the Workflow Configuration Assistant window as shown in Figure A70 and click 'Submit'.

- Workflow Account: **owf_mgr**
- Workflow Password: **trc513**
- Sys Password: **trc513**
- TNS Connect Descriptor: **localhost:1521:steward**

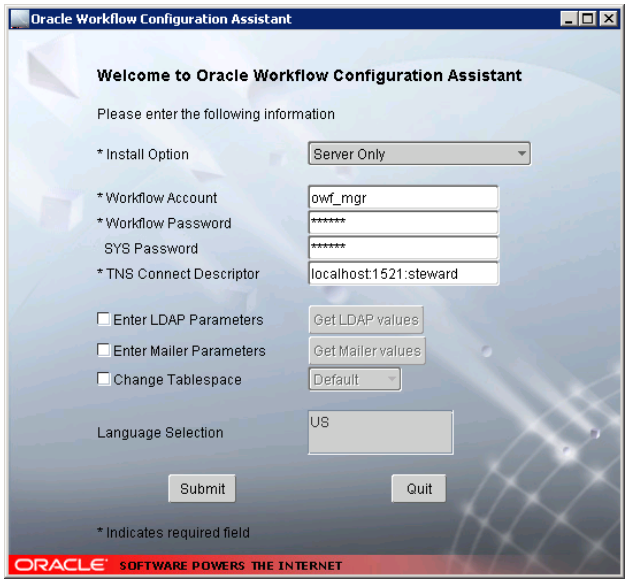


Figure A70. Oracle Workflow Configuration Assistant – Account Window

As shown in Figure A71, the Workflow Configuration Assistant account will be configured. This window will close after a while.

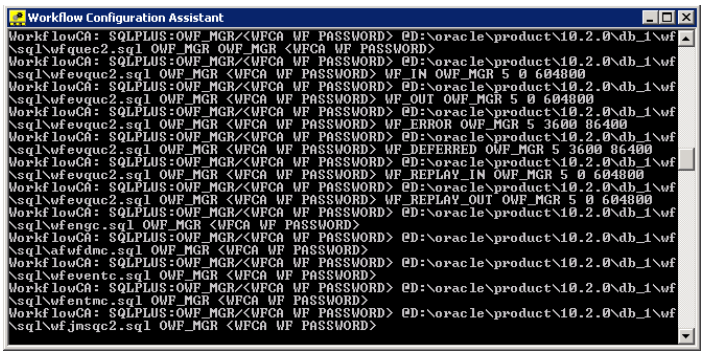


Figure A71. Oracle Workflow Configuration Assistant – Installation Process

A2.7.3 Unlock the owf_mgr Account

After the Workflow Configuration Assistant has completed successfully, the owf_mgr account is unlocked. This is done through the SQL Plus application as shown in Figure A72.

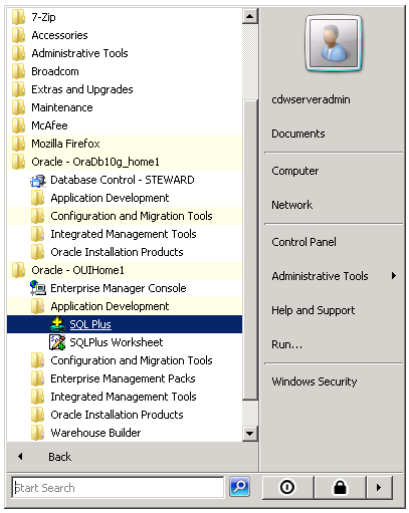


Figure A72. Oracle workflow configuration – SQL Plus

Log into SQL Plus as a SYSDBA with

user id: **system**
password: **trc513**
host/service name: **steward**

After login, execute the following commands as shown in Figure A73.

- alter user owf_mgr account unlock;
- grant execute any procedure to owf_mgr;
- grant owb_o_steward_owner to owf_mgr;

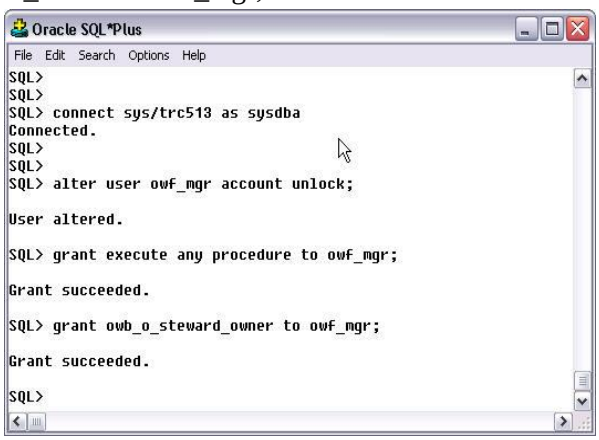


Figure A73. Oracle workflow configuration - Unlock the owf_mgr

A3 STEWARD Deployment

A3.1 First Step – Login to Design Center

The first in step in STEWARD deployment is to login at the OWB Design Center with user id 'steward_owner' (since all the tasks are performed under 'steward_owner' authority). As shown in Figure A74, login to the Design Center with the following credentials:

- User Name: **steward_owner**
- Password: **trc513**
- Host: **cce-trc-cdwserv.ce.ufl.edu**
- Service Name: **steward**

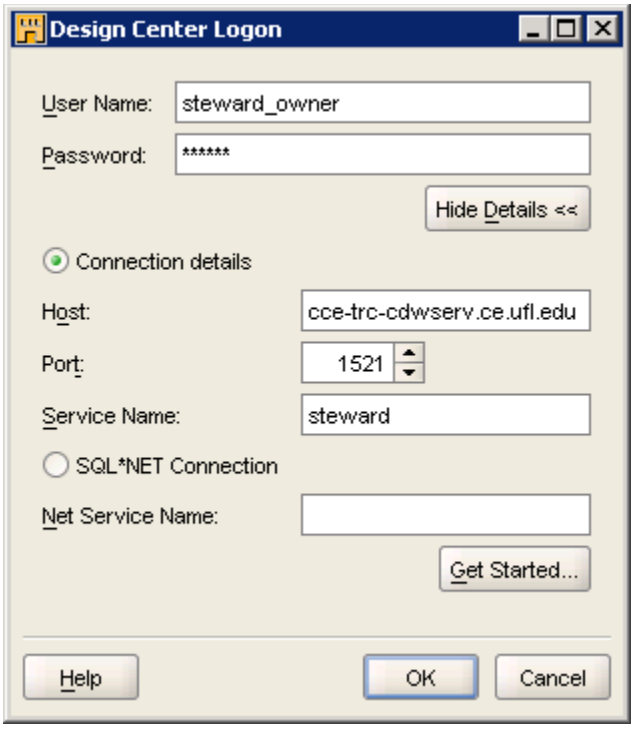


Figure A74. Oracle Warehouse Builder – Design Center Login

A3.2 Prerequisites- Create a Target User

The target schema in the Oracle Warehouse Builder is a target where the traffic data is loaded and data objects like cubes, dimensions, views, or mappings are created. This target schema stores the data objects physically after the deployment process. The target module consists of target schema and gives reference to the target schema by an assigned location on a system. Every target module must be mapped to a target user schema. To create the STEWARD target schema users, perform the following steps.

In the Design Center, expand the Security node of the Global Explorer panel and select the 'Users' node by right-click and then select 'New' as shown in Figure A75.

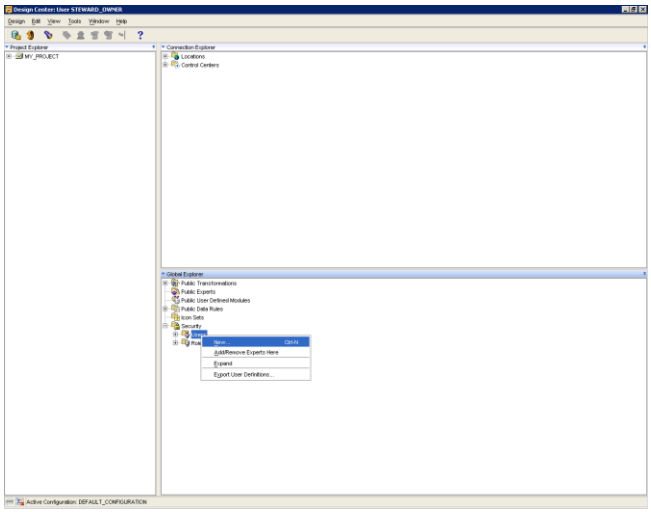


Figure A75. Oracle Warehouse Builder – Select DB User

As shown in Figure A76, click the Create DB User.

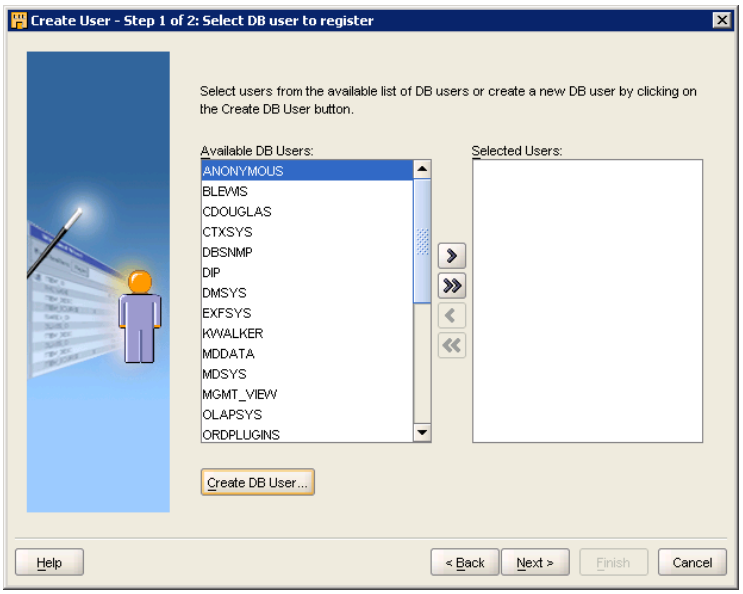


Figure A76. Oracle Warehouse Builder - Create DB User

The window ‘Create Database User’ appears. As shown in Figure A77, enter the information for the new DB user as below and click ‘Ok’.

Name: **Gator**

Password: **trc513**

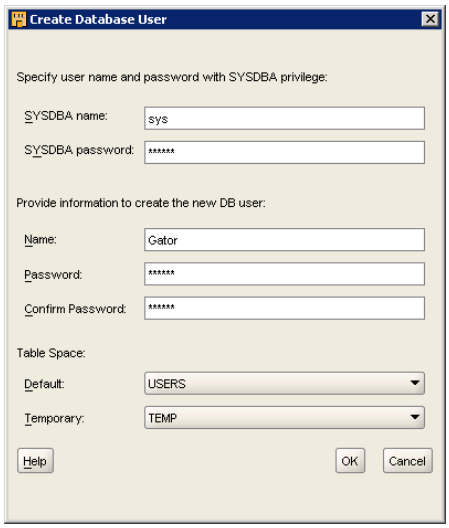


Figure A77. Oracle Warehouse Builder - Create Database User Window

After the user is created, check the user ‘Gator’ as the target schema and click ‘Next’ as shown in Figure A78.

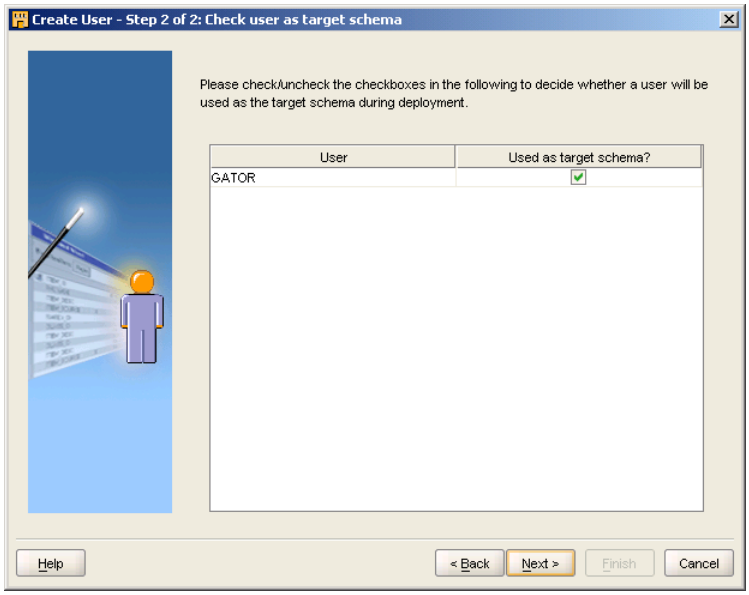


Figure A78. Oracle Warehouse Builder - Check User as Target Schema

Enter the password ‘trc513’ when prompted as shown in Figure A79 and click ‘OK’.

Figure A79. Oracle Warehouse Builder - Target User Password

As shown Figure A80, the “create user” procedure is done. Click ‘Finish’.

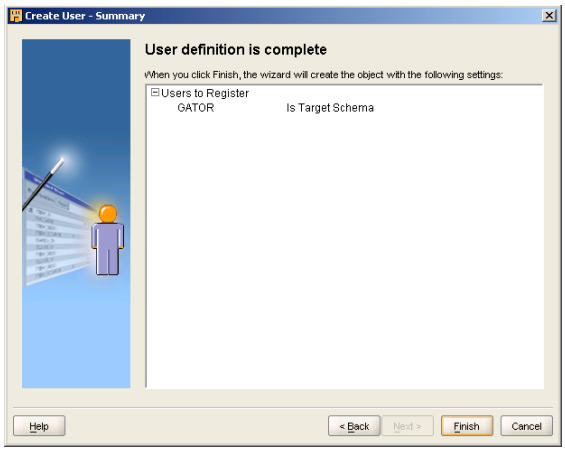


Figure A80. Oracle Warehouse Builder - Check User Summary

As shown Figure A81, the ‘Register User’ procedure is completed in few minutes.

Figure A81. Oracle Warehouse Builder - Register Users Progress

After the ‘Gator’ user is registered, expand the Oracle Locations in the Connection Explorer in the OWB Design Center. Verify that the ‘STEWARD_USER_LOCATION’ has been added. This is shown below in Figure A82.

The verification is done by checking the STEWARD user location at:

Connection Explorer\Locations\Databases\Oracle

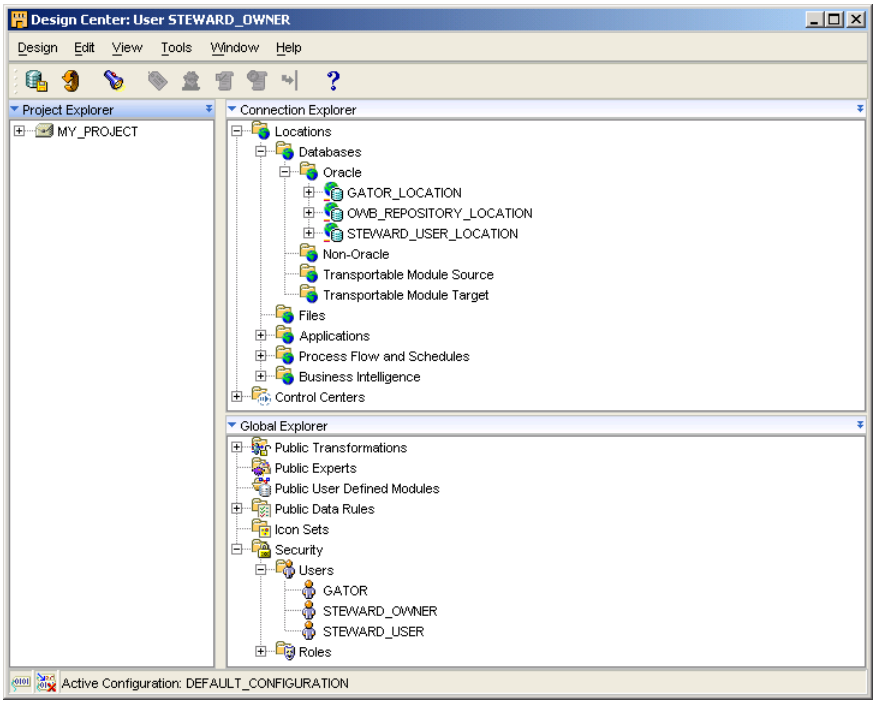


Figure A82. Oracle Warehouse Builder - Connection Explorer

After verifying the ‘STEWARD_USER_LOCATION’, it is observed that ‘GATOR_LOCATION’ is also created while registering Gator as a user. This should be deleted from the Locations before the metadata is imported to this new system. Navigate to the Connection Explorer\Locations\Databases\Oracle and right click on ‘GATOR_LOCATION’ to delete it as shown in Figure A83.

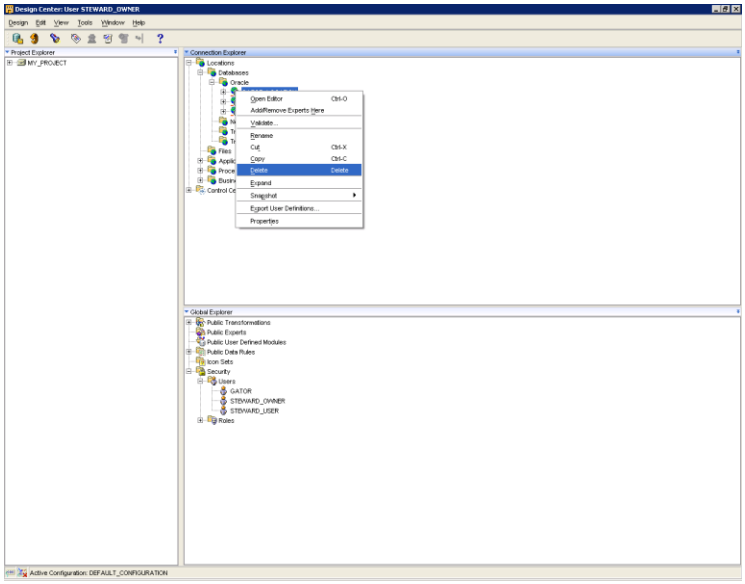


Figure A83. Oracle Warehouse Builder - Delete Gator_Location

As shown in Figure A84, a warning message will appear to confirm the deletion of this location. Click ‘Ok’ and continue to importing metadata.

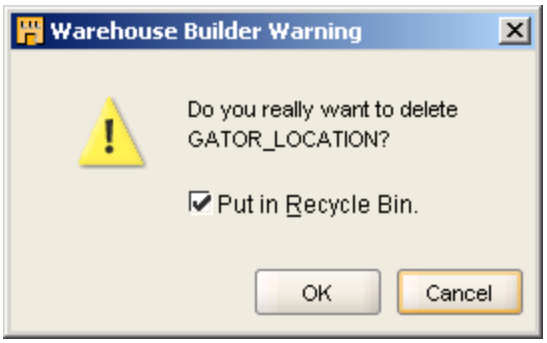


Figure A84. Oracle Warehouse Builder – Confirm Deleting Gator_Location

A3.3 Prerequisites- Create New Files

The files in the Oracle Warehouse Builder are used to load the external files into the database. The new STEWARD system would require external file locations for loading the traffic data files. Hence, for each traffic data file, a 'new' file is created with the help of Oracle OWB. As the STEWARD system is replicated, these files need to be defined before the STEWARD schemas for a respective file is moved/imported to the new system.

To replicate the STEWARD system, the following folders are created:

- D:\Steward
- D:\Steward\Scripts
- D:\Steward\TSS_FacilityData
- D:\Steward\TSS_LANE_DATA
- D:\Steward\TSS_STATION_DATA
- D:\Steward\TSS_STATION_DATA\TSS_5Min_Data
- D:\Steward\TSS_STATION_DATA\TSS_15Min_Data
- D:\Steward\TSS_STATION_DATA\TSS_1HR_Data
- D:\Steward\TSS_LANE_DATA\TSS_5Min_Data
- D:\Steward\TSS_LANE_DATA\TSS_15Min_Data
- D:\Steward\TSS_LANE_DATA\TSS_1HR_Data

Since the data format for all the traffic data files, including the station, lane, facility are already pre-defined, it is necessary that a sample file is kept at the respective file folder locations, so that the file structure could be easily created. The next step is to create these 'new' files in the OWB. These are listed as below:

1) TIME_REF
2) TSS_15MIN_DATA
3) TSS_15MIN_LANE_DATA
4) TSS_1HR_DATA
5) TSS_1HR_LANE_DATA
6) TSS_5MIN_DATA
7) TSS_5MIN_LANE_DATA
8) TSS_ETL_REPORTS
9) TSS_FACILITY

Sections, A3.3.1 to A3.3.18 gives an example on creating a new file for TSS_15Min_Data. For all the other file names, the same steps have to be repeated again.

A3.3.1 Oracle Warehouse Builder - Create New File

From the Design Center, go to the Project Explorer pane, navigate to Files and right click on it to select 'New' as shown in Figure A85.

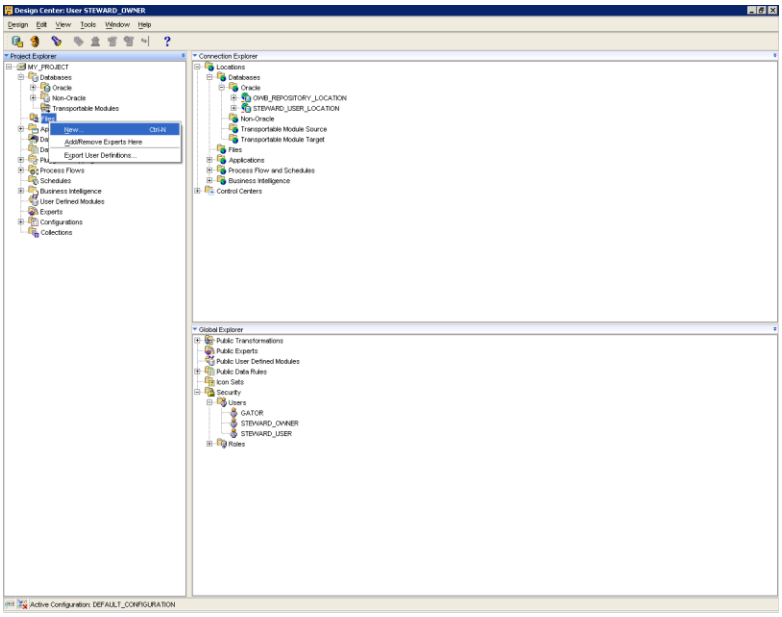


Figure A85. Oracle Warehouse Builder - Create New File

A3.3.2 Oracle Warehouse Builder - Create Module Wizard

As shown in Figure A86, the *create module wizard* will appear. Click 'Next' to continue.

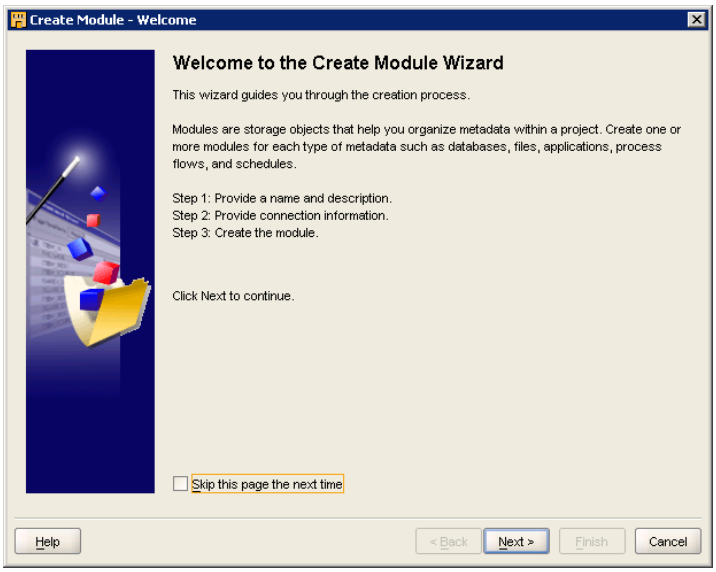


Figure A86. Oracle Warehouse Builder - Create Module Wizard

A3.3.3 Oracle Warehouse Builder - Name and Description

As shown in Figure A87, add the name of the new file "TSS_15MIN_DATA" and if required, the description of this new module.

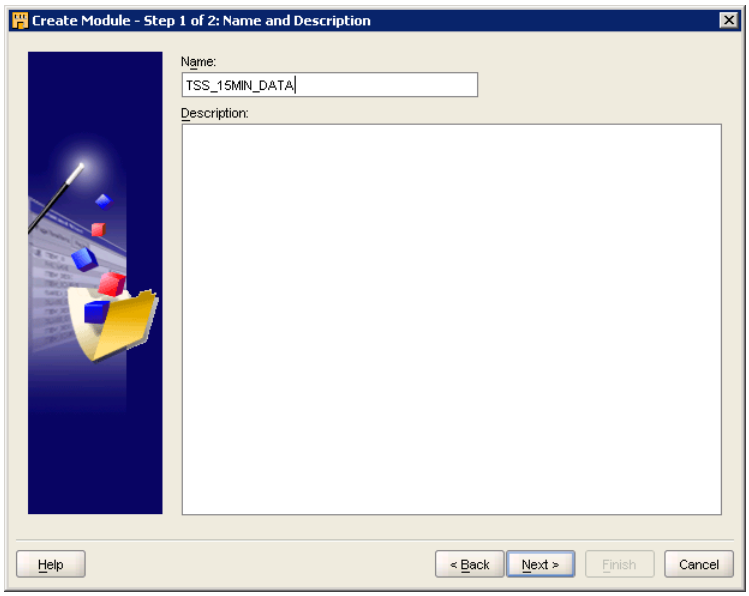


Figure A87. Oracle Warehouse Builder – Create Module, Name and Description

A3.3.4 Oracle Warehouse Builder - Select Location

As shown in Figure A88, select the location for the new file and click 'Edit' to continue. By default the location for a new file would be the name of the file followed by 'Location1'. Click 'Next' to continue.

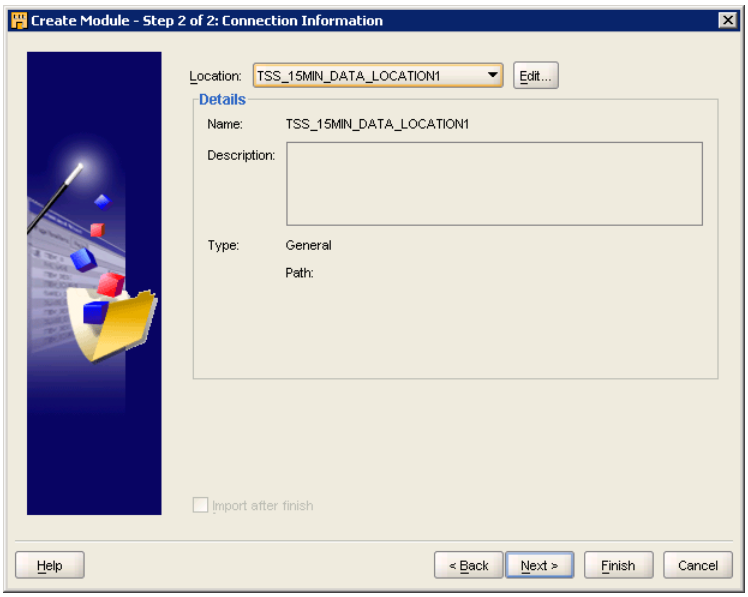


Figure A88. Oracle Warehouse Builder - Connection Information

A3.3.5 Oracle Warehouse Builder - Select the Location Path

As shown in Figure A89, click 'Browse' and select the path of the folder that was created in Section A3.3. While creating all the files, the respective folder location should be selected. After this is selected, click 'Ok' to continue.

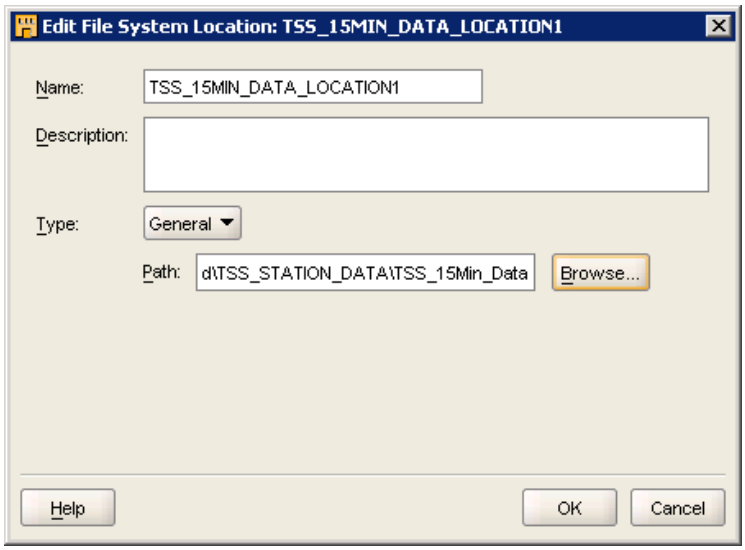


Figure A89. Oracle Warehouse Builder - Select Location Path

A3.3.6 Oracle Warehouse Builder - Finish Connection Information

As shown in Figure A90, the connection information is verified. Click 'Next' to continue. (Note: The path in Figure A86 is an example and should be D:\ for the current installation)

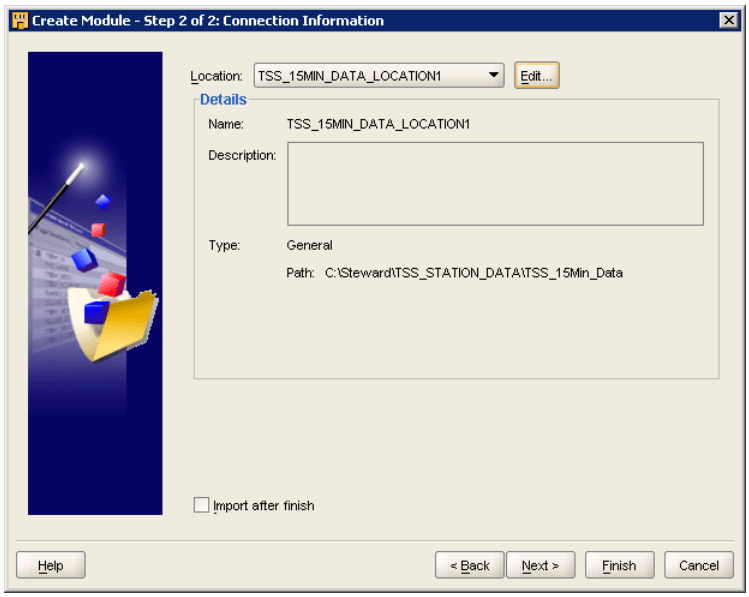


Figure A90. Oracle Warehouse Builder - Finish Connection Information

A3.3.7 Oracle Warehouse Builder - Module is Created

As shown in Figure A91, the module/file creation is finished.

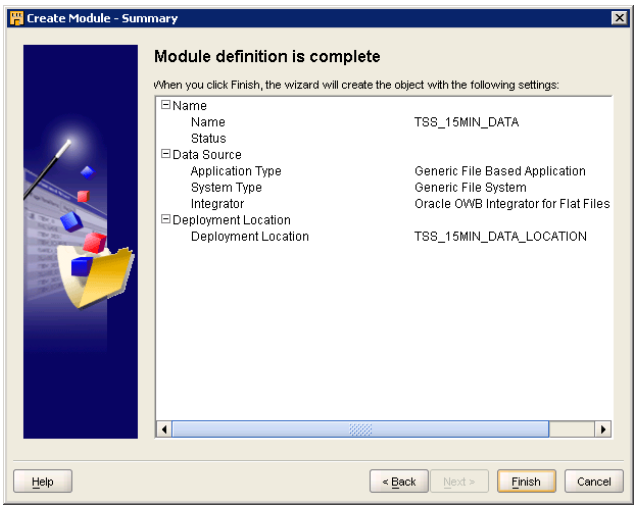


Figure A91. Oracle Warehouse Builder - Finishing Module Creation

After the file creation is finished, the next step is to import the metadata for these files. This is done by placing the sample pre-defined files in comma separated format (.csv) in the respective folders created in Section A3.3. The pre-defined files (see below in **BOLD**) are stored at these locations:

- *TIME_REF:* D:\Steward\TSS_FacilityData**time.txt**
- *TSS_15MIN_DATA:* D:\Steward\TSS_STATION_DATA\TSS_15Min_Data**tss_15min_data.csv**
- *TSS_15MIN_LANE_DATA:* D:\Steward\TSS_LANE_DATA\TSS_15Min_Data**tss_lane_data_15L.csv**
- *TSS_1HR_DATA:* D:\Steward\TSS_STATION_DATA\TSS_1HR_Data**tss_1hr_data.csv**
- *TSS_1HR_LANE_DATA:* D:\Steward\TSS_LANE_DATA\TSS_1HR_Data**tss_lane_data_60L_lane.csv**
- *TSS_5MIN_DATA:* D:\Steward\TSS_STATION_DATA\TSS_5Min_Data**tss_5min_data.csv**
- *TSS_5MIN_LANE_DATA:* D:\Steward\TSS_LANE_DATA\TSS_5Min_Data**tss_lane_data_5L.csv**
- *TSS_ETL_REPORTS:* D:\Steward\TSS_STATION_DATA**ETL_Reports.csv**
- *TSS_FACILITY:* D:\Steward\TSS_FacilityData**TSSStationData.csv**

The placement of these files becomes critical for importing metadata. It should be noted that it is assumed that the STEWARD moderator/user is aware of the format of this files. However, the original structure could also be edited and modified. This could be performed at this stage or at a later stage. The following sections provide the step by step information on metadata import.

A3.3.8 Oracle Warehouse Builder – Importing from the New Module

After all the files are created, right click the file 'TSS_15MIN_DATA' and select *Import*.

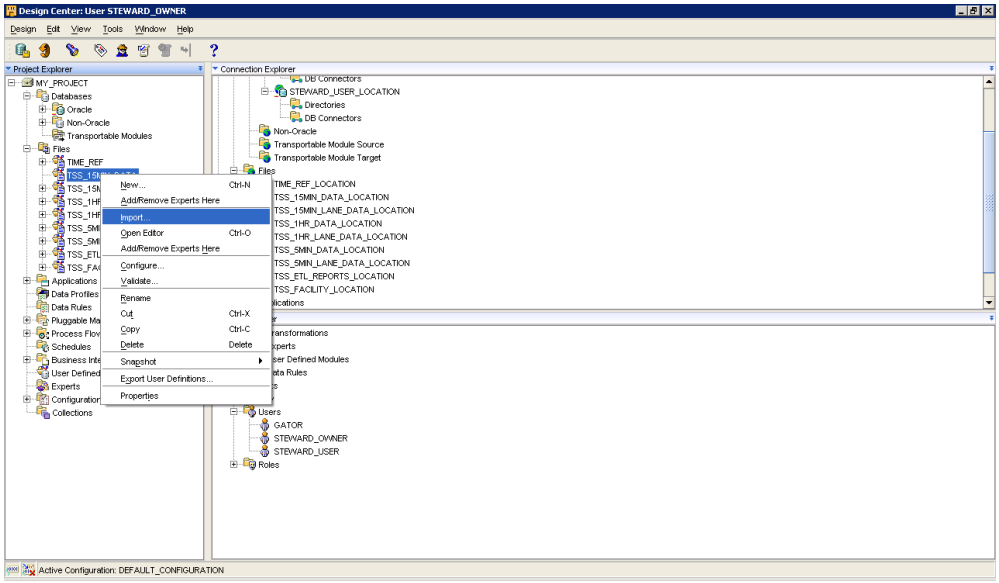


Figure A92. Oracle Warehouse Builder - Importing Files

A3.3.9 Oracle Warehouse Builder - Import Metadata Wizard

As shown in Figure A93, the Import Metadata Wizard will open. Click 'Next' to continue.

Figure A93. Oracle Warehouse Builder - Import Metadata Wizard

A3.3.10 Oracle Warehouse Builder - Filter Information

As shown in Figure A94, select '*All Data Fields*' for the filter information. Click 'Next' to continue.

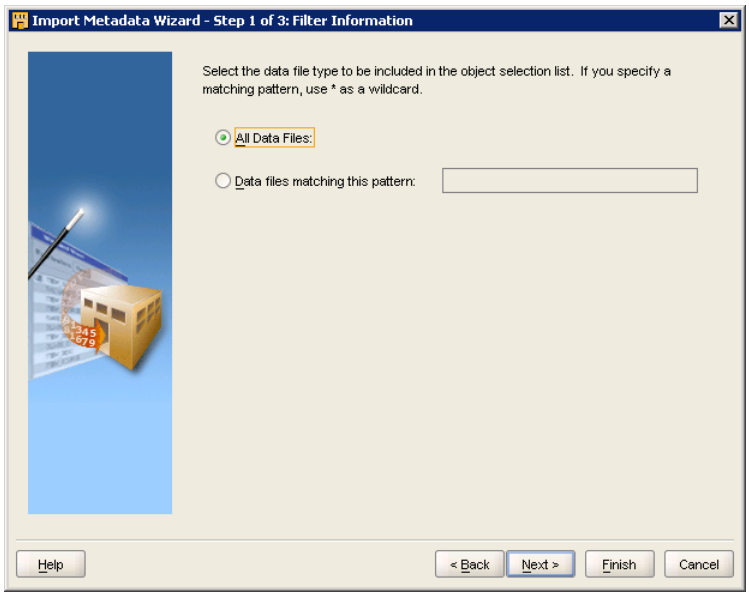


Figure A94. Oracle Warehouse Builder - Filter Information

A3.3.11 Oracle Warehouse Builder - Object Selection

As shown in Figure A95, select the pre-defined file on the '*Selected*' section and click 'Next' to continue. Since the pre-defined files are kept at the location as specified in Section A3.3.7, all the respective files would appear here.

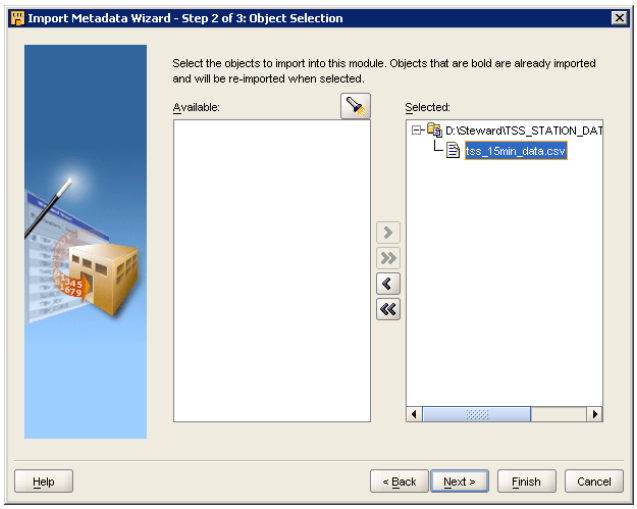


Figure A95. Oracle Warehouse Builder - Select the Right Object for the Module

A3.3.12 Oracle Warehouse Builder - Run the Sample

As shown in Figure A96, after the file is selected from previous step, click the 'Sample' option to continue. This will enable the user to configure the structure and properties of the pre-defined files.

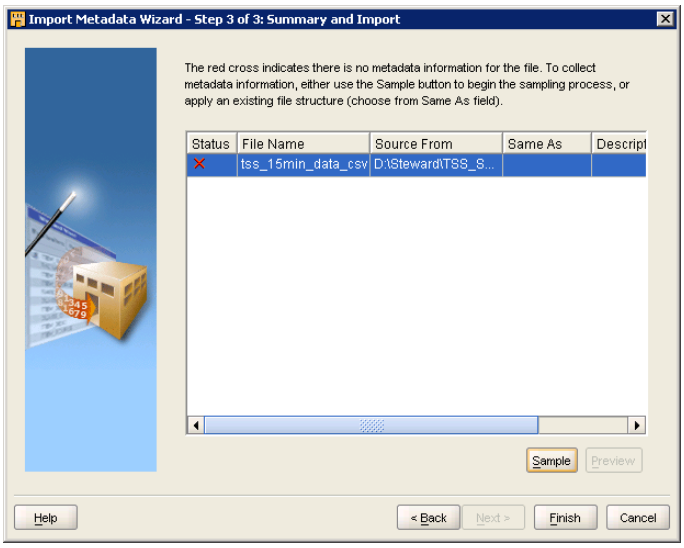


Figure A96. Oracle Warehouse Builder - Run the Sample

A3.3.13 Oracle Warehouse Builder - Flat File Sample Wizard

As shown in Figure A97, run the Flat File Sample Wizard by clicking the 'Next' option.

Figure A97. Oracle Warehouse Builder - Flat File Sample Wizard

A3.3.14 Oracle Warehouse Builder - Check the File Name

As shown in Figure A98, the name of the file is verified. From this window, the user should be able to see the sample of the pre-defined file. This should also be verified at this point.

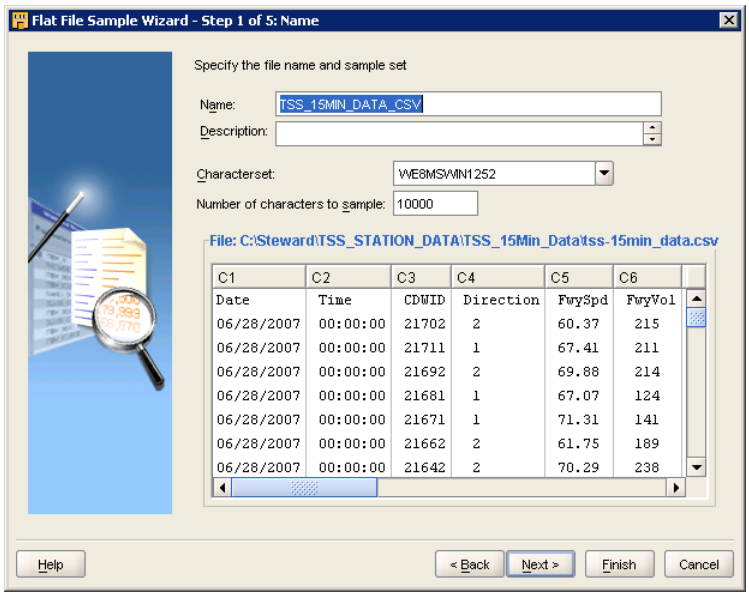


Figure A98. Oracle Warehouse Builder - Check the File Name

A3.3.15 Oracle Warehouse Builder - Record Organization, File Format and File Layout

As shown in Figure A99, the record organization is displayed. The default options are advisable to be used. Click 'Next' to continue.

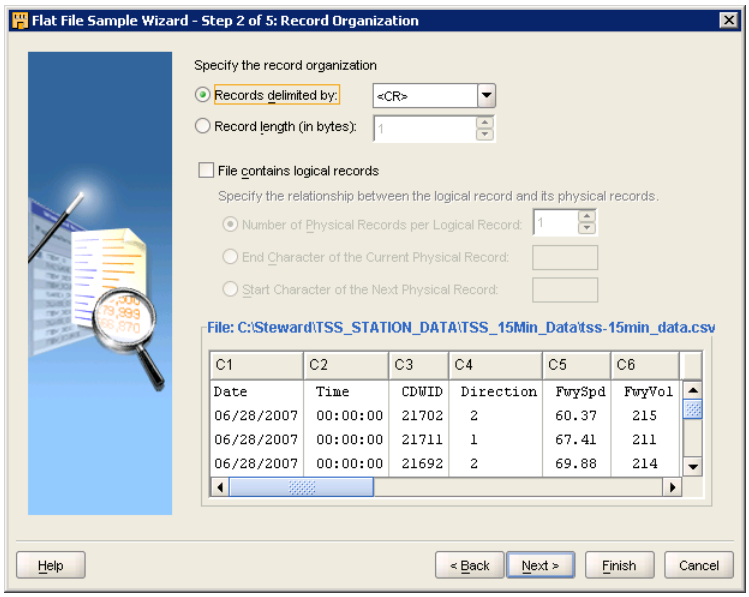


Figure A99. Oracle Warehouse Builder - Report Organization

As shown in Figure A100, select the file format option as delimited and click ‘Next’ to continue.

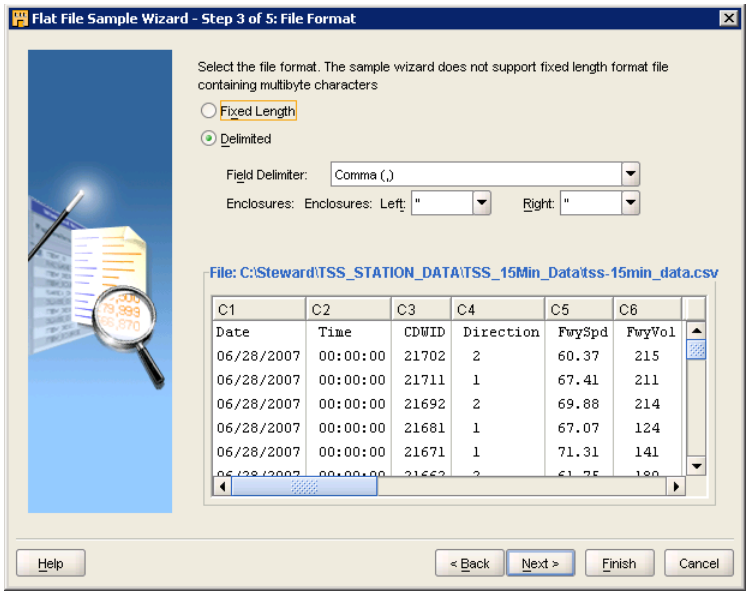


Figure A100. Oracle Warehouse Builder - File Format

As shown in Figure A101, select the file layout and specify ‘Single Record’ format option as delimited and click ‘Next’ to continue.

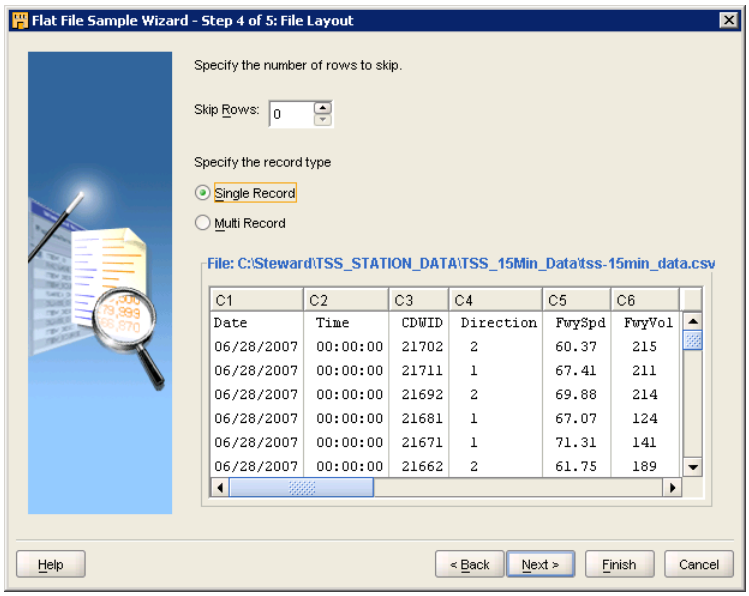


Figure A101. Oracle Warehouse Builder - File Layout

A3.3.16 Oracle Warehouse Builder - Field Properties

As shown in Figure A102, this step sets the field properties. The field *Name*, *Type* and *Length* need to be updated for all the columns as shown in Figure A103. Also, select the "*Use the first record as the field names*" option before clicking 'Next'. The type of all the names should be carefully selected as this information becomes very valuable when the tables are created in the database.

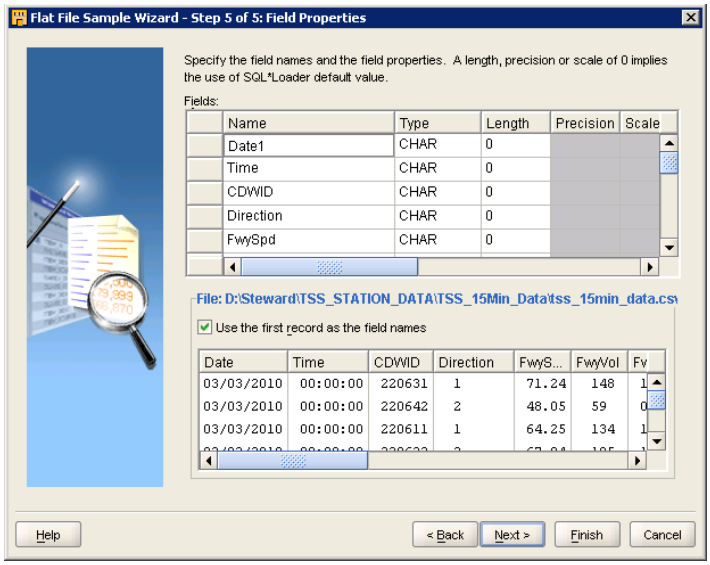


Figure A102. Oracle Warehouse Builder - Field Properties

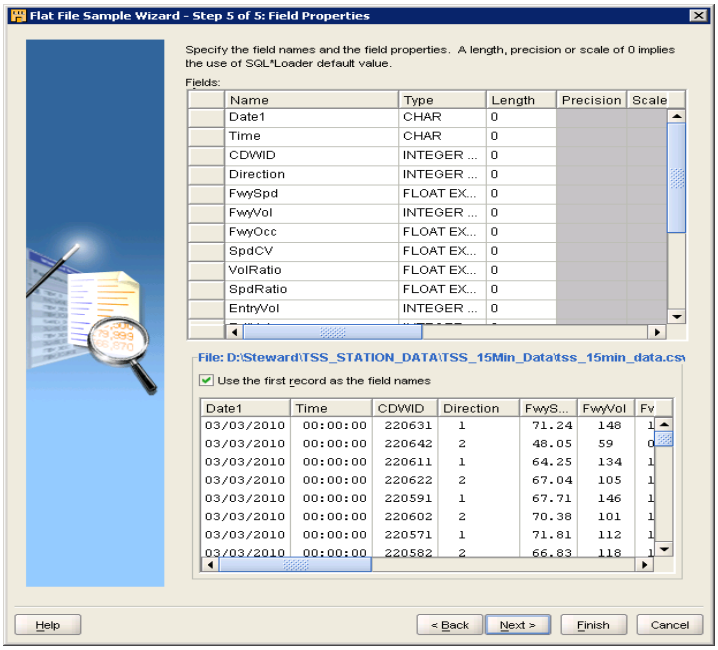


Figure A103. Oracle Warehouse Builder - Updated Field Properties

A3.3.17 Oracle Warehouse Builder - Summary

As shown in Figure A104, verify the name and type of the file in the Flat File definition. Click 'Finish' to continue.

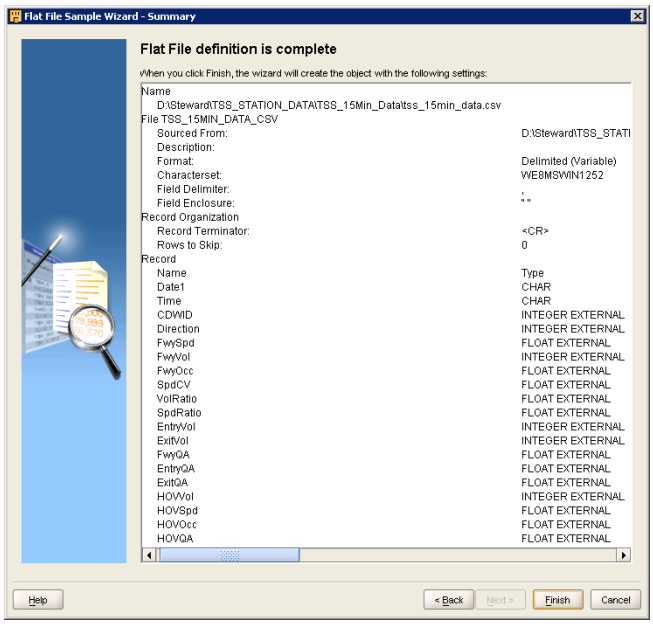


Figure A104. Oracle Warehouse Builder – Flat File Summary

A3.3.18 Oracle Warehouse Builder - Import Data Wizard Summary

As shown in Figure A105, the Import Data Wizard is performed for the file. Verify the status as compared to Figure A92. Click 'Finish' to exit the window. All the files should be created as this stage. The next step is to import the metadata from the previous STEWARD system.

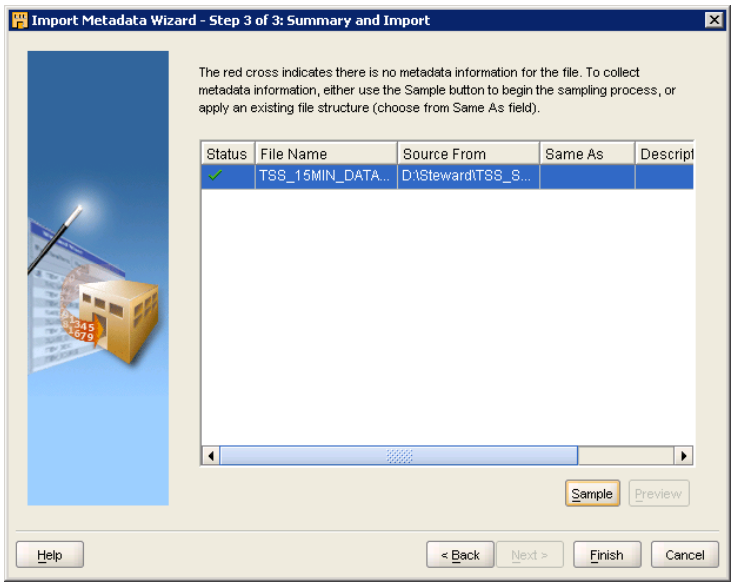


Figure A105. Oracle Warehouse Builder - Import Data Wizard Summary

A3.4 Importing Metadata

The CASE tools such as Oracle Designer and Computer Associates "ERwin" generally provides a metadata repository to store definitions for the database structure and information. In the Oracle Warehouse Builder 10g, the CASE tool for the design of business intelligence systems also provides a metadata store.

The metadata in the Oracle Warehouse Builder includes information about the entire dataset and it can be exported and imported using the Metadata Loader (MDL) utility. For the STEWARD system migration, the STEWARD project metadata needs to be exported from current STEWARD system and imported into the new STEWARD system during the setting up of the new system.

A3.4.1 Prerequisite- Export Warehouse Builder Metadata

The first step before importing the metadata to a new system is to export the metadata from the current STEWARD which is at the Transportation Research Center, University of Florida. As shown in Figure A106, the *Export Warehouse Builder Metadata* process is run from the Design Center to export the metadata. The metadata includes the structure of all the files and the information about the data. This also becomes a critical step during the migration of the STEWARD system. Navigate through the 'Design' tab and locate the Warehouse Builder Metadata from the Export option.

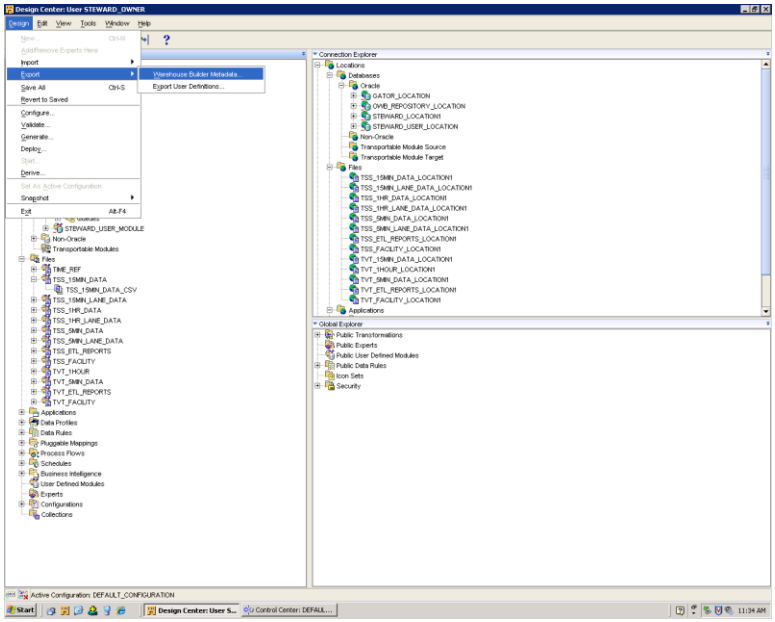


Figure A106. Oracle Warehouse Builder -Export Warehouse Builder Metadata

A3.4.2 Prerequisite- Export Warehouse Builder Metadata (Step 2)

Figure A107 appears after the option is selected. This is the metadata export window. Check the 'Export all object dependencies' and click the 'Export' button. Save the log (.log) file at a location and transfer this to the new system machine.

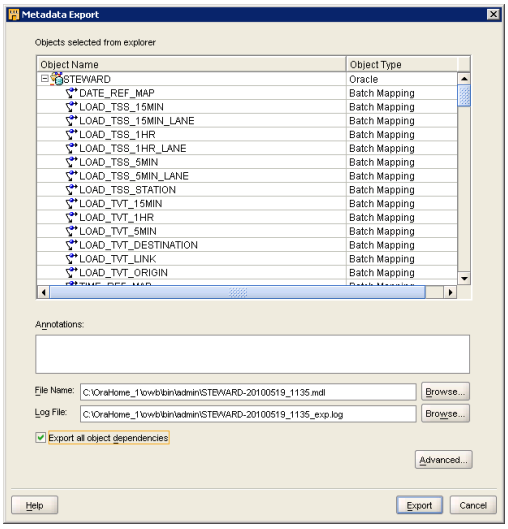


Figure A107. Oracle Warehouse Builder - Metadata Export Window

As shown in Figure A108, the progress of the metadata appears. Wait for a while it finishes up the final exporting the metadata. Click the 'Close' button to continue. The next step is to import the (.log) that was recently exported into the new system.

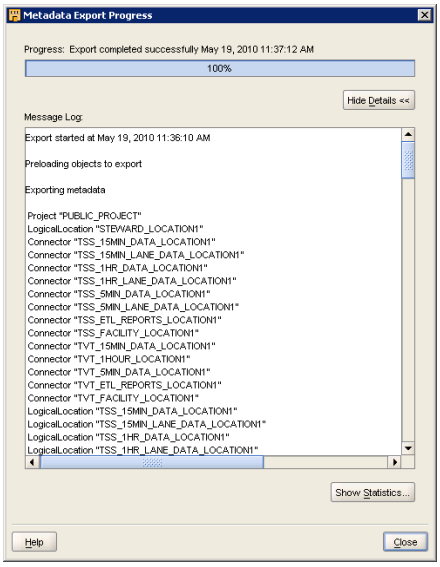


Figure A108. Oracle Warehouse Builder - Metadata Export Progress Window

A3.4.3 Importing Metadata - Import Warehouse Builder Metadata Step 1

To import the metadata into the new system, select the 'Design' tab in the Oracle Data Warehouse Builder Design Center and navigate to the Import\Warehouse Builder Metadata option as shown in Figure A109.

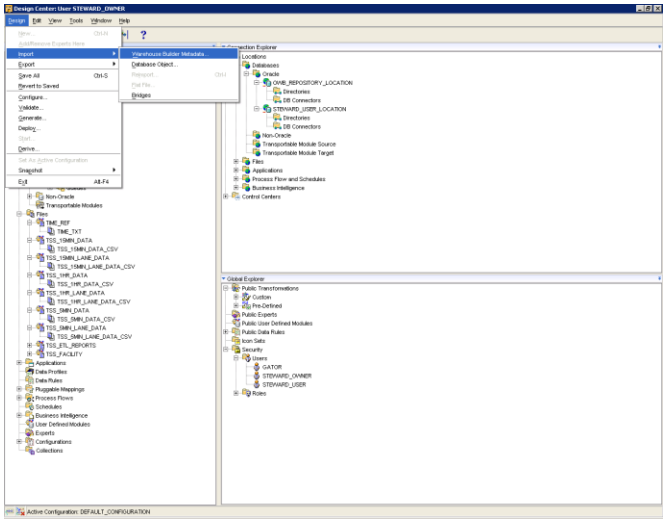


Figure A109. Oracle Warehouse Builder – Metadata Import Selection

Figure A110 appears after the Import option is selected. The file names and the Log file that were exported from the last section are specified here under the respective path locations. All the options are chosen as default and a new metadata is created. Click 'Import' to start the importing process. It should be noted that if the import is not successful, the option 'Match By' *"Names"* should be selected.

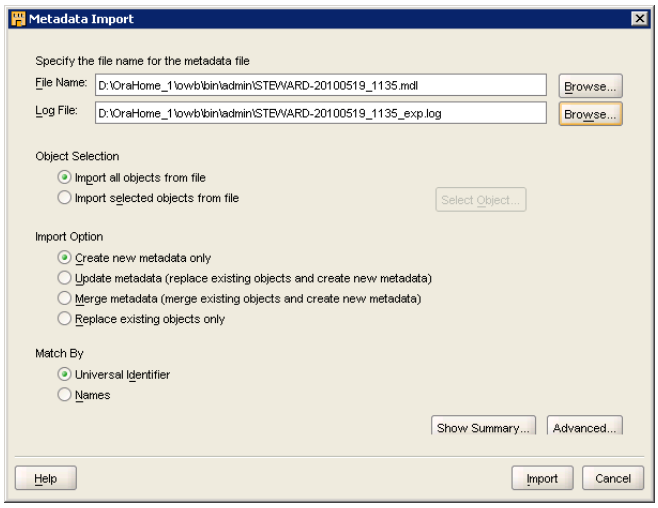


Figure A110. Oracle Warehouse Builder – Metadata Import File Names

As shown in Figure A111, the metadata import is under progress. If the metadata import is not successful, it is recommended to visit the *Show Details* option and the problems should be thoroughly checked and rectified.

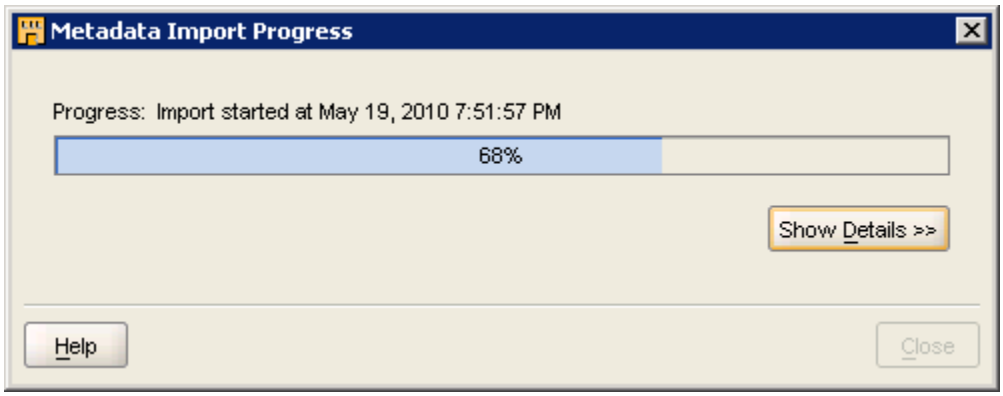


Figure A111. Oracle Warehouse Builder – Metadata Import Progress

A3.5 Registration of the Control Center Manager

The next step is to register the metadata. This step registers the imported module to the newly created depository. This is performed under the Control Center in the Oracle Data Warehouse Builder. This is accessible through the 'Tools' tab in the OWB. As shown in Figure A112, the STEWARD_LOCATION1 is registered. The name LOCATION1 should not be confused with LOCATION as this is system generated. In a similar way all the other locations present in the left side pane of the Control Center should be registered.

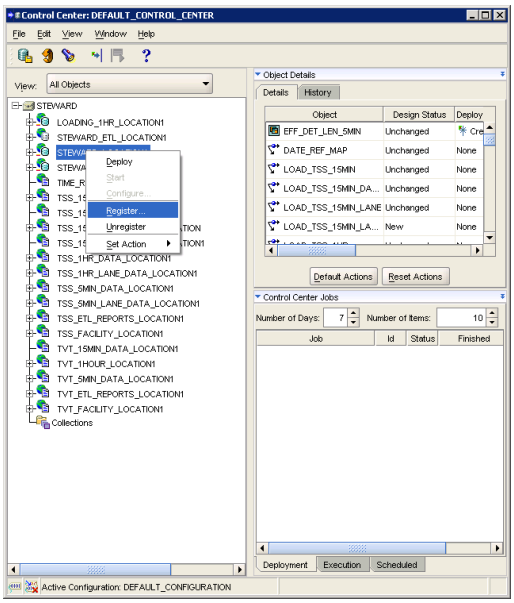


Figure A112. Oracle Warehouse Builder – Register Locations in Control Center

The next step is to check the registry of all the locations. This check is performed in the Design Center's *Connection Explorer*. Right Click on the STEWARD_LOCATION1 and select the Open Editor option. This is shown in Figure A113.

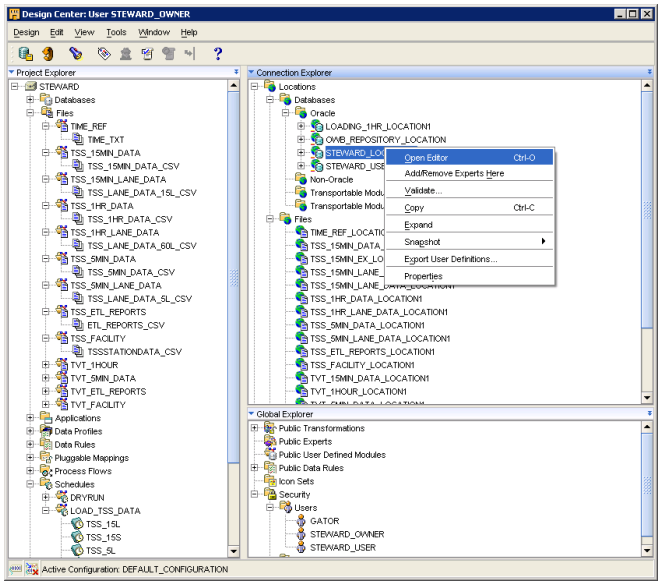


Figure A113. Oracle Warehouse Builder – Registration in Connection Explorer

As shown in Figure A114, the following window appears. Select Open Editor menu and enter the user name, password, host, and service name. Verify the connection details and type the password 'trc513'. To test the connection, click 'Test Connection'. A successful message is displayed.

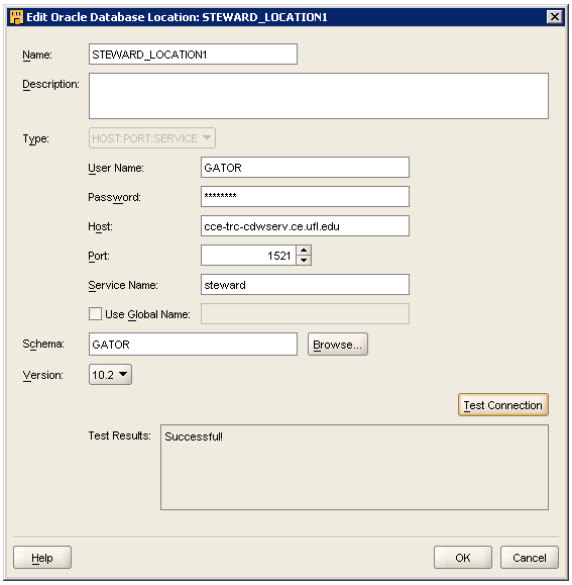


Figure A114. Oracle Warehouse Builder – Database Locations Test Connection

It can be verified again by performing the same step as in Figure A112. If it is already registered, the following screen will appear as shown in Figure A115.

Figure A115. Oracle Warehouse Builder – Database Locations Registration Status

All the other locations under the Oracle tab of the Connection Explorer should be registered. As shown in Figure A116, (*.locations) including all the files locations are registered and verified in the Connection Explorer in a similar. If a location is not registered, a window showing the error message (Figure A117) will appear.

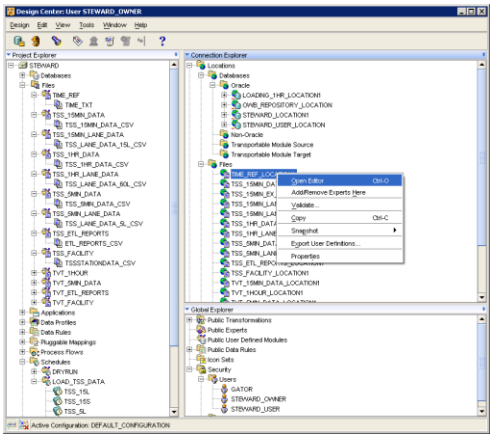


Figure A116. Oracle Warehouse Builder – File Locations Registration

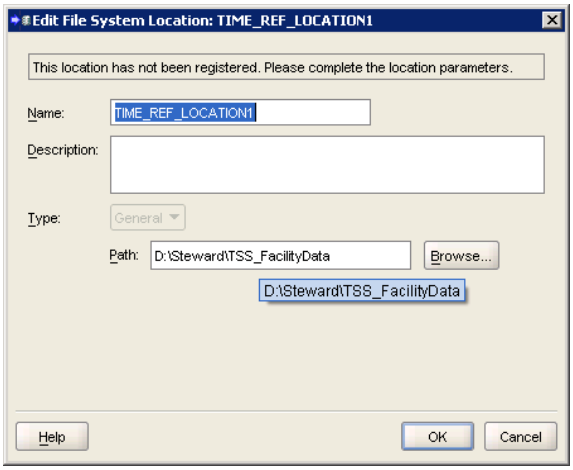


Figure A117. Oracle Warehouse Builder – File Locations Registration Status

After all the locations are registered, the left pane of the Control Center is refreshed. All the file locations that are registered previously appear here. This could be verified further by right-click and selecting the register option.

The updated Control Center is shown in Figure A118.

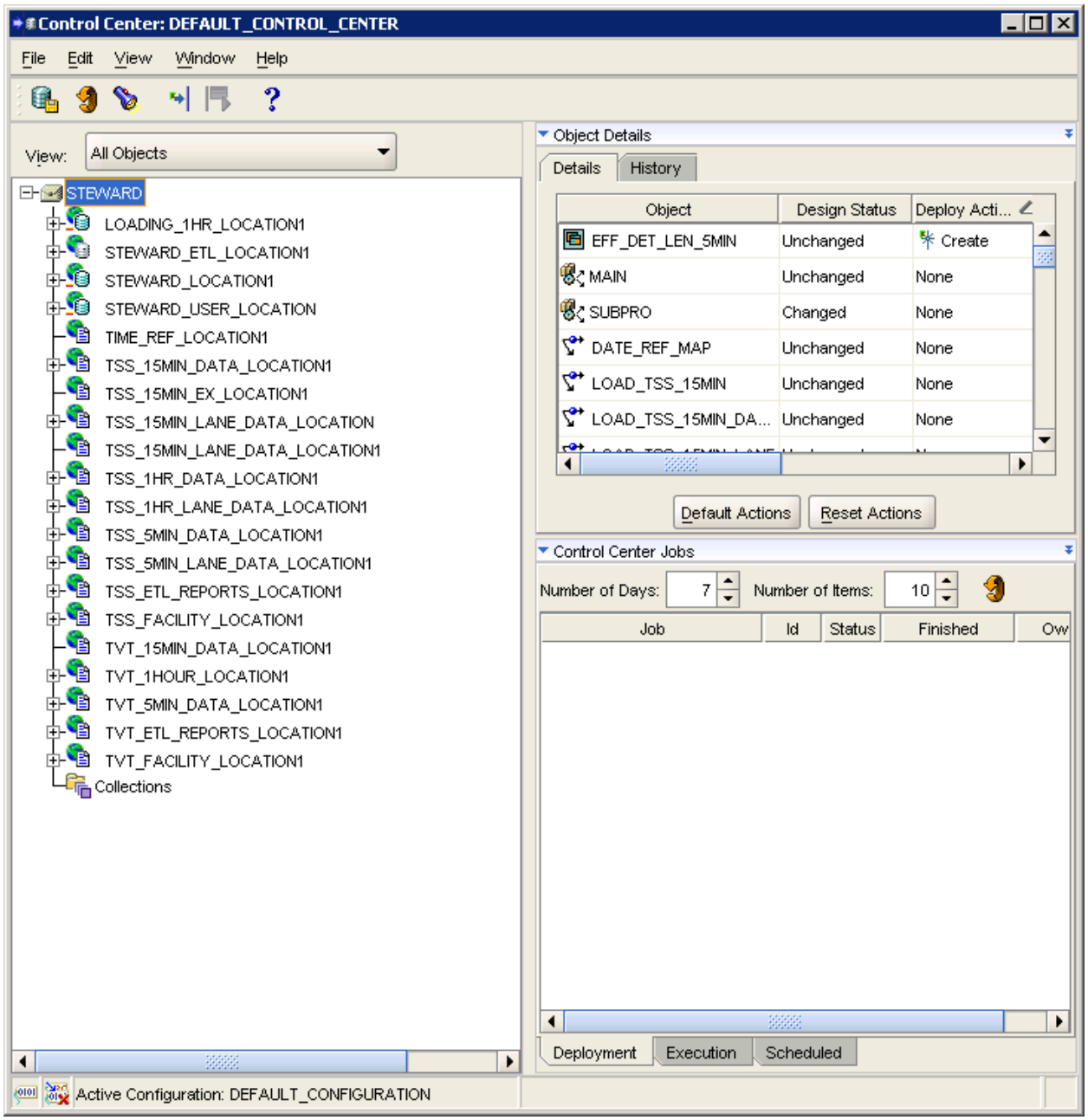


Figure A118. Oracle Warehouse Builder - Updated Control Center

A3.6 Data Deployment Process

After the registration for all the files and the respective locations are done, the next step is to deploy the newly imported schemas. These includes the External Tables, Cubes (Fact Tables), Mappings, Dimensions and Materialized Views.

A3.6.1 External Table Deployment

The first step is to deploy all the imported modules. As shown in Figure A119, select all *External Tables* and click *Default Actions* or *Create* in the Control Center and then, click the *Deploy* button to deploy the DB.

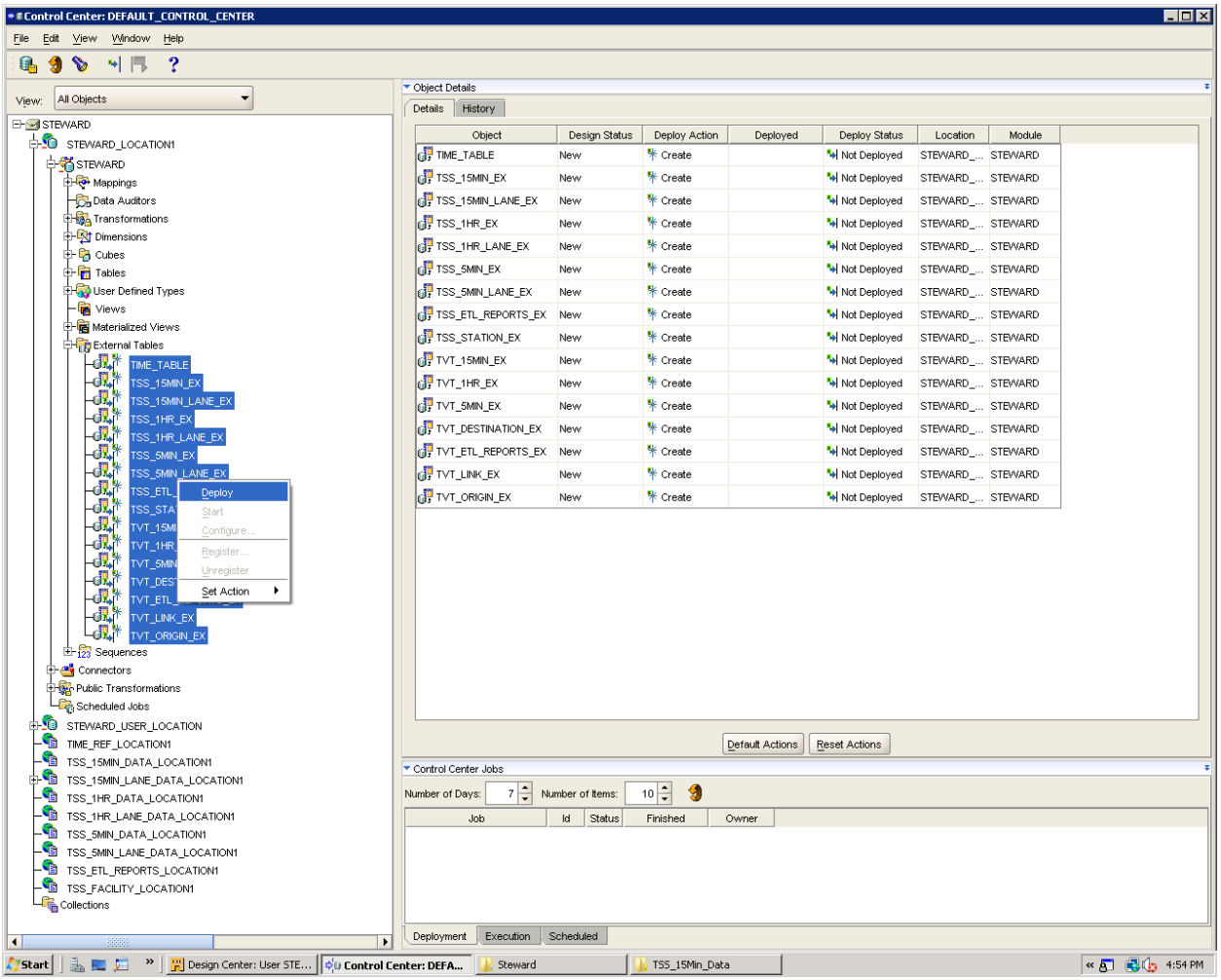


Figure A119. Oracle Warehouse Builder - External Table Deployment

After all the files in the Control Center are deployed, check the deployment status as *Success.*

A3.6.2 Dimension Table Deployment

As shown in Figure A120, select all dimension tables and all sequences and right-click the selected files. Click the *Default Actions* button or by setting *Set Action* or Deploy Action as *Create* to deploy the DB dimension tables and the sequences.

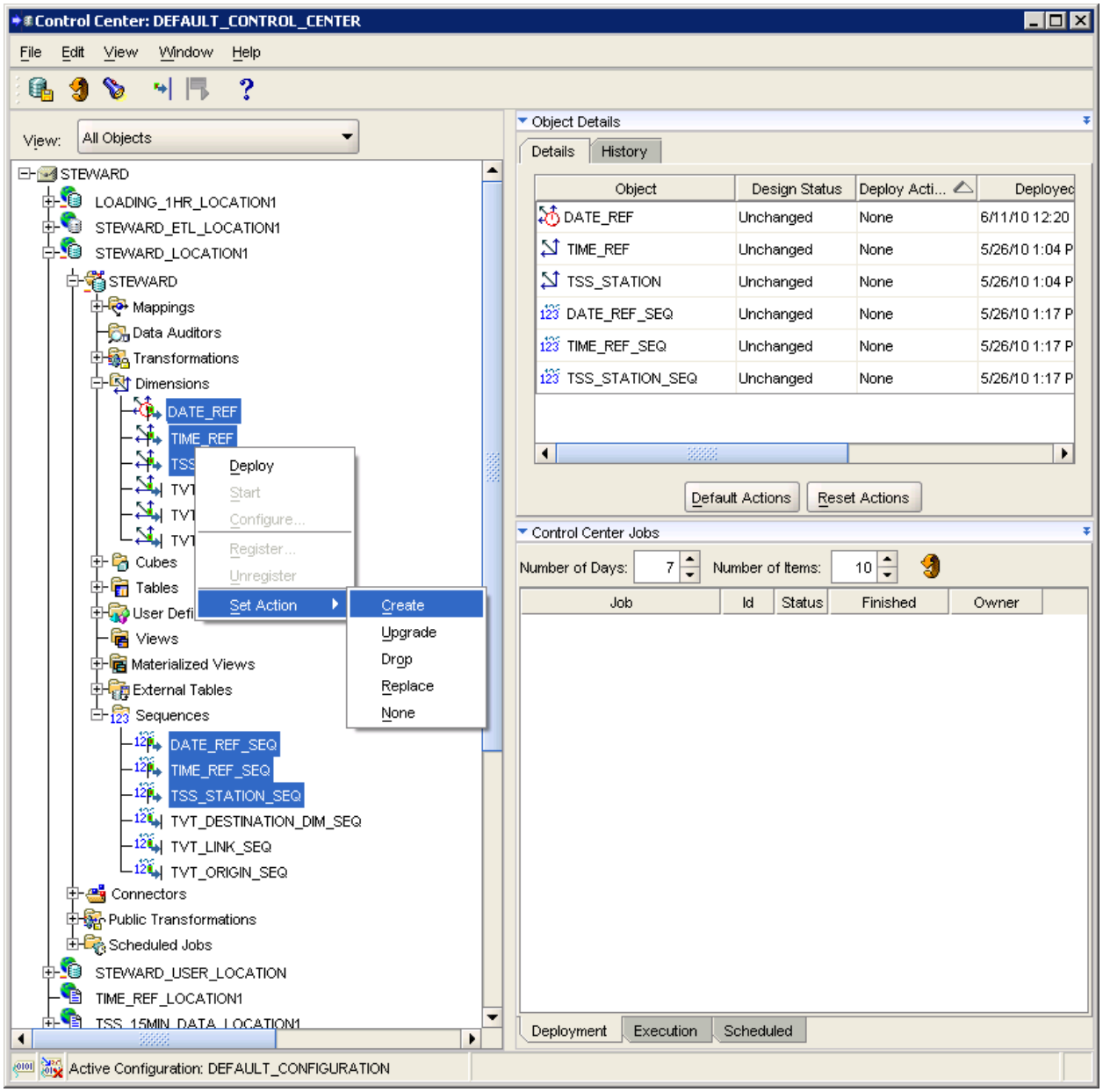


Figure A120. Oracle Warehouse Builder - Dimension Table Deployment

After all the files in the Control Center are deployed, check the deployment status as *Success.*

A3.6.3 Fact Table Deployment

As shown in Figure A121, select all Cubes and all tables for Fact Table and right-click the selected files. Click the *Default Actions* button or by setting *Set Action* or Deploy Action as *Create* to deploy the DB cubes and tables for the Fact Table.

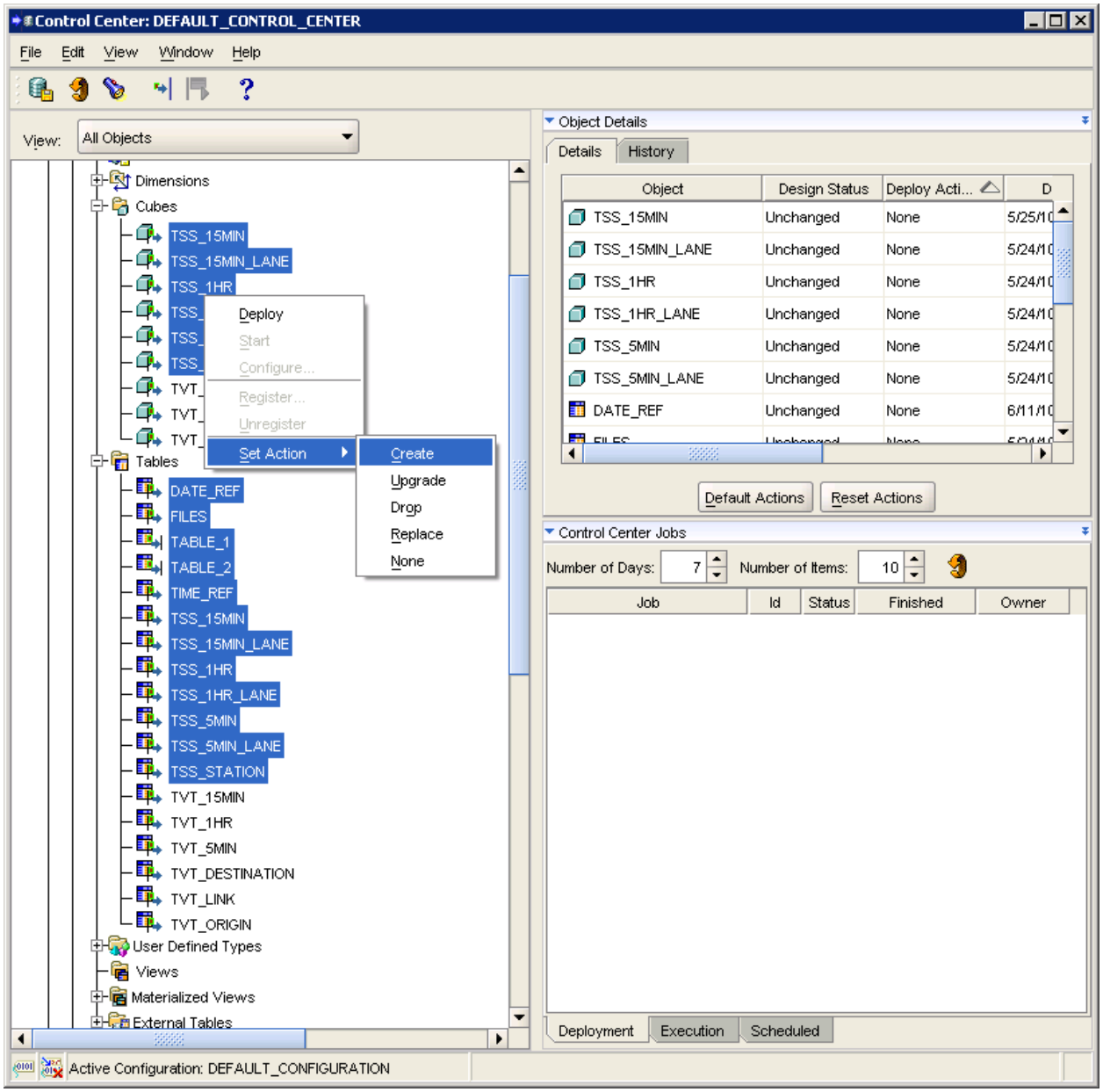


Figure A121. Oracle Warehouse Builder - Fact Table Deployment

After all the files in the Control Center are deployed, check the deployment status as *Success.*

A3.6.4 Mappings Deployment

As shown in Figure A122, select all the files under Mappings and right-click the selected files. Click the *Default Actions* button or by setting *Set Action* or Deploy Action as *Create* to deploy the mappings.

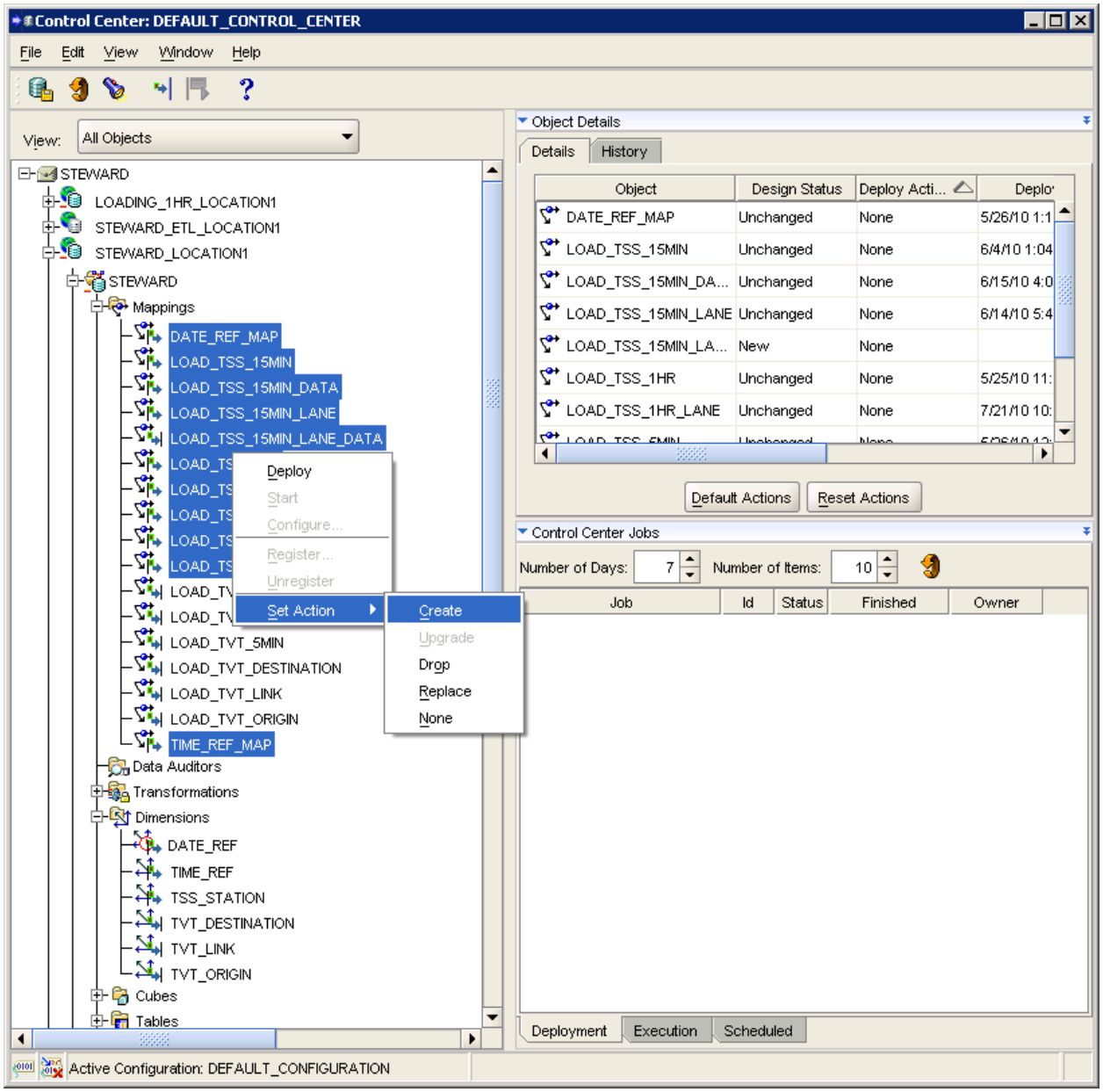


Figure A122. Oracle Warehouse Builder - Mappings Deployment

After all the files in the Control Center are deployed, check the deployment status as *Success.*

A3.7 Data Loading Process

The database or the target schema can be loaded with real daily traffic data using the data loading processes. Two sets of data loading procedures are possible: one, through mapping where only one file is uploaded at a time and the other, through process flows where multiple files are uploaded at the same time.

A3.7.1 Data Loading Through Mapping

To load a single data file into the database, the Mappings table that was deployed in the previous section is used. The files to be uploaded are kept at the same location where the pre-defined files were located in Section A3.3.7. The data loading procedure is then started by navigating to the respective 'Mapping' file (corresponding to the data file that needs to be uploaded) and then by selecting the 'Start' option by right-clicking. This is shown in Figure A123. After the 'Start' option is selected, the data is uploaded to the database.

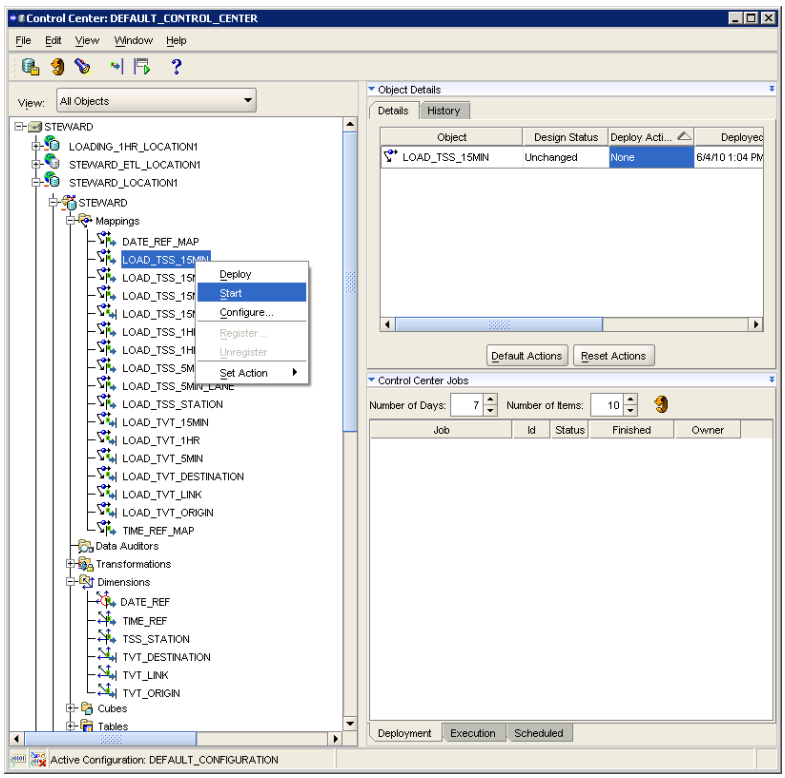


Figure A123. Oracle Warehouse Builder – Data Loading Through Mapping

The progress of the data loading process can be seen on the lower right side of the Control Center window within the Control Center Jobs section. If the status of this process is successful (indicated by a green color tick mark), it means that the file has been uploaded to the database. This is shown in Figure A124. In case, a red color cross appears, it is inferred that the mapping of that particular file or the file did not get loaded successfully. The error message should be carefully read and the changes should be made accordingly.

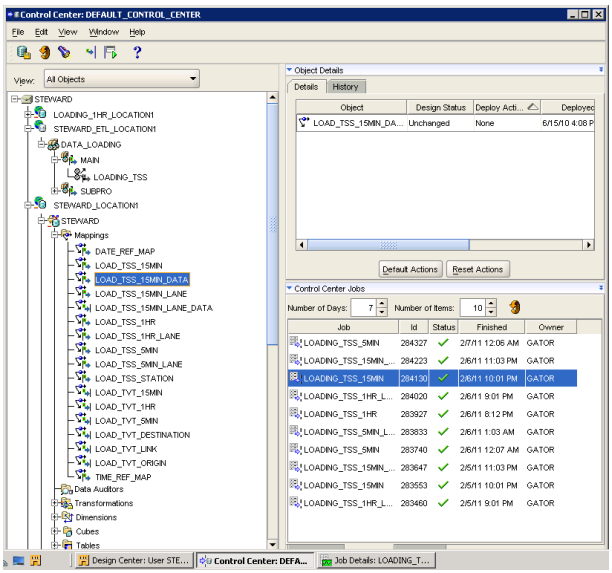


Figure A124. Oracle Warehouse Builder – Data Loading Through Mapping – Job Progress

The status of the data loading can also be verified by navigating to the Job Details of that particular file. This is done by double-clicking the job status on the Control Center Jobs section and checking the 'Execution Results' status. If the numerical value associated with the 'Merged' column appears, it means that the file has been uploaded to the database. This value indicates the number of rows that have been added in the database. This number should be same as the number of the rows in that particular file. This is shown in Figure A125.

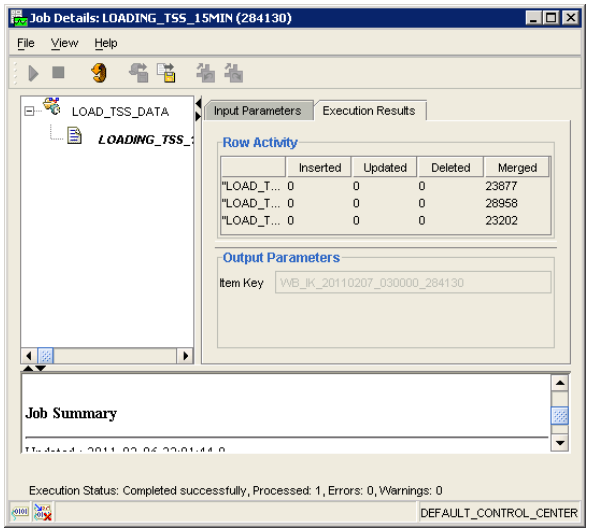


Figure A125. Oracle Warehouse Builder – Data Loading Through Mapping – Job Details

Similarly, this process is repeated for all the other aggregated station and lane data files. With this procedure, the data for a day of a particular district can be uploaded. Multiple files could also be uploaded simultaneously, but it is advisable that only one file is executed and uploaded at a single time. With the help of the mappings, the daily operations could be performed for STEWARD and data for districts can be uploaded accordingly. The data uploading can also be verified with the help of SQL Plus by checking the number of files that have been added during this process.

However, this method of uploading is not efficient if an automated process has to be configured for the database system. As this step is performed manually for a single file, it has to be performed multiple number of times within a day to ensure that all the levels of aggregated data for all the districts are uploaded.

A3.7.2 Data Loading Through Process Flows

To upload multiple files simultaneously, the process flows function is used in the OWB. This enables the users to upload multiple files of the same category for multiple days. For e.g., the 15-minute station data can be uploaded for all the districts at the same time for multiple number of days. This is performed by creating and writing an algorithm for data uploading. For the current STEWARD system, this algorithm was imported during the metadata import from the previous STEWARD system. A brief description on how to build these algorithms is provided in this section. As shown in Figure A126, a new Process Flow procedure is created from the Design Center of the OWB.

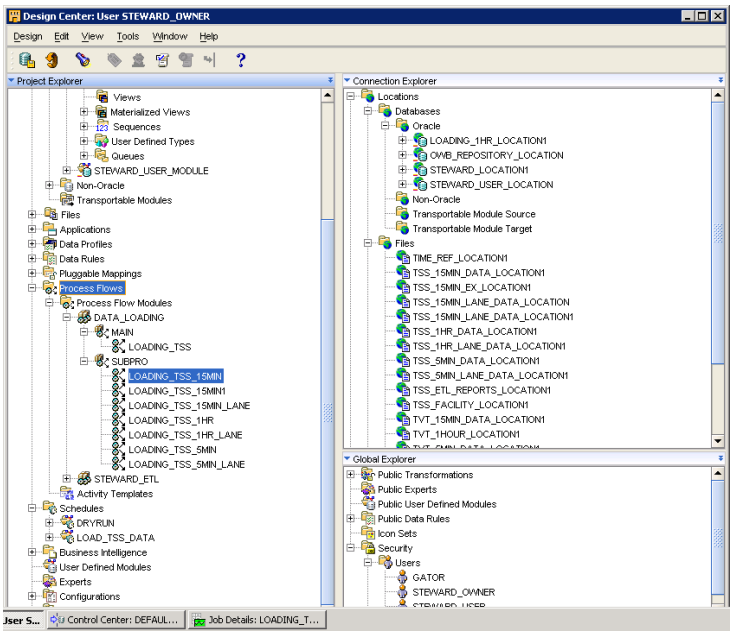


Figure A126. Oracle Warehouse Builder – Data Loading Through Process Flows Creation

The explanation of building and creating the algorithms is beyond the scope of this document. The algorithms in the new system are generally the same as compared to the previous system and this is ensured while the metadata import step. In case, the algorithms are not imported successfully, these algorithms are created again by referring to the previous system. This algorithm enables the users to repeat the process of data uploading of a particular file category multiple times until all the files associated with that mappings are uploaded.

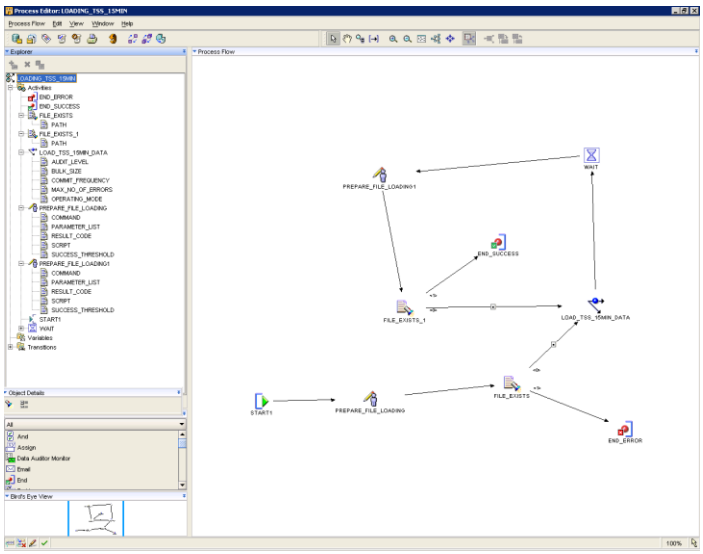


Figure A127. Oracle Warehouse Builder – Data Loading through Process Flows – Algorithm

As shown in Figure A128, the parameters associated with a respective algorithm are given the respective values. The configuration of the parameter values becomes a critical part of the data uploading process, so all the parameters should be same as given in the previous system's algorithms.

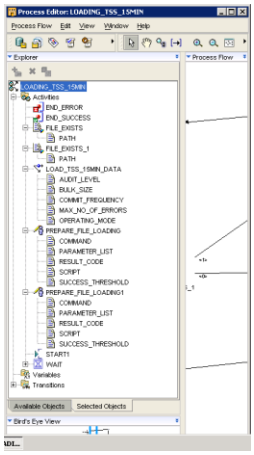


Figure A128. Oracle Warehouse Builder – Data Loading through Process Flows – Parameter

Similarly, the algorithms or the data loading procedures are created for all the mappings in the STEWARD database. All the loading procedures under the process flows are then registered and deployed. After the loading procedures are successfully deployed, the procedures are started in the Control Center of the OWB. The progress of the data loading could be verified as described in Section A3.7.1 of this appendix. With the creation of all loading procedures, it is possible for the users to upload multiple days of data for a particular categorical file. The next step of the STEWARD process is to schedule these jobs so that they can be a part of the automated process. The scheduling of these jobs is briefly discussed in next chapter of this appendix.

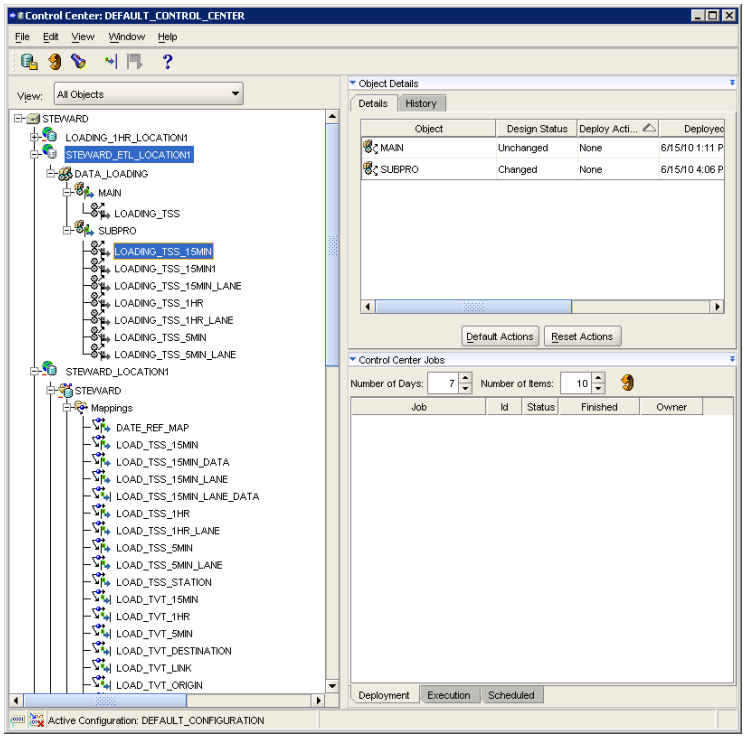


Figure A129. Oracle Warehouse Builder – Data Loading through Process Flows – Deployment

A3.7.3 Data Loading Issues

This section briefly describes the issues that were encountered by the STEWARD moderators during the migration of previous STEWARD server to the new system.

First, an issue may be encountered during the deployment of the External Tables in the Control Center as performed in Section A3.6. This occurs during the metadata import step as all the information regarding the External Tables may not have been imported successfully. This issue arises because of the un-matched variable names and variable types. This issue is resolved by navigating to the External Tables in the Design Center and verifying the properties of the variable names and its properties. After this is performed, if the External Tables are not successfully deployed, these external tables need to be created manually. As shown in Figure A130, the External Tables in OWB Design Center is navigated and a new External Table is created.

The users are advised to create the new tables with same field names and their types as appear in the External Tables of the previous system. This process of creating is simple and is not included in this manual as it may or may not occur during the installation of the new STEWARD system. After the External Tables are created, these need to be linked to the same location where the pre-defined files are kept. After configuring these steps, the tables are deployed again from the OWB Control Center. It is observed that the creation of the new tables lead to successful deployment of tables.

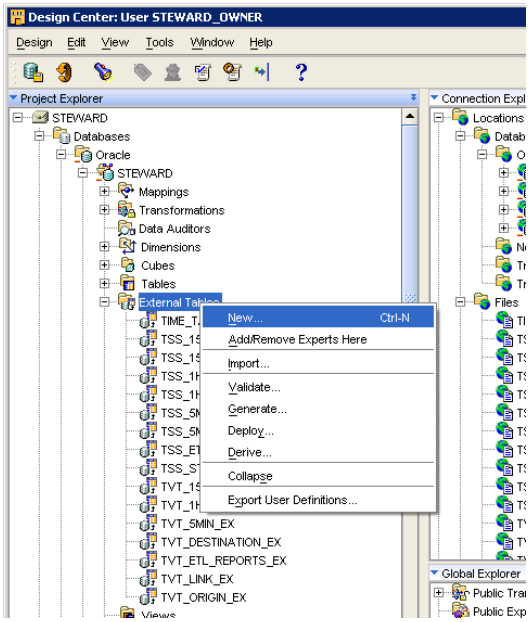


Figure A130. Oracle Warehouse Builder – Data Loading Issues – External Tables

Second, an issue during the deployment of the Mappings might occur. The error for an unsuccessful deployment would appear as a result of un-matched and inconsistent parameters and fields information given in the mapping structure. This issue is resolved by navigating to the Mappings file in the OWB Design Center and by selecting the respective mapping file.

As shown in Figure A131, the following screen would appear. The table structure would appear and the users are advised to check the properties of the left and right tables and their variables. The mapping schema should also be carefully checked and verified. After the verification of these mapping structures, it is assumed that the deployment of the Mappings will occur successfully.

However, if the problem persists, the users are advised to create new 'Mappings' file for all the cases. This is a simple process and it has not been described in this manual in details. During the creating of these new files, all the variables on the left side and right side are required to match perfectly with each other. All the input and output properties of these parameters and the fields are required to be same as specified in the current STEWARD system. The current STEWARD moderators have tested the creation of the new Mappings and its functionality.

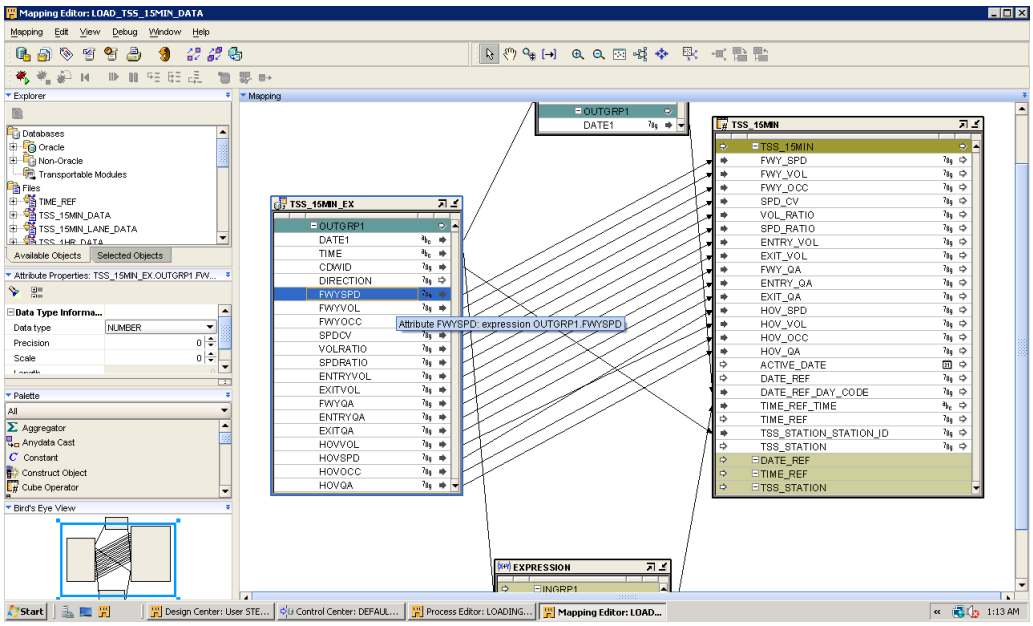


Figure A131. Oracle Warehouse Builder – Data Loading Issues - Mapping

For all the other issues, including the above mentioned, the users are advised to visit the Oracle Corporation meta-link, as different solutions are possible for similar issues.

A4 STEWARD Web Installation

After the STEWARD database is created and data is uploaded, the web configuration of the STEWARD is completed. This is performed for the outside users that wish to access the STEWARD database and extract the data or the reports. For this communication of the web-users or the outside users with the Oracle database, the Net configuration for Oracle data needs to be set up. This is accessed and accomplished through the Net Configuration Assistant for Oracle as shown in Figure A132. These steps have been performed already in Section A2.5.2.

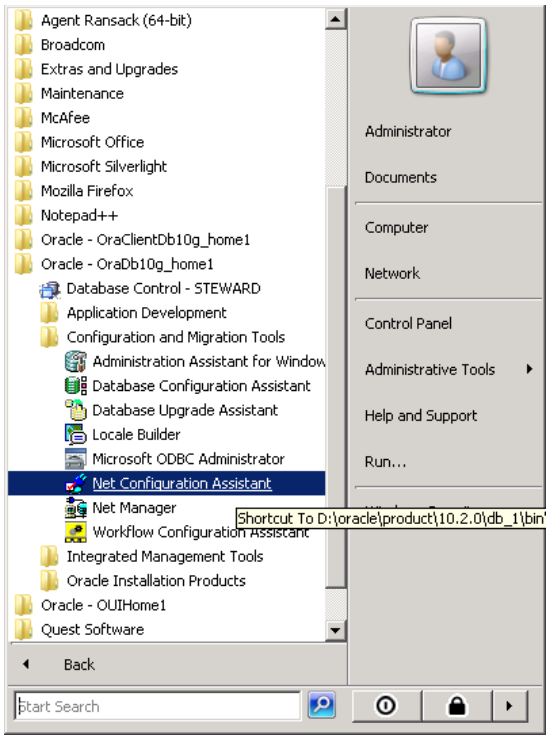


Figure A132. STEWARD Web Installation - Web Configuration Assistant

A4.1 Net Configuration Assistant Steps

The following steps must be performed using the Oracle Net Configuration Assistant before the web-configuration could be performed.

As shown in Figure A133, select local net service name configuration and click 'Next'.

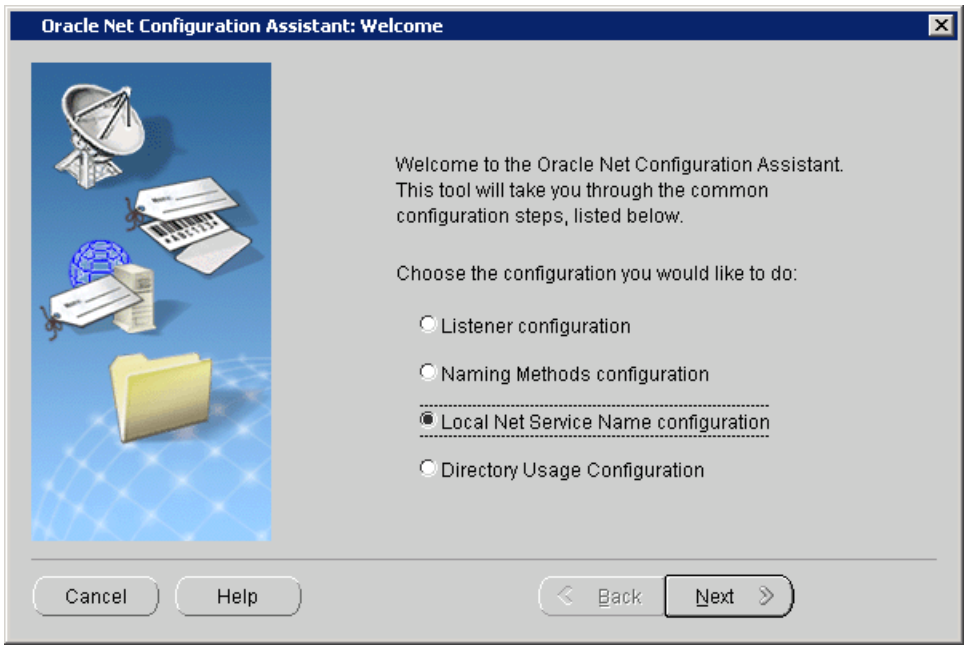


Figure A133. STEWARD Web Installation - Oracle Net Configuration Welcome

As shown in Figure A134, select service name as 'Steward' and click 'Next'.

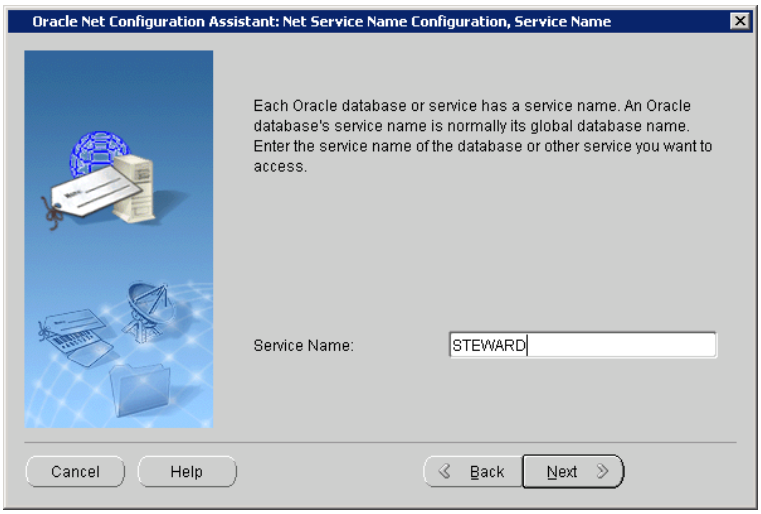


Figure A134. STEWARD Web Installation - Oracle Net Configuration Service Name

As shown in Figure A135, select the communication protocol as TCP and click ‘Next’.

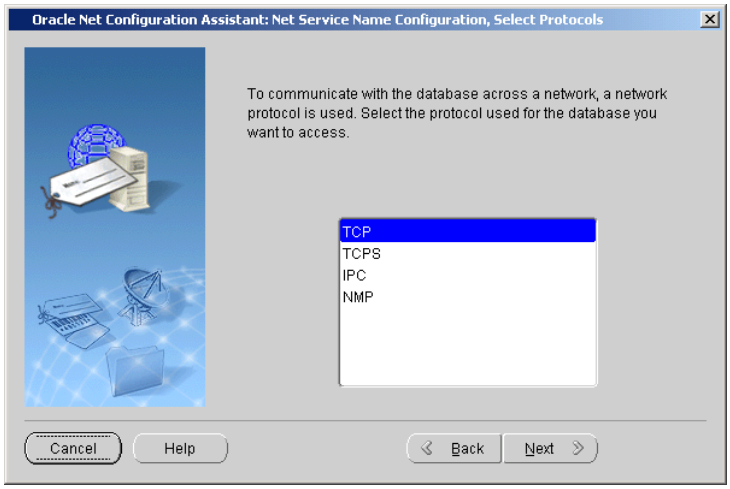


Figure A135. STEWARD Web Installation - Oracle Net Configuration Select Protocols

As shown in Figure A136, edit the host name as ‘cce-trc-cdwserv.ce.ufl.edu’ and use the standard port number of 1521. Click ‘Next’ to continue.

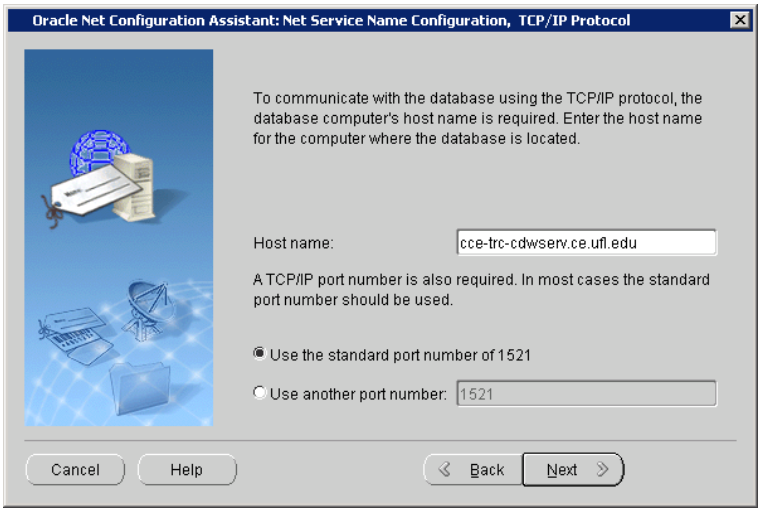


Figure A136. STEWARD Web Installation - Oracle Net Configuration TCP/IP Protocol

As shown in Figure A137, perform the test for the new Oracle net configuration and click ‘Next’.

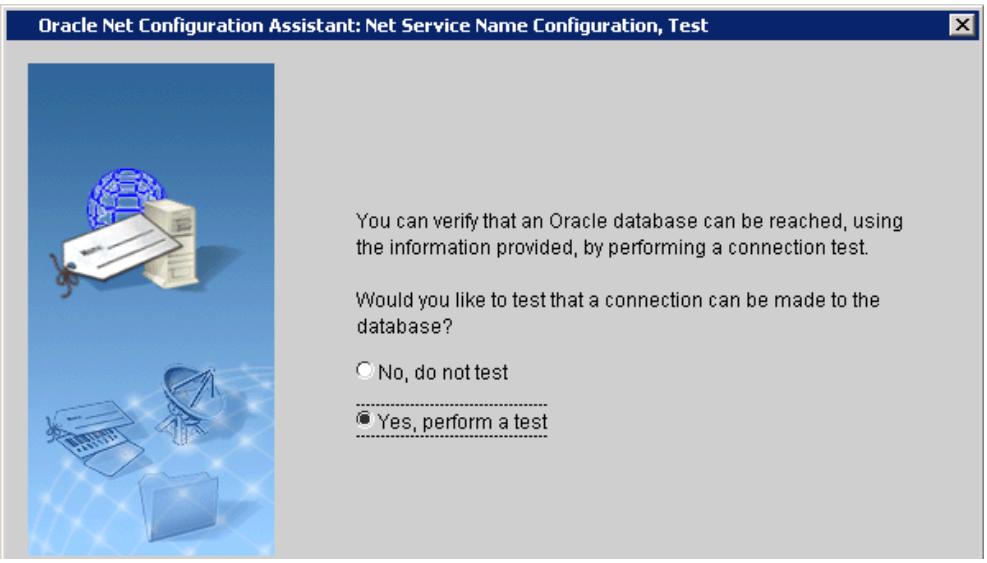


Figure A137. STEWARD Web Installation - Oracle Net Configuration Test

As shown in Figure A138, if the test fails for an invalid ID or password, change the login ID and password by clicking the ‘Change Login’ option.

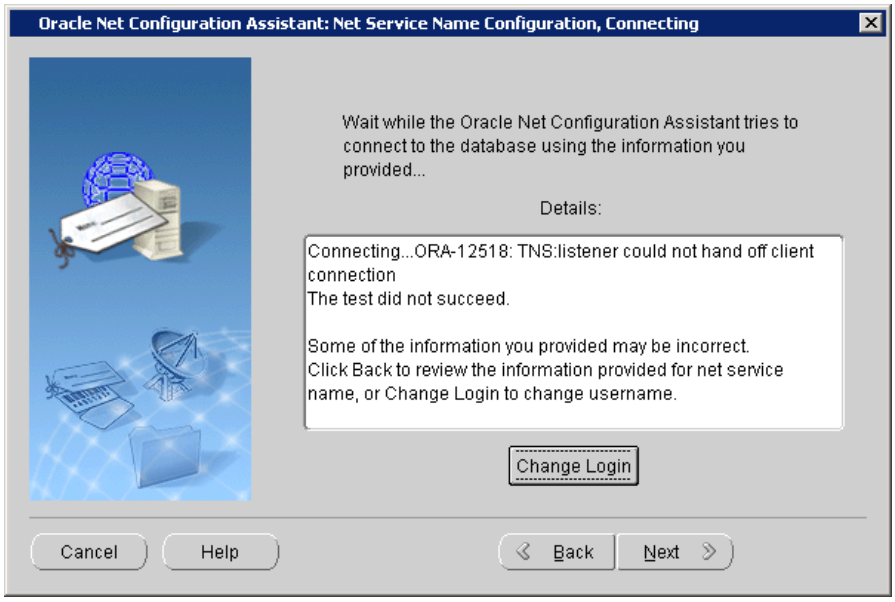


Figure A138. STEWARD Web Installation - Oracle Net Configuration Connecting

As shown in Figure A139, enter the following credentials:

Username: **gator**

Password: **trc513**

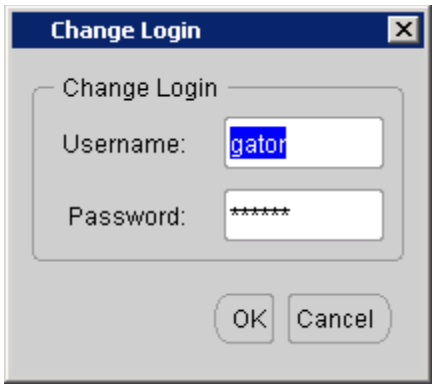


Figure A139. STEWARD Web Installation - Oracle Net Configuration Login Details

As shown in Figure A140, the remote login should be successful. Click 'Next' and exit the Net Configuration Assistant.

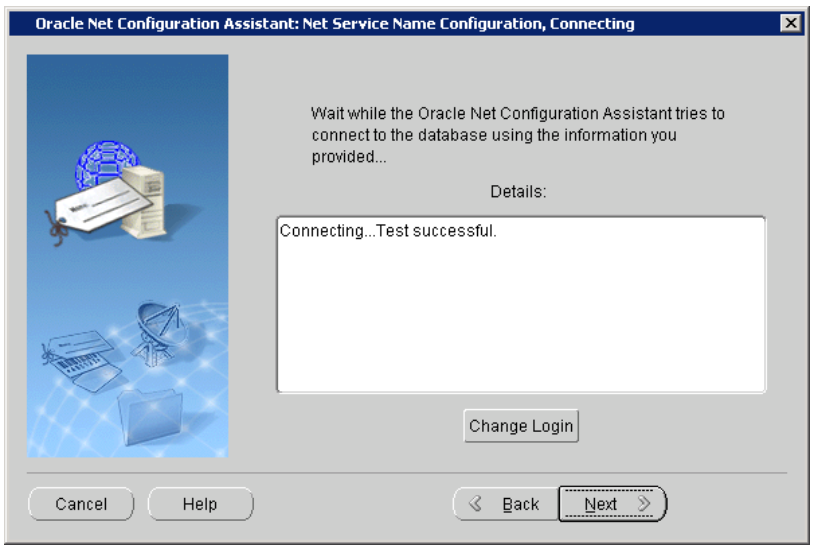


Figure A140. STEWARD Web Installation - Oracle Net Configuration Connection Successful

The ASP codes and Java scripts do not require any specific installation procedures. The STEWARD web-directory is copied from the previous system to the new system (that hosts the web-server). For the new system, the Microsoft Internet Information Services 7.0 is configured and the default root directory is located at C:\Inetpub\wwwroot. The web-site folders are located at this location of the database/web server.

A4.2 STEWARD Web Program Installation

This section describes the step by step process of the installation process of Microsoft Internet Informative Services 7.0 on Windows Server 2008. After the configuration of the IIS 7.0 on the system, the STEWARD web-directory from the previous server is copied to the new system. It is assumed that with the successful configuration of the IIS 7.0 on the system along with other parameter settings, the database can be accessed through any browser. The step by step process for this configuration is given below:

The first step is to access the server manager at the following location:

Start Menu\All Programs\Administrative Tools\Server Manager. As shown in Figure A141, the server manager window opens.

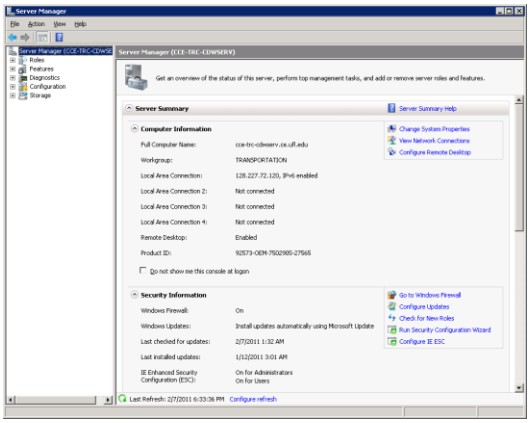


Figure A141. STEWARD Web Installation – IIS Set up Server Manager

As shown in Figure A142, select the Roles, in the Server Manager. The Role Summary View is displayed next. Click 'Add Roles' to continue.

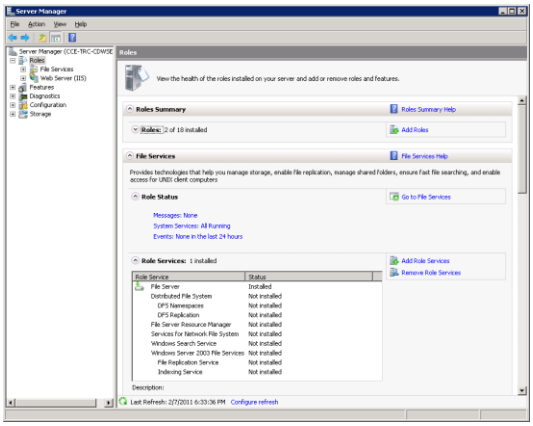


Figure A142. STEWARD Web Installation – IIS Set up Select Roles

As shown in Figure A143, the Add Roles Wizard opens. Click 'Next' to select roles to install in the current system.

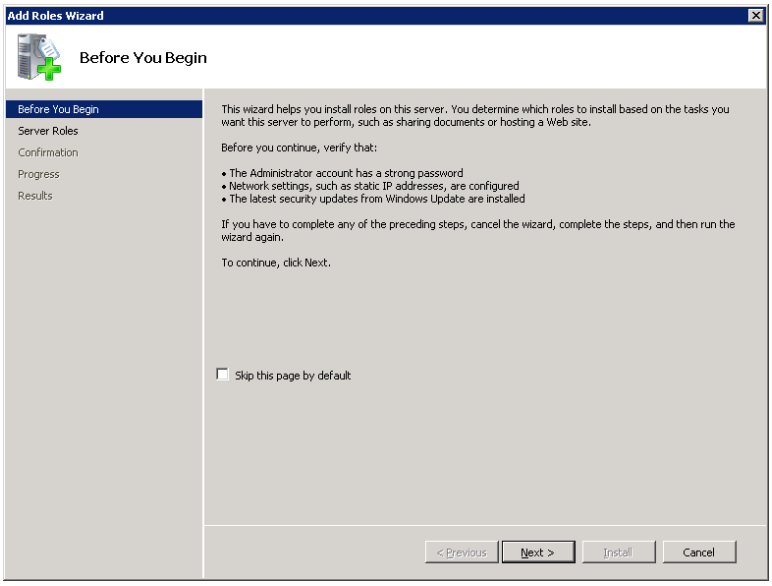


Figure A143. STEWARD Web Installation – IIS Set up Add Summary View

As shown in Figure A144, select the Web Server (IIS) and File Services and click 'Next' to continue.

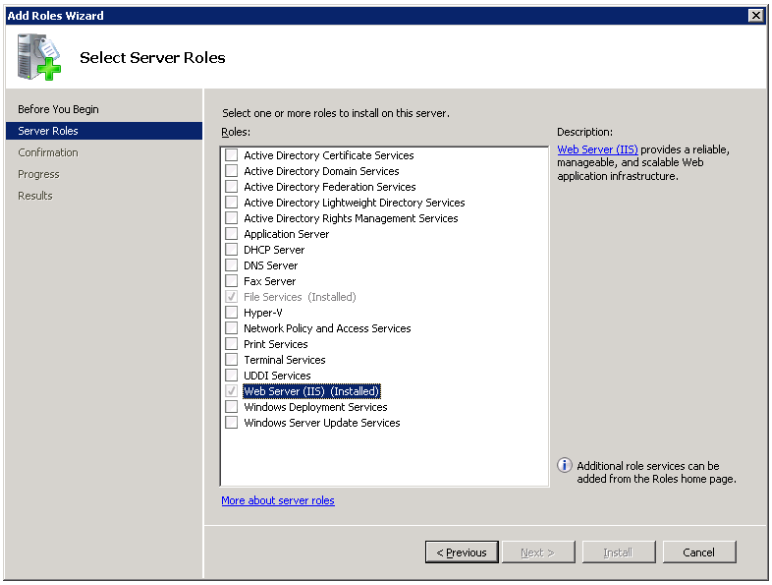


Figure A144. STEWARD Web Installation – IIS Set up Add Roles

The Add Roles Wizard will notify you if any dependency is required as IIS depends on the Windows Process Activation Service (WAS) feature. As shown in Figure A145, the following dialog box will appear. Click all the required roles services and click 'Next' to continue. It is advisable that the user configuring IIS should select all the basic role services including the ASP.NET.

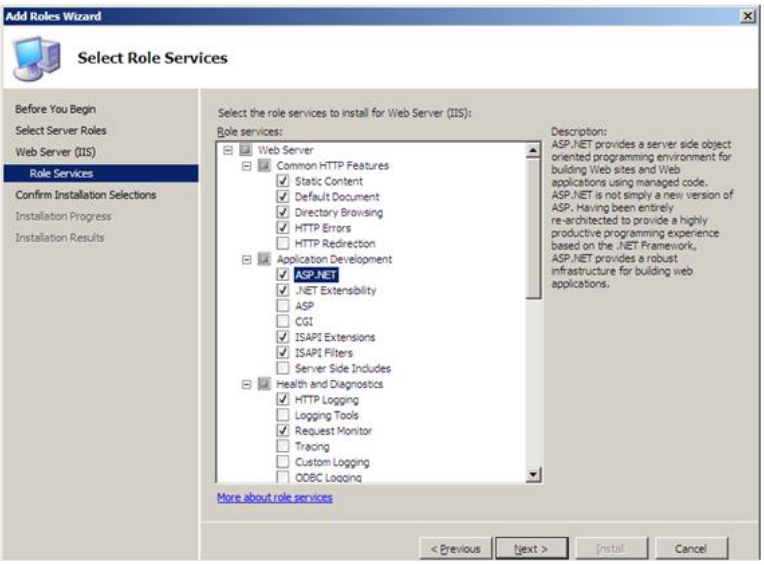


Figure A145. STEWARD Web Installation – IIS Set up Add Roles Services

A summary of what will be installed will appear as shown in Figure A146. Click 'Install' to continue.

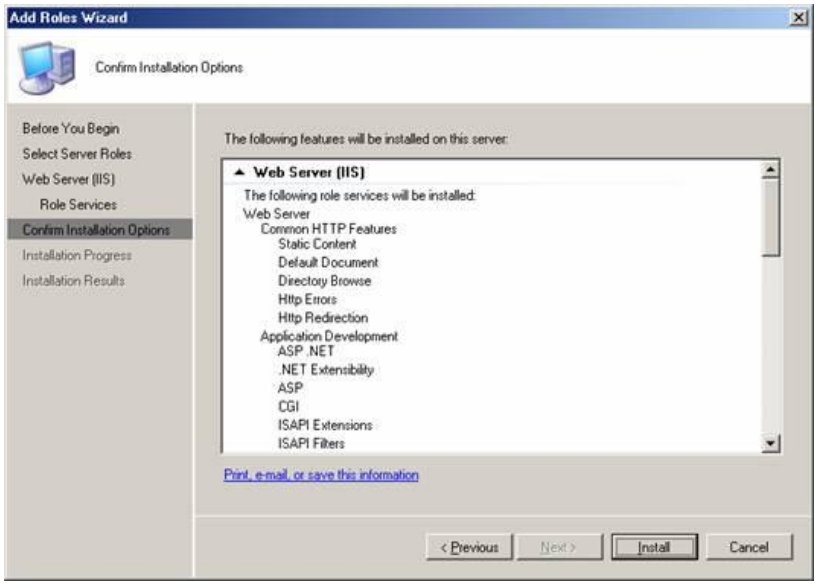


Figure A146. STEWARD Web Installation – IIS Set up Confirm Roles

Click the ‘Install’ button to continue and the installation progress dialog box appears as shown in Figure A147.

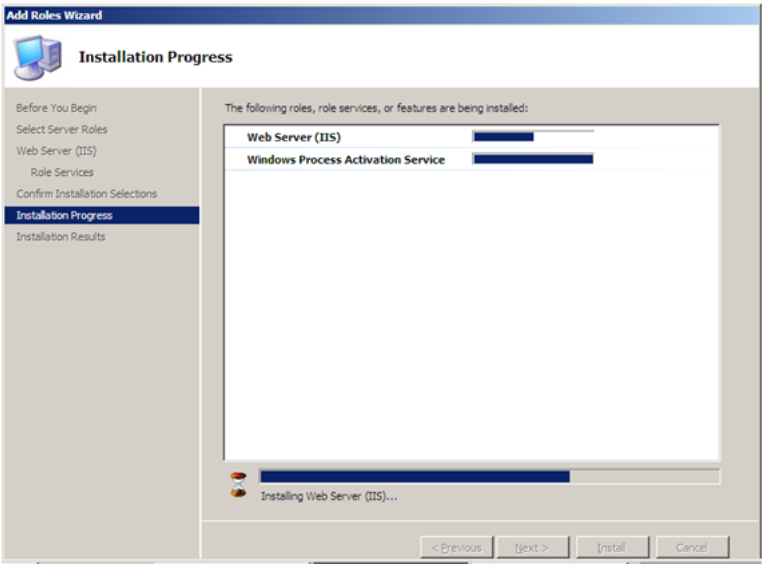


Figure A147. STEWARD Web Installation – IIS Set up Installation Progress

After the installation is succeeded, the following dialog box will appear as shown in Figure A148. Click the ‘Close’ button to return to the Server Manager.

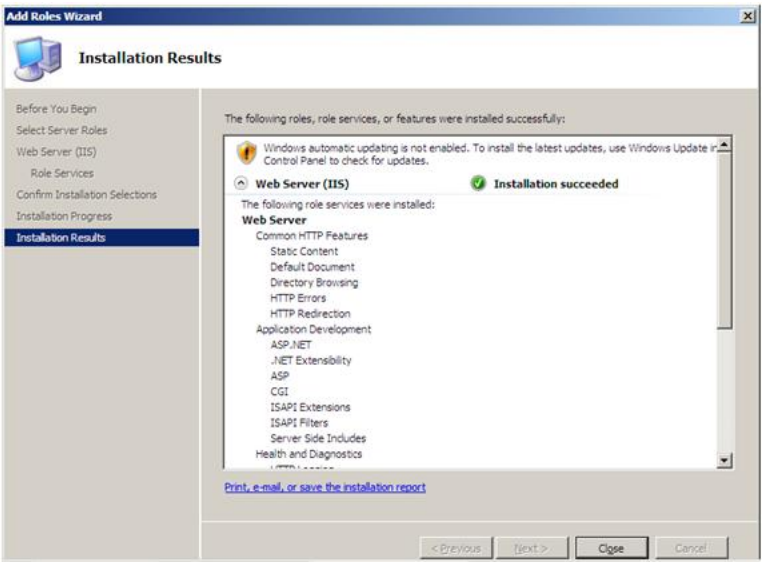


Figure A148. STEWARD Web Installation – IIS Set up Installation Results

As shown in Figure A149, the summary of all the role services that were installed can be seen in the Server Manager window. A quick check could also be performed to verify the IIS 7.0 installation. Start any browser and enter the address as http://localhost. The default IIS 'Welcome Page' appears.

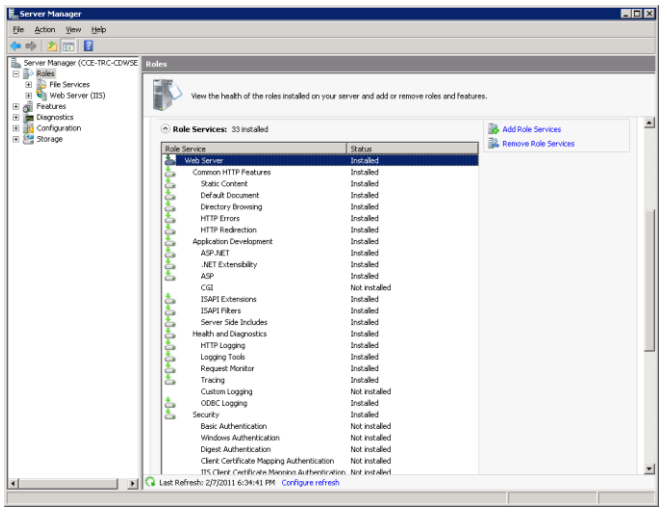


Figure A149. STEWARD Web Installation – IIS Set up Role Services Installed

After the IIS 7.0 is installed, the folder associated with the STEWARD system is placed under the Default Web Site location as shown in Figure A150. This will lead to the functioning of the STEWARD web-site. The next step is to configure system settings for the web-site to be hosted on the Internet.

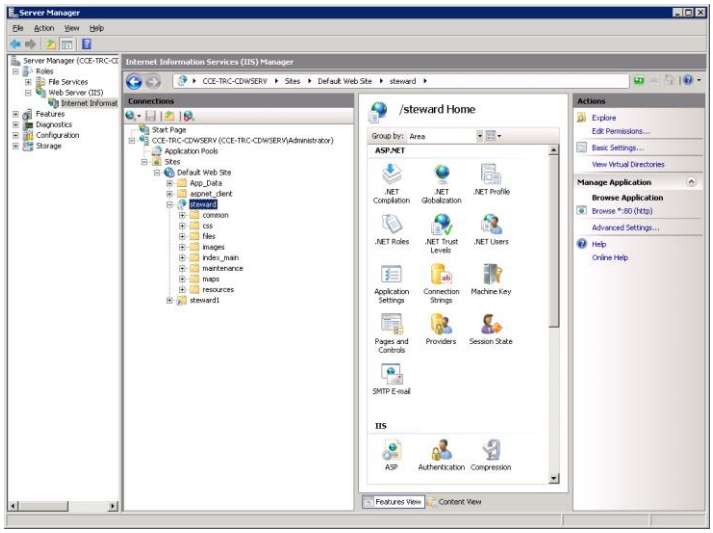


Figure A150. STEWARD Web Installation – IIS Set up Web-Site

A4.3 System Configuration

After the STEWARD web program or web-server is installed using the IIS 7.0, the system configuration needs to be updated. There are two main configuration operations. First, is to enable the web-server on the system firewall and second, is to add the sharing permission for public users to access the traffic data directory.

A4.3.1 Firewall Setting

The STEWARD web-server is installed on Windows Server 2008 machine with the default firewall. For the web-service to be enabled through the firewall, the local area connection needs to be updated. This is performed by opening the Windows Firewall icon from the Control Panel as shown in Figure A151.

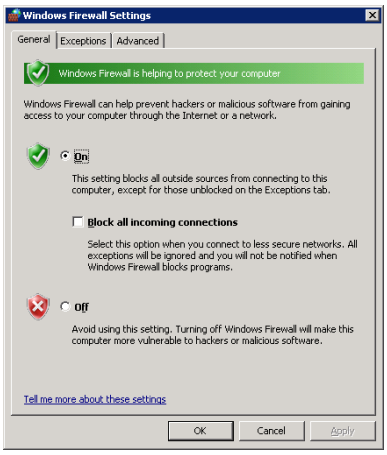


Figure A151. STEWARD Web Installation – Server 2008 Firewall

As shown in Figure A152, select the 'Advanced' tab and check the local area connections.

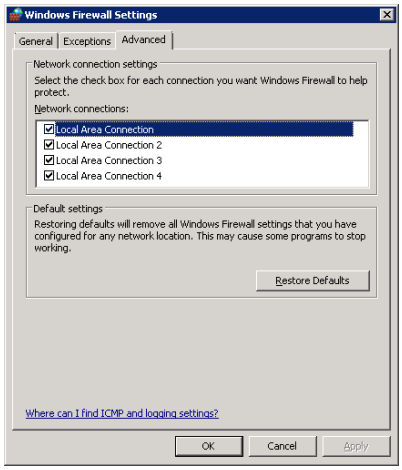


Figure A152. STEWARD Web Installation - Server 2008 Local Connections

Select the ‘Exceptions’ tab and check the ‘World Wide Web Services (HTTP)’ as shown in Figure A153 and click ‘Ok’ to continue.

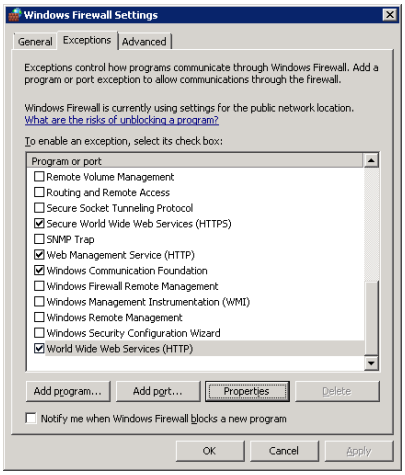


Figure A153. STEWARD web Installation - Server 2008 Firewall Exceptions

A4.3.2 Permission for File Sharing

The customized reports that are created by STEWARD are downloaded at a local directory ‘files’ on the machine hosting the website. For the outside users to access and download these files, it is necessary that sufficient file permissions are given to this folder so that the outside users can access this folder. The files folder is located in the web-site directory of STEWARD at C:\Inetpub\wwwroot\Steward\files. This directory requires the web sharing options for the web user to open or download data file. This is configured by selecting the properties and navigating to the Advanced Security Settings for files as shown in Figure A154 and giving the permissions as shown in the dialog box.

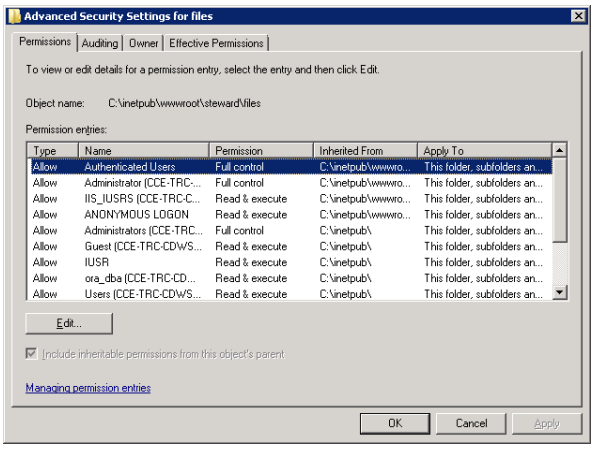


Figure A154. STEWARD web Installation - File Sharing Permission

This step will complete the set up of the STEWARD Internet access.

A4.4 Web Program Configuration

After the STEWARD web program is hosted on the new system, some other settings are also required to be changed. These changes are all specific to the system hosting the web-server.

A4.4.1 Web Server Address

The server address within the web-site directory needs to be updated with the latest STEWARD's url. At present, the server address is defined as:

server_url = http://cce-trc-cdwserv.ce.ufl.edu/ at

\wwwroot\Steward\common\asp\server_url.asp

The 'server_url' needs to be updated as necessary for all the files that has this function.

A4.4.2 Google Map API Key

The Google Maps (Source: Google) is used to display all the detector locations covered by the STEWARD system. For using the Google Map features, it is necessary to have an API key. Each web-server is assigned with only one API key, so when the STEWARD system is transferred, the new API key is required. The current Google Map API key that is used in all the maps is:

<script src= http://maps.google.com/maps?file=api&v=2&key=ABQIAAAAPk-oBJy_fmL1wtiD0Pnp-RSFEumW1SRDGm-XPvjC2Mk-0W4bkhQ3R7ch2a8bNyDJv_F1yn-5tDjv8w

type="text/javascript"></script>

After the previous server is moved to new server, the new key is updated for all the maps on the STEWARD server. These are the interactive maps for each respective district and an interactive map for the entire state of Florida

\wwwroot\Steward\maps\TSS_interactive_map.html

\wwwroot\Steward\maps\dX_interactive_map.html

A4.4.3 STEWARD DB Login Information

For the web-server to communicate to the database server, it is necessary to identify the location where the database connection credentials are given. These are accessed and updated if required, at the following locations.

\wwwroot\Steward\common\asp\db_open.asp

\wwwroot\Steward\common\asp\db_close.asp

A5 STEWARD Management

A5.1 STEWARD Daily Operations

The STEWARD system receives archive data from SunGuide system in Districts 2, 4, 5, 6, and 7 on a daily basis. The data received from the districts are first processed and then loaded onto the STEWARD's database for users to access it in form of several customized reports. A schematic diagram of various components that are involved in the daily operations is provided in Figure A155. These components are incorporated with each other with the help of two computers and one external hard disk. One of the computers hosts the FTP server and the other host the database and the web server. The external hard disk is used to transfer the processed data from the FTP server to the database server. The extraction and transformation is performed on the FTP server with the help of Windows Scheduler and the data loading is done on the database server with the help of the Oracle Scheduler. The remainder of this section describes in details the steps involved with the daily STEWARD operations.

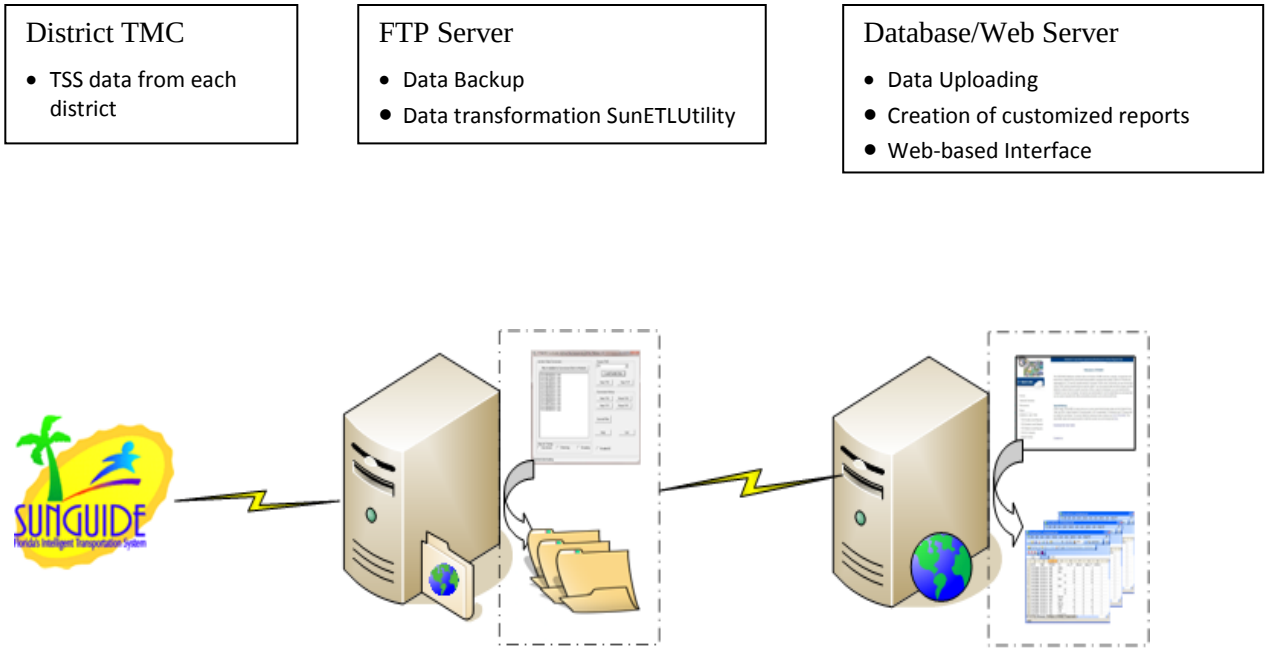


Figure A155. STEWARD System Architecture

A5.1.1 Data Transfer from District SunGuide Systems

The SunGuide system collects the 20-, 30-, or 60-seconds speed, count, and occupancy data from its detector network every day. The collected data are then archived in a raw format with pre-defined file names and formats through the SunGuide archive subsystem every day. The collected data are then transferred from the FDOT districts to the STEWARD's FTP server by 11:59 pm every day through a secure FTP connection. The archive data are received at the FTP

server in a text file format, typically named as '*TSS-MMDDYYYY—1.DAT'*. The contents of this file are presented in a comma separator format, so that the data entries could be read and extracted by the SunETLUtility.exe program. To utilize less space in the system, the transferred files are zipped by the SunGuide archive subsystem with the help of a zip utility and are send to the STEWARD FTP server. To receive these files at the FTP server, the STEWARD operators creates/created connections for all the participating districts through the 'FileZilla Server' software. Any other FTP server software could also be used to enable these transfers. Different connections are required for every district with unique login credentials to transfer these zipped files. In the new server, the following credentials are used to configure the connections in the 'FileZilla Server' software:

- FTP or host address: cdw880.ce.ufl.edu
- login ID's: cdw_user_d*x* (*x* denotes the district ID number, i.e., 2, 4, 5, 6, or 7)
  (*Passwords are not provided due to security reasons)

After these connections are configured with the appropriate credentials, the archived files are transferred to a specific folder location on the FTP server computer. The data for all the districts are received at a different folder location as the archived files do not include any district ID. These files are pushed into the FTP server the very next day after the data files are archived. The timeline for the transfer of these files from each district is given below:

- District 2 TSS data posted at 3:30 AM the next day
- District 4 TSS data posted at 2:00 AM the next day.
- District 5 TSS data posted at 12:00 PM the next day.
- District 6 TSS data posted at 2:00 AM the next day.
- District 7 TSS data posted at 3:00 AM the next day

All files are transferred and archived into the STEWARD FTP server at the following folders:

- District 2: C:\Steward_FTP\District2
- District 4: C:\Steward_FTP\District4
- District 5: C:\Steward_FTP\District5
- District 6: C:\Steward_FTP\District6
- District 7: C:\Steward_FTP\District7

A5.1.2 Data Backup and Transformation in the STEWARD FTP Server

After the raw data are transferred to the FTP server, the data are transformed/converted into the aggregated 5-, 15-, and 60-minutes station and lane data. This task is performed by the automatic utility which is run on the FTP server. This utility extracts the raw data and transforms the data into the processed format for all the districts. The utility is executed from the following location on the FTP server:

…Desktop\Automation utility\Utility

A snapshot of this utility is provided in Figure A156. To run this utility, the user is asked to specify folders locations at the two tabs of the utility. The first tab, i.e. 'General' tab asks the users to specify the paths for different folders on the local computer or the FTP server where the utility is running. For the new STEWARD system, the paths that are provided on the first tab are:

ETLUtility Path: The folder location where the SunETLUtility is placed on the FTP server. This is located in

…Desktop\Automation utility\AutomaticApplication\SunETLUtility

Log File Path: The text (.txt) file location that summarizes the data processing for each district. The folder location on the FTP server is

…Desktop\Automation utility\Utility\Daily Log Files

RAW Backup: This field specifies the location where the raw data is backed up. This is located at:

…Desktop\Automation utility\AutomaticApplication\RawBackup

Processed Files: The folder location where the processed data are located after the data transformation. This is referred as 'Master Folder' in this report and is located on the new server at:

…Desktop\Automation utility\AutomaticApplication\ProcessedDataBackup

Processed Files Backup: This path specifies the location where the processed data are expected to be backed up.

…Desktop\Automation utility\AutomaticApplication\ProcessedDataBackup

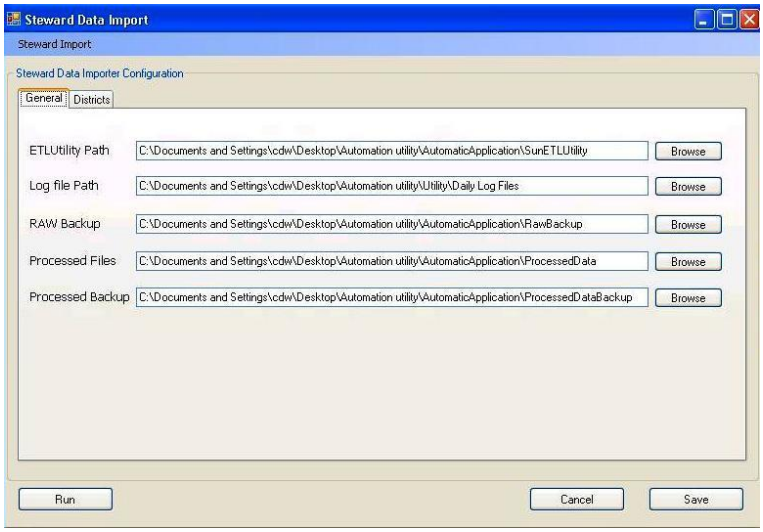


Figure A156. GUI for first tab of the automatic utility

Second, the 'Districts' tab that ask the users to specify the paths where the raw data for all districts are located on the FTP server. In general, these locations are the same where the raw data are received from the TMC's during the file transfer process. These paths enable the automatic utility to read and extract the raw data during the data transformation for each district. Since, the raw data are received in a zipped format, it is essential that this utility also unzip the raw data. To accommodate this, the "7 Zip" software is embedded with the new utility and included in the package of this utility. A snapshot of this tab is provided in Figure A157. After all the fields are specified, the utility is ready to run. In the current STEWARD system, these fields are located at:

C:\Steward_FTP\District*x* (where *x* denotes the district number)

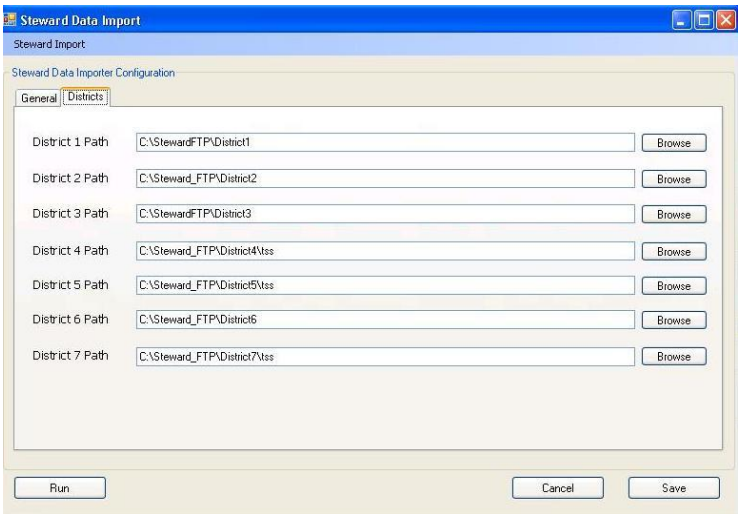


Figure A157. GUI for second tab of the automatic utility

After all the fields of the automatic utility are specified, the utility is run through the Windows Scheduler. In the new server, the last TSS or raw data file is received at 12 noon from District 5 every day, so the utility is scheduled to run every day at 2:00 PM. After the data transformation, the aggregated data are saved on the processed folder location as mentioned above. During this transformation, one input TSS data file creates 10 new output files and update one status file. For example, one sample input data file, *TSS-03032009--1.dat* from District 2 would be transformed into the following output files

- ConvertedData\D2_TSS-03032009-0.csv
- ConvertedData\D2_TSS-03032009-DailyReport.Log
- GroupData\D2_TSS-03032009-1S.csv
- GroupData\D2_TSS-03032009-1L.csv
- GroupData\D2_TSS-03032009-5S.csv
- GroupData\D2_TSS-03032009-5L.csv
- GroupData\D2_TSS-03032009-15S.csv
- GroupData\D2_TSS-03032009-15L.csv
- GroupData\D2_TSS-03032009-60S.csv
- GroupData\D2_TSS-03032009-60L.csv
- TSSConversionHistory-D2.csv (updated)

The processed data are then organized at a location given in the first tab of the utility. This folder is called as the 'master folder' for general reference. The processed data are organized such that all the 5-minutes station level files are clubbed under the 5-minutes station data folder and similarly, other data files are organized. The folders within this master folder are named exactly as same as described in Section A3.3 of this appendix. After all the data files are organized within this folder, the next step of the daily operations of STEWARD is configured and scheduled. As the automatic utility runs for an average of three-four hours every day, the next activity of the daily operations is scheduled to run at 7:00 PM, which is the data transfer from FTP server to the external hard disk.

A5.1.3 Data Loading into the STEWARD Database

After the data are processed on the FTP server through the automatic utility, the processed data files are organized in the master folder. This master folder is then copied to an external hard disk with the help of "SynBack" software. This operation is scheduled to occur at the FTP server machine at 7:00 PM every day.

The next step of the daily operations is the transfer of the master folder from the external hard disk to the database server computer. This is performed through a remote connection at 7:30 PM every day with the help of "SynBack" software. The location where the master folder is copied on the database server is the same as provided in Section A3.3 of this appendix. After the master folder is transferred to the database server, the data uploading of the processed data, i.e. the

station/lane data onto the STEWARD database is scheduled. This is performed on the database server through the OWB as follows:

First, the data algorithms as shown in Figure A127 of this appendix are configured. These algorithms are written for every aggregated time period, i.e. the 5-, 15- and 60-minutes for both the station and lane data. The following mapping tables are integrated with these algorithms:

- LOAD_TSS_5MIN: to load the 5-minute station data
- LOAD_TSS_15MIN: to load the 15-minute station data
- LOAD_TSS_1HR: to load the 60-minute station data
- LOAD_TSS_5MIN_LANE: to load the 5-minute lane data
- LOAD_TSS_15MIN_LANE: to load the 5-minute lane data
- LOAD_TSS_1HR_LANE : to load the 5-minute lane data

These algorithms are then associated with the respective data folders within the master folder. For e.g., the algorithm to load the 5-minutes station data would be associated with the 5-minutes station data folder inside the master folder and similarly, for all the other algorithms as explained in Section A3.7 of this appendix. The completeness of the algorithms is determined by the Oracle OWB itself. The data loading procedures are then started according to the procedure described in Section A3.7.2 of this appendix. After the successful deployment of the data loading procedures under the process flows, the loading procedures are integrated with the automated process. The data loading procedures are then scheduled to start at a particular time of the day through the OWB Design Center. As shown in Figure A158, new schedules are created through the OWB Design Center. This is performed by right-clicking the 'Schedules' option and selecting 'New'.

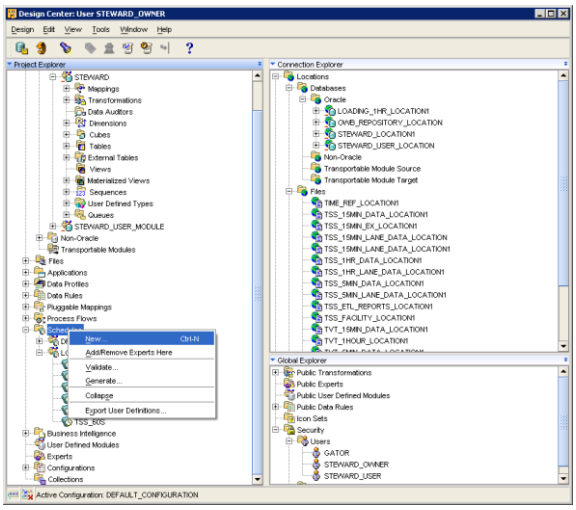


Figure A158. STEWARD Operations – New Schedule for a job

Each data loading procedure or just 'job' is scheduled to run at a particular time. The scheduling of these jobs is performed next for the daily STEWARD operations. As shown in Figure A159, the jobs can be scheduled after defining the start date, end date and the time run.

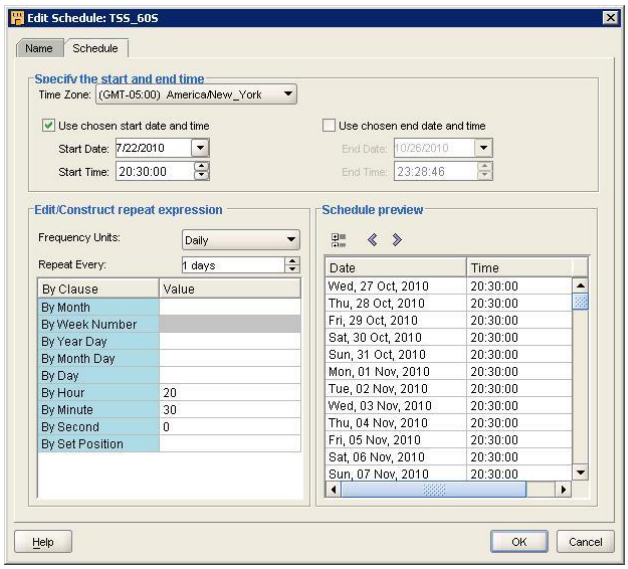


Figure A159. STEWARD Operations – New Schedule Dialog Box

In the new server, these jobs are scheduled to run after the master folder is transferred from the external hard disk to the database server. As the last process of the automated process ended just shortly after 7:30 PM, the first loading procedure is scheduled to run at 8:10 PM. The timeline for all the loading procedures for all categorical data are given below:

At 8:10 PM: TSS_60S, 60 minutes station data

At 9:00 PM: TSS_60L, 60 minutes lane data

At 10:00 PM: TSS_15S, 15 minutes station data

At 11:00 PM: TSS_15L, 15 minutes lane data

At 00:05 AM*: TSS_5S, 5 minutes station data

At 01:00 AM*: TSS_5L, 5 minutes lane data

(*-following next day)

As the processed data from all the districts are uploaded at the same time for the respective loading procedure, a significant time gap is given between two consecutive jobs to avoid an overlap. After all the jobs are scheduled and verified through the OWB Design Center, the OWB

Control Center is refreshed. After the Control Center is refreshed, all the jobs that are created appear on the left pane under the 'Scheduled Jobs' as shown in Figure A160.

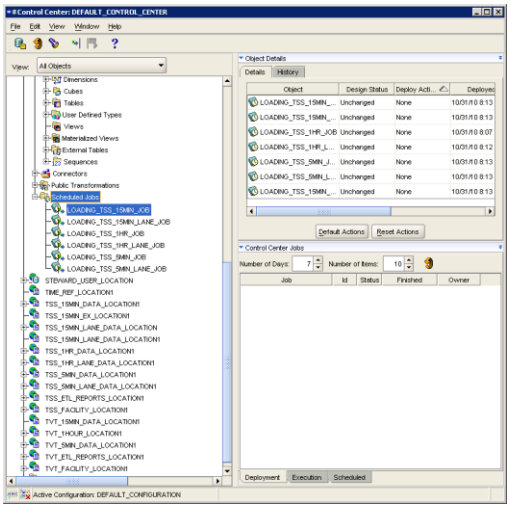


Figure A160. STEWARD Operations – Scheduled Jobs in Control Center

The scheduled jobs are then started in the Control Center of the OWB. Before the scheduled jobs are started, they are first deployed in the Control Center as described in Section A3.6 of this appendix. This is performed by right-clicking a particular job and by selecting the 'Start' button. As shown in Figure A161, the scheduled jobs are started.

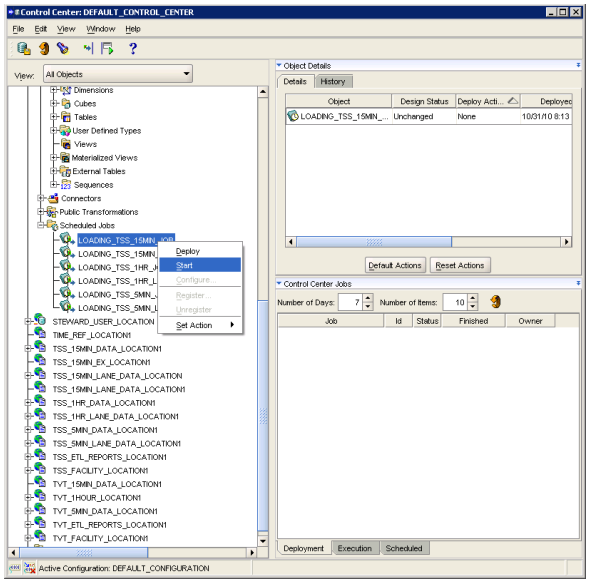


Figure A161. STEWARD Operations – Deploying and Starting Jobs

Once started, the data loading procedures will automatically occur for a data file at its respective time. It should be noted that the loading procedures and the job schedules are created manually only for the first time during the STEWARD implementation. The job scheduling is expected to run automatically and continuously after it is started. As shown in Figure A162, the status of the jobs can be seen on the right side of the Control Center Jobs.

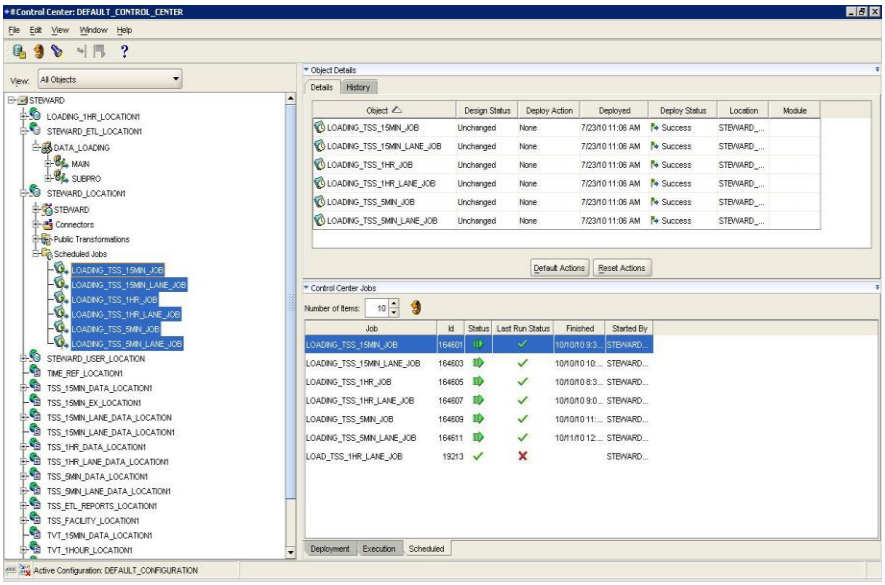


Figure A162. STEWARD Operations – Jobs Status

These jobs are expected to be started by the STEWARD operators every time the system or the database server has to be temporarily shut down. The possibility of starting the jobs to be a part of the automated process has not been studied yet.

After all the jobs are finished, only the station and the lane data are available at the STEWARD database. The next section describes the generation of the customized reports or the materialized views and its integration with the automated process.

A5.1.4 Daily updating of Reports on STEWARD

The last step in the automated process of STEWARD's daily operations is the creation/updating of materialized views. The materialized views are primarily the alternative name for the reports that are generated by the STEWARD system in the Oracle database environment. For the generation of these reports and their availability on the STEWARD system, it is necessary to update these materialized views. As new traffic data gets uploaded to the database on a daily basis, these customized reports are also expected to be updated on a daily basis. To update these materialized views, the OWB is configured and scheduled.

The first step in updating these materialized views is to configure the parameters associated with the respective views. After the system's performance was tested for different parameter settings, the refresh-on-demand method and the "refresh" FORCE method was used. These configurations are provided for a sample materialized view in Figure A163.

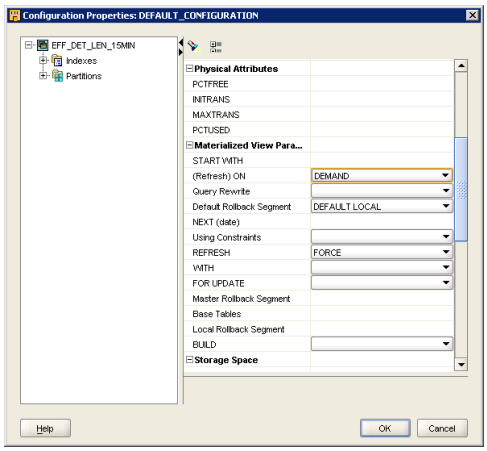


Figure A163. STEWARD Operations – Parameter Settings of Materialized Views

After the parameter settings are over, the materialized views are created using the Control Center of Design Center in the OWB. To create these, the Deploy Action on the right pane is changed to "Create" and the materialized views are deployed. After the creation of the materialized views, the views can be updated in two different ways. First, by changing the "Deploy Action" to "Replace" manually and then by deploying the respective action. This process deletes the last materialized view and creates a new materialized view as defined. This step is performed manually for all the materialized views that are updated and hence, this step becomes a highly undesirable way of becoming a part of the automated process of STEWARD's daily operations. Hence, a second way or an alternate way is used to update the materialized views.

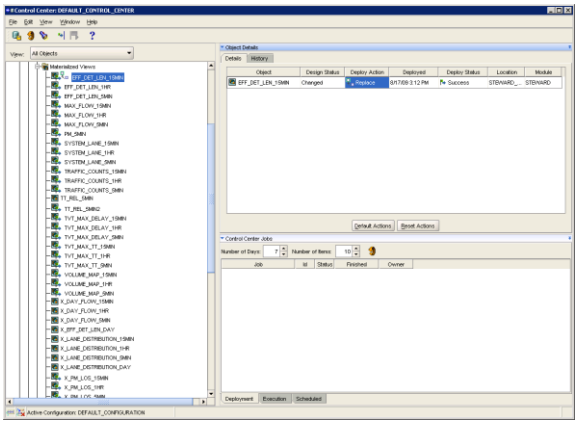


Figure A164. STEWARD Operations – Creating Materialized Views

To include the updating of materialized views within the automated process, it is necessary that these tables update automatically. To enable this, the FAST "refresh" option is used in the parameter settings as described in the previous step. After testing this option, it was realized that even after the FAST option was provided within the configuration settings, the materialized views were not updated automatically. Therefore a PL/SQL statement is run through the SQL software to enable the automatic updating of these materialized views. A sample code for this PL/SQL statement is provided at the end of this section. Within this statement, the refresh method and the time scheduler is configured. It came to the observation that enabling the FAST option within the PL/SQL statement led to the automatic updating of the materialized view.

In the current STEWARD system, the "Traffic Counts" reports or materialized views are updated with the help of these PL/SQL statements and these statements are executed at 3:00 AM every day after the station and lane traffic data are loaded onto the database. At this stage, only the "Traffic Counts" reports are provided to the users as they are updated daily. However, in future if the demand for other reports increases, the automatic updating of all the materialized views will be enabled except the "Performance Measures".

PL/SQL Create Program for Refreshing Materialized View:

```
BEGIN
DBMS_SCHEDULER.create_program (
   program_name   => 'mv_TRAFFIC_COUNTS_5MIN_refresh',
   program_type   => 'PLSQL_BLOCK',
   program_action => 'BEGIN DBMS_MVIEW.REFRESH("TRAFFIC_COUNTS_5MIN", "F","", TRUE, FALSE,
0,0,0, FALSE); END;',
   enabled        => TRUE,
   comments       => 'Program to refresh the materialized view TRAFFIC_COUNTS_5MIN.');
   END;

BEGIN
  DBMS_SCHEDULER.create_schedule (
   schedule_name   => 'TRAFFIC_COUNTS_5MIN_schedule',
   start_date      => SYSTIMESTAMP,
   repeat_interval => 'FREQ=DAILY; BYHOUR=3;',
   end_date        => NULL,
   comments        => 'Repeats daily, on the hour, forever.');
END;

BEGIN
```

```
DBMS_SCHEDULER.create_job (
   job_name      => 'TRAFFIC_COUNTS_5MIN_job',
   program_name  => 'mv_TRAFFIC_COUNTS_5MIN_refresh',
   schedule_name => 'TRAFFIC_COUNTS_5MIN_schedule',
   enabled       => TRUE,
   comments      => 'Job defined for refreshing TRAFFIC_COUNTS_5MIN_job.');
END;
```

This forms the last part of the automated process of the STEWARD operations. After the data is uploaded and the reports are generated, the users are able to access these reports through Internet.

A5.1.5 Data Backup for STEWARD System

Different categories of data backups are performed in the STEWARD system. These are the traffic data, and the STEWARD system that includes the database, and the web-interface.

Traffic Data Backup:

The different kinds of data that are backed up under this category are:

- **Raw Data (I):** The raw data received by the FTP server from the SunGuide systems are backed up. The backup takes place after all the archived data files are transferred. This is scheduled through the "SynBack" software at the following locations on the FTP server:
  F:\FTP_Backup\Districtx (*x* denotes the district number)

- **Raw Data (II):** Another backup is taken for the archived raw data during the execution time of the automatic utility. The location of this backup is specified in the automatic utility. This backup may be similar and identical to the previous backup and may utilize more space on the hard disks, but it has been proved useful for the STEWARD operators for keeping a track on the daily archived data.
- **Processed Data:** The processed data files or the aggregated data files are backed up during the execution time of the automatic utility. The location of this backup is specified in the automatic utility. These are archived to replace the missing data files during the data uploading process. The processed data are also archived to support requests from other research projects. The data also include 1-minute station or lane data that are not loaded to the database but are useful for research purposes.

STEWARD system backup:

The different kind of data back up under this category are:

- **Web Server:** The directory that contains all the files and folders used to host the STEWARD's web-interface are also backed up on a regular basis. This back up is performed manually on the database server at the following location: D:\STEWARD_Website
- **Database:** The entire STEWARD's database is backed up using the Oracle Enterprise Manager (EM) and this is performed automatically with the help of the EM graphical user interface. These are stored at the Oracle Home folder of the database server.
- **System Configuration:** The database configuration that includes several processes and schedules are backed up regularly through the OWB. The backup is performed manually using the *Export Warehouse Builder Metadata* function in the OWB as shown in Figure A165. These files are archived in Oracle Home folder of the database server.

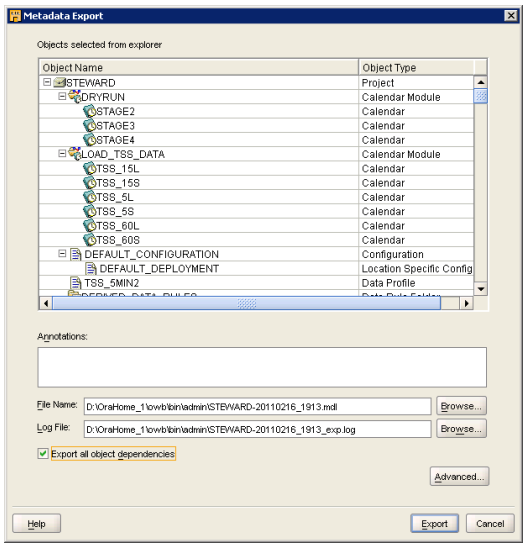


Figure A165. STEWARD Export Warehouse Builder Metadata

It should be noted that most of the traffic data backups are done on the new internal hard drives of the FTP server through the automatic utility or the "SynBack" software, whereas the STEWARD system back-ups are done at the dedicated drive for the STEWARD database.

A5.2 STEWARD Operations - Adding a New District

To add a new district to the STEWARD database, the facility file containing all the detector information is prepared first. The next portion of this section describes the steps taken to prepare/configure the new facility files, followed by changes that are made to accommodate the additional district.

A5.2.1 Steps to Configure a New Facility file

The steps that are followed to prepare new facility files are briefly described next:

1. Develop a list of all stations and lanes from the TSS archive files of the new district. This is performed by a special utility program called TSSBuilder which reads these archive files and compiles a list of all the station and lane ID's that exist in the TSS raw data files. A detailed description of TSSBuilder application is available at the resources section of the STEWARD web-interface.

2. All the stations that are identified in the first step are assigned to a facility of the respective district. Different roadways in a particular district are named as the respective facilities. For e.g., I-95 in District 6 is one facility and SR-826 is another facility. The facility assignments are carried by the respective district personnel. In the STEWARD system, every district may have up to 10 facilities, numbered from 0 through 9.

3. After the facility is assigned to each station, the station ID's are given to each station. The station numbers are given in a sequence. The order of this sequence is not important as long as each number represents a unique station ID. The last digit of the station ID's are given according to the direction of traffic. If the direction is NB/EB, the last digit becomes 1 else, for SB/WB, the last digit is 2.

4. The positions of each station are established. For this, the user must have the coordinates, state milepost, RCI road number and county milepost from the respective districts. These are provided by the TMC's in the form of facility information files.

5. The station status of each station is established. Typically these are 0 for a normal working detector and 1 for an offline station.

6. The lane ID's are assigned to each lane. This process is the most time-consuming and the most detailed part of creating a facility file. The station number in Step 3 forms the first 6 digits of the lane ID's. Two more digits are assigned to the lane ID's after the 6 digits. The following steps determine the last two digits of the lane ID.

   The penultimate digit of the lane ID's are given according to the type of lane. These numbers are given to each lane according to the following convention:

   1 for left entrance ramp, 2 for left exit ramp, 3 for normal freeway mainline, 4 for right entrance ramp, 5 for right exit ramp

The last digit of the lane ID's is given to every lane according to the location of the respective lane from the median or the shoulder. In general, the lane numbers are assigned from left to

right. For e.g., the lane next to the median or the leftmost lane is given as value 1 and the lane next to the leftmost lane is given as 2 and so on until the number of lanes at a particular station location. However, this convention is different for every district and their facilities. It is necessary that the STEWARD operators communicate with the respective TMC's for detailed information on the lane configurations and create most accurate files.

A5.2.2 Changes in ETL Process

The main impact of adding new district or facility would be to guarantee the quality of TSS data and to update the facility file into good working condition. In general, there needs to be several communications between the STEWARD operators and respective district while receiving new district information, so that the STEWARD operators can set up a new facility file. In some case, the new facilities could be found in the TSS data files without any notifications. In this case, the increased number of orphan stations in the ETL report file could be used as the indicator of new facility data in the TSS data file.

There were several issues on the ETL process to include the new district data into STEWARD:

- Basic quality checks are executed on new data. If it fails, STEWARD holds the data until this problem is resolved. One of the issues is negative or zero scan intervals for its TSS data. For e.g., District 2, 5 and 6 had more than 10% of negative or zero scan intervals in its daily data. It is expected that the TSS data files have all the traffic data logged in a time order but some records may appear to have older time stamps than the previous record (negative scan) or a case where duplicated time stamps from the same location at the same time (zero scan) are observed. All these case are very important in resolving the data checks issue. In the case of District 2 or 6, updating RTMS driver appears to resolve the issue.

- Facility data files might need to be created, updated and verified. These data would have a great impact on the quality of the STEWARD data. Therefore, roadway configuration data from the FDOT Roadway Characteristic Inventories (RCI) and Google satellite maps should be used as necessary to verify the facility data.

Adding a new district will involve the following actions:

- Updates in the FTP server: Currently, STEWARD is set up for the District 2, 4, 5, 6, 7, and the Turnpike. It might be necessary to update this configuration.
- Each district must have a configuration workbook (TSSMaster-Dx.xls) file for its facility data. This file includes all of the station and lane information for the district TMC. This file needs to be created and verified with other data sources, such as FDOT RCI database or Google satellite maps.

Adding new facility will require the following action:

- Update the facility data file (TSSMaster-Dx.xls) to accommodate all of the stations and lanes in the new facility.

A5.2.3 Updates to the STEWARD Database

While a new district or a facility file is added, there are no design changes in the STEWARD database scheme. Only the TSS facility table of the database, i.e. TSS_STATION should be updated by adding the new stations to the master table of the STEWARD database.

A5.2.4 Updates to the STEWARD Web Interface

There might be two types of changes to incorporate the new district or facility to update the STEWARD web interface and to update the STEWARD web report interface.

Updating the STEWARD web interface would be relatively easy by updating an HTML/ASP program as follows:

- Adding or replacing new/update facility file in the resources page
  Facility file location: C:\Inetpub\wwwroot\steward\resources

  Resource web page: C:\Inetpub\wwwroot\steward\resources.html

- Adding or replacing new/update facility file in the maps page
  Map file location: C:\Inetpub\wwwroot\steward\maps

  Maps web page: C:\Inetpub\wwwroot\steward\maps.html

- Activating the new district in the floating menu
  Left pane web page: C:\Inetpub\wwwroot\steward\left_menu.asp

Updating STEWARD web reports might some new code in the following folders.

- Update TSS facility level reports
  Main folder for TSS section level reports

  C:\Inetpub\wwwroot\steward\index_main\TSS_Facility_Level_Reports

  TSS facility level reports folder

  C:\Inetpub\wwwroot\steward\index_main\TSS_Facility_Level_Reports\volume_map

  C:\Inetpub\wwwroot\steward\index_main\TSS_Facility_Level_Reports\traffic_counts

  C:\Inetpub\wwwroot\steward\index_main\TSS_Facility_Level_Reports\lane_volumes

- Update TSS section level reports

  Main folder for TSS section level reports

  C:\Inetpub\wwwroot\steward\index_main\TSS_Section_Level_Reports

  TSS section level reports folder

  C:\Inetpub\wwwroot\steward\index_main\TSS_Section_Level_Reports\performance_measure

  C:\Inetpub\wwwroot\steward\index_main\TSS_Section_Level_Reports\TT_Reliability

- Update TSS station level reports

  Main folder for TSS section level reports

  C:\Inetpub\wwwroot\steward\index_main\TSS_Station_Level_Reports

  TSS section level reports folder

  C:\Inetpub\wwwroot\steward\index_main\TSS_Station_Level_Reports\volume_data

  C:\Inetpub\wwwroot\steward\index_main\TSS_Station_Level_Reports\traffic_counts

  C:\Inetpub\wwwroot\steward\index_main\TSS_Station_Level_Reports\max_flow_rate

  C:\Inetpub\wwwroot\steward\index_main\TSS_Station_Level_Reports\effective_detector_length

APPENDIX – B

Script for creating a Job for MView TRAFFIC_COUNTS for 5 mins:

B1

```
BEGIN
DBMS_SCHEDULER.create_program (
   program_name   => 'mv_TRAFFIC_COUNTS_5MIN_refresh',
   program_type   => 'PLSQL_BLOCK',
   program_action => 'BEGIN DBMS_MVIEW.REFRESH("TRAFFIC_COUNTS_5MIN", "F","", TRUE, FALSE,
0,0,0, FALSE); END;',
   enabled        => TRUE,
   comments       => 'Program to refresh the materialized view TRAFFIC_COUNTS_5MIN.');
   END;
```

B2

```
BEGIN
  DBMS_SCHEDULER.create_schedule (
   schedule_name   => 'TRAFFIC_COUNTS_5MIN_schedule',
   start_date      => SYSTIMESTAMP,
   repeat_interval => 'FREQ=DAILY; BYHOUR=3;',
   end_date        => NULL,
   comments        => 'Repeats daily, on the hour, for ever.');
END;
```

B3

```
BEGIN
DBMS_SCHEDULER.create_job (
   job_name      => 'TRAFFIC_COUNTS_5MIN_job',
   program_name  => 'mv_TRAFFIC_COUNTS_5MIN_refresh',
   schedule_name => 'TRAFFIC_COUNTS_5MIN_schedule',
   enabled       => TRUE,
   comments      => 'Job defined for refreshing TRAFFIC_COUNTS_5MIN_job.');
END;
```