DriveRight® Fleet Management Software

Database Selection Guide

Use this guide to help you select the right database to use with your DriveRight Fleet Management Software (FMS), and to help you perform any required database installation procedures before you run DriveRight FMS for the first time.

DriveRight Fleet Management Software (FMS) supports the following databases.

- Single-User Databases:
  - MS Access (Microsoft Access) (all required software included)
  - MSDE (Microsoft Desktop/Data Engine, all required software included)
- Multi-User Databases:
  - MySQL
  - MS SQL Server (Microsoft SQL Server)
  - Oracle

Note: If you are using a MS SQL Server or an Oracle Server, you must run stored procedures from the FMS installation disk once you have installed FMS. If installing the software directly to the server or from a secured client, the stored procedures should run automatically. If the software can not be installed directly to the server, the stored procedures should be run separately.

When you first run your DriveRight FMS, you will be required to select a database to use with the software. Each of the supported databases for DriveRight FMS have their pluses and minuses. First, we'll look at the selection considerations for each of the databases. Later in this document we present the installation procedure for each of the databases.

Selection Considerations: Single-User

So, which database should you use? Review the following considerations for each database and then select the best database for your situation.

MS Access Considerations

Of the supported databases, MS Access is the easiest to install and use.

- There is no additional software to buy if you use MS Access. DriveRight FMS will automatically install all the necessary components when you first run the program.
- MS Access does have limitations to the number of users connected simultaneously to the database, the size of database, and reliability of database.
- MS Access is only recommended for installations that have one user.

- Maximum database size for MS Access is 2 gigabytes.
- MS Access is only recommended for small fleets of five vehicles or less. Be aware that if you are logging GPS data (required the DriveRight 600 with optional GPS), the database size increases very rapidly since GPS data is logged much more frequently than data from the DriveRight console. In such cases, if you decide to go with MS Access database, it is advised to backup your data at least once a month.

MSDE Considerations

MSDE stands for Microsoft Desktop/Data Engine, which is a scaled down version of SQL Server. It is the basic database with Client/Server architecture from Microsoft.

- There is no additional software to buy if you use MSDE. MSDE setup is included as an option during DriveRight FMS installation.
- MSDE works well with a maximum of five simultaneous connections.
- Since in some instances DriveRight FMS software maintains more than one connection to the database server, MSDE is also recommended for single-user installations.
- In contrast to MS Access, MSDE runs as a Window’s service. So if your computer has performance issues, we recommend MS Access instead of MSDE.

Note: In our database performance tests, MS Access performed better than MSDE.

Active Database Size Recommendations

The recommendations displayed in the table below are based on tests and calculations made with these assumptions:

- 10 trips per day per vehicle.
- Trip duration 1 hour (10 hours driving time per day per vehicle).
- GPS data logged every 10 seconds.

Scenario	Number of Vehicles	Active Data stored in Database	Remarks and Recommendations
No GPS	25	Up to 1 year	Backup data once a month for application performance. Using automation options you can set maximum of 1 year of data in the active database.
	50	Up to 9 months	Backup data once a month for application performance. Using automation options you can set maximum of 9 months of data in the active database.

No GPS	100	Up to 6 months	Backup data once a month for application performance. Using automation options you can set maximum of 6 months of data in the active database.
MSDE with GPS	5	Up to 1 month	Backup data every month for application performance. Using automation options you can set maximum of 1 months of data in the active database.
MS Access with GPS	10	Up to 1 month	Backup data every month for application performance. Using automation options you can set maximum of 1 months of data in the active database.
Multi-User Database with/without GPS	Unlimited	3-6 months	Backup data every month without GPS or once a week with GPS. Using automation options you can set a maximum of 6 months data in the active database.

Note: If your fleet size exceeds the scenarios shown above, we recommend that you use a multi-user database. For multi-user databases, theoretically there is not fleet size limit. Still, it is recommended that you back up data once a month, keeping only the amount of data required for reporting purposes in the active database, typically 3-6 months.

Selection Considerations: Multi-User

We recommend using either MySQL, MS SQL Server, or Oracle if more than one user will be accessing the data at a time, if the fleet has more than five vehicles, or if you are logging GPS data for more than one vehicle.

MySQL, MS SQL Server & Oracle Considerations

MySQL, Oracle and MS SQL Server database servers are very powerful databases, which can support large amounts of data and multiple simultaneous connections with good performance.

- Of these three powerful databases, MySQL is the most economical server with a cost of approximately $220 (as of April 2003). It is by far the easiest to setup and maintain and is also the only multi-user database used by Drive Right FMS that does not require the support of a database administrator. We strongly recommend MySQL for customers with large fleets and potentially large databases that do not already have a license for MS SQL Server or for Oracle.
- MS SQL Server is a powerful database server from Microsoft. It is also easy to setup, but MS SQL Server is more expensive than MySQL.
- If you decide to go with MS SQL Server or Oracle, it is assumed that you have IT personnel who have knowledge to maintain these database servers.

- In the case of MySQL, in the future most of the maintenance commands will be integrated through the DriveRight FMS software, to eliminate any administrative overhead.
- Though Oracle is a common database server among large applications, it is both expensive and difficult to maintain.
- In order to connect to an Oracle server with DriveRight FMS running on multiple machines, Oracle Client software must be installed on each machine. Each installation of Oracle Client software must be individually licensed.

Multi-User Database Server Setup

Follow this sequence when setting up a multi-user database for DriveRight FMS:

1. Server Installation.
2. Client Installation.
3. Database Configuration.

Multi-User Database Client Setup

Follow this sequence when installing DriveRight FMS on a user's computer:

1. Client Installation.
2. Database Connection.

MS Access Installation

Of all the databases, MS Access is the easiest to setup. DriveRight FMS automatically installs all the required components for this database.

Server Installation

Not applicable. The server is automatically installed if it is selected during DriveRight FMS initial configuration.

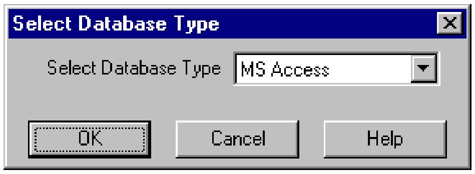
Database Configuration

Not applicable. The database is utomatically configured if it is selected during DriveRight FMS initial configuration.

Database Connection

Please refer to the "DriveRight Fleet Management Software Getting Started Guide" for DriveRight FMS installation and configuration instructions:

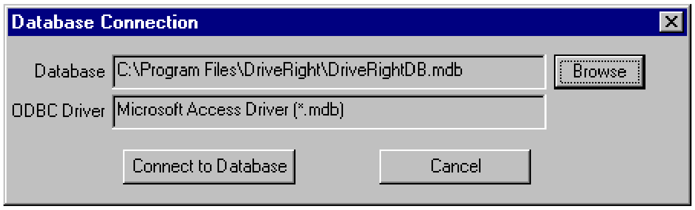
1. During the initial DriveRight FMS configuration, select MS Access as database type.



Select Database Type

Select Database Type: MS Access

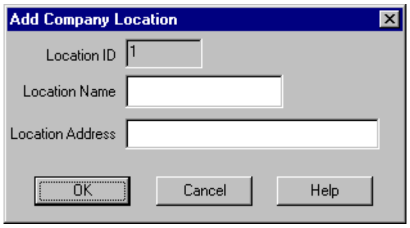
A dialog box is displayed with the path in which the database will be created. This defaults to the DriveRight FMS install directory.



Database: C:\Program Files\DriveRight\DriveRightDB.mdb

ODBC Driver: Microsoft Access Driver (*.mdb)

2. If you want to change the directory used for the MS Access database, click the Browse button and select the desired directory.
3. To finish the database connection, click **Connect to Database**. DriveRight FMS creates a new database if the database doesn't exist, or it connects to the specified database if the database already exists.
4. If the database is empty when you connect, you are prompted to enter the location name and address to be associated with all data at your location. After entering the location name and address, click Add and the DriveRight software will start.



Location ID: 1

Location Name:

Location Address:

5. We recommend that only one user at a time access the MS Access database.

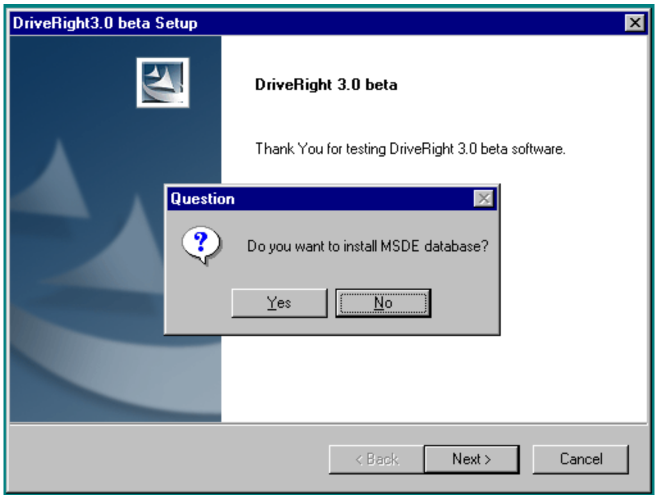
MSDE Installation

MSDE is another option that is included during the DriveRight FMS installation.

Server Installation

1. During DriveRight FMS installation, you will be asked if you want to install the MSDE database. Click **Yes**.

It will take approximately 10 minutes to install MSDE on your computer.


DriveRight3.0 beta Setup

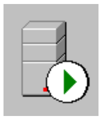
DriveRight 3.0 beta

Thank You for testing DriveRight 3.0 beta software.

Question

Do you want to install MSDE database?

2. You need to make sure the MSDE Server is running. To do this go to: Startup->Programs->MSDE->Service Manager. Press the green arrow “Start/Continue” button to start MSDE. When MSDE is running, you will see a green indicator in the MSDE icon on the task bar.

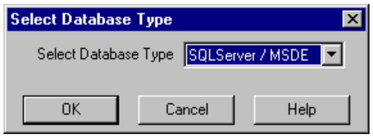


Database Configuration

Not applicable. Automatically configured when MSDE is selected during DriveRight FMS initial configuration.

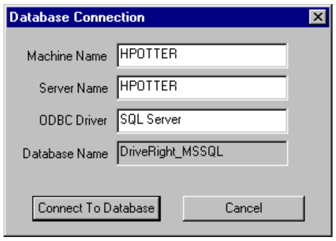
Database Connection

1. When you run DriveRight FMS the database type selection dialog appears. Select SQL Server / MSDE and Click OK.


Select Database Type

Select Database Type: SQLServer / MSDE

2. The **Database Connectio**n dialog box displays with the following fields after selecting MSDE as your database:


Database Connection

Machine Name: HPOTTER

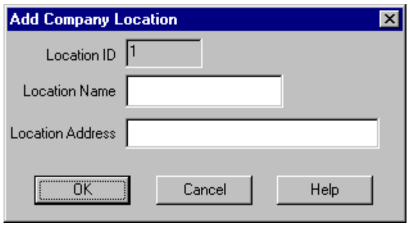
Server Name: HPOTTER

ODBC Driver: SQL Server

Database Name: DriveRight_MSSQL

- **Machine Name, Server Name —** Both these are defaulted to the local machine name. You do not need to change these.
- **ODBC Driver —** This need not be changed in almost all cases. If you have version 2.5 of SQL Server driver (unlikely), then you need to edit it to look like {SQL Server}.
- **Database Name —** Defaults to DriveRight_MSSQL and cannot be changed.

3. If the database is empty when you connect, you are prompted to enter the location name and address to be associated with all the data at your location. After entering the location name and address, click Add and the DriveRight software will start.



Location ID 1

4. It should be noted that the MSDE database server is optimized for a maximum of 5 simultaneous connections. Considering the fact that the DriveRight FMS software may keep more than one connection alive, it is advisable to limit the number of users to one at a time.

MySQL Installation

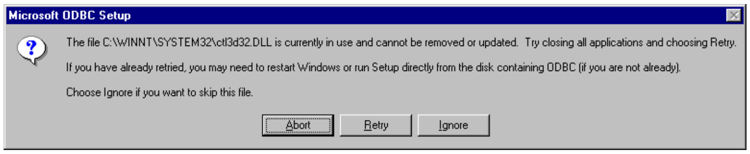
MySQL is a database server developed by a company named 'MySQL AB' as an Open Source Project. It is considered fast, stable, and is easy to setup compared to database servers like Oracle. Also, at a price tag of about $220 (as of April 2003), it is by far the least expensive multi-user database server for use with the DriveRight FMS. For medium to large fleets, we strongly recommend MySQL server, especially if you do not already have a license for either Oracle or MS SQL server.

Server Installation

1. If you don’t already have a license for MySQL, you need to buy one from their website: www.mysql.com. MySQL Classic version is adequate for all DriveRight FMS applications.
2. Install MySQL Server. During installation of MySQL server, the setting of user names and passwords is optional.
3. You need to make sure MySQL is running first. To do this go to Startup->Settings->Control Panel->Administrative Tools->Services. You will see a service name titled MySQL. If it is not running, select it and right click to start it.

Client Installation

1. The client software, MySQL Connector/ODBC 2.50, must be installed on all computers that will be running DriveRight FMS. This client software can be downloaded for free if you have purchased a license for MySQL server, and can be installed on any number of computers.
2. During the installation of the MySQL Connector/ODBC 2.50, the **Microsoft ODBC Setup** dialog box displays.Click **Ignore**.



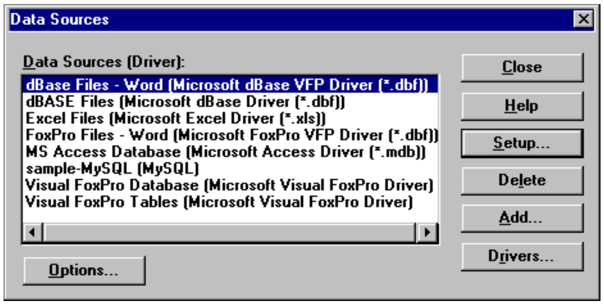
Microsoft ODBC Setup

The file C:\WINNT\SYSTEM32\ctl3d32.DLL is currently in use and cannot be removed or updated. Try closing all applications and choosing Retry.

If you have already retried, you may need to restart Windows or run Setup directly from the disk containing ODBC (if you are not already).

Choose Ignore if you want to skip this file.

3. Click **Close** in the **Data Sources** dialog box. You are finished installing the client software.



Data Sources

Data Sources (Driver):

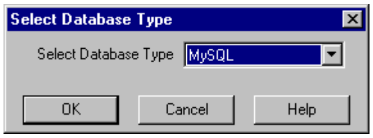
dBase Files - Word (Microsoft dBase VFP Driver (*.dbf))
dBASE Files (Microsoft dBase Driver (*.dbf))
Excel Files (Microsoft Excel Driver (*.xls))
FoxPro Files - Word (Microsoft FoxPro VFP Driver (*.dbf))
MS Access Database (Microsoft Access Driver (*.mdb))
sample-MySQL (MySQL)
Visual FoxPro Database (Microsoft Visual FoxPro Driver)
Visual FoxPro Tables (Microsoft Visual FoxPro Driver)

Database Configuration

1. With MySQL running, install and run the DriveRight FMS on the same machine as the database server. This is only required for database configuration. After the database is configured, remove DriveRight FMS from the server. When the database configuration is complete, you will be able to connect to the MySQL server from any PC on the local area network.

Note: If you are running MySQL on a non-Windows server, you need to manually execute the SQL scripts that configure your database with the necessary database tables manually. The scripts are located in the "mySQL_TablesCreation.sql" file on the install CD.

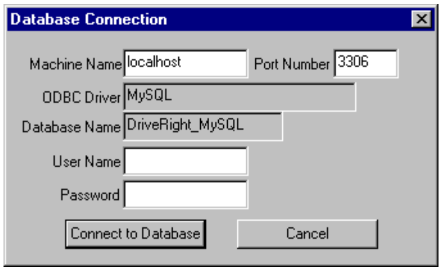
2. When you run the DriveRight FMS, the **Database Type Selection** dialog displays. Select **MySQL**, and Click **OK**.



Select Database Type

Select Database Type: MySQL

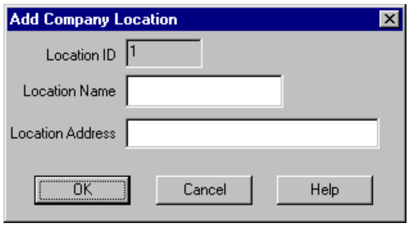
3. Click **Connect to Database** in the **Database Connection** dialog box.



Database Connection

Machine Name: localhost
Port Number: 3306
ODBC Driver: MySQL
Database Name: DriveRight_MySQL

4. If the database is empty when you connect, you are prompted to enter the location name and address to be associated with all the data at your location. After entering the location name and address, click Add and the DriveRight software will start.

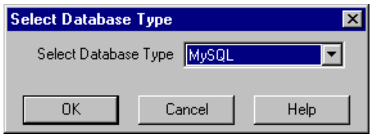


Add Company Location

Location ID: 1

Database Connection on User’s Computer

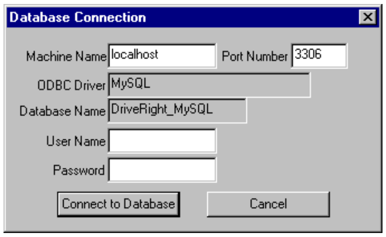
1. From this point on, anyone who installs DriveRight FMS needs to enter the IP address noted above in the 'Machine Name' edit box of the Database Connection dialog.
2. When you run DriveRight FMS for the first time, the **Database Type Selection** dialog box displays. Select **MySQL** and Click **OK**.



Select Database Type

Select Database Type: MySQL

3. The **Databse Connection** dialog box displays. Enter the machine name of MySQL Server in the **Machine Name** text box. Contact your system administrator for the MySQL machine name.



Database Connection

Machine Name: localhost
Port Number: 3306
ODBC Driver: MySQL
Database Name: DriveRight_MySQL

4. Click **Connect to Database** to finish the database connection.

MS SQL Server Installation

MSDE and MS SQL Server are compatible databases developed by Microsoft. MSDE is a single-user database while MS SQL Server is a multi-user, server based database. MS SQL Server also requires the services of a database administrator to install and maintain.

Server Installation

1. If you don't already have a license for MS SQL Server, buy one from Microsoft.
2. Install MS SQL Server.
3. You need to make sure MS SQL Server is running. To do this go to: Startup->Programs->Microsoft SQL Server->Service Manager. Press the green arrow "Start/Continue" button to start MS SQL Server. When the server is running, you will see a green indicator in the MS SQL Server icon on the task bar.

Client Installation

Not applicable.

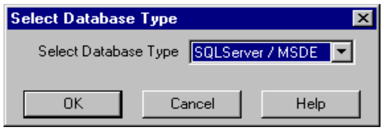
Database Configuration

The MS SQL server can be configured two ways, on the server itself or from a user's computer.

Note: The server can also be configured without installing DriveRight FMS on the server, by performing the necessary stored procedure tasks on the server itself. see "Appendix A: Installing MS SQL Stored Procedures Manually" on page 16.

To Configure the Database on the Server:

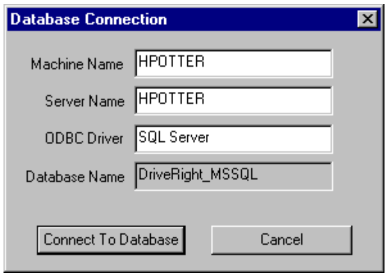
1. With MS SQL Server running, install and run the DriveRight FMS on the same machine as the database server. This is only required for the database configuration. After the database is configured you can uninstall the DriveRight FMS from the server. When the database configuration is complete, you will be able to connect to the MS SQL Server from any PC on the local area network.
2. When you run the DriveRight FMS for the first time, the **Select Database Type** dialog box displays. Select SQL Server / MSDE and Click **OK**.



Select Database Type

Select Database Type: SQLServer / MSDE

3. When FMS is run for the first time, the **Database Connection** dialog box displays after selecting SQL Server / MSDE as your database.



Machine Name: HPOTTER

Server Name: HPOTTER

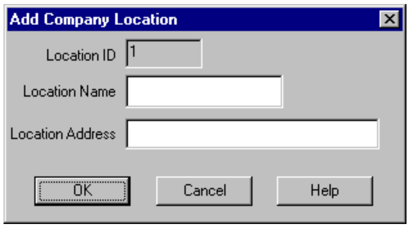
ODBC Driver: SQL Server

Database Name: DriveRight_MSSQL

The **Database Connection** dialog box displays the following fields:

- **Machine Name, Server Name** — Both of these default to the local machine name. You will need to change these.
- **ODBC Driver** — This will not need to be changed in most installations. If version 2.5 of the SQL Server driver is installed (unlikely), then you will need to edit it to look like {SQL Server}.
- **Database Name —** Defaults to DriveRight_MSSQL and cannot be changed.

4. Once you have entered the pertinant information, click **Connect to Database.** The FMS software launches and connects to the database.
5. If the database is empty when you connect, you will be prompted to enter location name and address to be associated with all the data at your location. After entering the location name and address, click Add and the DriveRight FMS software will start.

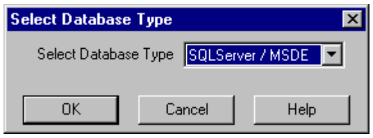


Location ID: 1

6. Once the software has been configured on the MSDE/SQL Server, give permissions to users to connect from multiple machines. This depends on how your SQL Server is setup. A simple approach would be to grant permissions to all Windows users to connect. In this case, the Windows Login/ Password is used to authenticate. This can be setup by running SQL Server Enterprise Manager, going to Security->Logins and adding either individual users or domain users.
7. From this point on, anyone who installs DriveRight FMS needs to enter the MS SQL Server machine name in the **Machine Name** edit box in the **Database Connection** dialog box. Contact you database administrator for the MS SQL Server machine name.

To Configure the Database on a User's Computer:

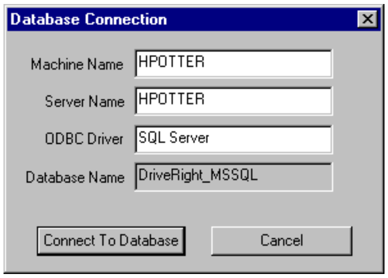
1. When you run the DriveRight FMS software for the first time, the **Database Type Selection** dialog box displays. Select SQL Server / MSDE and Click **OK**.



Select Database Type

Select Database Type: SQLServer / MSDE

2. The **Database Connection** dialog box displays:



Database Connection

Machine Name: HPOTTER

Server Name: HPOTTER

ODBC Driver: SQL Server

Database Name: DriveRight_MSSQL

3. Enter the **Machine Name** and **Server Name**.
   - **Machine Name, Server Name** — Both these default to your machine name. You need to change these. Contact your system administrator for the MS SQL Server Machine Name and Server Name. In most cases, these names are the same.
   - **ODBC Driver —** This need not be changed in most cases. If you have version 2.5 of the SQL Server driver installed (unlikely), then you will need to edit it to look like {SQL Server}.
   - **Database Name —** Defaults to DriveRight_MSSQL and cannot be changed.
4. Once the information is entered, click **Connect to Database.** The FMS software launches and connects to the database.

Oracle Installation

Since Oracle server installation and administration are complicated, you will require an Oracle database administrator to use Oracle with DriveRight FMS.

Server Installation

1. If you don't already have a license for Oracle server, buy one from Oracle.
2. The DriveRight FMS software has been tested with the Oracle9i server and with Oracle9i client. Since the DriveRight FMS software executes only the standard SQL statements, the software should work with previous versions of Oracle server and client.

3. Create a database with SID as *DRight* using the Database Configuration Assistant.
4. Create a Net Service Name called *DRight*, with the SID created in Step 3.
5. Create a user name and password for FMS users to connect to Oracle server.

Note: Once a connection to the Oracle server is made, all the stored procedures for creating the FMS database tables must be run in the server.

Client Installation

The Oracle client must be installed on all computers running DriveRight FMS. Please check with your Oracle database administrator for client licensing.

The following steps are for installing Oracle Client 9i. If you install a different version of Oracle Client, some steps might vary.

Note: Please follow these steps carefully and in the sequence in which they are written. Oracle installation is a time consuming process, and if you go wrong at some place you may have to reinstall the whole Oracle Client again.

Oracle Client 9i Installation:

1. Locate and run the Oracle Client CD. Click **autorun.exe**. The Oracle9i Client installation screen displays.
2. Click **Install/Deinstall Products**. The **Welcome** dialog box displays.
3. Click **Installed Products**. If the dialog box says you have no installed products, go to step 4. If an older version exists, a dialog box pops up with the details. Remove these before installing 9i. The dialog box displays the installed products as a tree structure. Expand the tree to display the check boxes. Expand and check all the shown items, and click **Remove**. Wait until every thing is deinstalled.
4. Now you are ready to install 9i Client. Click **Next**. In the **Next** dialog box, click **Next**. This step takes a few seconds.
5. In the **Installation Types** dialog box, select **Administrator** and click **Next**.
6. In the **Summary** dialog box, click **Install**.
7. If the installation is successful, the **Oracle Net Configuration Assistant** displays. This creates the **Net Service Name**, which is used during the creation of the DSN. Select the **No, I will create net service names…** and click **Next.**
8. Select **Oracle8i or later…** and click **Next**.
9. Enter the **Service Name** as **DRight** and click **Next**.
10. Select **TCP** and click **Next**.
11. Enter your Oracle server IP address as the **Host Name** and click **Next**. Contact your database administrator for the Oracle server IP address. If

this step takes more than 10 minutes, please skip the remaining steps and see “Creating Listener and Net Service Name Manually” on page 14.

12. Click **Yes, perform a test** and then click Next.
13. You will most likely see a success message. Click **Change Login** and change both user name and password to the user name and password given by your system administrator. Go back and click **Next**, it performs a test connection with the new user name and password.
14. Leave **Net Service Name** as it is and click **Next**.
15. When you are prompted to configure another Net Service Name, select **No** and click **Next.** Then click **Finish**.
16. Click **Exit**. The Oracle 9i client installation is finished. See “Creating and Configuring Oracle DSN” on page 15 for the next steps in the process.

Creating Listener and Net Service Name Manually

If you are in this section, you had problems creating a service name while creating Oracle Client. If every thing worked fine in the previous section move on to the next section.

1. First close the present **Oracle Client Installation** dialog box. A message may display that states that some things are not installed correctly. Use the following steps to correct the process.
2. Go to Programs->Oracle-OraHome90->Configuration and Migration Tools->Net Manager.
3. Expand the directory to Local->Listeners. If LISTENER is displayed, a listener was created during the 9i Client installation. If so, see step 5 below.
4. Click on Listeners, and then click the '+' icon on the left side if LISTERNER is not displayed. Leave the name as 'LISTENER'. Then click **Add Address** and leave the defaults alone.
5. Click **Add Address** again, and select **Protocol as IPC** from the **Protocol** drop down box. Enter the key as 'EXTPROC0' (The last character is a zero, and the character before C is capital letter O).
6. Expand **Service Naming**. If you see a child node, with a name similar to dright.davisnet.com, continue on to Step 9.
7. Click **Service Naming** and then click the **+** icon. Add **Net Service Name** as 'DRight'. Click **Next** twice. In the **Host Name**, enter the Oracle server IP address. Click **Next** to continue.
8. Select **Oracle8i or later**, and enter **DRight** in the text box. Click **Next**.
9. Click **Test**. If this step fails, please contact your database administrator. If this step succeeds, click **Change Login** and change both the user name and password to the user name and password provided by your database administrator.
10. Click **OK** and click **Test** again. Then close the dialog box.
11. Click **Finish**.

12. Expand **Service Naming**, select the child node which looks similar to: (DRight.*your_company_name*.com). Check for the parameters. Host Name should be the IP address of your Oracle server.
13. Close **Net Manager**, and click **Save** when it prompts you. See “Creating and Configuring Oracle DSN” on page 15 for the next steps in the process.

Creating and Configuring Oracle DSN

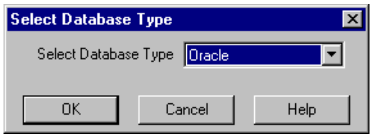
1. Go to Start->Settings->Control Panel.
2. Double click **Administrative Tools**, and select **Data Sources**.
3. In the **Data Sources** dialog box, select the **System DSN** tab.
4. Click **Add**, and in the list of drivers select **Oracle in OraHome90** (located almost at the bottom of the list) and click **Finish.**
5. In **Data Source Name**, enter **DriveRightOracle**. In **TNS Service Name**, select DRIGHT. Enter the user name and password provided by your database administer and click **OK**. The **Connection Successful** dialog box displays. If the **Connection Successful** dialog box does not display, check for any spelling mistakes and try again.
6. Click **OK**. The **Data Source Name** displays in the list of **System Data Sources**.

Database Configuration

1. With the Oracle server running, install and run the DriveRight FMS software on the same machine. This is only required for database configuration. After the database is configured you will be able to connect to the Oracle Server from any PC on the local area network.

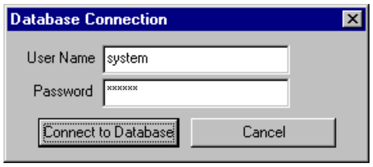
Note: If you are running Oracle on a non-Windows server, you need to manually execute the SQL scripts to configure your database. The scripts are located in the “oracle_TablesCreation.sql” file on the install CD.

2. When you run **DriveRight FMS** for the first time, the **Database Type Selection** dialog box displays. Select **Oracle** and click **OK**.



Select Database Type: Oracle

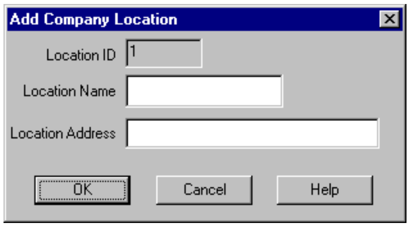
3. The **Database Connection** dialog box displays:



User Name: system

Password: ******

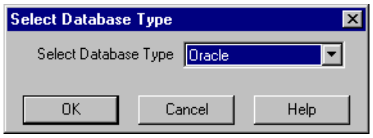
4. Enter the user name and password created in step 4 of the **Server Installation** procedure.
5. Click **Connect to Database** to finish the connection or click **Cancel** to exit.
6. If the database is empty when you connect, you are prompted to enter the location name and address to be associated with all the data at your location. After entering the location name and address, click **OK** and the DriveRight software will start.



Location ID 1

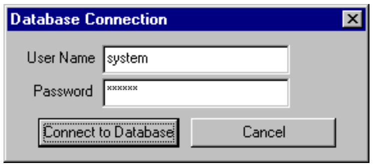
Database Connection on the User's Computer

1. When you run DriveRight FMS for the first time, the **Database Type Selection** dialog box displays. Select Oracle and Click **OK**.



Select Database Type Oracle

2. The **Database Connection** dialog box displays:



User Name system
Password ******

3. Enter the user name and password provided by your database administrator.
4. Click **Connect to Database** to finish the connection or click **Cancel** to exit.

Appendix A: Installing MS SQL Stored Procedures Manually

If a direct configuration of the database files on the MS SQL server can not be permitted, the stored procedures can be installed manually by using the following instructions. Use the following steps to create stored procedures in the database server.

The stored procedures can be entered manually using the script for each stored procedure below or they can be cut and pasted from the DriveRight_StoredProcedure_SQLServer.doc file located in the C:\Program Files\DriveRight directory.

1. Go to Start->Programs->Microsoft SQL Server->Enterprise Manager. The **SQL Server Enterprise Manager** displays.
2. On the left hand side pane, expand the directory displayed to:
   SQL Server Group->Local-> Databases->DriveRight_MSSQL.
3. Right click on **Stored Procedures** and select **New Stored Procedure**.
4. Paste or type in the first stored procedure (AccidentLogs Table) below in the text area, and click **Check Syntax** to verify that there are no errors in the stored procedures. If there are no errors, the **Syntax check successful** dialog box is displayed. Click **OK** to exit.
5. Repeat above steps for all the stored procedures displayed below.

For each stored procedure, enter or paste all text starting at CREATE PROCEDURE and ending with END.

AccidentLogs Table

```
CREATE PROCEDURE dr_insert_accidentLog @field1 smallint,
@field2 smallint,@field3 datetime,@field4 smallint,
@field5 char(1),@field6 char(1),@field7 smallint,@field8 char(1),
@field9 smallint,@field10 char(1),@field11 smallint,
@field12 char(1),@field13 smallint,@field14 char(1),
@field15 smallint,@field16 char(1),@field17 smallint,
@field18 char(1),@field19 smallint,@field20 char(1),
@field21 smallint,@field22 char(1),@field23 smallint,
@field24 char(1),@field25 smallint,@field26 char(1),
@field27 smallint,@field28 char(1),@field29 smallint,
@field30 char(1),@field31 smallint,@field32 char(1),
@field33 smallint,@field34 char(1),@field35 smallint,
@field36 char(1),@field37 smallint,@field38 char(1),
@field39 smallint,@field40 char(1),@field41 smallint,
@field42 char(1),@field43 smallint,@field44 char(1),
@field45 smallint,@field46 char(1),@field47 smallint,
@field48 char(1),@field49 smallint,@field50 char(1),
@field51 smallint,@field52 char(1),@field53 smallint,
@field54 char(1),@field55 smallint,@field56 char(1),
@field57 smallint,@field58 char(1),@field59 smallint,
@field60 char(1),@field61 smallint,@field62 char(1),
@field63 smallint,@field64 char(1),@field65 smallint,
@field66 char(1),@field67 smallint,@field68 char(1),
@field69 smallint,@field70 char(1),@field71 smallint,
@field72 char(1),@field73 smallint,@field74 char(1),
@field75 smallint,@field76 char(1),@field77 smallint,
@field78 char(1),@field79 smallint,@field80 char(1),
@field81 smallint,@field82 char(1),@field83 smallint,
@field84 char(1),@field85 smallint,@field86 char(1),
@field87 real,@field88 real

AS

BEGIN

SET NOCOUNT ON

INSERT INTO ACCIDENTLOGS VALUES
(@field1,@field2,@field3,@field4,@field5,@field6,@field7,@field8,
@field9,@field10,@field11,@field12,@field13,@field14,@field15,
@field16,@field17,@field18,@field19,@field20,@field21,@field22,
@field23,@field24,@field25,@field26,@field27,@field28,@field29,
@field30,@field31,@field32,@field33,@field34,@field35,@field36,
@field37,@field38,@field39,@field40,@field41,@field42,@field43,
@field44,@field45,@field46,@field47,@field48,@field49,@field50,
@field51,@field52,@field53,@field54,@field55,@field56,@field57,
@field58,@field59,@field60,@field61,@field62,@field63,@field64,
@field65,@field66,@field67,@field68,@field69,@field70,@field71,
@field72,@field73,@field74,@field75,@field76,@field77,@field78,
@field79,@field80,@field81,@field82,@field83,@field84,@field85,
@field86,@field87,@field88);

END
```

CarChips Table

```
CREATE PROCEDURE dr_insert_carChip @field1 smallint,
@field2 smallint, @field3 char(20), @field4 smallint,
@field5 smallint, @field6 smallint, @field7 smallint,
@field8 smallint, @field9 smallint, @field10 real, @field11 real,
@field12 real, @field13 real, @field14 smallint, |
@field15 smallint, @field16 smallint, @field17 smallint,
@field18 smallint, @field19 smallint, @field20 smallint,
@field21 smallint, @field22 smallint, @field23 smallint, @field24
char(1), @field25 char(1)

AS

BEGIN

SET NOCOUNT ON

INSERT INTO CARCHIPS VALUES (@field1, @field2, @field3, @field4,
@field5, @field6, @field7, @field8, @field9, @field10, @field11,
@field12, @field13, @field14, @field15, @field16, @field17,
@field18, @field19, @field20, @field21,@field22, @field23,
@field24, @field25);

END
```

CompanyLocations Table

```
CREATE PROCEDURE dr_insert_companyLocation @field1 smallint,
@field2 char(40), @field3 char(80), @field4 smallint, @field5
```

```
smallint, @field6 smallint, @field7 char(40)

AS

BEGIN

SET NOCOUNT ON

INSERT INTO COMPANYLOCATIONS VALUES (@field1, @field2, @field3,
@field4, @field5, @field6, @field7);

END
```

Days Table

```
CREATE PROCEDURE dr_insert_day @field1 smallint,@field2 smallint,
@field3 datetime,@field4 smallint,@field5 smallint, @field6 real,
@field7 smallint,@field8 smallint,@field9 smallint,
@field10 smallint,@field11 smallint,@field12 smallint,
@field13 smallint,@field14 smallint, @field15 smallint,
@field16 smallint,@field17 smallint,@field18 smallint,
@field19 smallint,@field20 smallint,@field21 smallint

AS

BEGIN

SET NOCOUNT ON

INSERT INTO DAYS VALUES (@field1, @field2, @field3, @field4,
@field5, @field6, @field7, @field8, @field9, @field10, @field11,
@field12,@field13,@field14,@field15,@field16,@field17,@field18,
@field19,@field20,@field21);

END
```

DownloadDates Table

```
CREATE PROCEDURE dr_insert_downloadDate @field1 smallint, @field2
smallint, @field3 datetime, @field4 smallint, @field5 smallint

AS

BEGIN

SET NOCOUNT ON

INSERT INTO DOWNLOADDATES VALUES (@field1, @field2, @field3,
@field4, @field5);

END
```

DriverGroups Table

```
CREATE PROCEDURE dr_insert_driverGroup @field1 smallint, @field2
smallint, @field3 char(60), @field4 char(60)

AS

BEGIN

SET NOCOUNT ON
```

```
INSERT INTO DRIVERGROUPS VALUES (@field1, @field2, @field3,
@field4);

END
```

DriveRights Table

```
CREATE PROCEDURE dr_insert_driveRight @field1 smallint,
@field2 smallint,@field3 smallint, @field4 smallint,
@field5 smallint, @field6 smallint, @field7 smallint,
@field8 smallint, @field9 smallint, @field10 smallint,
@field11 int, @field12 smallint, @field13 int, @field14 char(1),
@field15 char(1), @field16 char(1), @field17 char(1),
@field18 char(1), @field19 char(1), @field20 char(1),
@field21 char(80)

AS

BEGIN

SET NOCOUNT ON

INSERT INTO DRIVERIGHTS VALUES (@field1, @field2, @field3,
@field4, @field5, @field6, @field7, @field8, @field9, @field10,
@field11, @field12, @field13, @field14, @field15, @field16,
@field17, @field18, @field19, @field20, @field21);

END
```

Drivers Table

```
CREATE PROCEDURE dr_insert_driver @field1 smallint,
@field2 smallint, @field3 smallint, @field4 char(60),
@field5 char(5), @field6 char(80), @field7 char(60),
@field8 char(20), @field9 char(20),@field10 char(80),
@field11 char(80), @field12 char(15), @field13 char(1),
@field14 char(80)

AS

BEGIN

SET NOCOUNT ON

INSERT INTO DRIVERS VALUES (@field1, @field2, @field3, @field4,
@field5, @field6, @field7, @field8, @field9, @field10, @field11,
@field12, @field13, @field14);

END
```

Fleets Table

```
CREATE PROCEDURE dr_insert_fleet @field1 smallint, @field2
smallint, @field3 char(40)

AS

BEGIN
```

```
SET NOCOUNT ON

INSERT INTO FLEETS VALUES (@field1, @field2, @field3);

END
```

GPS Table

```
CREATE PROCEDURE dr_insert_gps @field1 smallint,@field2 smallint,
@field3 datetime,@field4 smallint,@field5 smallint, @field6
smallint, @field7 real, @field8 smallint, @field9 real, @field10
real, @field11 int, @field12 char(1)

AS

BEGIN

SET NOCOUNT ON

INSERT INTO GPS VALUES (@field1,@field2,@field3,@field4,@field5,
@field6,@field7,@field8,@field9,@field10,@field11,@field12);

END
```

OdometerLogs Table

```
CREATE PROCEDURE dr_insert_odometerLog @field1 smallint,@field2
smallint,@field3 datetime,@field4 smallint,@field5 real, @field6
char(40)

AS

BEGIN

SET NOCOUNT ON

INSERT INTO ODOMETERLOGS VALUES
(@field1,@field2,@field3,@field4,@field5, @field6);

END
```

Readiness Codes Table

```
CREATE PROCEDURE dr_insert_readinessCode @field1 smallint,@field2
smallint,@field3 datetime,@field4 char(1),@field5 char(1),@field6
char(1),@field7 char(1),@field8 char(1),@field9 char(1),@field10
char(1),@field11 char(1),@field12 char(1),@field13
char(1),@field14 char(1),@field15 char(1),@field16
char(1),@field17 char(1),@field18 char(1),@field19
char(1),@field20 char(1),@field21 char(1),@field22
char(1),@field23 char(1),@field24 char(1),@field25
char(1),@field26 char(1),@field27 char(1) ,@field28
char(1),@field29 char(1),@field30 char(1),@field31
char(1),@field32 char(1),@field33 char(1),@field34
char(1),@field35 char(1),@field36 char(1)

AS

BEGIN
```

```
SET NOCOUNT ON

INSERT INTO READINESSCODES VALUES
(@field1,@field2,@field3,@field4, @field5, @field6,@field7,
@field8, @field9,@field10, @field11, @field12,@field13, @field14,
@field15,@field16, @field17, @field18,@field19, @field20,
@field21,@field22, @field23, @field24,@field25, @field26,
@field27,@field28, @field29, @field30,@field31, @field32,
@field33,@field34, @field35, @field36);

END
```

SafetyScore Table

```
CREATE PROCEDURE dr_insert_safetyScore @field1 smallint,@field2
smallint,@field3 smallint,@field4 smallint,@field5 char(1),
@field6 smallint,@field7 real, @field8 char(1)

AS

BEGIN

SET NOCOUNT ON

INSERT INTO SAFETYSCORE (locationID, driverID, year, month,
exemptDriver, score, mileage, vehicleType) VALUES
(@field1,@field2,@field3,@field4,@field5,@field6,@field7,
@field8);

END
```

TamperLogs Table

```
CREATE PROCEDURE dr_insert_tamperLog @field1 smallint,@field2
smallint,@field3 datetime,@field4 datetime,@field5 smallint,
@field6 smallint

AS

BEGIN

SET NOCOUNT ON

INSERT INTO TAMPERLOGS VALUES
(@field1,@field2,@field3,@field4,@field5,@field6);

END
```

TripAddresses Table

```
CREATE PROCEDURE dr_insert_tripAddress @field1 smallint,
@field2 smallint, @field3 char(80), @field4 char(40),
@field5 char(40),@field6 char(80), @field7 char(40),
@field8 char(40), @field9 char(40),@field10 char(40),
@field11 char(20), @field12 char(20), @field13 char(80),
@field14 real, @field15 real, @field16 real, @field17
smallint, @field1 char(80),@field19 smallint, @field20 smallint

AS
```

```
BEGIN

SET NOCOUNT ON

INSERT INTO TRIPADDRESSES VALUES (@field1, @field2, @field3,
@field4, @field5, @field6, @field7, @field8, @field9, @field10,
@field11, @field12, @field13, @field14, @field15, @field16,
@field17, @field18, @field19, @field20);

END
```

Trips Table

```
CREATE PROCEDURE dr_insert_trip @field1 smallint,@field2
smallint, @field3 smallint,@field4 datetime,@field5
datetime,@field6 smallint,@field7 real,@field8 smallint,@field9
smallint,@field10 smallint,@field11 int,@field12 real,@field13
real,@field14 smallint,@field15 smallint,@field16
char(1),@field17 smallint,@field18 smallint,@field19
char(80),@field20 char(1),@field21 char(1)

AS

BEGIN

SET NOCOUNT ON

INSERT INTO TRIPS VALUES
(@field1,@field2,@field3,@field4,@field5,@field6,@field7,@field8,
@field9,@field10,@field11,@field12,@field13,@field14,@field15,
@field16,@field17,@field18,@field19,@field20,@field21);

END
```

TroubleCodes Table

```
CREATE PROCEDURE dr_insert_troubleCode @field1 smallint,@field2
smallint,@field3 datetime,@field4 int, @field5 int, @field6 int,
@field7 real, @field8 real, @field9 real, @field10 real, @field11
real, @field12 real, @field13 real, @field14 real, @field15 real,
@field16 real, @field17 char(80)

AS

BEGIN

SET NOCOUNT ON

INSERT INTO TROUBLECODES VALUES (@field1, @field2, @field3,
@field4, @field5, @field6, @field7, @field8, @field9, @field10,
@field11, @field12, @field13, @field14, @field15, @field16,
@field17);

END
```

Vehicles Table

```
CREATE PROCEDURE dr_insert_vehicle @field1 smallint, @field2
smallint, @field3 char(80), @field4 smallint, @field5 smallint,
@field6 smallint, @field7 char(80), @field8 char(20), @field9
char(10), @field10 datetime, @field11 real, @field12 smallint,
@field13 char(1), @field14 char(1), @field15 char(80)

AS

BEGIN

SET NOCOUNT ON

INSERT INTO VEHICLES VALUES (@field1, @field2, @field3, @field4,
@field5, @field6, @field7, @field8, @field9, @field10, @field11,
@field12, @field13, @field14, @field15);

END
```

Appendix B: Installing Oracle Stored Procedures Manually

Once FMS is installed on an Oracle server, the stored procedures can be installed manually using the following instructions. Use the following steps to create stored procedures in the Oracle database server. The stored procedures can be entered manually using the script for each stored procedure below or they can be cut and pasted from the DriveRight_StoredProcedure_ORACLEServer.doc file located in the C:\Program Files\DriveRight directory.

1. Go to Start->Programs->Oracle-OraHome90->Enterprise Manager Console. The **Oracle Enterprise Manager Console** displays.
2. Select **Launch StandAlone** and click **OK**.
3. Expand the Databases directory to display DRIGHT.(Domain Name).COM.
4. Enter login information. The default user name is **system** and the password is **system**. Click **OK**.
5. On the left hand side pane, expand the directory to display Schema->Procedure->System.
6. Right-click on **System** and select **Create**.
7. Enter the name of procedure in the Name field, and paste the respective procedure info for the table. All the procedures displayed below list the name the stored procedure should be labeled as and the resulting stored procedure information that should be pasted/entered. Click **Create**.
8. Repeat the above steps for all the stored procedures.
9. You’ll be able to see “Procedure created successfully” message after creating each procedure on the database.
10. Verify the **Status** column for each procedure on the right pane. It should say **Valid** for all the procedures.

Accident Logs Table

Procedure Name: dr_insert_accidentLog

```
(
field1 IN NUMBER, field2 IN NUMBER,field3 IN DATE,
field4 IN NUMBER, field5 IN CHAR, field6 IN CHAR,
field7 IN NUMBER, field8 IN CHAR, field9 IN NUMBER,
field10 IN CHAR, field11 IN NUMBER, field12 IN CHAR,
field13 IN NUMBER, field14 IN CHAR, field15 IN NUMBER,
field16 IN CHAR, field17 IN NUMBER, field18 IN CHAR,
field19 IN NUMBER,field20INCHAR, field21 IN NUMBER,
field22 IN CHAR, field23IN NUMBER, field24 IN CHAR,
field25 IN NUMBER,field26 IN CHAR, field27IN NUMBER,
field28 IN CHAR, field29 IN NUMBER,field30 IN CHAR,
field31 IN NUMBER,field32 IN CHAR, field33 IN NUMBER,
field34 IN CHAR, field35 IN NUMBER,field36 IN CHAR,
field37 IN NUMBER,field38 IN CHAR, field39 IN NUMBER,
field40 IN CHAR, field41 IN NUMBER,field42 IN CHAR,
field43 IN NUMBER,field44 IN CHAR, field45 IN NUMBER,
field46 IN CHAR, field47 IN NUMBER,field48 IN CHAR,
field49 IN NUMBER,field50 IN CHAR, field51 IN NUMBER,
field52 IN CHAR, field53 IN NUMBER,field54 IN CHAR,
field55 IN NUMBER,field56 IN CHAR, field57 IN NUMBER,
field58 IN CHAR, field59 IN NUMBER,field60 IN CHAR,
field61IN NUMBER,field62 IN CHAR, field63 IN NUMBER,
field64 IN CHAR, field65 IN NUMBER,field66 IN CHAR,
field67 IN NUMBER,field68 IN CHAR, field69 IN NUMBER,
field70 IN CHAR, field71 IN NUMBER,field72 IN CHAR,
field73 IN NUMBER,field74I N CHAR, field75 IN NUMBER,
field76 IN CHAR, field77 IN NUMBER,field78 IN CHAR,
field79 IN NUMBER,field80 IN CHAR, field81 IN NUMBER,
field82 IN CHAR, field83 IN NUMBER,field84 IN CHAR,
field85 IN NUMBER, field86 IN CHAR,
field87 IN REAL,field88 IN REAL
)

AS

BEGIN

INSERT INTO ACCIDENTLOGS VALUES (field1, field2, field3, field4,
field5, field6, field7, field8, field9, field10, field11,
field12, field13, field14, field15, field16, field17, field18,
field19, field20, field21, field22, field23, field24, field25,
field26, field27, field28, field29, field30, field31, field32,
field33, field34, field35, field36, field37, field38, field39,
field40, field41, field42, field43, field44, field45, field46,
field47, field48, field49, field50, field51, field52, field53,
field54, field55, field56, field57, field58, field59, field60,
field61, field62, field63, field64, field65, field66, field67,
field68, field69, field70, field71, field72, field73, field74,
field75, field76, field77, field78, field79, field80, field81,
field82, field83, field84, field85, field86, field87, field88);
END
```

CarChips Table

Procedure Name: dr_insert_carChip

```
(
field1 IN NUMBER,field2 IN NUMBER,field3 IN CHAR,
field4 IN NUMBER,field5 IN NUMBER,field6 IN NUMBER,
field7 IN NUMBER,field8 IN NUMBER,field9 IN NUMBER,
field10 IN FLOAT,field11 IN FLOAT,field12 IN FLOAT,
field13 IN FLOAT,field14 IN NUMBER,field15 IN NUMBER,
field16 IN NUMBER,field17 IN NUMBER,field18 IN NUMBER,
field19 IN NUMBER,field20 IN NUMBER,field21 IN NUMBER,
field22 IN NUMBER,field23 IN NUMBER,field24 IN CHAR,
field25IN CHAR
)

AS

BEGIN

INSERT INTO CARCHIPS VALUES
(field1,field2,field3,field4,field5,field6,field7,field8,field9,
field10,field11,field12,field13,field14,field15,field16,field17,
field18, field19, field20, field21, field22, field23, field24,
field25);

END
```

CompanyLocations Table

Procedure Name: dr_insert_companyLocation

```
(
field1 IN NUMBER,field2 IN CHAR,field3 IN CHAR,
field4INNUMBER,field5 IN NUMBER,field6 IN NUMBER,
field7INCHAR
)

AS

BEGIN

INSERT INTO COMPANYLOCATIONS VALUES (field1, field2, field3,
field4, field5, field6, field7);

END
```

Days Table

Procedure Name: dr_insert_day

```
(
field1 IN NUMBER,field2 IN NUMBER,field3 IN DATE,
field4 IN NUMBER,field5 IN NUMBER,field6 IN FLOAT,
field7 IN NUMBER,field8 IN NUMBER,field9 IN NUMBER,
field10 IN NUMBER,field11 IN NUMBER,field12 IN NUMBER,
field13 IN NUMBER,field14 IN NUMBER,field15 IN NUMBER,
field16 IN NUMBER,field17 IN NUMBER,field18 IN NUMBER,
field19 IN NUMBER,field20 IN NUMBER,field21 IN NUMBER
)

AS

BEGIN

INSERT INTO DAYS VALUES
(field1,field2,field3,field4,field5,field6,field7,field8,field9,
field10,field11,field12,field13,field14,field15,field16,field17,
field18,field19, field20, field21);

END
```

Download Dates Table

Procedure Name: dr_insert_downloadDate

```
(
field1 IN NUMBER,field2 IN NUMBER,field3 IN DATE,
field4 IN NUMBER,field5 IN NUMBER
)

AS

BEGIN

INSERT INTO DOWNLOADDATES VALUES (field1, field2, field3, field4,
field5);

END
```

Drivers Table

Procedure Name: dr_insert_driver

```
(
field1 IN NUMBER,field2 IN NUMBER,field3 IN NUMBER,
field4 IN CHAR,field5 IN CHAR,field6 IN CHAR,
field7 IN CHAR,field8 IN CHAR,field9 IN CHAR,field10 IN CHAR,
field11 IN CHAR,field12 IN CHAR,field13 IN CHAR,field14 IN CHAR
)

AS

BEGIN

INSERT INTO DRIVERS VALUES
(field1,field2,field3,field4,field5,field6,field7,field8,field9,
field10,field11,field12, field13,field14);

END
```

DriversGroup Table

Procedure Name: dr_insert_driverGroup

```
(
field1 IN NUMBER,field2 IN NUMBER,field3 IN CHAR,field4 IN CHAR
)
```

```
AS

BEGIN

INSERT INTO DRIVERGROUPS VALUES (field1, field2, field3, field4);

END
```

DriveRights Table

Procedure Name: dr_insert_driveRight

```
(
field1 IN NUMBER,field2 IN NUMBER,field3INNUMBER,
field4 IN NUMBER,field5 IN NUMBER,field6 IN NUMBER,
field7 IN NUMBER,field8 IN NUMBER,field9 IN NUMBER,
field10 IN NUMBER,field11 IN NUMBER,field12 IN NUMBER,
field13 IN NUMBER,field14 IN CHAR,field15 IN CHAR,
field16 IN CHAR,field17 IN CHAR,field18 IN CHAR,
field19 IN CHAR,field20 IN CHAR,field21 IN CHAR
)

AS

BEGIN

INSERT INTO DRIVERIGHTS VALUES
(field1,field2,field3,field4,field5,field6,field7,field8,field9,
field10,field11,field12,field13,field14,field15,field16,field17,
field18,field19, field20, field21);

END
```

Fleets Table

Procedure Name: dr_insert_fleet

```
(
field1 IN NUMBER,field2 IN NUMBER,field3 IN CHAR
)

AS

BEGIN

INSERT INTO FLEETS VALUES (field1, field2, field3);

END
```

GPS Table

Procedure Name: dr_insert_gps

```
(
field1 IN NUMBER,field2 IN NUMBER,field3 IN DATE,field4 IN NUMBER,
field5 IN NUMBER,field6 IN NUMBER,field7 IN FLOAT,
field8 IN NUMBER,field9 IN FLOAT,field10 IN FLOAT,field11 IN INT,
field12 IN CHAR
)
```

```
AS

BEGIN

INSERT INTO GPS VALUES (field1, field2, field3, field4, field5,
field6, field7, field8, field9, field10, field11, field12);

END
```

OdometerLogs Table

Procedure Name: dr_insert_odometerLog

```
(
field1 IN NUMBER,field2 IN NUMBER,field3 IN DATE,field4 IN NUMBER,
field5 IN FLOAT,field6 IN CHAR
)

AS

BEGIN

INSERT INTO ODOMETERLOGS VALUES (field1, field2, field3, field4,
field5, field6);

END
```

ReadinessCodes Table

Procedure Name: dr_insert_readinessCode

```
(
field1 IN NUMBER, field2 IN NUMBER, field3 IN DATE,
field4 IN  NUMBER, field5 IN CHAR, field6 IN CHAR,
field7 IN  NUMBER,field8 IN  CHAR, field9 IN CHAR,
field10 IN  NUMBER, field11 IN CHAR,field12 IN CHAR,
field13 IN  NUMBER, field14 IN CHAR, field15 IN CHAR,
field16 IN  NUMBER, field17 IN CHAR, field18 IN CHAR,
field19 IN  NUMBER, field20 IN CHAR, field21 IN CHAR,
field22 IN  NUMBER,field23 IN CHAR, field24 IN CHAR,
field25 IN  NUMBER,field26 IN CHAR, field27 IN CHAR,
field28 IN  NUMBER,field29 IN CHAR, field30 IN CHAR,
field31 IN  NUMBER,field32 IN CHAR, field33 IN CHAR,
field34 IN  NUMBER,field35 IN  CHAR, field36 IN CHAR
)
AS

BEGIN

INSERT INTO READINESSCODES VALUES (field1,field2,field3,field4,
field5,field6,field7,field8,field9,field10,field11,field12,
field13,field14,field15,field16,field17,field18,field19,field20,
field21,field22,field23,field24,field25,field26,field27,field28,
field29,field30,field31,field32,field33,field34,field35,field36);

End
```

SafetyScore Table

Procedure Name: dr_insert_safetyScore

```
(
field1 IN NUMBER,field2 IN NUMBER,field3 IN NUMBER,
field4 IN NUMBER,field5 IN CHAR,field6 IN NUMBER,field7 IN FLOAT,
field8 IN CHAR,field9 IN CHAR,field10 IN CHAR,field11 IN NUMBER,
field12 IN NUMBER,field13 IN FLOAT
)

AS

BEGIN

INSERT INTO SAFETYSCORE VALUES (field1, field2, field3, field4,
field5, field6,field7, field8, field9, field10, field11, field12,
field13);

END
```

TamperLogs Table

Procedure Name: dr_insert_tamperLog

```
(
field1 IN NUMBER,field2 IN NUMBER,field3 IN DATE,
field4 IN DATE,field5INNUMBER,field6INNUMBER
)

AS

BEGIN

INSERT INTO TAMPERLOGS VALUES (field1, field2, field3, field4,
field5, field6);

END
```

Trips Table

Procedure Name: dr_insert_trips

```
(
field1IN NUMBER,field2 IN NUMBER,field3 IN NUMBER,
field4 IN DATE,field5 IN DATE,field6 IN NUMBER,
field7 IN FLOAT,field8 IN NUMBER,field9 IN NUMBER,
field10 IN NUMBER,field11 IN NUMBER,field12 IN FLOAT,
field13 IN FLOAT,field14 IN NUMBER,field15 IN NUMBER,
field16 IN CHAR,field17 IN NUMBER,field18 IN NUMBER,
field19 IN CHAR,field20 IN CHAR,field21 IN CHAR
)

AS

BEGIN

INSERT INTO TRIPS VALUES (field1, field2, field3, field4, field5,
field6, field7, field8, field9, field10, field11, field12,
field13, field14, field15, field16, field17, field18, field19,
field20, field21);

END
```

TripAddresses Table

Procedure Name: dr_insert_tripAddresses

```
(
field1 IN NUMBER,field2 IN NUMBER,field3 IN CHAR,
field4 IN CHAR,field5 IN CHAR,field6 IN CHAR,
field7 IN CHAR,field8 IN CHAR,field9 IN CHAR,
field10 IN CHAR,field11 IN CHAR,field12 IN CHAR,
field13 IN CHAR,field14 IN FLOAT,field15 IN FLOAT,
field16 IN FLOAT,field17 IN NUMBER,field18 IN CHAR,
field19 IN NUMBER,field20 IN NUMBER
)

AS

BEGIN

INSERT INTO TRIPADDRESSES VALUES (field1, field2, field3, field4,
field5, field6, field7, field8, field9, field10, field11,
field12, field13, field14, field15, field16, field17, field18,
field19, field20);

END
```

TroubleCodes Table

Procedure Name: dr_insert_troubleCode

```
(
field1 IN NUMBER, field2 IN NUMBER, field3 IN DATE,
field4 IN NUMBER, field5 IN NUMBER, field6 IN NUMBER,
field7 IN FLOAT, field8 IN FLOAT, field9 IN FLOAT,
field10 IN FLOAT,field11 IN FLOAT, field12 IN FLOAT,
field13 IN FLOAT, field14 IN FLOAT, field15 IN FLOAT,
field16 IN FLOAT, field17 IN CHAR
)

AS

Begin

INSERT INTO TROUBLECODES VALUES (field1, field2, field3, field4,
field5, field6, field7, field8, field9, field10, field11,
field12, field13, field14, field15, field16, field17);

End
```

Vehicles Table

Procedure Name: dr_insert_vehicle

```
(
field1 IN NUMBER, field2 IN NUMBER, field3 IN CHAR,
field4 IN NUMBER, field5 IN NUMBER, field6 IN NUMBER,
field7 IN CHAR, field8 IN CHAR, field9 IN CHAR,
field10 IN DATE, field11 IN FLOAT, field12 IN NUMBER,
field13 IN CHAR, field14 IN CHAR, field15 IN CHAR
)

AS

Begin

INSERT INTO VEHICLES VALUES (field1, field2, field3, field4,
field5, field6, field7, field8, field9, field10, field11,
field12, field13, field14, field15);

End
```

Contacting Davis Technical Support

If you have questions about DriveRight FMS, or encounter problems installing or using the software, please contact Davis Technical Support. Most questions can be answered while you're on the phone.

Sorry, we are unable to accept collect calls.

(510) 732-7814 – Monday through Friday, 7:00 a.m. to 5:30 p.m. Pacific Time.

(510) 670-0589 – Fax to Technical Support.

support@davisnet.com – E-mail to Technical Support.

info@davisnet.com – E-mail to Davis Instruments.

www.davisnet.com – Product documentation is available on the DriveRight Support section of our website. Watch for FAQs and other updates.

Product Number: 8186 Part Number: 7395.195
DriveRight® Fleet Management Software Database Selection Guide Rev. C Manual (March 10, 2006)
This product complies with the essential protection requirements of the EC EMC Directive 89/336/EC.

DAVIS
Davis Instruments
3465 Diablo Avenue, Hayward, CA 94545-2778 U.S.A.
510-732-9229 • Fax: 510-732-9188
E-mail: info@davisnet.com • www.davisnet.com