Retail Pro® 9

Database and Custom Programming Guide

Retail Pro Interrnational, LLC
400 Plaza Dr., Suite 200
Folsom, CA 95630 USA

USA 1-800-738-2457
International +1-916-605-7200
www.retailpro.com

About this Guide

The purpose of this documentation is to provide overview and detailed information for custom programmers and database administrators who are using Retail Pro 9.

If you believe the information presented here is incomplete or inaccurate, we encourage you to contact us at emanuals@retailpro.com.

The software described herein is furnished under a license agreement.

Retail Pro International, LLC
400 Plaza Dr., Suite 200
Folsom, CA 95630 USA

USA 1-800-738-2457)
International +1-916-605-7200
www.retailpro.com

Retail Pro Internationjal, LLC
400 Plaza Dr., Suite 200
Folsom, CA 95630
U.S.A.

Trademarks

Document Revision History

Date	Description
05/20/2011	Updated with new tables for R5 (Loyalty, Passwords, etc.)
12/01/2011	Added Tender Types and info on Tender Type 12 (Foreign Currency).
05/08/2012	Updated INVN_SBS with text1-text10 and other new fields.
07/27/2012	Removed XML Files info. XML File information can be found in the Retail_Pro_9_XML_Files.pdf document

Table Of Contents

Introduction

About the Database and Custom Programming Guide

This document provides overview and detailed information for custom programmers and database administrators who are using Retail Pro 9.

The first section of the` documentation explains how to install the Retail Pro server and clients, basic database concepts, and how to use custom programming to retrieve data from Retail Pro.

For example, custom programmers will learn how to:

- Populate a Microsoft Excel® spreadsheet with selected Retail Pro data.
- Create custom reports using Microsoft Access®.
- Incorporate Retail Pro data into a Microsoft Word® mail merge document.

All examples use an ODBC connection to access the Retail Pro database.

Key Benefits of Oracle11*g*

The data management system for Retail Pro 9-Series is a relational database management system (RDBMS) utilizing an embedded Oracle11*g* database engine. Oracle is the industry leader in database management technology and the Oracle11*g* engine has a proven reputation for outstanding functionality, reliability, and security. It can scale virtually without limit and has built-in safeguards to ensure security and data integrity. Additionally, the open architecture of Oracle11*g* provides easy access to Retail Pro's database for custom development and interfacing with other applications. Some key benefits of Oracle11*g* are described below.

Performance

Oracle continues to lead the database technology industry, significantly outperforming its competitors. Oracle11*g* efficiently handles large volumes of data and provides simultaneous access by multiple users in virtually any size operation. In addition to its cutting-edge data storage and retrieval methods, the Oracle11*g* database engine outperforms other databases in data access speed by allowing multiple users to access the same data table simultaneously. Other databases lock the entire table for a user to access its data and require all other transactions to wait. With Oracle11*g*, only the individual row that contains the data is locked, enabling multiple users to get fast, accurate results simultaneously.

Scalability

Oracle11*g* utilizes a multi-tiered architecture, so program components can be distributed across multiple machines. For example, one server might host the database and perform data access functions, another server might host a reporting server and perform business analytic functions, and a workstation might run Retail Pro and print reports. As your business grows, Oracle11*g* allows for easy scalability by distributing the system across multiple servers as necessary.

Analytical Power

The database is designed to quickly and efficiently calculate analytical and aggregated information in the database without taking up extra disk space from storing values. Average daily values are easily accessed and transformed into statistical information. For example, using the Stock Ledger feature, you can quickly view the inventory on hand values for any store at any point in history.

Reliability and Security

The data that is generated during day-to-day operations must be protected at all costs. Oracle11*g* has built-in safeguards to protect data against unplanned system failure, storage failure, site failure or, human error.

Security was a primary consideration in all aspects of the design of our database engine. Oracle11*g* addresses security vulnerabilities by securing transmitted data, encrypting sensitive information, restricting user access at the record level, and providing a single point of entry to authorized applications.

Data Integrity

Oracle11*g* contains several built-in mechanisms to protect data integrity. In addition, information is stored relationally, which allows much more flexibility and efficiency in maintaining data. For example, an item's historical data will remain linked even if item information, such as the description, is changed.

Interoperability and Open Architecture

Retail Pro is probably not the only system in your retail environment. The Oracle 11*g* database is designed to maximize interoperability, the ability of one computer application to communicate with another application. In addition, Oracle 11*g* is built with an open architecture, which allows open access to the Retail Pro database for custom application development or custom reporting.

Low Administration

Perhaps the most important reason why Island Pacific has chosen to embed the Oracle 11*g* database is that it virtually eliminates database administration. Through its built-in utilities, the database can be tuned, backed up or rebuilt, dramatically reducing the level of maintenance needed.

Open Data Access

All merchandise data, transactional history and other data from all subsidiaries are available for viewing and reporting. Reports can be generated to include store-by-store detail for any subsidiary, or consolidated data from any combination of stores and subsidiaries. With proper authorization, these reports can be utilized by any person at any time, anywhere in the world.

Data Access for Reporting Packages

Retail Pro 9 was engineered to provide open data access for robust reporting through third-party reporting packages or custom programming. Retail Pro can support any reporting package or custom application that uses SQL/ODBC for database access. One example might be Crystal Reports from Business Objects.

Multiple Users

Give multiple users simultaneous access to the same reports without any decrease in system performance by using a multi-tiered architecture. Retail Pro is designed to maintain a high level of performance with multiple users running reports and analysis simultaneously.

Retail Pro 9 utilizes a client-server model, the best architecture for a robust, multi-user environment. Users running Retail Pro (the clients) make requests to the server(s) for report data. Each server then accesses the database. The Retail Pro database management system processes the requests and controls access to the information in the database. The bulk of the data processing occurs on the server, which allows each client running a report to have access to the processed data quickly and concurrently.

Stock Values and Statistics

Obtain fast, efficient access to specialized data and key statistics. Retail Pro provides rapid access to specialized information not normally available. Its relational database has been specifically architected to provide advanced business analytical information quickly and accurately:

- Stock values (quantity, cost, pricing) at a specific time in the past, for example, End of Month values.
- Key statistics, such as Average Daily Inventory, that can be used as a basis for calculating Key Performance Indicators (KPIs).

Retail Pro Reports

Retail Pro includes a basic report server called Retail Pro Reports, which uses SQL (Structured Query Language) to generate reports. Reports provides some default report designs and filters. You can group reports together and run them as a single batch for added convenience.

Restrictions on Accessing Oracle Tables and Views

Due to licensing constraints, direct access to the Oracle tables is not allowed. For custom reporting and development, read-only access to the data is granted through Views, and data can be imported into Retail Pro through ECM using XML.

Database FAQs

What is SQL?

Structured Query Language, developed in the 1970s.

What is PL/SQL?

Procedural language for SQL. Allows you to control the flow of operations as with any development language such as Visual Basic, C, PowerBuilder, Delphi etc.

What is DML?

Data manipulation language used to Select, Insert, Update and Delete data.

What is a table?

The basic storage object in a database.

What is a standard view?

A standard view is a stored SQL statement that can be accessed by a user in the same manner as they would access a table. When a view in queried, Oracle will execute the stored SQL and return the result set to the user.

What is an object view?

An object view is similar to a standard view in the way that it is a stored SQL statement but it also contains built in relationships between data that would have to be manually formulated if you were using a standard view.

What is a stored procedure?

A stored procedure is a block of PL/SQL code stored in the database that can be used by other applications. A stored procedure can be packaged into an Oracle package so that procedures and functions can be grouped together. A stored procedure can optionally return data back to the calling application.

Need to install the Retail Pro Client (Oracle Net)

Both native connections and ODBC connections require the Oracle client.

How do you connect to Retail Pro?

Connecting to Retail Pro via the ‘reportuser’ user when developing reports. The ‘reportuser’ user is a special Oracle user account created for custom report development.

Which ODBC driver should I use?

Use the driver from Microsoft, Crystal Reports or Oracle. Each one behaves slightly differently so you will need to find the driver that best suits your needs

Why use an ODBC driver instead of a native connection?

ODBC is more flexible and does not tie to a specific data source. A native connection provides the richest set of Oracle 11*g* features.

What is a database session?

A database session is a single connection to an Oracle database. Most reports will only require a single connection to the Retail Pro database. A database session consumes Oracle and Windows resources.

How do I manage sessions?

You can use the Connection Manager to manage sessions created by both Retail Pro and any other application that connects to the Retail Pro database.

How do I get data out of the Oracle database?

Use SQL and PL/SQL to directly extract data from the Retail Pro database. It is important to understand the underlying database structure.

How do I write SQL and PL/SQL?

Use Windows Notepad to create SQL scripts and execute them with SQL*Plus – an Oracle-provided tool.

Or use the Retail Pro Database Explorer to write and execute SQL SELECT statements.

If you do a lot of development, then you should invest in a SQL development tool that will make development easier.

Reference: See *Retail Pro 9.2 Standard Table Views*

Controllers

Each installation of the Retail Pro 9 database is called a controller and is assigned an identifying number.

Retail Pro requires you to assign a Controller when you restart your computer after installing Retail Pro.

The Controller enables Retail Pro to track the origin of a document, as well as the location of edits made to that document after its initial creation. The originating Controller remains on that document and can not be updated or changed.

The Controller, combined with other key identifiers such as Subsidiary, Store, and Workstation, identify each part of your company down to the individual workstation.

Controllers vs. Stations

The **Controller** field replaces the **Station Number**, which is used in earlier versions of Retail Pro, as the key field for identifying the origin of documents.

Each workstation in Retail Pro 9 is assigned to a controller. This is different from 8 Series, where each workstation is assigned to a station.

Assigning a Controller

Typically, you specify a Controller (as well as Subsidiary and Workstation) when you restart your computer after installing a new Retail Pro® installation.

Note: You can specify the controller in the Workstation Preferences > General later, but you will not be able to create documents or customers until you specify the identifiers.

Within a subsidiary, you can use a Controller only once. You can assign up to 999 Controllers per subsidiary/store combination.

Planning Controllers

Before installing Retail Pro, plan your Controllers so that there is no uncertainty about which Controller will be assigned which workstations.

Additional Tracking Fields

Retail Pro uses these additional fields to track document history:

- Orig Controller
- Orig Global Store Code
- Orig Store
- Orig Station

Updating the Delta Table

The Delta Table provides the basis for all statistical calculations in Retail Pro, storing old values, new values and differences.

You need to update the Delta Table on a regular basis. Delta Table updates accumulate the latest data and update statistical calculations for every item on every document that enters the database. For example, every sale, return, transfer or adjustment of that item and the quantity involved (-2, -5, 1, 20, etc.)

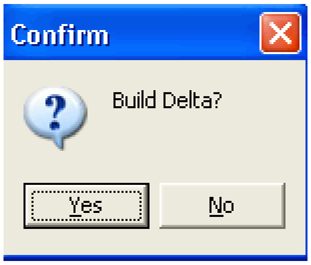
Your statistics will only be as accurate and up-to-date as your last update (and the documents posted at the time of that update).

You can manually launch an update of the Delta Table from the Tools menu or from within the Audit area. You can view the Delta Table update within the Technician's Tool Kit.

To update the Delta Table:

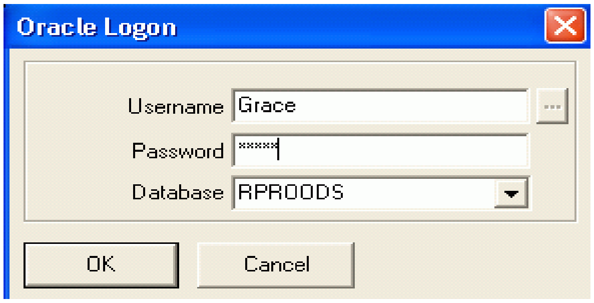
1. Select Tools **>** Update Delta from the top menu of the Home Screen.

   Alternatively, select Merchandise **>** Audit from the Home Screen to access the Audit area., and then click **Update Delta**. A confirmation dialog displays.

   Build Delta?

2. Click **Yes** to confirm the build. The Oracle Logon dialog displays.

   Username Grace
   Password *****
   Database RPROODS

3. Enter your **Username** and **Password**, then click **OK**.

You can also build the delta table from within the Audit area. This ensures your audits are performed using the latest information available.

Scheduling Delta Table Updates

You can schedule Delta Table updates to occur at the times that are most convenient and cause the least amount of disruptions to your operations.

When scheduling a Delta Table updates, keep the following things in mind:

- You can schedule the update of the Delta Table for a specific subsidiary (using a command-line parameter)
- Improper configuration can result in Delta Table updates occurring at inopportune times.
- Potential slowdowns can occur if multiple overlapping Delta Table updates are performed.

To schedule a Delta Build:

1. Set up a new task for **techtoolkit.exe** in Schedule Editor. Assign any of the command line parameters listed below:

Parameter	Allowed Values	Description and Rules	Expected Format
/u[sername] Alternative: -u[sername]		Retail Pro username	/u:sysadmin
/p[assword Alternative: -p[assword]		Retail Pro password	/p:mypassword
/m[ode]	deltab[uild]	Run the delta build process.	/m:anymode
/s[ilent]		Run in silent mode. If silent mode is specified, then the GUI will not be displayed wile the requested process is performed.	/s
/sub	Any existing subsidiary number All	If All is specified, then the delta build will run for all subsidiaries.	/sub:001,002,005,006 /sub:all

Parameter	Allowed Values	Description and Rules	Expected Format
/deltamode	C[lear] F[ull] U[pdate]	Clears and deletes all existing delta table records. Performs a full delta build Performs update from last delta build. This parameter is ignored unless the mode is set to deltabuild.	/deltamode:f

Example: techtoolkit.exe /u:sysadmin /p:sysadmin /m:deltab /deltamode:u /s sub:001

Clearing the Delta Table

If the delta table was incorrect, and you cannot correct it through a full rebuild, you can clear the old delta table values and then do a new rebuild.

Why would I clear the delta table?

You might use this strategy, for example, if the table contains erroneous documents or documents that were added to the delta table in the past but no longer exist in Retail Pro®. When you run a full rebuild, Retail Pro® reviews everything currently in the database and updates all the delta tables; however, it does not remove documents that are no longer in the system.

Important!: If you use the following operation to clear the delta table, the delta_src will be destroyed for all periods, even closed periods.

To clear the delta table:

1. Run **Techtoolkit.exe** using the **-clear** command line parameter.

   Example: Techtoolkit.exe -c -full -clear

Abbreviations

The following table lists common abbreviations that are used in the database schema and in the XML files. This information is useful for 3rd-party developers who are using the XML files to develop reports and other 3rd-party applications.

Word	Abbreviation
Accounts Payable	AP
Accounts Receivable	AR
Acknowledge	Ack
Address	Addr
Adjustment	Adj
Allocation	Alloc
Alternate	Alt
Alternate Lookup	ALU
Amount	Amt
Amount with Tax	Amtwt or AmtWT
Application	App
Approved	Approv
Attribute	Attr
Audit	Audt
Authorize	Auth
Automatic	Auto
Average	Avg
Calendar	Cal
Check	Chk
Commission	Comm
Company	Cmp
Component	Cmpnt
Configuration	Cfg
Count	Cnt

Word	Abbreviation
Credit Card	Crd
Currency	Curr
Customer	Cust
Database	DB
Date of Birth	DOB
Default	Deflt
Definition	Def
Delete	Del
Denomination	Denom
Deposit	Depst
Description	Desc
Destination	Dest or Dst
Dictionary	Dict
Discontinued	Discon
Discount	Disc
Document	Doc
Driver's License	DL
Electronic Funds Transfer	EFT
Employee	Empl
End of Month	EOM
Error	Err
Estimated Time of Arrival	EFT
Excluded	Excl
Expiration	Exp
Extended	Ext
Field	Fld
Fields	Flds
Filter	Fltr
First	Fst

Word	Abbreviation
Foreign Currency	FC
Function	Func
Gift Card	Gft_Crd
Gift Certificate/Store Credit	GCSC
Global	Glob
Group	Grp
High Security	Hisec
History	Hist
Import	Imp
Included	Incl
Information	Info
Inspected	Insp
Instructions	Instr
Internal	Int
Invoice	Invc
Language	Lang
Last	Lst
Length	Len
Level	Lvl
Location	Loc
Maximum	Max
Message	Msg
Minimum	Min
Movement	Mov
Movement Note	MN
Number	No
On Hand	OH
Open to Buy	OTB
Operating System	OS

Word	Abbreviation
Order	Ord
Original	Orig
Package	Pkg
Parameters	Params
Pattern	Ptrn
Payment	Pmt
Percent	Perc
Percent with Tax	PercWT
Permission	Perm
Physical Inventory	PI
Point of Sale	POS
Position	Pos
Postal Code/ZIP Code	Zip
Preference	Pref
Price with Tax	PriceWT
Processed	Proc
Purchase	Purch
Purchase Order	PO
Quantity	Qty
QuickBooks	QB
Rebate	Reb
Received	Rcvd
Reference	Ref
Region	Rgn
Repair	Rpr
Repair Order	RO
Repository	Rp
Required	Req
Resolved	Resolv

Word	Abbreviation
Responsible	Resp
Sales Order	SO
Salesperson Incentive Fee	SPIF
Schedule	SC
Security	Sec
Sequence	Seq
Size	Siz
Source	Src
Statistic	Stat
Stock Keeping Unit	SKU
Sublocation	Subloc
Subsidiary	Sbs
Supplemental	Suppl
Target	Trgt
Trademark	TM
Transaction	Trx
Transfer	Trans
Transfer Order	TO, Tord
Transfer Slip	Slip
Transit	Trans
Universal Product Code	UPC
User-defined Field	UDF
Value	Val
Vendor	Vend
Verified	Verif
Voucher	Vou
Wait List	WL
With Tax	WT
Workstation	WS

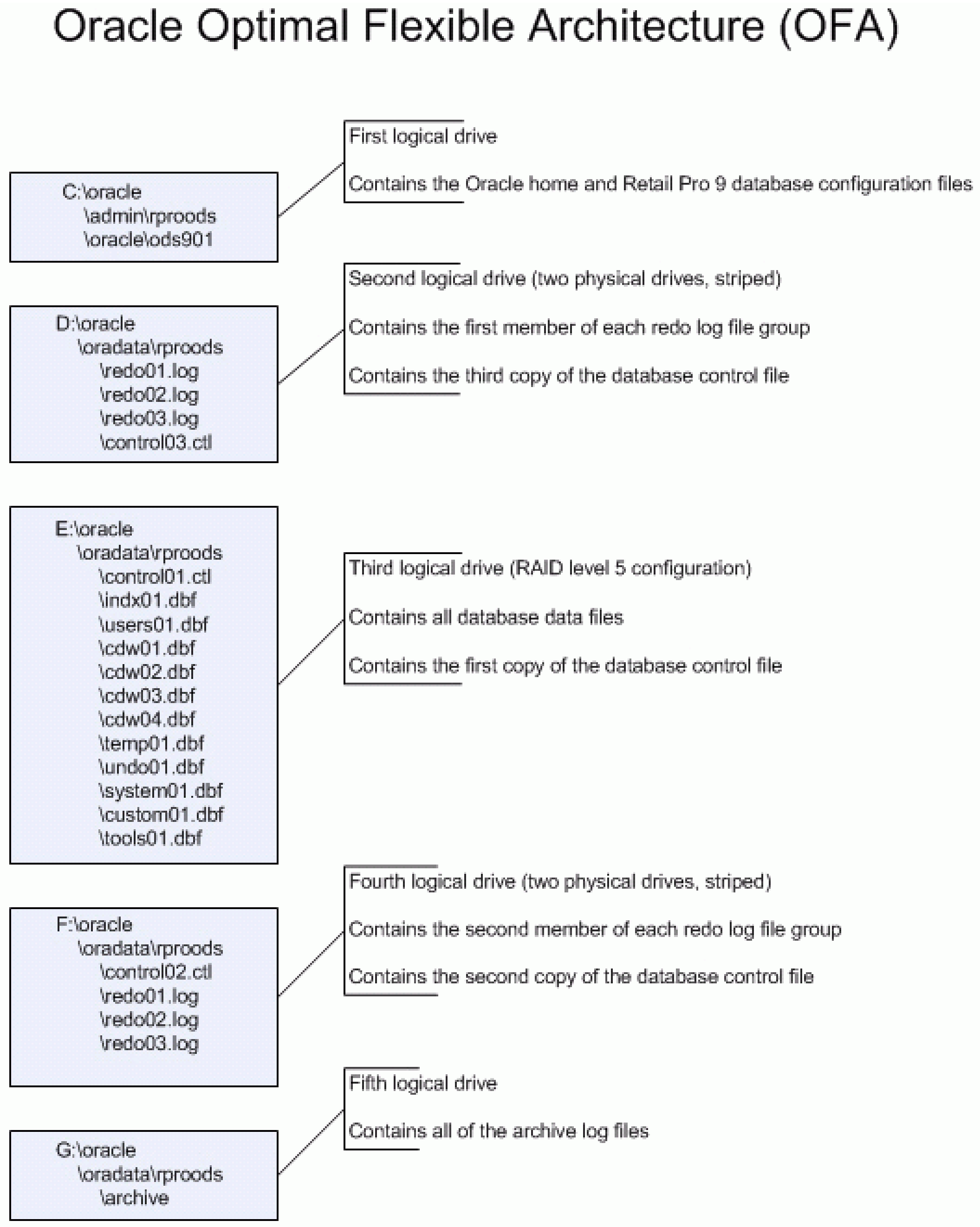
Database File Layout

There are several layouts that you can use for the database files. Important considerations are:

- How easy will it be to recover the database files?
- How much is the client willing to spend on hard disk arrays?

Layout Options

Option	Comments
Place all schema objects on one or more large RAID 10 arrays	The performance advantage of hardware striping far outweighs any disadvantage caused by combining data and index, or any other schema types.
Place redo logs on RAID 1 or RAID 10 arrays	Place each log thread on a separate RAID 1 pair. This allows sequential read optimization algorithms on the storage array to be activated. An alternative is to include redo log files with the main RAID 10 disk set. This is useful where log write activity is low, or where there are a limited number of disks.
Place archive log files on RAID 1, RAID 10, or RAID 5 disks	Both for performance and availability reasons, archive logs should be physically separated from other data types.



Oracle Optimal Flexible Architecture (OFA)

```
C:\oracle
    \admin\rproods
    \oracle\ods901
```

First logical drive

Contains the Oracle home and Retail Pro 9 database configuration files

```
D:\oracle
    \oradata\rproods
        \redo01.log
        \redo02.log
        \redo03.log
        \control03.ctl
```

Second logical drive (two physical drives, striped)

Contains the first member of each redo log file group

Contains the third copy of the database control file

```
E:\oracle
    \oradata\rproods
        \control01.ctl
        \indx01.dbf
        \users01.dbf
        \cdw01.dbf
        \cdw02.dbf
        \cdw03.dbf
        \cdw04.dbf
        \temp01.dbf
        \undo01.dbf
        \system01.dbf
        \custom01.dbf
        \tools01.dbf
```

Third logical drive (RAID level 5 configuration)

Contains all database data files

Contains the first copy of the database control file

```
F:\oracle
    \oradata\rproods
        \control02.ctl
        \redo01.log
        \redo02.log
        \redo03.log
```

Fourth logical drive (two physical drives, striped)

Contains the second member of each redo log file group

Contains the second copy of the database control file

```
G:\oracle
    \oradata\rproods
        \archive
```

Fifth logical drive

Contains all of the archive log files

RAID Levels

Oracle can take advantage of each type of RAID configuration. RAID levels are described below.

RAID	Description	Oracle Usage
0	With RAID 0, data is striped across disks, but no data protection is offered. This is not truly RAID, since there is no redundancy. RAID 0 offers high performance, since there is only one Physical I/O per logical read or write. However, any error on any disk causes the whole RAID array to fail. RAID 0 can be appropriate for temp or scratch space.	
1	RAID 1 consists of pairs of mirrored disks. The two disks together offer a single logical disk. If one disk fails, the other disk takes over until the disk is replaced. RAID 1 is relatively expensive since for every two disks purchased, only the storage capacity of one disk is usable. RAID 1 offers good sequential read and write performance. However, it offers only moderate random read and write performance, due to lower spindle count than multi-disk striped configurations.	RAID 1 is commonly used for redo logs and archive logs, due to the sequential reads and writes performed by logging processes. RAID 1 redo and archive logs are particularly good for OLTP performance. Note that each redo thread should be separated on separate disk pairs to maintain sequential access. Since redo logs rarely fill an entire disk, using a RAID 1 disk pair for redo logs balances the need for higher sequential I/O performance with the less efficient storage utilization that can be afforded to transient files.
5	RAID 5 utilizes a different algorithm for redundancy. Instead of mirroring, parity-based error correction is striped across all disks in the array. If data on any disk (or a whole disk) is lost, the data can be recalculated from the remaining disks. The net effect is the number of accessible disks is one less than the total disks in the RAID group. This is a relatively cost effective form of RAID. After a disk failure, a replacement disk can be automatically synched with the remaining disks. However, please note that rebuild speeds may be slow. Performance will be degraded during rebuilds, although the RAID group continues to function. RAID 5 tends to suffer from a write	RAID 5 offers good performance for read-only databases such as Data Warehouses, due to data striping across multiple disks (reads are performed in parallel). RAID 5 is generally considered poor for OLTP databases, due to the write penalty. However, it is the least expensive form of protected RAID storage, and cost considerations often trump performance considerations. If you have to use RAID 5 for OLTP databases, make sure you maximize the amount of write cache.

	performance penalty. Each logical write requires four I/Os: read the disk stripe, read the parity, write the disk stripe, calculate and write the new parity. Hardware cache can be used to mitigate the write penalty.	We are recommending that RAID 5 be used for the Retail Pro 9 database data files because of the solid reliability RAID 5. There is a very small sacrifice in performance because of this but well worth it in the event of a disk failure.
10	RAID 10 (often called RAID 1 + 0, or RAID 0 + 1) offers a combination of mirroring and striping techniques. Each individual disk is mirrored, but the data is striped across multiple disks. A minimum of 4 disks is required for a RAID 10 group. There is a hardware specific maximum for each. RAID 10 groups are commonly implemented as a 10 disk RAID group.	RAID 10 offers the highest write performance for any form of protected disks. It is commonly used for Oracle data files. RAID 10 is especially good for OLTP databases.

Importing Documents

You can import any document in xml format into Retail Pro using the Enterprise Communication Manager (ECM). This enables you to import documents from outside or legacy systems into Retail Pro. When importing documents, make sure to include the required fields for that document type.

Important! To import the Invoice.xml file, the INVN_BASE_ITEM section is required.

To import documents into Retail Pro:

1. Add the transaction information to the appropriate xml file. For example, add sales transaction information to the **Invoice.xml** file. (Make sure the file includes INVN_BASE_ITEM)
2. Place the file in the **ECM\Polling\[Station]\In\Recvd** directory.
3. Perform a Process In operation using the Enterprise Communications Manager (ECM).

INVN_BASE_ITEM Info Required for New Items on Documents

When a receipt (invoice) or other transactional document is transmitted from one station to another, it cannot be guaranteed that the receiving station will have the items that are listed on that receipt in the stations' inventory. When ECM is importing a receipt or other document into Retail Pro, if an item that is on the document is not in inventory, then that item will be created by ECM and marked as inactive. This enables the document to be created successfully at the importing station. To do that, ECM needs to know basic details about the item and this information is contained in the <INVN_BASE_ITEM> section of the XML file.

Flag Field Must Be Set to "0" for Receipts to Update On-Hand Quantity

If importing items into Retail Pro, be sure to set the **Flag** field in the **Inventory.xml** file to "0". If you don't set the **Flag** field to "0" in the **Inventory.xml** file, receipts that are imported into Retail Pro will not update the item's on-hand quantity.

Do Not Import Receipts with Tender_Type=12

Do not try to import documents that have " tender_type=12"

Tender type 12 is foreign currency inside of Rpro9 CORE CODE in memory. In the database, though, foreign currency is identified by two conditions - tender type is CASH (0) and currency does not match base currency for that subsidiary.

Required Fields for Retail Pro Document Types

The required fields for each Retail Pro document type are listed below:

Adjustment Memos (ADJUSTMENT table)

ADJ_SID

SBS_NO

STORE_NO

CREATED_DATE

MODIFIED_DATE

CMS_POST_DATE

Purchase Orders (PO table)

PO_SID

SBS_NO

STORE_NO

CREATED_DATE

MODIFIED_DATE

CMS_POST_DATE

Receipts (INVOICE table)

INVN_BASE_ITEM

CMS_POST_DATE

INVC_SID

SBS_NO

STORE_NO

INVC_NO

INVC_TYPE

CREATED_DATE

MODIFIED_DATE

Sales Orders (SO table)

MODIFIED_DATE

CMS_POST_DATE

SO_SID

SBS_NO

STORE_NO

Slips (SLIP table)

SLIP_SID

SBS_NO

OUT_STORE_NO

IN_SBS_NO

IN_STORE_NO

UNVERIFIED

CREATED_DATE

MODIFIED_DATE

CMS_POST_DATE

Transfer Orders (TORD table)

TO_TYPE

CREATED_DATE

MODIFIED_DATE

CMS_POST_DATE

TO_SID

SBS_NO

STORE_NO

Vouchers (VOUCHERS table)

VOU_SID

SBS_NO

STORE_NO

VOU_NO

VOU_TYPE

VOU_CLASS

CREATED_DATE

MODIFIED_DATE

CMS_POST_DATE

Importing Items

You can import items in xml format into Retail Pro using the Enterprise Communication Manager (ECM). This enables you to import items from outside or legacy systems into Retail Pro 9.

This section explains how to import new items into the Retail Pro v9 database from an external system and how to perform updates.

If you are importing a receipt (Invoice.xml) or other document, and the items on the document do not exist in the target inventory, ECM will create those items and mark them with a status of Inactive.

To import new items into Retail Pro:

1. Add the item information to the Inventory.xml file.

   Reference: See Required Information for New Items

2. Place the file in the **ECM\Polling\[Station]\In\Recvd** directory, and then perform a Process In operation using the Enterprise Communications Manager (ECM).

Importing New Items

Item Definition/Description

The first time you import items from an outside system into Retail Pro, you must enter values in the following fields:

sbs_no	dcs_code	description1
created_date	modified_date	tax-code
flag	ext_flag	kit_type
max_disc_perc1	max_disc_perc2	unorderable
print_tag	active	mark_for_del
cms_post_date	eci_flag	regional
qty	store_no	

Important!: Do not leave a required field blank or use empty quotation marks (“ “)

Active Field Must Be Set to 1

When importing items, it is crucial that the **Active** field is set to "1" .

Note: You can import departments and vendors whose active field is set to "0", but not items.

Defining Store Quantities

In the INVN_SBS_QTYS section of the Inventory.xml file, enter a **store_no** for each store and the qty of the item at the store. You do not need to enter a **qty** of "0" for those stores where the on-hand quantity is zero.

For example, if you want to import an item and set its quantity to zero at all stores, the XML will look like this:

```
<INVN_SBS_QTYS>
<INVN_SBS_QTY store_no="1"/>
<INVN_SBS_QTY store_no="2"/>
<INVN_SBS_QTY store_no="3"/>
<INVN_SBS_QTY store_no="4"/>
....
<INVN_SBS_QTY store_no="250" qty="0"/>
</INV_SBS_QTYS>
```

In this example, after the item is in Retail Pro, the retailer could create adjustment memos to adjust the quantity.

If you want to define (or update) non-zero item quantities, list every store_no and qty.

Example:

```
<INVN_SBS_QTYS>
<INVN_SBS_QTY store_no="1" qty="10"/>
<INVN_SBS_QTY store_no="2" qty="10"/>
<INVN_SBS_QTY store_no="3" qty="10"/>
<INVN_SBS_QTY store_no="4" qty="10"/>
<INVN_SBS_QTY store_no="250" qty="40"/>
</INV_SBS_QTYS>
```

Store 250

Store 250 is used to store the on-hand quantity for the entire company. Be sure to enter the correct company quantity for store 250 when importing items into Retail Pro for the first time.

You must either pass correct store 250 quantities, OR use Technician's Toolkit after the import to recalculate company quantities. If you don't enter the correct quantity for store 250 or run Technician's Toolkit, Retail Pro inventory will display incorrect quantity values for the company.

INVN_BASE_ITEM Required if Transactions include New Items

When a receipt (invoice) or other transactional document is transmitted from one station to another, it cannot be guaranteed that the receiving station will have the items that are listed on that document in the stations' inventory. When ECM is importing a receipt or other transactional document into Retail Pro, if an item that is on the receipt is not in inventory, then that item will be created by ECM and marked as inactive. This enables the document to be created successfully at the importing station. To do that, ECM needs to know basic details about the item and this information is contained in the <INVN_BASE_ITEM> section of the XML file.

Flag Field Must Be Set for Transactions to Update On-Hand Quantity

When importing new items into Retail Pro, be sure to set the **Flag** field in the **Inventory.xml** file. If you don't set the **Flag** field in the **Inventory.xml** file, receipts that are imported into Retail Pro will not update the item's on-hand quantity.

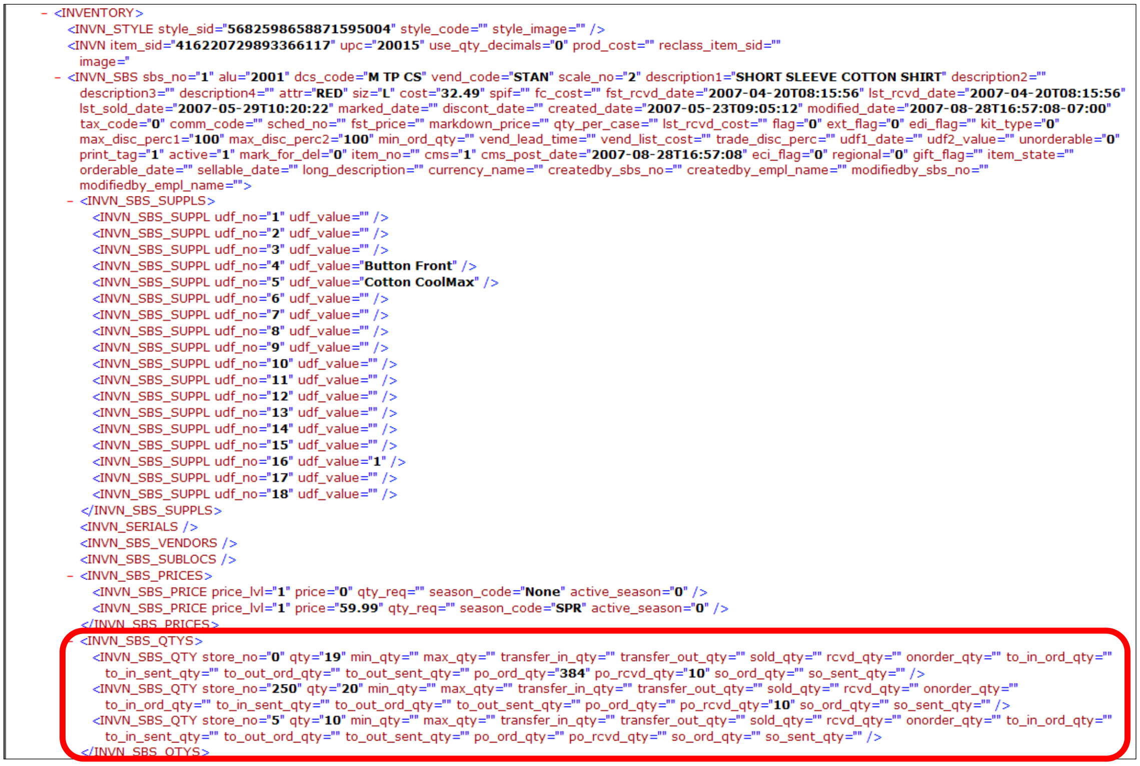
The flag field is a bit array field:

Bit	**Function**
0	Skip Committed flag
1	Skip Inventory flag (i.e. Non-inventory item)
2	Non-editable flag. An example of this would be the CMS flag, used in v9 but not used in v8.
3-7	Reserved for future use.

The following table lists the integer value that should be entered in the **Flag** field to achieve the desired setting.

Integer Value	**Description**
0	All flags deselected (0000 0000). An item with this setting would be included in Committed, would not be a non-inventory item, and would be editable.
1	Enables the Skip Committed flag (0000 0001)
2	Enables the Non-Inventory flag (0000 0010)
3	Enables both the Skip Committed and Non-Inventory flags. (0000 0011)
4	Enables the CMS flag (0000 0100)(
5	Enables the CMS and Skip Committed flag (0000 0101)
6	Enables the CMS flag and the Non-Inventory flag (0000 0110)
7	Enables the CMS flag, Non-Inventory flag, and the Skip Committed flag (0000 0111)

Sample Inventory.xml file:



```
- <INVENTORY>
    <INVN_STYLE style_sid="5682598658871595004" style_code="" style_image="" />
    <INVN item_sid="416220729893366117" upc="20015" use_qty_decimals="0" prod_cost="" reclass_item_sid=""
      image="
  - <INVN_SBS sbs_no="1" alu="2001" dcs_code="M TP CS" vend_code="STAN" scale_no="2" description1="SHORT SLEEVE COTTON SHIRT" description2=""
      description3="" description4="" attr="RED" siz="L" cost="32.49" spif="" fc_cost="" fst_rcvd_date="2007-04-20T08:15:56" lst_rcvd_date="2007-04-20T08:15:56"
      lst_sold_date="2007-05-29T10:20:22" marked_date="" discont_date="" created_date="2007-05-23T09:05:12" modified_date="2007-08-28T16:57:08-07:00"
      tax_code="0" comm_code="" sched_no="" fst_price="" markdown_price="" qty_per_case="" lst_rcvd_cost="" flag="0" ext_flag="0" edi_flag="" kit_type="0"
      max_disc_perc1="100" max_disc_perc2="100" min_ord_qty="" vend_lead_time="" vend_list_cost="" trade_disc_perc="" udf1_date="" udf2_value="" unorderable="0"
      print_tag="1" active="1" mark_for_del="0" item_no="" cms="1" cms_post_date="2007-08-28T16:57:08" eci_flag="0" regional="0" gift_flag="" item_state=""
      orderable_date="" sellable_date="" long_description="" currency_name="" createdby_sbs_no="" createdby_empl_name="" modifiedby_sbs_no=""
      modifiedby_empl_name="">
    - <INVN_SBS_SUPPLS>
        <INVN_SBS_SUPPL udf_no="1" udf_value="" />
        <INVN_SBS_SUPPL udf_no="2" udf_value="" />
        <INVN_SBS_SUPPL udf_no="3" udf_value="" />
        <INVN_SBS_SUPPL udf_no="4" udf_value="Button Front" />
        <INVN_SBS_SUPPL udf_no="5" udf_value="Cotton CoolMax" />
        <INVN_SBS_SUPPL udf_no="6" udf_value="" />
        <INVN_SBS_SUPPL udf_no="7" udf_value="" />
        <INVN_SBS_SUPPL udf_no="8" udf_value="" />
        <INVN_SBS_SUPPL udf_no="9" udf_value="" />
        <INVN_SBS_SUPPL udf_no="10" udf_value="" />
        <INVN_SBS_SUPPL udf_no="11" udf_value="" />
        <INVN_SBS_SUPPL udf_no="12" udf_value="" />
        <INVN_SBS_SUPPL udf_no="13" udf_value="" />
        <INVN_SBS_SUPPL udf_no="14" udf_value="" />
        <INVN_SBS_SUPPL udf_no="15" udf_value="" />
        <INVN_SBS_SUPPL udf_no="16" udf_value="1" />
        <INVN_SBS_SUPPL udf_no="17" udf_value="" />
        <INVN_SBS_SUPPL udf_no="18" udf_value="" />
      </INVN_SBS_SUPPLS>
      <INVN_SERIALS />
      <INVN_SBS_VENDORS />
      <INVN_SBS_SUBLOCS />
    - <INVN_SBS_PRICES>
        <INVN_SBS_PRICE price_lvl="1" price="0" qty_req="" season_code="None" active_season="0" />
        <INVN_SBS_PRICE price_lvl="1" price="59.99" qty_req="" season_code="SPR" active_season="0" />
      </INVN_SBS_PRICES>
    - <INVN_SBS_QTYS>
        <INVN_SBS_QTY store_no="0" qty="19" min_qty="" max_qty="" transfer_in_qty="" transfer_out_qty="" sold_qty="" rcvd_qty="" onorder_qty="" to_in_ord_qty=""
          to_in_sent_qty="" to_out_ord_qty="" to_out_sent_qty="" po_ord_qty="384" po_rcvd_qty="10" so_ord_qty="" so_sent_qty="" />
        <INVN_SBS_QTY store_no="250" qty="20" min_qty="" max_qty="" transfer_in_qty="" transfer_out_qty="" sold_qty="" rcvd_qty="" onorder_qty=""
          to_in_ord_qty="" to_in_sent_qty="" to_out_ord_qty="" to_out_sent_qty="" po_ord_qty="" po_rcvd_qty="10" so_ord_qty="" so_sent_qty="" />
        <INVN_SBS_QTY store_no="5" qty="10" min_qty="" max_qty="" transfer_in_qty="" transfer_out_qty="" sold_qty="" rcvd_qty="" onorder_qty="" to_in_ord_qty=""
          to_in_sent_qty="" to_out_ord_qty="" to_out_sent_qty="" po_ord_qty="" po_rcvd_qty="" so_ord_qty="" so_sent_qty="" />
      </INVN_SBS_QTYS>
```

Importing Updated Item Information

To import updated item information into Retail Pro:

1. Add the item information to the Inventory.xml file.
2. Place the file in the **ECM\Polling\[Station]\In\Recvd** directory, and then perform a Process In operation using the Enterprise Communications Manager (ECM).

Updating Item Quantities

If you want to import updated item quantities, list each store_no and qty in the IVNV_SBS_QTY section.

Example:

```
<INVN_SBS_QTYS>
<INVN_SBS_QTY store_no="1" qty="10"/>
<INVN_SBS_QTY store_no="2" qty="10"/>
<INVN_SBS_QTY store_no="3" qty="10"/>
<INVN_SBS_QTY store_no="4" qty="10"/>
<INVN_SBS_QTY store_no="250" qty="40"/>
</INV_SBS_QTYS>
```

Importing Employees

Security Information for New Employees Imported into Retail Pro 9

When you import new employees into Retail Pro 9 via the Employees.xml file, you must import the employees without password or security settings. After the basic information for the employee is added, you must go into Employee Mgmt > Groups to assign passwords, security rights, and privileges.

Security information is stored in the <SECURITY_SETTINGS> area of the XML file. A sample of which is shown below:

```
<SECURITY_SETTINGS sig1="80095CC838C28504CBB5E89A20985A10FFF9187F">
<EMP_SBSS>
<EMP_SBS sbs_no="1" access_all="1" stores="0-1"/>
</EMP_SBSS>
<CMS_USER user_id="97" user_name="USER2" sbs_no="1" active="1" password="C498F7B745"
sysadmin="0" sig2="49E5E008F0B1F5E306701BE393602844A2FFA5B">
<USER_GROUPS>
</USER_GROUPS>
</CMS_USER>
<SIGNATURE sig3="D9FBBA855F6D5BFFB73E2C08134377B9FB11AAFC"/>
</SECURITY_SETTINGS>
```

These are not required fields. If you are importing new employees into Retail Pro 9, do not enter values in these fields for the new employees. When you assign password and security information, the fields will be populated by the system.

Adding Custom Buttons for Outside Applications

The **Usermenu.ini** file stores information for the buttons that appear on the Retail Pro user menus, including all the applications that can be accessed from the Tools menu.

You can change the **Usermenu.ini** file to add buttons that link to outside applications. For example, you might want to add a button that links to your QuickBooks accounting software. The buttons that you add can be accessed by clicking Tools on the Retail Pro Home Screen.

To add custom buttons to the Retail Pro Tools menu:

1. Using Notepad or other text editor, create a file called **Usermenu.txt**.
2. In the **Usermenu.txt** file, enter information for each button in the following format:

   [1]

   UpdateUserMenu [Application_Name], app1.exe

   [2]

   UpdateUserMenu [Application_Name], app2.exe
3. Save the file to the **\Programs\** folder of the Retail Pro 9 Maintenance Pack.
4. Run the Maintenance Pack.

   Result: The Maintenance Pack updates the **Usermenu.ini** file.
5. After running the Maintenance Pack, start Retail Pro, and then click **Tools**. Buttons that link to the programs added via the **Usermenu.txt** file will be displayed. Select a button to launch the application.

Multiple Databases from Single Workstation

This topic explains how to access more than one database from the same workstation. This is useful if you need to access a test database as well as a production database but don't want to manage two installations of the Retail Pro 9 executable (RetailPro9.exe) on your workstation.

To access multiple databases from a single workstation, you need to:

1. Edit the TNSNames.ora file to include entries for each database.
2. Create a separate shortcut for each database to which the workstation will connect.

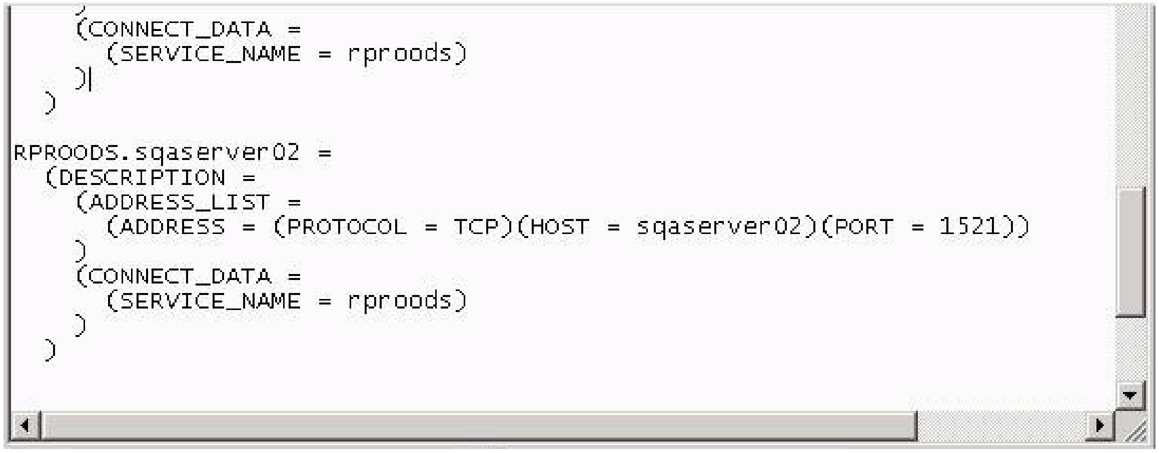
Important: To access multiple databases from one workstation, the same version of Retail Pro 9 must be installed on both the test database and production database. If the versions are different, you will need two installations of Retail Pro 9 executables on his workstation - one for each version of Retail Pro 9.

To edit the TNSNames.ora file:

1. Navigate to the workstation's **\RetailPro9** folder.
2. Right-click the **TNSNames.ora** file and select **Open With ...**, and then select Notepad.

   Result: The file opens in Notepad.
3. Add code for each additional database, similar to the illustration below.

   Warning: The syntax of this file is important to Oracle. Do not add extra space characters or rearrange the format of the TNS entry.



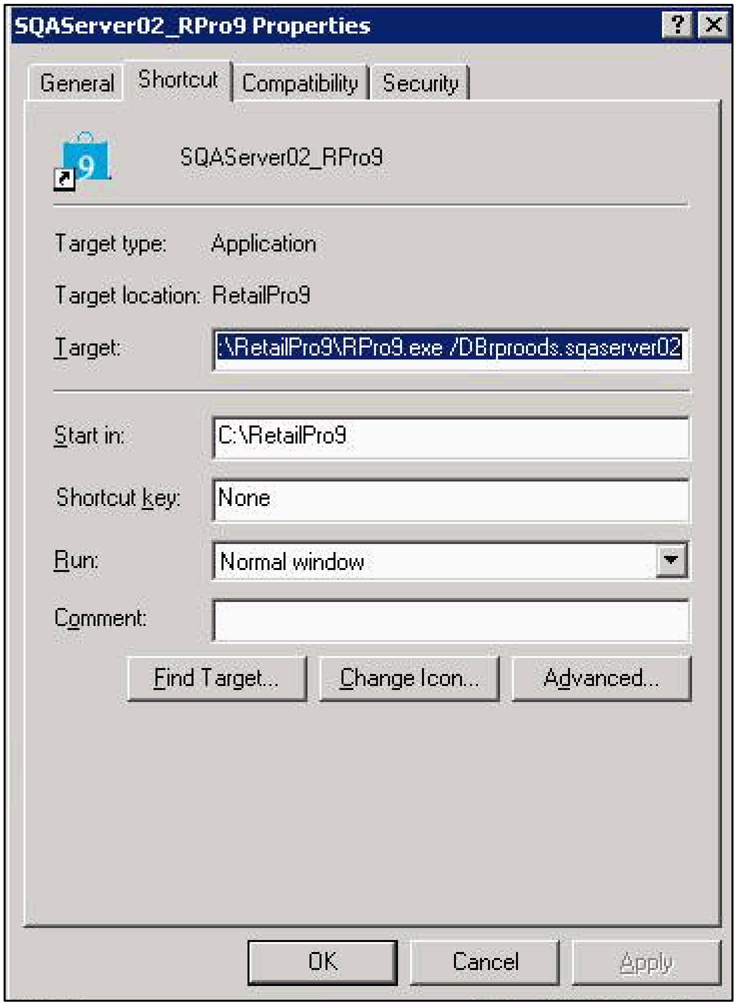
```
    (CONNECT_DATA =
      (SERVICE_NAME = rproods)
    )
  )

RPROODS.sqaserver02 =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = sqaserver02)(PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = rproods)
    )
  )
```

4. Select **File > Save**, and then **File > Exit**.

To create a shortcut to a database:

1. Right-click in an empty area of the desktop, and select **New > Shortcut**. The Create Shortcut Wizard displays.
2. Type or browse to the location of Retail Pro 9 (RPRO9.exe), and then click Next.
3. Type a name for the shortcut, and then click Finish.
4. Right-click in the shortcut, and select **Properties**. The Properties dialog is displayed.



SQAServer02_RPro9

Target type: Application

Target location: RetailPro9

Target: :\RetailPro9\RPro9.exe /DBrproods.sqaserver02

Start in: C:\RetailPro9

Shortcut key: None

Run: Normal window

Comment:

5. Select the Shortcut tab. Modify the Target field to point to the appropriate server, and then click OK.
6. Repeat for each database to which you want to connect.
7. To access a particular database, double-click the shortcut for that database.

Custom Programming Guide

About Custom Programming in Retail Pro 9

Retail Pro uses open architecture that allows you to use Standard SQL to generate your own data queries and import the data to your own custom programs and reports. For example, you can:

- Populate a Microsoft Excel spreadsheet with selected Retail Pro data.
- Create custom reports using Microsoft Access.
- Incorporate Retail Pro data into a Microsoft Word mail merge document.

All examples use an ODBC connection to access the Retail Pro database (see Creating an ODBC Connection to Retail Pro for instructions).

Views

Views provide a standardized interface to the underlying database and are vital to the custom development process. Developers do not need to understand all the referential integrity relationships of the relational database.

Once the complexity of the underlying database structure is abstracted via views, custom applications rely on the Oracle standard view interface. Because the interface to the database stays the same, applications do not require modification (or re-compilation) when the database structure changes.

What is a Standard View?

A standard view (or view) is a tailored presentation of the data contained in one or more tables or other views. A view takes the output of a query and puts it in a table format. Therefore, a view can be thought of as a stored query or a virtual table. You can use views in most places where a table can be used.

For example, the Invoice table has several columns and numerous rows of information. If you want users to see only five of these columns or only specific rows, then you can create a view of that table for other users to access.

What is an Object View?

An object view is an extension of the basic relational view mechanism. Just as a view is a virtual table, an object view is a virtual object table. By using object views, you can create virtual object tables from data — using either built-in or user-defined data types — stored in the columns of relational or object tables in the database.

Object views provide the ability to offer specialized or restricted access to the data and objects in a database. For example, you can use an object view to provide a version of an employee object table that does not have attributes containing sensitive data and does not have a deletion method.

Standard Views vs. Object Views

Using object views can lead to better performance. Relational data that make up a row of an object view traverse the network as a unit, potentially saving many round trips.

Object views also provide a gradual migration path for legacy data. They provide for co-existence of relational and object-oriented applications, and make it easier to introduce object-oriented applications to existing relational data without having to make a drastic change from one paradigm to another.

Object views provide the flexibility of looking at the same relational or object data in more than one way. Thus you can use different in-memory object representations for different applications without changing the way you store the data in the database.

The Role of SIDs (System Identifiers)

An SID (System Identifier) is assigned to inventory items, documents, customers and vendor records. The SID is used to uniquely identify records in the database. SIDs are used to link document header and detail information together.

Database Objects

A database object is any object created in the Oracle database. This includes tables, views, types, table spaces, etc.

Retail Pro Schema

To fully exploit the power of Retail Pro for reporting, you must first understand the structure of the Retail Pro database. There are recurring structures that have been implemented throughout the database:

- Logical groupings of tables with parent-child relationships
- Foreign-key constraints between tables to ensure the validity of data and prevent dangling references
- Lookup tables that contain various codes and preferences settings
- Security, preference, versioning and licensing tables
- Summary tables like delta source, history trail and cost trail.
- A layer of standard views and object views to abstract your reports and applications from the underlying database structure which is subject to change.

Creating an ODBC Connection to Retail Pro

In Microsoft Windows 2000:

1. Select Start > Settings > Control Panel > Administrative Tools > Data Sources (ODBC).
2. Select the System DSN tab
3. Click Add.
4. Select the Microsoft ODBC for Oracle driver and select Finish.
5. Enter information in the Microsoft ODBC for Oracle Setup dialog box.

   Note: For the **Data Source Name**, enter **RPROODS**.
6. Click **OK** when finished.

Result: The ODBC connection is created and can now be used to access the Retail Pro® database (RPROODS). An Oracle account has been defined for use by developers to create third-party applications based on the Retail Pro database. The account has specific database privilege. The tables below describe the rights that have been granted to the Oracle account.

The “reportuser” user has privileges to create all necessary database objects in his/her own schema. This means that objects created by the “reportuser” user will reside in the ods_custom table space. The “reportuser” user shares the temp table space with all other Business Analytics users.

Deselect Image Field for Inventory and Vendors

When connecting to the database, be sure to de-select the Image field for inventory and vendor records. If the Image field is selected, the connection will fail.

Connecting to the Database

There are multiple methods that can be used to connect to the database. ODBC, JDBC and direct access via Oracle’s Net protocol are all acceptable connection methods. Regardless of your connection method, the default user name, password and database connect string will be the same.

Connect string syntax: username/password@databasename

Sample connect string: reportuser/report@rproods

Reportuser User Account

Username = reportuser

Password = report

Default user that can access the views. This user has rights to select data from the views but cannot write changes back to the database. The “reportuser” user will have the ability to create database objects in the Report schema. The “reportuser” user will be able to insert, select, update and delete data from objects in the Report schema. The default password should be changed after installation of the database.

Installing Microsoft ODBC for Oracle Driver

If you do not have the Microsoft ODBC for Oracle driver installed on your system, then you can download the Microsoft Data Access Components from the following Web page:

http:\\www.microsoft.com/data/download 26sp2.htm.

Entering Username and Password

You may be prompted for a username and password the first time you connect to Retail Pro from within an application (Microsoft Access, Microsoft Excel, Microsoft Query, etc.). Refer to the Database Name and Reportuser User Account tables for connection information.

General Examples

To help custom developers, the following examples are provided to show the various methods of accessing Retail Pro views. Examples are provided for illustrative purposes only. In addition to these general examples, you can also examples for Microsoft Excel, Microsoft Access, and Microsoft Word.

Refer to the View Descriptions section for more information.

Extracting Document Header Information

This example queries the **Invoice_ov** view to extract header information from a sales receipt in Retail Pro. The **Where** clause of the SQL statement specifies that only normal sales and return receipts from Subsidiary 1 will be retrieved.

Only one view is reference by this query.

```
Select ab.Modifier, ab.Status2, ab.Invc_Sid, ab.Invc_Type, ab.Invc_No,
ab.Created_Date, ab.Created_Date, ab.Customer_Ref.Company_Name,
ab.Cashier_Ref.Empl_Name, ab.Store_No, ab.Invc_Total.Ext_Price,
ab.Invc_Total.Ext_Orig_Price, ab.Invc_Total.Ext_Disc_Amt,
ab.Invc_Total.Shipping_AmtWT, ab.Invc_Total.Fee_AmtWT,
ab.Invc_Total.Amt,
ab.Invc_Total.Ext_Tax_Amt, ab.Tender_Type, ab.Sbs_No
From Invoice_ov ab
Where (ab.Status2 IN (0))
And (ab.Invc_Type IN (0, 2))
And (ab.Sbs_No IN (1));
```

Extracting Document Header and Detail Information

This example queries the Po_ov view and references the PO item and quantity information based on the document SID.

```
Select bt.Inventory_Ref.DCS_Ref.SubClass, bt.Inventory_Ref.DCS_Ref.Class,
bt.Inventory_Ref.DCS_Ref.Department, bt.Inventory_Ref.Siz,
bt.Inventory_Ref.Attr, bt.Inventory_Ref.Description1, bt.Item_Sid,
ds.PO_Sid, fi.Due_Qty, fi.Ord_Qty, fi.Ext_Ord_Cost, fi.Store_No
From Po_ov ds,  Table(ds.Po_items) bt,  Table(bt.PO_Qtys) fi
Where (ds.PO_Sid=:DOC_SID)
```

Extracting Data Using Filters

This example queries Retail Pro for quantity adjustments that have occurred during the current year. The query is filtered and only extracts year-to-date adjustments for stores 0 through 5 and subsidiary 1.

```
Select av.Adj_Sid, av.Adj_No, av.Created_Date, av.Created_Date,
av.Store_No,
v.Clerk_Ref.Empl_Name, av.Adj_Reason_Name, av.Post_Date, av.Sbs_No,
ff.Item_SID,
ff.Inventory_Ref.Item_No,
ff.Inventory_Ref.DCS_Ref.Department,ff.Inventory_Ref.DCS_Ref.Class,
ff.Inventory_Ref.DCS_Ref.SubClass,
ff.Inventory_Ref.Vendor_Ref.Vend_Code, ff.Inventory_Ref.Description1,
ff.Inventory_Ref.Attr,
ff.Inventory_Ref.Siz, ff.Orig_Qty, ff.Adj_Qty, ff.Dif_Qty
From Adjustment_ov av,  Table(av.Adj_Items) ff
Where (av.Store_No IN (0, 1, 2, 3, 4, 5))
And (av.Post_Date Between To_Date('01-01-2002 00:00:00','MM-DD-YYYY
HH24:MI:SS')
And To_Date('07-30-2002 23:59:59','MM-DD-YYYY HH24:MI:SS')) And
(av.Sbs_No IN (1))
```

List Inventory Items

This example extracts inventory information from Retail Pro®.

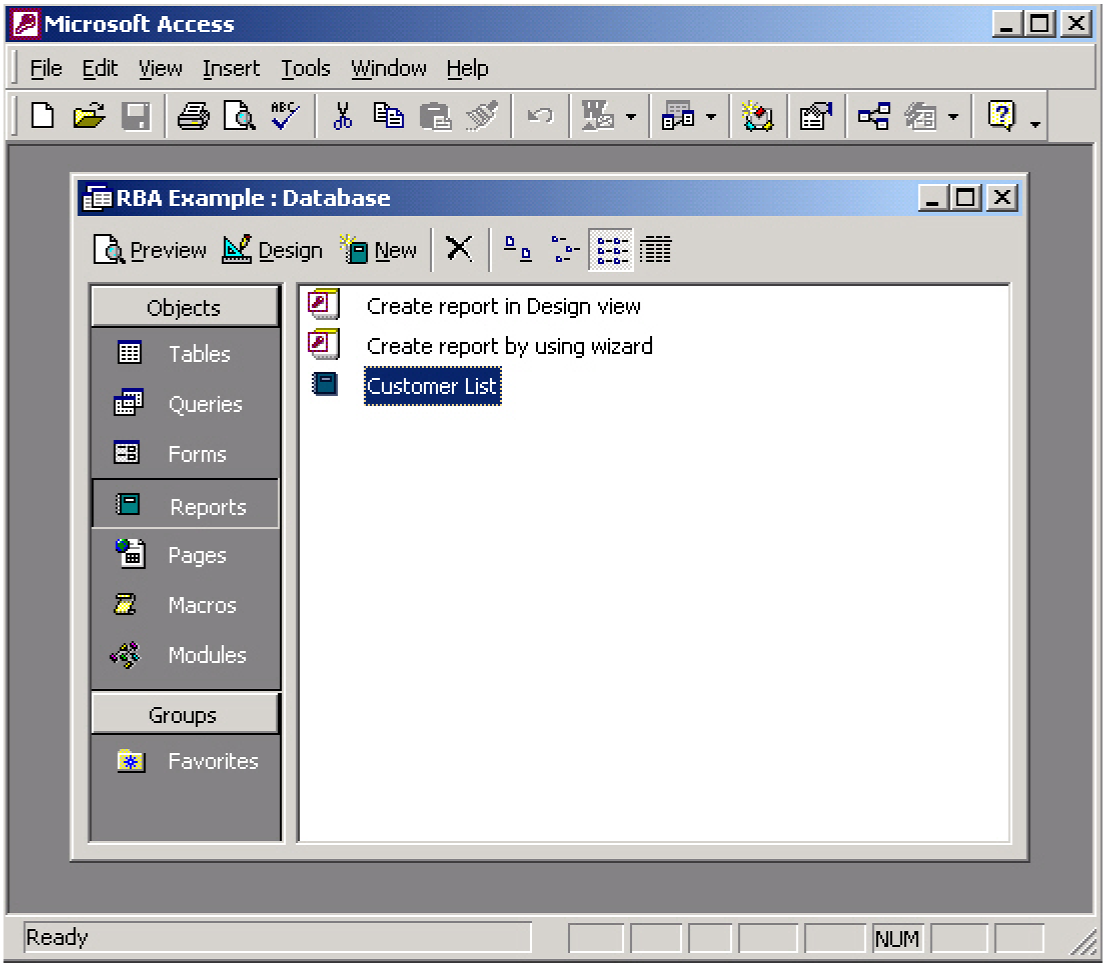
```
Select ck.Item_SID, ck.Store_No, ck.Sbs_No, ck.Price_Lvl,
ck.Inventory_Ref.Item_No,
ck.Inventory_Ref.DCS_Ref.Department, ck.Inventory_Ref.DCS_Ref.Class,
ck.Inventory_Ref.DCS_Ref.SubClass, ck.Inventory_Ref.Vend_Code,
ck.Inventory_Ref.Description1, ck.Inventory_Ref.Attr, ck.Inventory_Ref.Siz,
ck.Store_No
From Invn_Current_V ck
Where (ck.Sbs_No IN (1))
And (ck.Price_Lvl=1)
```

Microsoft Access Example

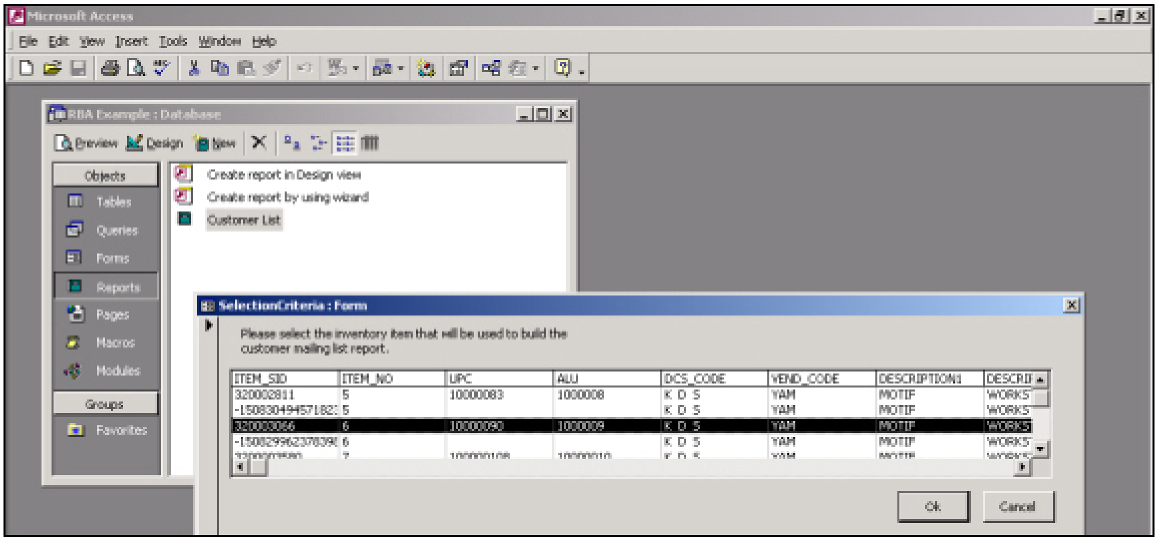
The example demonstrates how to create a simple custom report in Microsoft Access using an ODBC connection to Retail Pro.

In this example, you run an Access report (with parameters) that displays all customers who have purchased the specified inventory item. The report can be easily modified to accept different parameters such as DCS and style information instead of a single inventory item to use when producing the report. The report can then be used to send a customer mailing to all customers of a certain department to announce a sale event.

1. Open the Access example. Select the Reports object, and then double-click the Customer List report. The Selection Criteria form is displayed.



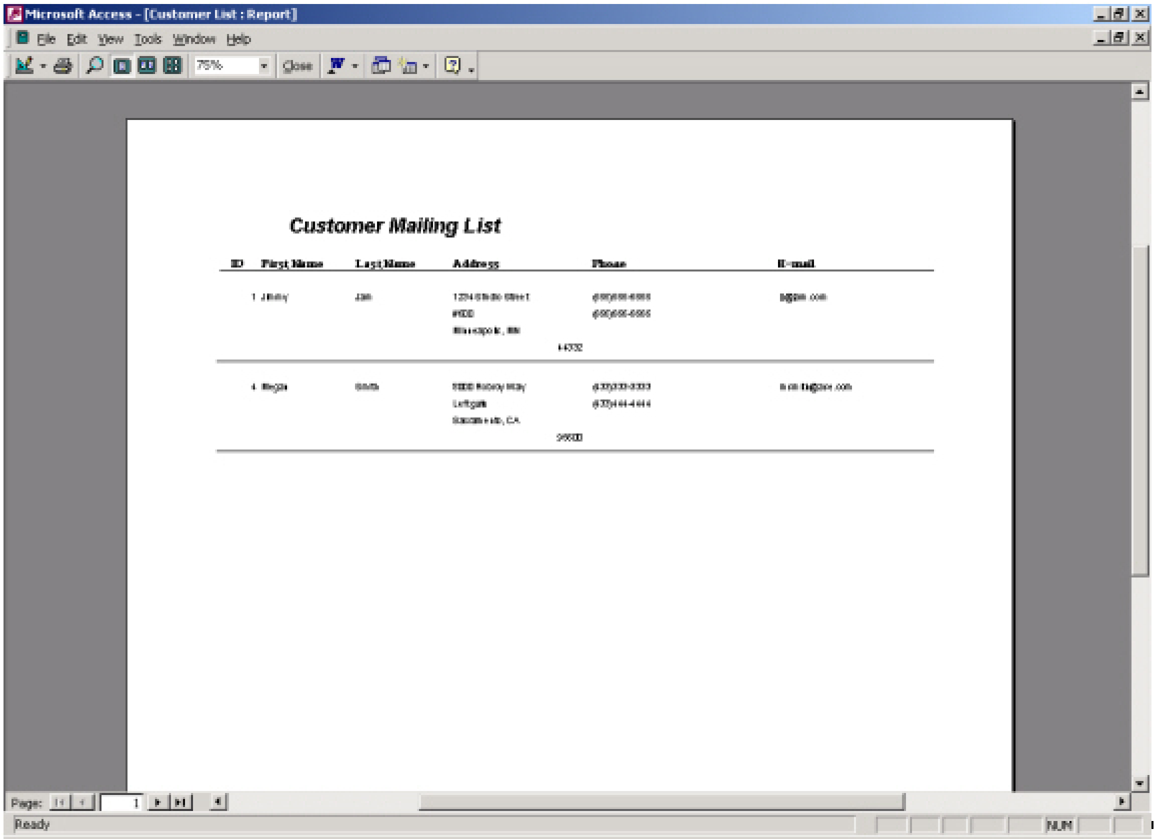
2. Select the inventory item that you are interested in.



Please select the inventory item that will be used to build the customer mailing list report.

ITEM_SID	ITEM_NO	UPC	ALU	DCS_CODE	VEND_CODE	DESCRIPTION1	DESCRIF
320002811	5	10000083	1000008	K D S	YAM	MOTIF	WORKS
-150830494571823	5			K D S	YAM	MOTIF	WORKS
320003066	6	10000090	1000009	K D S	YAM	MOTIF	WORKS
-150829962378390	6			K D S	YAM	MOTIF	WORKS
320003580	7	100000108	10000010	K D S	YAM	MOTIF	WORKS

3. Click **OK**.



Microsoft Access - [Customer List : Report]
File Edit View Tools Window Help
75%
Close
Page: 1
Ready
NUM

Customer Mailing List

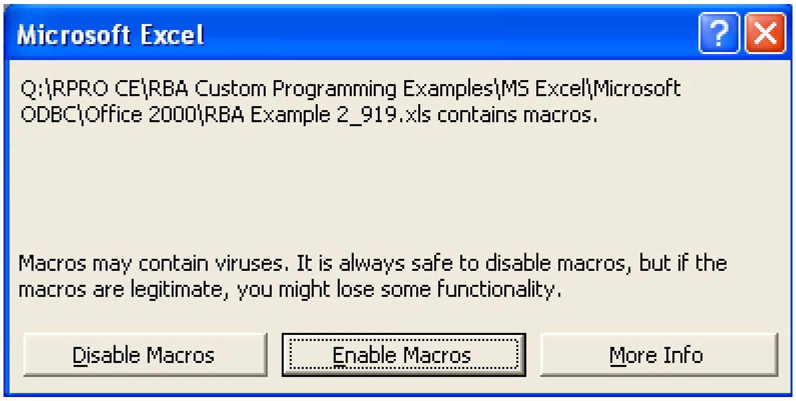
ID	First Name	Last Name	Address	Phone	E-mail
1	Jimmy	[illegible]	[illegible] #100 Minneapolis, MN 44332	[illegible] [illegible]	[illegible]
4	Megan	Smith	[illegible] Way [illegible] Sacramento, CA 95000	(333)333-3333 (333)444-4444	[illegible]

Microsoft Excel Example

This example demonstrates how to populate a Microsoft Excel worksheet with data from Retail Pro. The user can write an ad-hoc SQL query and execute the query. The user can also access the Microsoft Query Designer to visually design SQL queries.

This example was developed using standard Excel features and Visual Basic. All tools are standard Microsoft Office tools. This example should work correctly when run using Office 2000 (Excel 9) or Office 2002 (Excel 10).

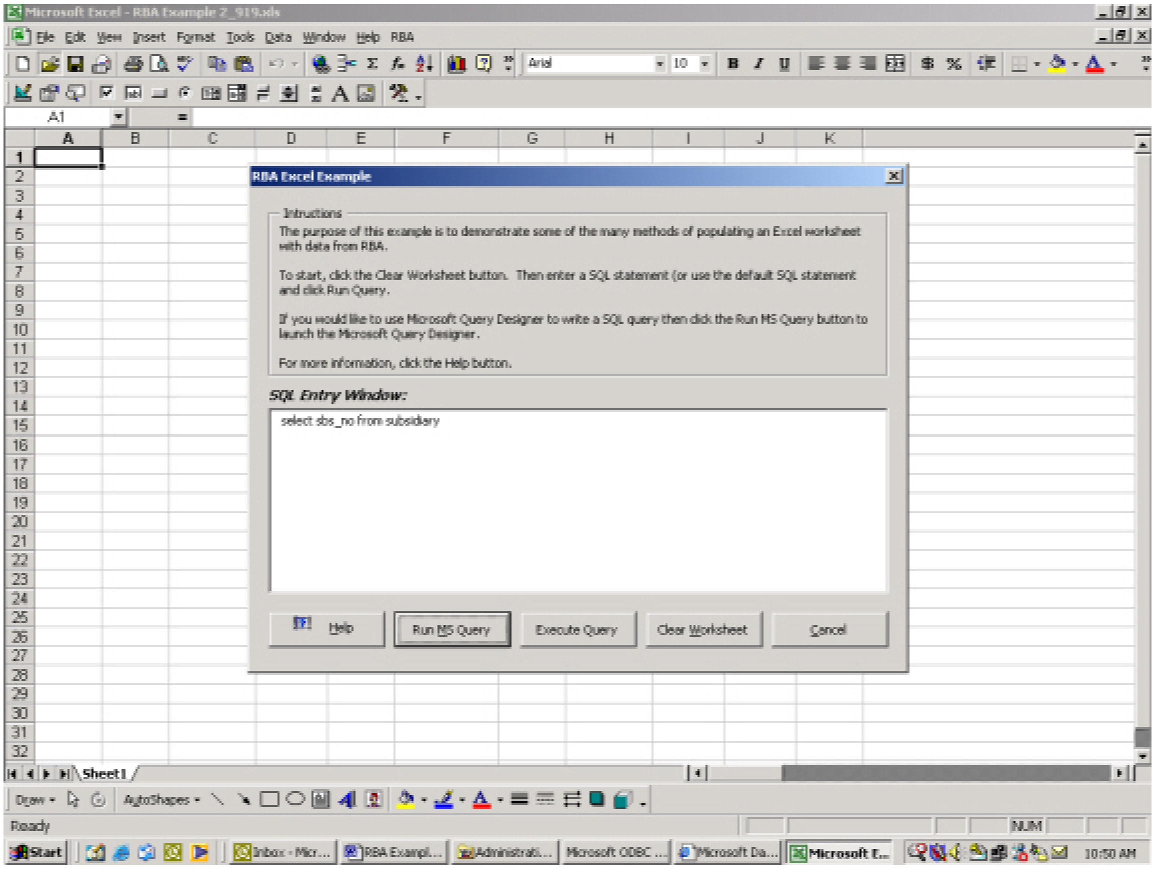
When you open the example project, depending on your Excel Macro Security Settings, you will be prompted to enable macros. Once you enable macros you will see the Retail Pro Excel Example main form. Retail Pro has also been added to the Excel menu bar.



Q:\RPRO CE\RBA Custom Programming Examples\MS Excel\Microsoft ODBC\Office 2000\RBA Example 2_919.xls contains macros.

Macros may contain viruses. It is always safe to disable macros, but if the macros are legitimate, you might lose some functionality.

1. To start, click **Clear Worksheet**. Then, enter a SQL statement (or use the default SQL statement and click **Execute Query**.



Microsoft Excel - RBA Example 2_919.xls
File Edit View Insert Format Tools Data Window Help RBA
RBA Excel Example

Intructions

The purpose of this example is to demonstrate some of the many methods of populating an Excel worksheet with data from RBA.

To start, click the Clear Worksheet button. Then enter a SQL statement (or use the default SQL statement and click Run Query.

If you would like to use Microsoft Query Designer to write a SQL query then click the Run MS Query button to launch the Microsoft Query Designer.

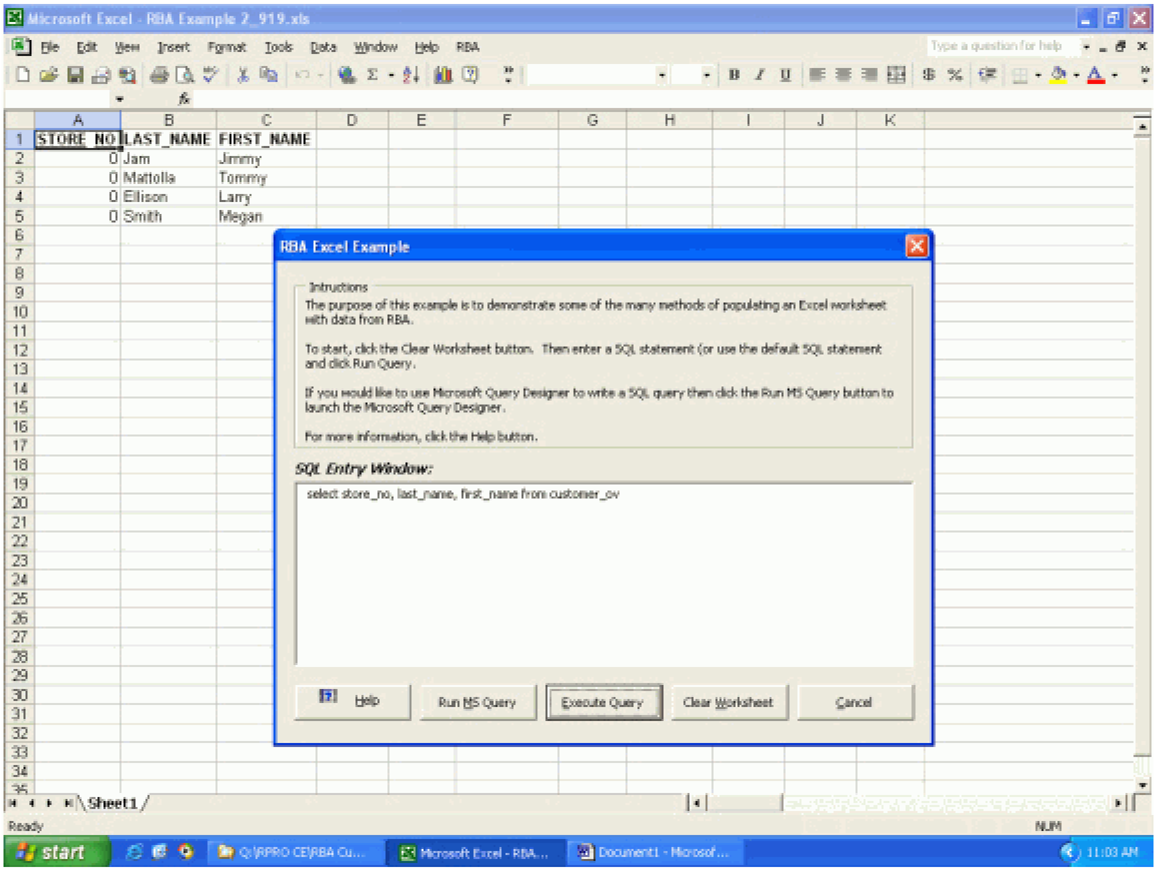
For more information, click the Help button.

SQL Entry Window:

```
select sbs_no from subsidiary
```

If you wanted to use Microsoft Query Designer to write a SQL query, select Run MS Query to launch the Microsoft Query Designer. This allows you to design a new query that can be used to populate the Excel worksheet. The example application does not support this functionality but you can refer to the MS Query and MS Excel documentation for instructions on how to do this. For more information, select the Help button. You can copy and paste the SQL statement created in MS Query into the example application's SQL Entry Window. When you select the Help button, the on-line help instructions are displayed in a pop-up dialog box.

2. After entering a SQL query and clicking **Execute Query**, the worksheet will be populated with the Retail Pro data returned from the query.



STORE_NO	LAST_NAME	FIRST_NAME
0	Jam	Jimmy
0	Mattolla	Tommy
0	Ellison	Larry
0	Smith	Megan

Intructions

The purpose of this example is to demonstrate some of the many methods of populating an Excel worksheet with data from RBA.

To start, click the Clear Worksheet button. Then enter a SQL statement (or use the default SQL statement and click Run Query.

If you would like to use Microsoft Query Designer to write a SQL query then click the Run MS Query button to launch the Microsoft Query Designer.

For more information, click the Help button.

SQL Entry Window:

```
select store_no, last_name, first_name from customer_ov
```

2. Before running the next query, clear the worksheet by clicking **Clear Worksheet**.

Microsoft Word Example

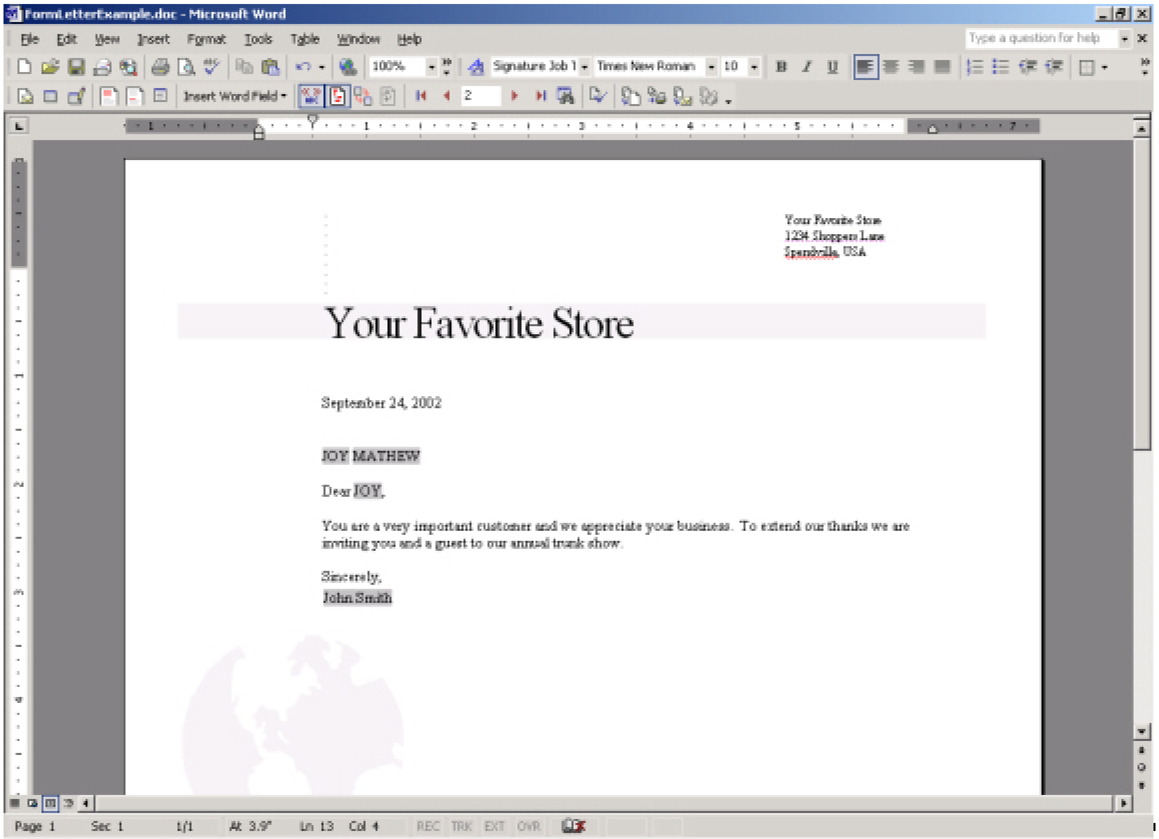
This example demonstrates how to incorporate data from Retail Pro into a Microsoft Word mail merge document.

The basic components of this example are a query written using Microsoft Query named Customer A-List.dqy and Microsoft Word's mail merge wizards. The query defines the SQL statement that will be used to retrieve customer sales information from Retail Pro. The query retrieves all customers who have spent $1000 or more in their lifetime at your store. The query results are then returned to Microsoft Word to be merged into the form letter that was designed using the mail merge wizards.

To Install the Microsoft Word example:

1. Copy the Microsoft Word file and the query file to any folder on your hard drive.
2. Double-click on FormLetterExample.doc.
3. When prompted, select 'Find datasource...' and open to Customer A-List.dqy.
4. You are prompted again to connect to the datasource each time the mail merge document is opened. Select Yes.
5. You are not prompted the next time that you run the example.
6. Word will connect to the data source and retrieve the data.

Once the data is retrieved Word will automatically merge the customer data into the form letter where the mail merge fields have been defined.


FormLetterExample.doc - Microsoft Word
File Edit View Insert Format Tools Table Window Help
Type a question for help
100%
Signature Job 1
Times New Roman
10
Insert Word Field
2
Page 1 Sec 1 1/1 At 3.9" Ln 13 Col 4 REC TRK EXT OVR

Your Favorite Store
1234 Shoppers Lane
Spendville, USA

Your Favorite Store

September 24, 2002

JOY MATHEW

Dear JOY,

You are a very important customer and we appreciate your business. To extend our thanks we are inviting you and a guest to our annual trunk show.

Sincerely,
John Smith

Key Performance Statistics

In this section of the document, the key performance statistics displayed in Profile reports are explained in more detail.

Average Daily Inventory

Average Daily Inventory is the average quantity on hand in inventory at the end of each day, after adjustment for all the day's transactional activities. By averaging inventory quantities over an entire period, retailers get a truer picture of inventory than from a single snapshot, such as on-hand quantities at the end of the period.

Knowing the average daily inventory helps retailers:

- Determine correct min/max levels
- Decide how often reorders will be necessary
- Calculate Turn and other performance statistics

How ADI is Calculated

To calculate ADI, the program starts with current inventory values and works backward in time, reversing each document's effect on inventory.

The average inventory for the period is then easily and accurately calculated by adding the daily totals and dividing by the number of days in the period.

Using a snapshot such as End of Period Qty could potentially show an average inventory value that is much greater for the month. Purchasing decisions based on this misleading data would result in the loss of potential sales if the merchandise sells out earlier than expected.

Note: Calculating ADI and Tracking Adjustments

To calculate ADI accurately, you must track quantity adjustments in Retail Pro.

Turn

Turn (also called Turnover) is a measure of how fast you are moving items through the door to customers. The advantages of having a high Turn rate include:

- **Increased sales volume** Fresh merchandise sells better and faster than old, shopworn merchandise.
- **Less risk of obsolete stock and markdowns** When inventory is selling quickly, merchandise isn't in the store long enough to become obsolete. As a result, markdowns are reduced and gross margins increase.

- **More money for market opportunities** When Turn is high, money previously tied up in inventory is freed to buy more merchandise.
- **Decreased operating expenses** An increase in Turn may mean that a lower level of inventory is supporting the same level of sales, which translates to lower inventory carrying costs.

However, an excessively high Turn rate can hurt retailers in the following ways:

- **Lowered sales volume** If customers can't find the size or color they seek—or even worse, if they can't find the product line at all—a sale is lost.
- **Increased cost of goods sold** By buying smaller quantities, the buyer can't take advantage of quantity and freight discounts.
- **Increased operating expenses** A buyer spends about the same amount of time meeting with vendors and writing orders whether the order is large or small. It also takes about the same amount of time, for both large and small orders, to print invoices, receive merchandise, and pay invoices.

How Turn by Quantity is Calculated

Sold_Qty * (365/# days in period, up to 365) / ADI

# days = (END_DATE - GREATEST(NVL(INVN_SBS.FST_RCVD_DATE, passed BEGIN_DATE), passed BEGIN_DATE) + 1)

How Turn by Cost is Calculated

(Sold_Qty * Cost) * (365/# days in period, up to 365) / ADC

# days = (END_DATE - GREATEST(NVL(INVN_SBS.FST_RCVD_DATE, passed BEGIN_DATE), passed BEGIN_DATE) + 1)

GMROI

GMROI reports the gross profitability of an item — the number of margin dollars returned in one year for each dollar invested in inventory stock. An easy way to think of GMROI is: "For every dollar I put into this item, I will get X amount back, on an annual basis."

GMROI helps retailers:

Put inventory dollars into merchandise that gives the greatest return on their investment.

Identify low-profit merchandise that can be culled from the inventory mix.

How GMROI Is Calculated:

(SOLD_EXT_PRICE - SOLD_EXT_COST) * (365/# days in period, up to 365) / RAW_ADC

# days in period = (END_DATE - GREATEST(NVL(INVN_SBS.FST_RCVD_DATE, passed BEGIN_DATE), passed BEGIN_DATE) + 1)

Improving GMROI

Gross margin is the value of sales minus the cost of goods sold. To increase gross margin, you must either increase sales revenue or reduce the cost of the merchandise.

The obvious way to increase sales revenue is simply to increase prices. Unfortunately, in a competitive environment, that is not so simple.

Here are three possible strategies to employ to increase GMROI:

- Raising Prices
- Careful Use of Markdowns
- Reducing Cost of Goods Sold

Raising Prices

A common rule of thumb is to avoid price increases on products that are known value items or those that your competitors focus on for price comparisons.

Sample Steps for Implementing a Price Increase

Define a representative sample selection of products from your stores. Each product should be of a particular size or be aimed at a particular demographic.

Select a control group that resembles as closely as possible the first group.

Increase prices in the sample stores and track how sales perform against the control group during a trial period.

If there is little or no effect on sales, you can roll out the price change throughout your stores.

Reducing Markdowns

Reducing markdowns is another strategy for improving GMROI. To do this, track the Sell Through % by store.

Suppose Store A is selling faster than plan and will sell out early. Store B is selling slower than plan but a markdown may help it along. Store C is doing really poorly and even a large markdown may not be enough to clear the problem. In this case, consider moving merchandise from Store C to Store A, where it can still sell at full price.

This capacity to apply markdowns selectively and transfer goods judiciously is an important component in increasing gross margin.

Reducing Cost of Goods Sold

You can, for example, analyze vendor performance and possibly order products from different vendors. Again, this is not a simple decision and must be considered with other factors. For example, one vendor may be a little more expensive but delivers on time and offers better quality with fewer returns. When all the factors are considered, the more expensive supplier may, in fact, be "cheaper."

The Role of Planning

Informed merchandise planning can also offer a valuable impact. In addition to sales and stock figures, this could include, for example, competitor prices and the promotional calendar.

GMROI Example

Here is an example of how GMROI works.

You purchase 1 pen on Jan. 1 from your vendor for $0.10 The pencil has a selling price of $1.00 and stays in your inventory the entire year until, finally, on Dec. 31, someone comes into your store and buys it. The Annualized Profit (**Margin $**) is 90 cents ($1.00 - $0.10 = $0.90). The Average Daily Cost is 10 cents (its Cost on Jan 1 was $0.10, on Jan.2 was $0.10 … all the way to Dec. 31).

When we divide the annualized profit by the average daily cost, we get a GMROI of 9.

$$\frac{\begin{array}{c}\textbf{\$0.90}\\ \textbf{(Annualized Profit)}\end{array}}{\begin{array}{c}\textbf{\$0.10}\\ \textbf{(Average Daily Cost)}\end{array}} = \mathbf{9}$$

If instead you buy 12 pens on Jan. 1 (for $1.20) and sell all 12 on Dec. 31, (for $12.00), the annualized profit increases, but the average daily cost increases at the same rate, so GMROI stays the same.

$$\frac{\begin{array}{c}\textbf{\$10.80}\\ \textbf{(Annualized Profit)}\end{array}}{\begin{array}{c}\textbf{\$1.20}\\ \textbf{(Average Daily Cost)}\end{array}} = \mathbf{9}$$

However, if you buy 12 pens on Jan. 1 and sell one pencil at the end of every month, the annualized profit is still $10.80 ($12.00 - $1.20), but the Average Daily Cost drops to $0.60.

$$\frac{\begin{array}{c}\textbf{\$10.80}\\ \textbf{(Annualized Profit)}\end{array}}{\begin{array}{c}\textbf{\$0.60}\\ \textbf{(Average Daily Cost)}\end{array}} = \mathbf{18}$$

Suppose that you don't want to pay storage costs throughout the year, so you buy only one pen at the beginning of every month and then sell it on the last day of every month. Your annualized profit is still $10.80 but the average daily cost drops to $0.10. This increased profitability is shown in the increase of GMROI to 108.

$$\frac{\textbf{Annualized Profit (\$10.80)}}{\textbf{Average Daily Cost (\$0.10)}} = \mathbf{108}$$

Stock / Sales

Stock/Sales is the ratio (for a specified period) of stock available for sale to what was sold.

If Stock/Sales is higher than usual, it means merchandise isn't selling as quickly as anticipated. To correct the imbalance, retailers will have to cut back on orders or mark down merchandise to avoid becoming overstocked.

How Stock / Sales is Calculated

```
if Sold_qty = 0 then
  STOCK_SALES := 1
else
  STOCK_SALES := 1 + EOP_OH_QTY / SOLD_QTY
if  (P_END_DATE = passed END_DATE then
  EOP_OH_QTY := SUM(OH_QTY)
else
  EOP_OH_QTY := 0
```

Sell Thru %

Sell Thru % is the percentage of items available for sale (for a specified period) that were actually sold, in other words, the inverse of Stock/Sales. The lower the Stock/Sales ration, the higher the Sell Thru %. For example, if you had 1,500 units available at the beginning of the period and sold 500, the Sell Thru % would be 33.33.

How Sell-Thru % Is Calculated

```
if SOLD_QTY + EOP_OH_QTY = 0 then
  SELL_THRU = 0
else
  SELL_THRU := SOLD_QTY * 100 / (SOLD_QTY + EOP_OH_QTY)
if  (P_END_DATE = passed END_DATE then
  EOP_OH_QTY := SUM(OH_QTY)
else
  EOP_OH_QTY := 0
```

Days of Supply

The Days of Supply (DOS), tells you the number of days until you sell out your remaining stock of the item.

Days of Supply (DOS) tells you the number of days remaining until you sell out your remaining stock of an item.

This statistic helps retailers decide when to reorder merchandise. For example, if a vendor requires a lead time of one week for reorders of a certain item, and the remaining days of supply is 10, a retailer needs to start making a purchase order soon or risk selling out of an item.

How Days of Supply is Calculated

```
if SOLD_QTY = 0 then
  DOS := 0
else
  DOS:=EOP_OH_QTY*(passedEND_DATE-
GREATEST(NVL(INVN_SBS.FST_RCVD_DATE, passed BEGIN_DATE), passed BEGIN_DATE) + 1) / SOLD_QTY)
```

Index