

R:BASE 11

for Windows

Database Conversion Guide





R:BASE 11 for Windows

Database Conversion Guide

by R:BASE Technologies, Inc.

Welcome to R:BASE 11 for Windows!

R:BASE 11 is the new major release from R:BASE Technologies showcasing the latest advances for your databases and applications.

R:BASE 11 combines pure performance and high-tech features with a practical interface for users of all levels. R:BASE 11 users can achieve peak productivity to overachieve, outperform, and over-deliver on data management goals.

R:BASE 11 for Windows Database Conversion Guide

Copyright © 1982-2025 R:BASE Technologies, Inc.

Information in this document, including URL and other Internet web site references, is subject to change without notice. The example companies, individuals, products, organizations and events depicted herein are completely fictitious. Any similarity to a company, individual, product, organization or event is completely unintentional. R:BASE Technologies, Inc. shall not be liable for errors contained herein or for incidental consequential damages in connection with the furnishing, performance, or use of this material. This document contains proprietary information, which is protected by copyright. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written consent of R:BASE Technologies, Inc. We reserve the right to make changes from time to time in the contents hereof without obligation to notify any person of such revision or changes. We also reserve the right to change the specification without notice and may therefore not coincide with the contents of this document. The manufacturer assumes no responsibilities with regard to the performance or use of third party products.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

The software described in this document is furnished under a license agreement. The software may be used or copied only in accordance with the terms of that agreement. Any unauthorized use or duplication of the software is forbidden.

R:BASE Technologies, Inc. may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from R:BASE Technologies, Inc., the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Trademarks

R:BASE®, Oterro®, RAdmin®, R:Scope®, R:Mail®, R:Charts®, R:Spell Checker®, R:Docs®, R:BASE Editor®, R:BASE Plugin Power Pack®, R:Style®, RBZip®, R:Mail Editor®, R:BASE Dependency Viewer®, R:Archive®, R:Chat®, R:PDF Form Filler®, R:FTPClient®, R:SFTPClient®, R:PDFWorks®, R:Magellan®, R:WEB Reports®, R:WEB Gateway®, R:PDFMerge®, R:PDFSearch®, R:Documenter®, RBInstaller®, RBUpdater®, R:AmazonS3®, R:GAP®, R:Mail Viewer®, R:Capture®, R:Synchronizer®, R:Biometric®, R:CAD Viewer®, R:DXF®, R:Twain2PDF®, R:Scheduler®, R:Scribbler®, R:SmartSig®, R:OutLink®, R:HASH®, R:JobTrack®, R:TimeTrack®, R:Manufacturing®, R:QBDataDirect®, R:QBSynchronizer®, and R:QBDBExtractor®, and Pocket R:BASE® are trademarks or registered trademarks of R:BASE Technologies, Inc. All Rights Reserved. All other brand, product names, company names and logos are trademarks or registered trademarks of their respective companies.

Windows, Windows 11-10, Windows Server 2025-2016, Azure Maps, Word, Excel, Access, SQL Server, and Outlook are registered trademarks of Microsoft Corporation. OpenOffice is a registered trademark of the Apache Software Foundation.

Printed: July 2025 in Murrysville, PA

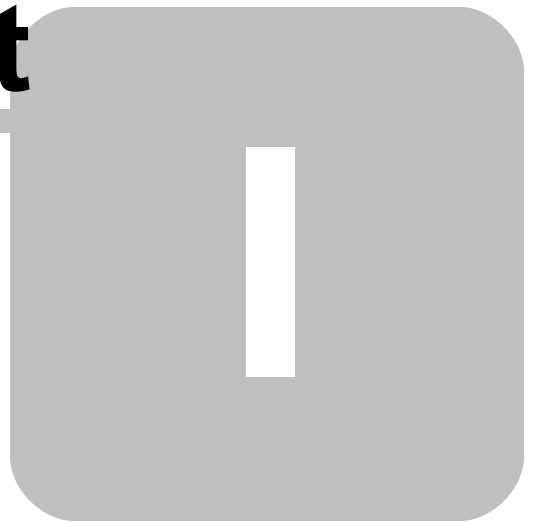
First Edition

Table of Contents

Part I Preparation	6
1 Make a Database Backup	7
2 Review Table and Column Names	7
Part II Converting the Database Files	10
1 Recognizing the Database Files	11
2 Conversion Steps	11
Converting from R:BASE 4000/5000	12
Converting from R:BASE System V to R:BASE 4.0	13
Converting from R:BASE 4.5 through 6.5++	15
Converting from R:BASE 7.x/9.x (32)/10.x	17
Converting from R:BASE Turbo V-8	18
Converting from R:BASE eXtreme 9.x (64)	20
Converting from R:BASE X/X.5 Enterprise (10.x)	22
Part III Converting Forms, Reports, & Labels	23
1 R:BASE 6.5++ and Lower Forms, Reports, & Labels	24
Converting Forms	24
Issues and Suggestions.....	29
Converting Reports	31
Issues and Suggestions.....	34
Converting Labels	37
Issues and Suggestions.....	39
2 R:BASE 7.x/Turbo V-8 Forms, Reports, & Labels	41
3 R:BASE eXtreme 9.x (32/64) Forms, Reports, & Labels	42
4 New Conversion Enhancements	42
Part IV Converting Applications	46
1 R:BASE 6.5++ and Lower Applications	47
Application Express Users (.APP)	47
Menu System Types.....	48
Group Bar	49
Menu Bar	51
Tree View	53
Split View	54
Tile Menu	57
Menu Action Types.....	60
Command Syntax Reference.....	62
Command Files	62
Updating the Command Syntax	63
Table and Column Name Changes.....	64
Obsolete Commands.....	65
DOS to Windows Conversion.....	69
Enhanced Commands.....	71

Unsupported Command Parameters.....	71
New Application Formats	72
2 R:BASE 7.x/Turbo V-8 Applications (.RBA)	72
3 R:BASE eXtreme 9.x Applications	74
4 PAGELOCK Setting	74
Part V ODBC Compliance	76
Part VI Y2K Compliance	78
Part VII Database Integrity Routine	81
Part VIII Useful Resources	84
Part IX Feedback	86
Index	88

Part



1 Preparation

When preparing to convert a database and use those files in R:BASE 11, it is important that you follow the proper steps to convert that database. Please review the following steps and recommendations to ensure you convert your legacy database(s) successfully.

1.1 Make a Database Backup

Before you begin, make a backup copy of your database. You should place the database on a storage medium that is different than the one which your original files are already stored upon. For example, if you are using a hard disk to store the data, you can back up your database to an external hard drive, a flash drive, a tape drive, or a CD-ROM. With these storage options, you can also copy the backup files back to your hard drive.

If users are still entering data into the live database, while the conversion is taking place, use a different database copy than the one in use. When the conversion is complete, the live data being added to can always be loaded into the converted database at a later time. Make sure no users are connected to the database when the copy is made.

During the conversion process, it is also recommended to make periodic backups. This will prevent you from starting from the beginning in case any problems occur.

1.2 Review Table and Column Names

You must review all table and column names to be sure that no invalid names are defined and no reserved words are used.

In the current version, a 1-18 character alpha-numeric name must be specified for a column. Spaces are NOT permitted. Valid names must start with an alpha character and can include the following symbols:

- Letters (A-Z)
- Numbers (0-9)
- _ (underscore)
- # (pound sign) *
- \$ (dollar sign) *
- % (percent sign) *

***Note:** For ODBC compliance, it is NOT recommended to use the #, \$, or % even though R:BASE permits it.

Review your table and column names that may begin with a number. Check for any characters that are not supported like a question mark "?" or a greater than character ">".

Also review your table and column names to verify that no reserved words are used. These words, commands, keywords, and other names are to be used exclusively by the R:BASE database. Names used by System Tables, System Views and System Columns are also reserved words. As a rule, all words beginning with "**SYS**" are reserved.

Do not use reserved words or any shorter forms of them as names for columns, tables, or views. As a rule, if the word is a reserved word, R:BASE will flag it. R:BASE will not allow you to use a reserved word as a column or table name, but this MAY NOT always be the case. For example in the table designer, R:BASE may not warn you about **REF**, short for **REFERENCES**, but **REF** will not be allowed in a command file. If a particular column or table is giving you problems, please check out the list below and consider all shortened versions of the words listed here.

The following is a list of all reserved words.

#DATE	LE
#PI	LEAD
#TIME	LIKE

ADA	LIMIT
ADD	LISTREL
ALL	LT
AND	MAX
ANY	MAXIMUM
AS	MIN
ASC	MINIMUM
ASCII	MODULE
AT	MPW
AUTHORIZE	NE
AUTONUMBER	NOCHECK
AVERAGE	NOFILL
AVG	NONE
BEGINS	NONUM
BETWEEN	NOT
BLINKING	NULL
BOTH	NUM
BY	OF
CASCADE	OFF
CHARACTER	ON
CHECK	OPTION
CMDHIST	OR
COLUMNS	ORDER
CONSTRAINT	OUTER
CONTAINS	OVER
CONTINUE	OWNER
COUNT	PARTITION
CURRENT	PLI
CURSORS	PRECISION
DATA	PRINTER
DBCONN	PRIVILEGE
DECIMAL	PROCEDURE
DESC	PUBLIC
DISTINCT	READ
DUPLICATE	REFERENCES
END	RPW
ENDC	SCHEMA
EQ	SCREEN
ESCAPE	SECTION
EXCEPT	SELECT
EXECUTE	SMALLINT
EXISTS	SOME
EXPLAIN	SORTED
FAILS	SOUNDS LIKE
FILL	SQLCODE
FLOAT	SQLERROR
FOR	STATICVAR
FOUND	STDEV
FROM	STRUCTURE
FULL	SUM
GE	TABLE
GO	TABLES
GOTO	TEMPORARY
GROUPED	TERMINAL
GT	TJOURNAL
HAVING	TO
IN	UNION
INDEX	UNIQUE
INDICATOR	USING
INTO	VALUES
IS	VARIANCE
KEY	VIEWS
KEYBOARD	WHERE
LAG	WITH
LANGUAGE	WORK

LAST	
------	--

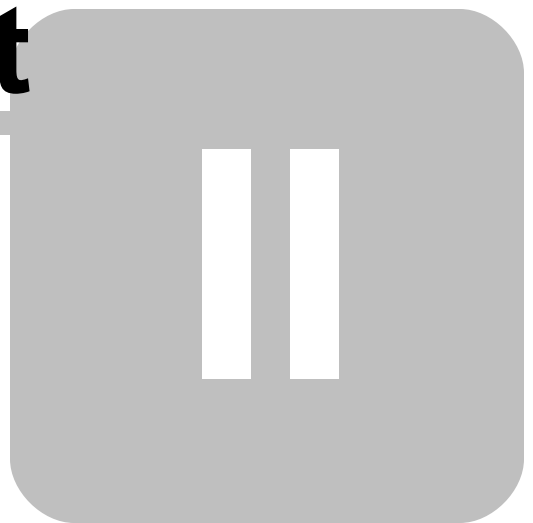
Of the above, the following are new reserved words added with the release of R:BASE 11.

- CMDHIST
- STATICVAR
- EXPLAIN
- DBCONN
- TJOURNAL

If you are still using a legacy version of R:BASE, it is recommended that you perform these table and column name changes in that version. After making the table and column names changes, R:BASE will update the form, report, and label column objects with the appropriate new name. However, if any of the columns and tables are listed in the form, report, and label variable expressions, you must manually edit these expressions.

On paper, record any table and column name changes. Later in the conversion process when you're updating your command files, you will need to make these necessary name changes for the command files as well.

Part



2 Converting the Database Files

2.1 Recognizing the Database Files

When locating and using the database name in the conversion instructions, be sure to remember that the "actual" database name for .RBS and .RBF files does not include the "1", "2", or "3", which is located before the file extension. For example, a R:BASE 3.1 database with the name SALES will have the database files SALES1.RBF, SALES2.RBF, and SALES3.RBF, but the database name is still just "SALES". In the steps below where the "dbname" is required in the command syntax, use only the "actual" database name.

To use this document you will need to know which version of R:BASE you are upgrading from. The table below lists the various R:BASE file formats. Once you find your R:BASE database version from the table below, follow the necessary conversion steps in the next section.

R:BASE Database File Formats

Version	File Format
R:BASE 4000 and R:BASE 5000	DBNAME1.RBS DBNAME2.RBS DBNAME3.RBS
R:BASE System V to R:BASE 4.0	DBNAME1.RBF DBNAME2.RBF DBNAME3.RBF
R:BASE 4.5 to R:BASE 5.0	DBNAME.RB1 DBNAME.RB2 DBNAME.RB3
R:BASE 5.1 to R:BASE 10.x	DBNAME.RB1 DBNAME.RB2 DBNAME.RB3 DBNAME.RB4
R:BASE Turbo V-8, R:BASE eXtreme 9.x (64), R:BASE 10.x Enterprise, and R:BASE 11	DBNAME.RX1 DBNAME.RX2 DBNAME.RX3 DBNAME.RX4

2.2 Conversion Steps

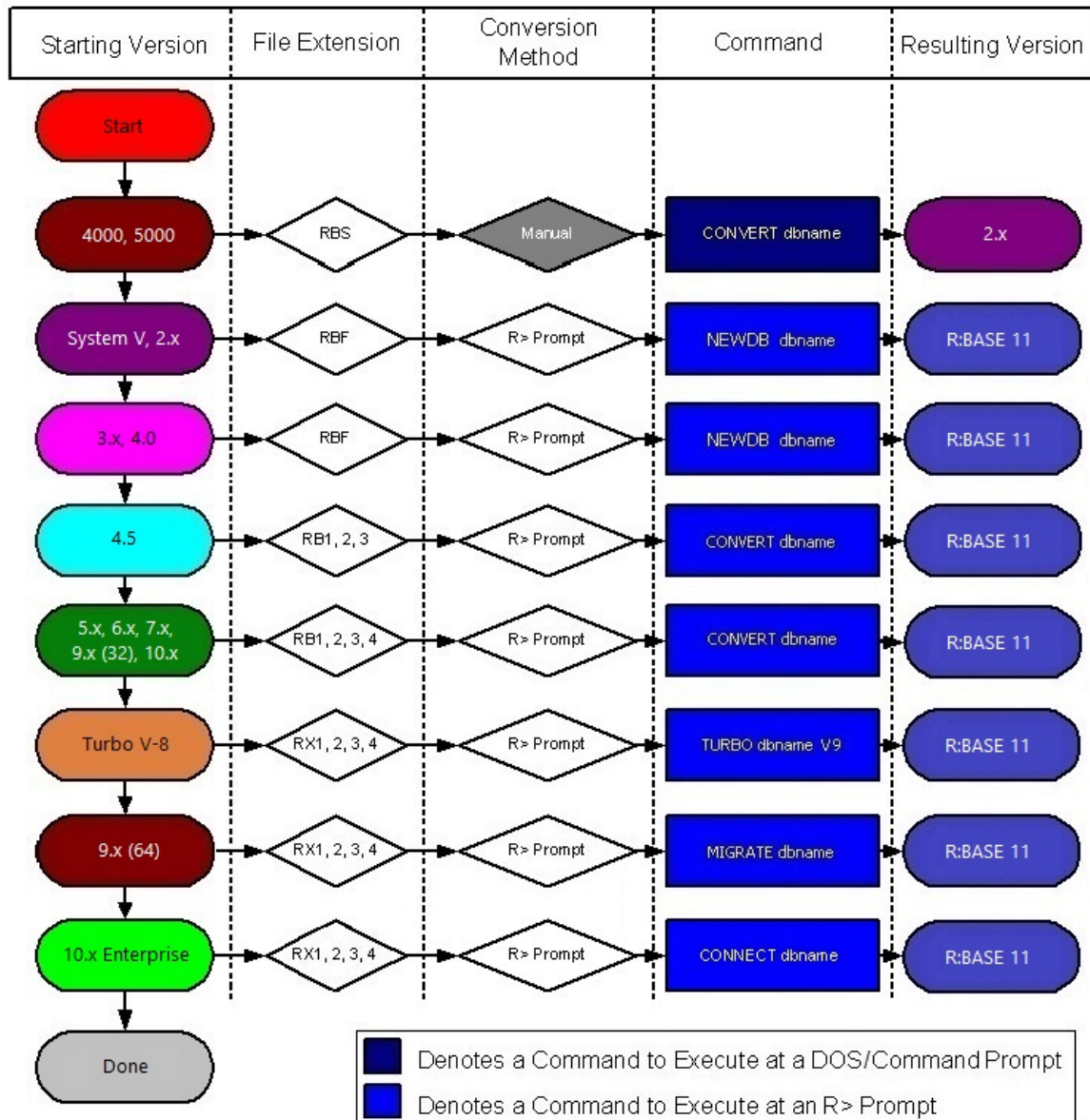
PLEASE REMEMBER TO BACK UP YOUR DATABASE!

The steps required to convert databases to work with R:BASE 11 for Windows are described in the following flow chart. Locate your current R:BASE database file "Starting Version" in the left column and complete the steps until you reach the bottom. If you are not sure of your database file version, refer to the [Recognizing the Database Files](#) chart.

In the sections that follow, proceed to the chapter that lists the version you are upgrading from. Your options include:

- [Converting from R:BASE 4000/5000](#)
- [Converting from R:BASE System V to R:BASE 4.0](#)
- [Converting from R:BASE 4.5 through 6.5++](#)
- [Converting from R:BASE 7.x/9.x \(32\)/10.x](#)
- [Converting from R:BASE Turbo V-8](#)
- [Converting from R:BASE eXtreme 9.x \(64\)](#)

- [Converting from R:BASE X/X.5 Enterprise \(10.x\)](#)



2.2.1 Converting from R:BASE 4000/5000

If you have not done so already, please contact R:BASE Technologies to receive the legacy conversion utility for migrating R:BASE 4000/5000 databases. The legacy conversion utility is a 16-bit application file that will not open on 64-bit operating systems.

IMPORTANT: The instructions below require the steps be executed on a 32-bit operating system.

1. Create a new directory; perhaps something near your root directory on the local drive (example: "C:\DBTEMP").
2. Copy the contents of the "CONVERT" directory provided by R:BASE Technologies into the directory created in Step 1.
3. Copy your .RBS [database files](#) to the new directory.

4. Open a Command Prompt or DOS Prompt window and navigate to the new directory created above in Step 1.
5. At the Prompt, type: `CONVERT dbname`
6. Repeat for all databases. In addition to the original .RBS database files, three new files will be created for each data that have the .RBF file extensions.
7. Exit the Command Prompt or DOS Prompt window. Use the EXIT command to exit.
8. Continue with the steps listed at [Converting from R:BASE System V to R:BASE 4.0](#)

2.2.2 Converting from R:BASE System V to R:BASE 4.0

At this point, you must have R:BASE 11 installed on your computer. Be sure to review the R:BASE 11 Getting Started Guide regarding the installation and set up for your R:BASE software. The Getting Started Guide is located in the R:BASE program directory.

If there is no OWNER or Password

1. Install and launch R:BASE 11
2. Navigate to your database directory using the "Change Current Folder" button (bottom left corner, folder icon with a green arrow, at the Database Explorer)
3. Go to the R> Prompt window (third button from the left on the Tool Bar, or press [Ctrl]+[R])
4. At the R> Prompt, type: `SET MULTI OFF` , then press [Enter]
5. At the R> Prompt, type: `SET STATICDB OFF` , then press [Enter]
6. At the R> Prompt type: `NEWDB DBNAME` , then press [Enter]
7. At the R> Prompt, type: `CONNECT DBNAME` , then press [Enter]
8. At the R> Prompt, type: `SET IDQUOTES=`` , then press [Enter] (the ` quote character is the reverse single quote, which is next to the "1" and under the tilde "~", on American keyboards)
9. At the R> Prompt, type: `DISCONNECT` , then press [Enter]
10. Perform the "Complete Database Rebuild" instructions below
11. Repeat "Steps 6-10" for all databases

If there is a Username and Password

1. Install and launch R:BASE 11
2. Navigate to your database directory using the "Change Current Folder" button (bottom left corner, folder icon with a green arrow, at the Database Explorer)
3. Go to the R> Prompt window (third button from the left on the Tool Bar, or press [Ctrl]+[R])
4. At the R> Prompt, type: `SET MULTI OFF` , then press [Enter]
5. At the R> Prompt, type: `SET STATICDB OFF` , then press [Enter]
6. At the R> Prompt, type: `SET USER UserName Password` , then press [Enter]
7. At the R> Prompt type: `NEWDB DBNAME` , then press [Enter]
8. At the R> Prompt, type: `CONNECT DBNAME` , then press [Enter]
9. At the R> Prompt, type: `SET IDQUOTES=`` , then press [Enter] (the ` quote character is the reverse single quote, which is next to the "1" and under the tilde "~", on American keyboards)
10. At the R> Prompt, type: `DISCONNECT` , then press [Enter]
11. Perform the "Complete Database Rebuild" instructions below
12. Repeat "Steps 6-11" for all databases

If there is a database OWNER

1. Install and launch R:BASE 11
2. Navigate to your database directory using the "Change Current Folder" button (bottom left corner, folder icon with a green arrow, at the Database Explorer)
3. Go to the R> Prompt window (third button from the left on the Tool Bar, or press [Ctrl]+[R])
4. At the R> Prompt, type: `SET MULTI OFF` , then press [Enter]
5. At the R> Prompt, type: `SET STATICDB OFF` , then press [Enter]
6. At the R> Prompt, type: `SET USER OwnerName` , then press [Enter]
7. At the R> Prompt type: `NEWDB DBNAME` , then press [Enter]
8. At the R> Prompt, type: `CONNECT DBNAME` , then press [Enter]
9. At the R> Prompt, type: `SET IDQUOTES=`` , then press [Enter] (the ` quote character is the reverse single quote, which is next to the "1" and under the tilde "~", on American keyboards)

10. At the R> Prompt, type: `DISCONNECT` , then press [Enter]
11. Perform the "Complete Database Rebuild" instructions below
12. Repeat "Steps 6-11" for all databases

Database Rebuild

In addition to the above conversion steps, the following steps are advised to completely rebuild the database. These steps will be performed in R:BASE 11.

1. Connect to the database and open the R> Prompt.
2. Check the IDQUOTE settings of connected database.

At the R> Prompt, type:

```
SHOW CHARACTERS
```

Notice the last parameter, **IDQUOTES**

If IDQUOTES is set to NULL, you'll need to change NULL to the appropriate character, such as ` (reversed quote) as following:

```
SET IDQUOTES= `
```

3. Unload the entire database. Use the SET USER command below if an OWNER name exists.

At the R> Prompt, type:

```
SET NULL -0-  
SET USER ownername  
OUTPUT filename.ALL  
UNLOAD ALL  
OUTPUT SCREEN
```

This process will create two files for Windows databases (filename.ALL and filename.LOB), and a single file for DOS databases.

4. Disconnect the database.

At the R> Prompt, type:

```
DISCONNECT
```

5. Now rename the database that was just unloaded in Steps 1-3.

Review the list of Databases in the R:BASE Database Explorer, and select the "Rename" option. Enter a new name for the database.

6. Finally, use R:BASE to rebuild the database by running the .ALL file.

At the R> Prompt type:

```
RUN filename.ALL
```

These steps will build a clean database!

If there is a considerable amount of errors with both rebuilding the structure and/or data, please refer to the "Database Repair Routine" within the Database Maintenance PDF document in the R:BASE program directory.

With the database file conversion complete, proceed to the chapter for [Converting Forms, Reports, & Labels](#).

2.2.3 Converting from R:BASE 4.5 through 6.5++

At this point, you must have R:BASE 11 installed on your computer. Be sure to review the R:BASE 11 Getting Started Guide regarding the installation and set up for your R:BASE software. The Getting Started Guide is located in the R:BASE program directory.

If there is no OWNER or Password

1. Install and launch R:BASE 11
2. Navigate to your database directory using the "Change Current Folder" button (bottom left corner, folder icon with a green arrow, at the database Explorer)
3. Go to the R> Prompt window (third button from the left on the Tool Bar, or press [Ctrl]+[R])
4. At the R> Prompt, type: `SET MULTI OFF` , then press [Enter]
5. At the R> Prompt, type: `SET STATICDB OFF` , then press [Enter]
6. At the R> Prompt type: `CONVERT DBNAME` , then press [Enter]
7. At the R> Prompt, type: `CONNECT DBNAME` , then press [Enter]
8. At the R> Prompt, type: `SET IDQUOTES=` , then press [Enter] (the ` quote character is the reverse single quote, which is next to the "1" and under the tilde "~", on American keyboards)
9. At the R> Prompt, type: `DISCONNECT` , then press [Enter]
10. Perform the "Complete Database Rebuild" instructions below
11. Repeat "Steps 6-10" for all databases

If there is a Username and Password

1. Install and launch R:BASE 11
2. Navigate to your database directory using the "Change Current Folder" button (bottom left corner, folder icon with a green arrow, at the Database Explorer)
3. Go to the R> Prompt window (third button from the left on the Tool Bar, or press [Ctrl]+[R])
4. At the R> Prompt, type: `SET MULTI OFF` , then press [Enter]
5. At the R> Prompt, type: `SET STATICDB OFF` , then press [Enter]
6. At the R> Prompt, type: `SET USER UserName Password` , then press [Enter]
7. At the R> Prompt type: `CONVERT DBNAME` , then press [Enter]
8. At the R> Prompt, type: `CONNECT DBNAME` , then press [Enter]
9. At the R> Prompt, type: `SET IDQUOTES=` , then press [Enter] (the ` quote character is the reverse single quote, which is next to the "1" and under the tilde "~", on American keyboards)
10. At the R> Prompt, type: `DISCONNECT` , then press [Enter]
11. Perform the "Complete Database Rebuild" instructions below
12. Repeat "Steps 6-11" for all databases

If there is a database OWNER

1. Install and launch R:BASE 11
2. Navigate to your database directory using the "Change Current Folder" button (bottom left corner, folder icon with a green arrow, at the Database Explorer)
3. Go to the R> Prompt window (third button from the left on the Tool Bar, or press [Ctrl]+[R])
4. At the R> Prompt, type: `SET MULTI OFF` , then press [Enter]
5. At the R> Prompt, type: `SET STATICDB OFF` , then press [Enter]
6. At the R> Prompt, type: `SET USER OwnerName` , then press [Enter]
7. At the R> Prompt type: `CONVERT DBNAME` , then press [Enter]
8. At the R> Prompt, type: `CONNECT DBNAME` , then press [Enter]
9. At the R> Prompt, type: `SET IDQUOTES=` , then press [Enter] (the ` quote character is the reverse single quote, which is next to the "1" and under the tilde "~", on American keyboards)
10. At the R> Prompt, type: `DISCONNECT` , then press [Enter]

11. Perform the "Complete Database Rebuild" instructions below
12. Repeat "Steps 6-11" for all databases

Database Rebuild

In addition to the above conversion steps, the following steps are advised to completely rebuild the database. These steps will be performed in R:BASE 11.

1. Connect to the database and open the R> Prompt.
2. Check the IDQUOTE settings of connected database.

At the R> Prompt, type:

```
SHOW CHARACTERS
```

Notice the last parameter, **IDQUOTES**

If IDQUOTES is set to NULL, you'll need to change NULL to the appropriate character, such as ` (reversed quote) as following:

```
SET IDQUOTES= `
```

3. Unload the entire database. Use the SET USER command below if an OWNER name exists.

At the R> Prompt, type:

```
SET NULL -0-  
SET USER ownername  
OUTPUT filename.ALL  
UNLOAD ALL  
OUTPUT SCREEN
```

This process will create two files for Windows databases (filename.ALL and filename.LOB), and a single file for DOS databases.

4. Disconnect the database.

At the R> Prompt, type:

```
DISCONNECT
```

5. Now rename the database that was just unloaded in Steps 1-3.

Review the list of Databases in the R:BASE Database Explorer, and select the "Rename" option. Enter a new name for the database.

6. Finally, use R:BASE to rebuild the database by running the .ALL file.

At the R> Prompt type:

```
RUN filename.ALL
```

These steps will build a clean database!

If there is a considerable amount of errors with both rebuilding the structure and/or data, please refer to the "Database Repair Routine" within the Database Maintenance PDF document in the R:BASE program directory.

With the database file conversion complete, proceed to the chapter for [Converting Forms, Reports, & Labels](#).

2.2.4 Converting from R:BASE 7.x/9.x (32)/10.x

Before converting your R:BASE 7.x/9.x (32)/10.x database(s) with R:BASE 11, the next step is advised if regular database maintenance (PACK/RELOAD) is not performed, to ensure a proper conversion.

The following will create a copy of the database that is completely rebuilt with the structure, data, and indexes reloaded.

1. Launch R:BASE version of 7.x/9.x (32)/10.x
2. Navigate to your database directory by selecting "Utilities" > "Set Current Working Directory..." from the main Menu bar.
3. Go to the R> Prompt window (third button from the left on the Tool Bar)
4. At the R> Prompt, type: `DISCONNECT` , then press [Enter]
5. At the R> Prompt, type: `RENAME DbName.RB* BckupDB.RB*` , then press [Enter]. "DbName" would be the actual name of your R:BASE database.
6. At the R> Prompt, type: `CONNECT BckupDB` , then press [Enter]
7. At the R> Prompt, type: `RELOAD DbName` , then press [Enter]

The database will now be rebuilt. If database errors occur at any point during the rebuild process, [check the integrity](#) of the database files.

8. At the R> Prompt, type: `CONNECT DbName` , then press [Enter]
9. Now connect back to your database after the rebuild, check your database to ensure the contents are intact.
10. After the database is checked, close R:BASE and continue with the steps below.

At this point, you must have R:BASE 11 installed on your computer. Be sure to review the R:BASE 11 Getting Started Guide regarding the installation and set up for your R:BASE software. The Getting Started Guide is located in the R:BASE program directory.

If there is **no** OWNER

1. Install and launch R:BASE 11
2. Navigate to your database directory using the "Change Current Folder" button (bottom left corner, folder icon with a green arrow, at the Database Explorer)
3. Go to the R> Prompt window (third button from the left on the Tool Bar, or press [Ctrl]+[R])
4. At the R> Prompt, type: `SET MULTI OFF` , then press [Enter]
5. At the R> Prompt, type: `SET STATICDB OFF` , then press [Enter]
6. At the R> Prompt, type: `CONVERT DBNAME` , then press [Enter]
7. At the R> Prompt, type: `CONNECT DBNAME` , then press [Enter]

If there is an Owner

1. Install and launch R:BASE 11
2. Navigate to your database directory using the "Change Current Folder" button (bottom left corner, folder icon with a green arrow, at the Database Explorer)
3. Go to the R> Prompt window (third button from the left on the Tool Bar, or press [Ctrl]+[R])
4. At the R> Prompt, type: `SET MULTI OFF` , then press [Enter]
5. At the R> Prompt, type: `SET STATICDB OFF` , then press [Enter]
6. At the R> Prompt, type: `CONVERT DBNAME IDENTIFIED BY ownername` , then press [Enter]
7. At the R> Prompt, type: `CONNECT DBNAME` , then press [Enter]

Complete Database Rebuild

In addition to the above conversion steps, the following steps are required to completely rebuild the database. These steps will be performed in R:BASE 11.

1. Connect to the database and open the R> Prompt.
2. Check the IDQUOTE settings of connected database.

At the R> Prompt, type:

```
SHOW CHARACTERS
```

Notice the last parameter, **IDQUOTES**

If IDQUOTES is set to NULL, you'll need to change NULL to the appropriate character, such as ` (reversed quote) as following:

```
SET IDQUOTES= `
```

3. Unload the entire database. Use the SET USER command below if an OWNER name exists.

At the R> Prompt, type:

```
SET NULL -0-  
SET USER ownername  
OUTPUT filename.ALL  
UNLOAD ALL  
OUTPUT SCREEN
```

This process will create two files (filename.ALL and filename.LOB)

4. Disconnect the database.

At the R> Prompt, type:

```
DISCONNECT
```

5. Now rename the database that was just unloaded in Steps 1-3.

Review the list of Databases in the R:BASE Database Explorer, and select the "Rename" option. Enter a new name for the database.

6. Finally, use R:BASE to rebuild the database by running the .ALL file.

At the R> Prompt type:

```
RUN filename.ALL
```

These steps will build a clean database!

With the database file conversion complete, proceed to the chapter for converting forms, reports, and labels. Choose either of the below:

[R:BASE 7.x/Turbo V-8 Forms, Reports, & Labels](#)
[R:BASE eXtreme 9.x \(32/64\) Forms, Reports, & Labels](#)

2.2.5 Converting from R:BASE Turbo V-8

Before attempting to convert your R:BASE Turbo V-8 database(s) with R:BASE 11, the next step is advised if regular database maintenance (PACK/RELOAD) is not performed, to ensure a proper conversion.

The following will create a copy of the database that is completely rebuilt with the structure, data, and indexes reloaded.

1. Launch R:BASE Turbo V-8
2. Navigate to your database directory by selecting "Utilities" > "Set Current Working Directory..." from the main Menu bar.
3. Go to the R> Prompt window (third button from the left on the Tool Bar)
4. At the R> Prompt, type: `DISCONNECT` , then press [Enter]
5. At the R> Prompt, type: `RENAME DbName.RX* BckupDB.RX*` , then press [Enter]. "DbName" would be the actual name of your R:BASE database.
6. At the R> Prompt, type: `CONNECT BckupDB` , then press [Enter]
7. At the R> Prompt, type: `RELOAD DbName` , then press [Enter]

The database will now be rebuilt. If database errors occur at any point during the rebuild process, [check the integrity](#) of the database files.

8. At the R> Prompt, type: `CONNECT DbName` , then press [Enter]
9. Now connect back to your database after the rebuild, check your database to ensure the contents are intact.
10. After the database is checked, close R:BASE and continue with the steps below.

At this point, you must have R:BASE 11 installed on your computer. Be sure to review the R:BASE 11 Getting Started Guide regarding the installation and set up for your R:BASE software. The Getting Started Guide is located in the R:BASE program directory.

If there is no OWNER

1. Install and launch R:BASE 11
2. Navigate to your database directory using the "Change Current Folder" button (bottom left corner, folder icon with a green arrow, at the Database Explorer)
3. Go to the R> Prompt window (third button from the left on the Tool Bar, or press [Ctrl]+[R])
4. At the R> Prompt, type: `SET MULTI OFF` , then press [Enter]
5. At the R> Prompt, type: `SET STATICDB OFF` , then press [Enter]
6. At the R> Prompt, type: `TURBO DBNAME V9` , then press [Enter]
7. At the R> Prompt, type: `CONNECT DBNAME` , then press [Enter]

If there is an Owner

1. Install and launch R:BASE 11
2. Navigate to your database directory using the "Change Current Folder" button (bottom left corner, folder icon with a green arrow, at the Database Explorer)
3. Go to the R> Prompt window (third button from the left on the Tool Bar, or press [Ctrl]+[R])
4. At the R> Prompt, type: `SET MULTI OFF` , then press [Enter]
5. At the R> Prompt, type: `SET STATICDB OFF` , then press [Enter]
6. At the R> Prompt, type: `TURBO DBNAME V9 IDENTIFIED BY ownername` , then press [Enter]
7. At the R> Prompt, type: `CONNECT DBNAME` , then press [Enter]

Complete Database Rebuild

In addition to the above conversion steps, the following steps are required to completely rebuild the database. These steps will be performed in R:BASE 11.

1. Connect to the database and open the R> Prompt.
2. Check the IDQUOTE settings of connected database.

At the R> Prompt, type:

```
SHOW CHARACTERS
```

Notice the last parameter, **IDQUOTES**

If IDQUOTES is set to NULL, you'll need to change NULL to the appropriate character, such as ` (reversed quote) as following:

```
SET IDQUOTES= `
```

3. Unload the entire database. Use the SET USER command below if an OWNER name exists.

At the R> Prompt, type:

```
SET NULL -0-  
SET USER ownername  
OUTPUT filename.ALL  
UNLOAD ALL  
OUTPUT SCREEN
```

This process will create two files (filename.ALL and filename.LOB)

4. Disconnect the database.

At the R> Prompt, type:

```
DISCONNECT
```

5. Now rename the database that was just unloaded in Steps 1-3.

Review the list of Databases in the R:BASE Database Explorer, and select the "Rename" option. Enter a new name for the database.

6. Finally, use R:BASE to rebuild the database by running the .ALL file.

At the R> Prompt type:

```
RUN filename.ALL
```

These steps will build a clean database!

With the database file conversion complete, proceed to the chapter for converting [R:BASE 7.x/Turbo V-8 Forms, Reports, & Labels](#).

2.2.6 Converting from R:BASE eXtreme 9.x (64)

Before attempting to convert your R:BASE eXtreme 9.x (64) database(s) with R:BASE 11, the next step is advised if regular database maintenance (PACK/RELOAD) is not performed, to ensure a proper conversion.

The following will create a copy of the database that is completely rebuilt with the structure, data, and indexes reloaded.

1. Launch R:BASE eXtreme 9.5 (64)
2. Navigate to your database directory by selecting "Utilities" > "Set Current Working Directory..." from the main Menu bar.
3. Go to the R> Prompt window (third button from the left on the Tool Bar)
4. At the R> Prompt, type: `DISCONNECT` , then press [Enter]
5. At the R> Prompt, type: `RENAME DbName.RX* BckupDB.RX*` , then press [Enter]. "DbName" would be the actual name of your R:BASE database.
6. At the R> Prompt, type: `CONNECT BckupDB` , then press [Enter]

7. At the R> Prompt, type: `RELOAD dbName` , then press [Enter]

The database will now be rebuilt. If database errors occur at any point during the rebuild process, [check the integrity](#) of the database files.

8. At the R> Prompt, type: `CONNECT dbName` , then press [Enter]
9. Now connect back to your database after the rebuild, check your database to ensure the contents are intact.
10. After the database is checked, close R:BASE and continue with the steps below.

At this point, you must have R:BASE 11 installed on your computer. Be sure to review the R:BASE 11 Getting Started Guide regarding the installation and set up for your R:BASE software. The Getting Started Guide is located in the R:BASE program directory.

If there is **no** OWNER

1. Install and launch R:BASE 11
2. Navigate to your database directory using the "Change Current Folder" button (bottom left corner, folder icon with a green arrow, at the Database Explorer)
3. Go to the R> Prompt window (third button from the left on the Tool Bar, or press [Ctrl]+[R])
4. At the R> Prompt, type: `SET MULTI OFF` , then press [Enter]
5. At the R> Prompt, type: `SET STATICDB OFF` , then press [Enter]
6. At the R> Prompt, type: `MIGRATE DBNAME` , then press [Enter]
7. At the R> Prompt, type: `CONNECT DBNAME` , then press [Enter]

If there is an Owner

1. Install and launch R:BASE 11
2. Navigate to your database directory using the "Change Current Folder" button (bottom left corner, folder icon with a green arrow, at the Database Explorer)
3. Go to the R> Prompt window (third button from the left on the Tool Bar, or press [Ctrl]+[R])
4. At the R> Prompt, type: `SET MULTI OFF` , then press [Enter]
5. At the R> Prompt, type: `SET STATICDB OFF` , then press [Enter]
6. At the R> Prompt, type: `SET USER OwnerName` , then press [Enter]
7. At the R> Prompt, type: `MIGRATE DBNAME` , then press [Enter]
8. At the R> Prompt, type: `CONNECT DBNAME` , then press [Enter]

Complete Database Rebuild

In addition to the above conversion steps, the following steps are required to completely rebuild the database. These steps will be performed in R:BASE 11.

1. Connect to the database and open the R> Prompt.
2. Check the IDQUOTE settings of connected database.

At the R> Prompt, type:

```
SHOW CHARACTERS
```

Notice the last parameter, **IDQUOTES**

If IDQUOTES is set to NULL, you'll need to change NULL to the appropriate character, such as ` (reversed quote) as following:

```
SET IDQUOTES= `
```

3. Unload the entire database. Use the SET USER command below if an OWNER name exists.

At the R> Prompt, type:

```
SET NULL -0-  
SET USER ownername  
OUTPUT filename.ALL  
UNLOAD ALL  
OUTPUT SCREEN
```

This process will create two files (filename.ALL and filename.LOB)

4. Disconnect the database.

At the R> Prompt, type:

```
DISCONNECT
```

5. Now rename the database that was just unloaded in Steps 1-3.

Review the list of Databases in the R:BASE Database Explorer, and select the "Rename" option. Enter a new name for the database.

6. Finally, use R:BASE to rebuild the database by running the .ALL file.

At the R> Prompt type:

```
RUN filename.ALL
```

These steps will build a clean database!

With the database file conversion complete, proceed to the chapter for converting [R:BASE eXtreme 9.x \(32/64\) Forms, Reports, & Labels](#).

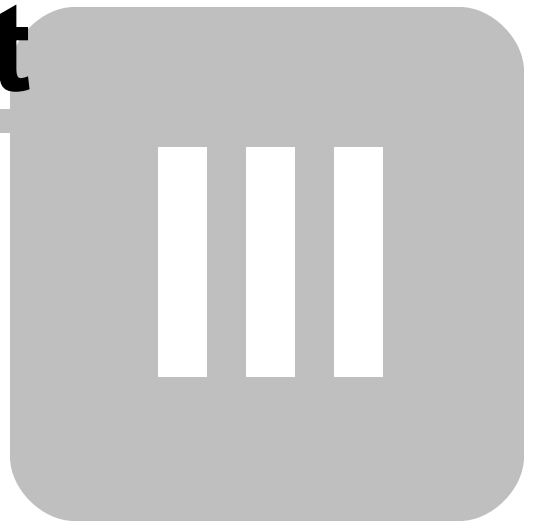
2.2.7 Converting from R:BASE X/X.5 Enterprise (10.x)

If you are upgrading to R:BASE 11 from R:BASE X/X.5 Enterprise (Version 10.x), you are only required to connect to the database in order to complete the conversion.

At this point in R:BASE use, R:BASE 11 and R:BASE X/X.5 Enterprise can connect to the same database. However, once any Version 11 specific features are implemented (such as static variables or database comments), R:BASE 10.x Enterprise versions can no longer connect to the database. A minor version number is stored within file 1 of the database files. R:BASE 11 will automatically alter the value if any new features of version 11 are implemented. Also, any new databases created in R:BASE 11 will start with the elevated minor version, and R:BASE 10.x Enterprise versions will not be able to connect to the database. Also, if R:BASE 11 is used to alter and save forms, reports, and labels, the objects will no longer be recognized in the prior R:BASE versions.

Please review the "What's New in R:BASE 11 for Windows" PDF document for the list of other new features.

Part



3 Converting Forms, Reports, & Labels

At this point in the conversion process, four database files should exist. When connected to a database, tables will be listed in the R:BASE Database Explorer window. You should also be able to browse the table data with no problems.

Now, steps can be taken to convert the existing forms, reports, and labels to R:BASE 11 for Windows. Whether the database originates from DOS or Windows all forms, reports, and labels can easily be converted to R:BASE 11.

Proceed to the next chapter for converting [6.5++ and Lower Version Forms, Reports, & Labels](#).

3.1 R:BASE 6.5++ and Lower Forms, Reports, & Labels

At this point in the database conversion process, R:BASE 11 for Windows must be installed on the computer to complete the instructions below.

1. Launch R:BASE 11 for Windows
2. Connect to the database by selecting "Database" > "Connect to Database..." from the main menu bar.

You will be prompted to locate your database files where they are stored on your computer. Once connected to your database, you will see the database name listed across the top of the R:BASE 11 program caption, in brackets.

Throughout the following instructions, there will be areas of the R:BASE interface that will require you to know what is being referred to (e.g. Database Explorer, Navigator, Tool Bar, menu bar, etc.). For a quick description as to what these objects are, please read the "Introduction" page of the R:BASE 11 Tutorial before continuing. To open the **Introduction** page, select "Help" > "R:BASE Tutorial" from the R:BASE main menu bar.

Once you have completed reading the Introduction page, close the window and go to the next section, **Converting Forms**.

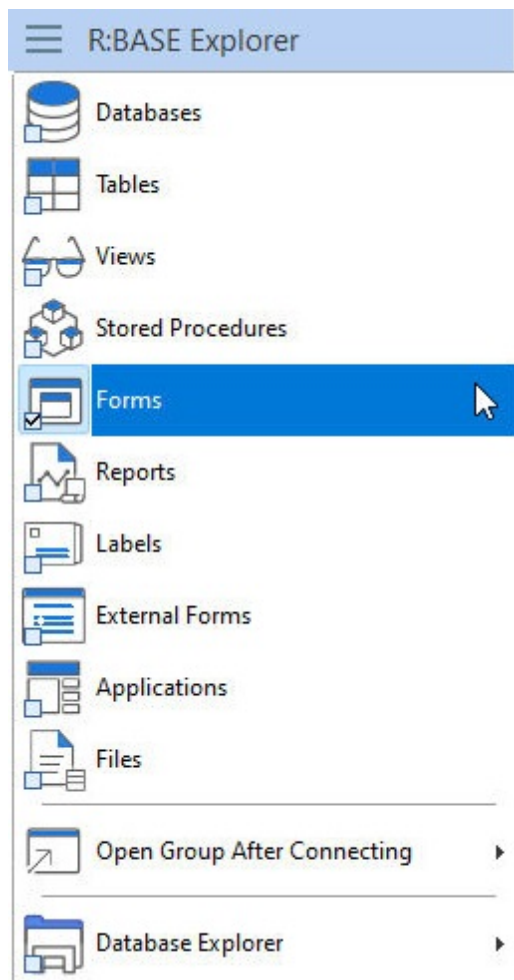
3.1.1 Converting Forms

You should now be looking at the R:BASE window with the Database Explorer displayed. To confirm, you will see the main R:BASE caption read the "R:BASE 11", then your database name listed in square brackets, then "R:BASE Database Explorer" listed in square brackets.

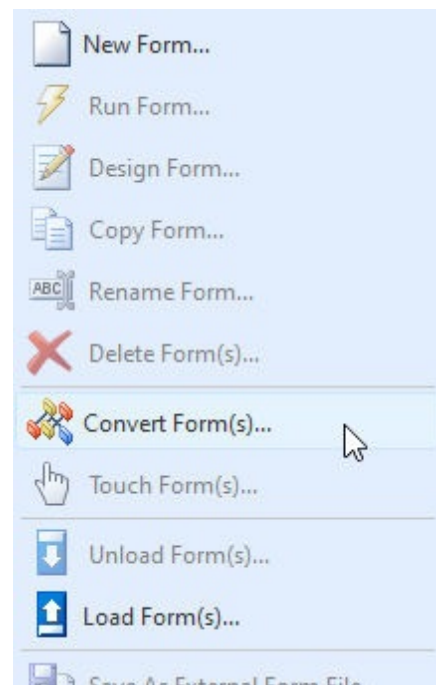
- If you do not see your database listed in brackets, then connect to it now by selecting "Database" > "Connect to Database..." from the main menu bar. You will be prompted to locate your database files where they are stored on your computer. Once connected to your database, you will see the database name listed in the R:BASE 11 program caption, in brackets.
- If you do not see "R:BASE Database Explorer" listed in square brackets, then select the "Database Explorer" button on the main Tool Bar. It is the first button on the left and the hint displayed will read "Database Explorer".

At the Database Explorer window, you will see the "Navigator" listed on the left side of the window containing various menu options of R:BASE database modules (e.g. Tables, Views, Forms, etc.)

1. Click on the Database Explorer button and choose "Forms" from the drop down menu (figure 1), then select the "Convert Form(s)..." menu option (figure 2).

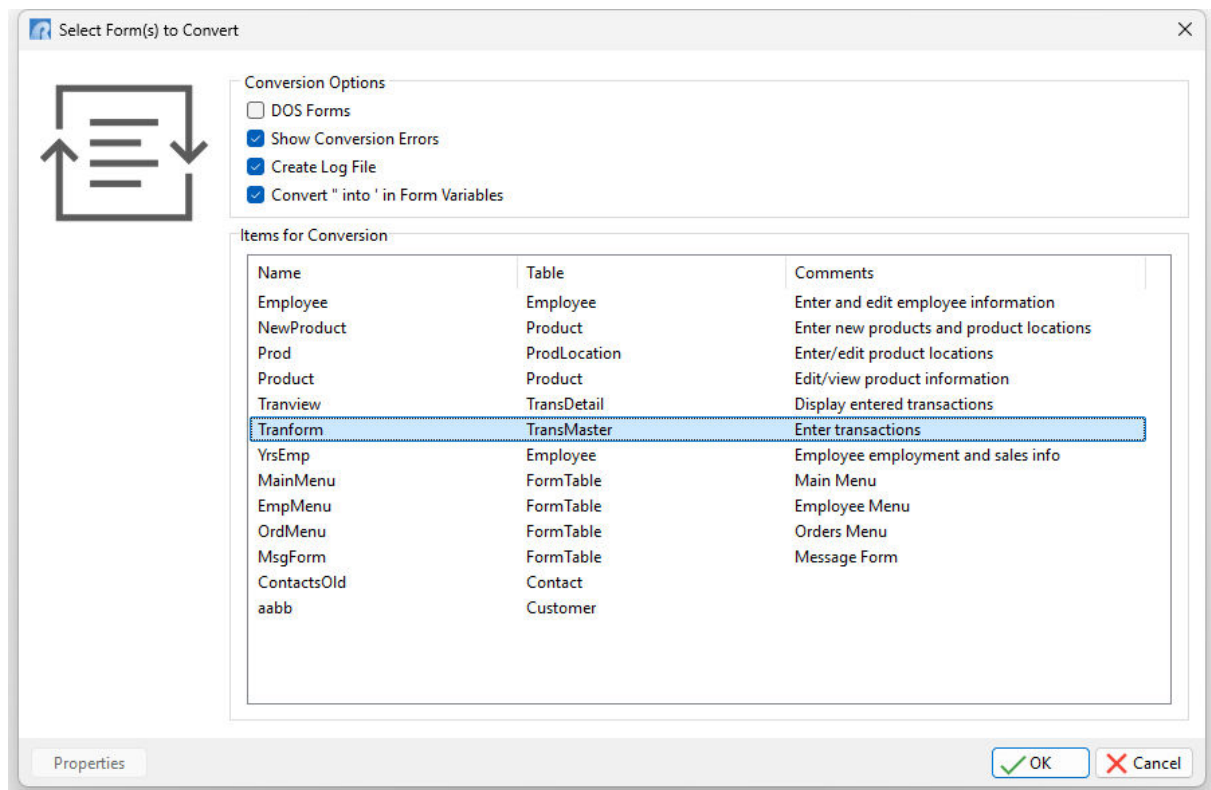


(figure 1)



(figure 2)

Upon selecting this option, a dialog will appear that may contain a list of the available forms to convert (figure 3). The forms listed are Windows-based forms that were created in a previous Windows version of R:BASE.

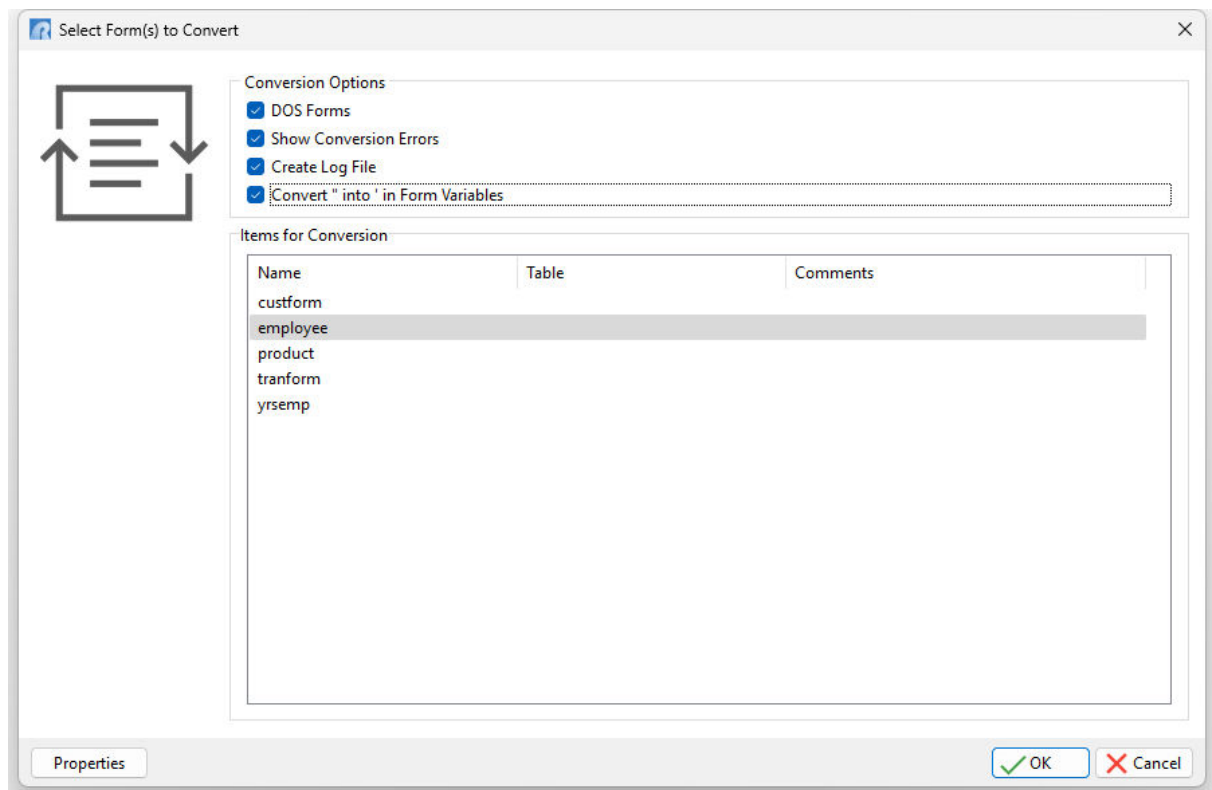


(figure 3)

- Individual forms can be selected and converted. Multiple forms can be converted by holding the [Ctrl] key and selecting individual form names. Or pressing the [Ctrl]+[A] keys will select all listed forms. Press the "OK" button to convert the selected forms. Conversion options include the ability to show errors, create a log file, and convert " into ' for form variables as check boxes. As the earlier steps advised to change the QUOTES setting from double quotes to an apostrophe, the last option will make legacy migrations much easier.

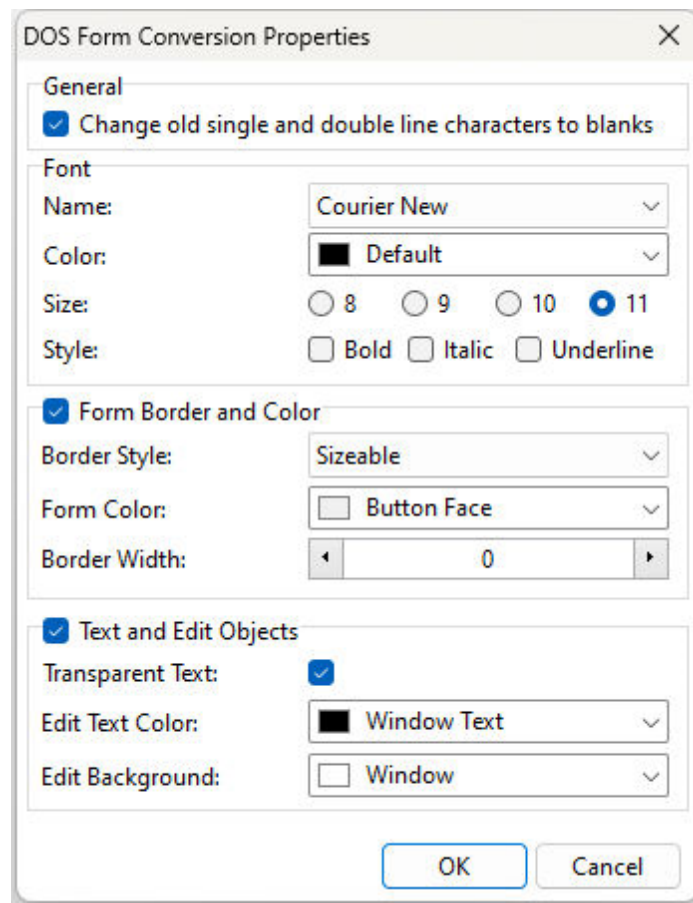
The form(s) will convert and will be displayed in the Database Explorer.

- The database may contain DOS forms. Select the "DOS Forms" check box. The dialog may now contain a list of the available DOS forms to convert (figure 4). Any forms listed are DOS-based forms that were created in a DOS version of R:BASE.



(figure 4)

4. The "Properties" button provides a form conversion option to change single and double line characters to blank spaces, alter the Font (name, color, size, and style), Form Border (style and width), Form Background Color, Text (transparent), and Edit Objects (text color and background) (figure 5).



(figure 5)

The following is an example of a form that was converted with the single and double line characters setting unchecked. This display may be beneficial in the conversion process to see how a form may have been divided into sections.

Entry Date: employee:e

Employee Information

ID Number: emp.

Hire Date: employee:h.

First Name: employee:e

Last Name: employee:emplnam

Title: employee:emptitle

Address: employee:empaddress

City: employee:empcity State: em Zip: employee:e

Home Phone: employee:emp

Office Extension: emp.

5. Like with the Windows-based forms, the DOS forms can be individually selected and converted. Multiple forms can be converted by holding the [Ctrl] key and selecting individual form names. Or pressing the [Ctrl]+[A] keys will select all listed forms. Press the "OK" button to convert the selected forms.

At this point, you can now enter the Form Designer to view and make changes to the forms. There are [Issues and Suggestions](#) to review for the converted forms. However, since many of the instructions below refer to specific control/object names and require a basic understanding of the Form Designer, it is recommended that you perform the conversions of the remaining modules (e.g. reports, labels), and then complete the R:BASE Tutorial before continuing.

From the Main Menu, choose "Help" > "R:BASE Tutorial". This tutorial covers step-by-step instructions on building a complete database and application from scratch providing the R:BASE relational logic as it adheres to Dr. Codd's relational model. This is where one would learn how to quickly adapt to the R:BASE interface!

3.1.1.1 Issues and Suggestions

NOTE: From inside the Form Designer, you can launch the Form Designer help file by pressing the [Shift] +[F1] hot keys.

- **Add an Enhanced DB Navigator Control**

The Enhanced DB Navigator is a form object that allows moving between, saving, and deleting rows in a table. To add the control, locate the button within the Database Controls (boxed in red below).



Mouse click on the button, then mouse click on the form where the object is to be placed. The properties dialog will appear to alter the settings for the control. Click OK to use the default settings.

- **Form Lookup Variables**

The "Lookup Variable" is now truly a "Lookup" variable, and is not to be changed. All expressions, if based on lookup variables, will be evaluated and refreshed INSTANTLY. Prior to version 7.x, this was not the case. Previously, the form variables must be refreshed with the RECALC VARIABLES command each time variables were to be refreshed. To achieve the same results in R:BASE 11, then do NOT use form defined variables, but rather EEPs to declare variables with the SET VARIABLE command, which can execute when needed and with the flexibility to change the variables accordingly.

- **Form Variables**

Form variables that perform calculations or combine a number of values should be enclosed in parenthesis. If error messages are displayed for variables performing calculations, like the one below,

```
2 : CURRENCY      vTotal = price * quantity
```

add parenthesis as follows:

```
2 : CURRENCY      vTotal = (price * quantity)
```

- **Multi-Page Form with Push Buttons**

When converting a multiple page form, the different "pages" will be placed on a Tab Control, with each page placed upon a separate "tab". Any Push Buttons on the form that use a predefined "Next Page" or "Previous Page" EEP will appear on the new form, however will have no action assigned to them. This is because there is technically no longer a "page" to move to with regards to the behavior of a Tab Control.

- **Lookup Variable Tip**

Within the R:BASE 11 Form Designer there is an option to run a series of commands (or Action) for a form before the Form Designer is launched. In the Form Designer, choose "Layout" from the Menu bar, then select "On Before Design Action...".

This will run a series of commands, which can include the creation of variables that Lookup Variables are based upon, to avoid annoying error messages when the form is opened in the designer. An example of a lookup variable based upon another variable would be...

```
1 : TEXT      vCompany = Company IN Customer WHERE CustID =
.vCustID
```

The error message appears because the variable vCustID is not defined. Adding the following SET VAR command to the "On Before Design Action..." will alleviate this error message:

```
SET VAR vCustID INTEGER
```

The "On Before Design Action..." is listed under "Layout" on the main menu bar.

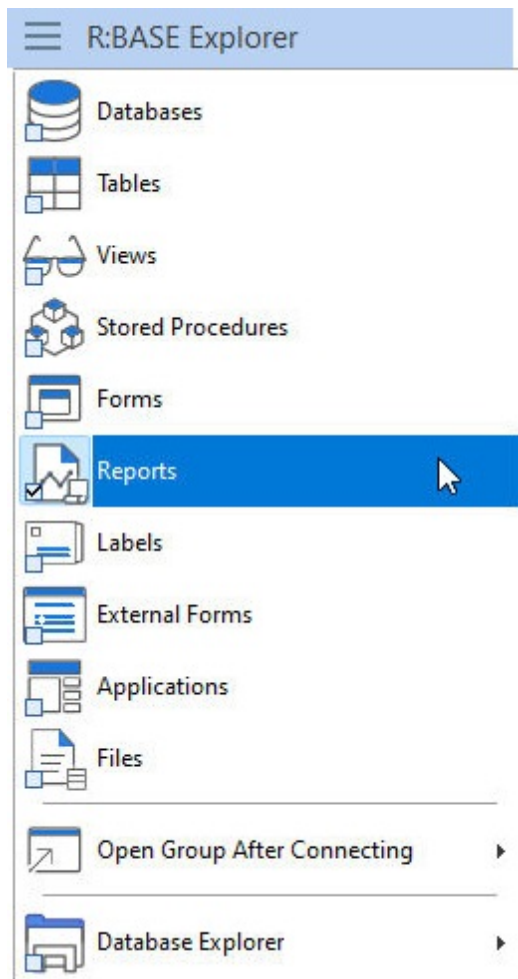
3.1.2 Converting Reports

You should now be looking at the R:BASE window with the Database Explorer displayed. To confirm, you will see the main R:BASE caption read the "R:BASE 11", then your database name listed in square brackets, then "R:BASE Database Explorer" listed in square brackets.

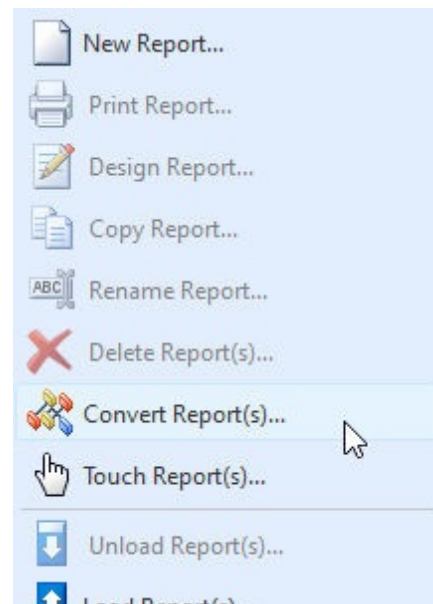
- If you do not see your database listed in brackets, then connect to it now by selecting "Database" > "Connect to Database..." from the main menu bar. You will be prompted to locate your database files where they are stored on your computer. Once connected to your database, you will see the database name listed in the R:BASE 11 program caption, in brackets.
- If you do not see "R:BASE Database Explorer" listed in square brackets, then select the "Database Explorer" button on the main Tool Bar. It is the first button on the left and the hint displayed will read "Database Explorer".

At the Database Explorer window, you will see the "Navigator" listed on the left side of the window containing various menu options of R:BASE database modules (e.g. Tables, Views, Forms, etc.)

1. Click on the Database Explorer button and choose "Reports" from the drop down menu (figure 1), then select the "Convert Report(s)..." menu option (figure 2).

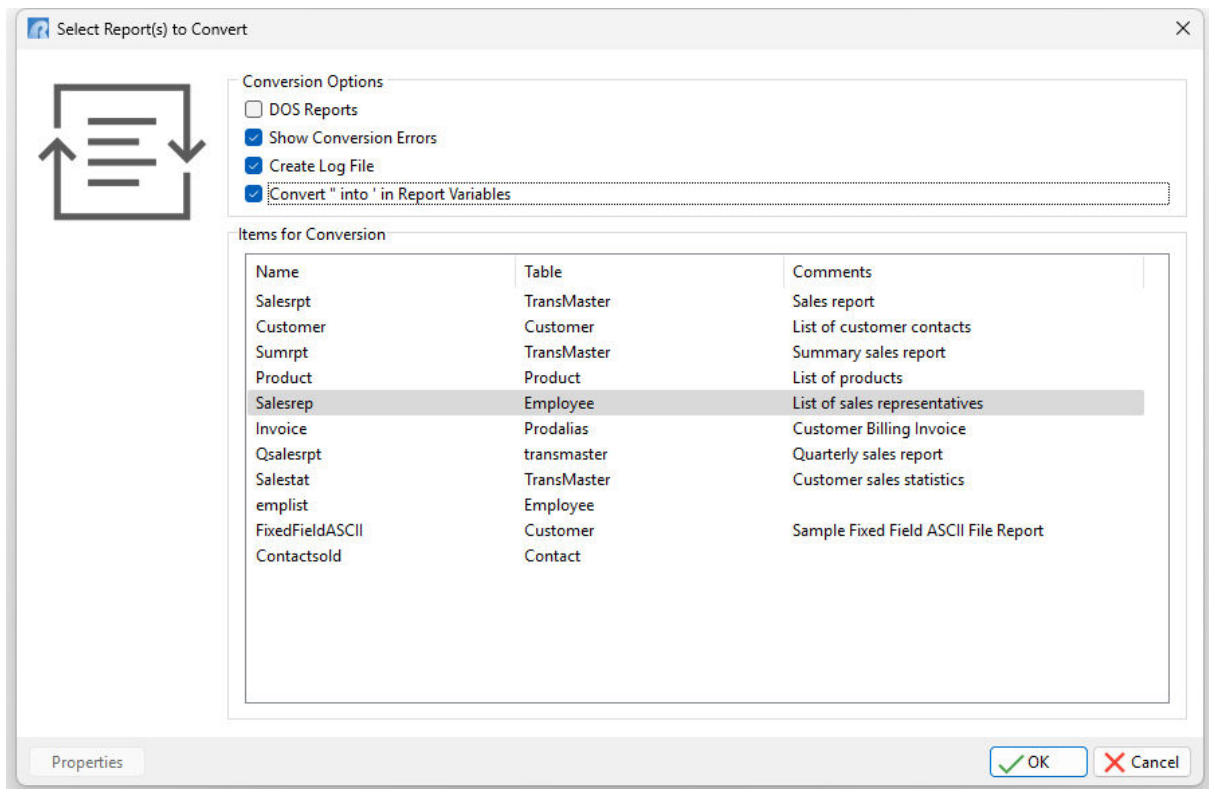


(figure 1)



(figure 2)

Upon selecting this option, a dialog will appear that may contain a list of the available reports to convert (figure 3). The reports listed are Windows-based reports that were created in a previous Windows version of R:BASE.

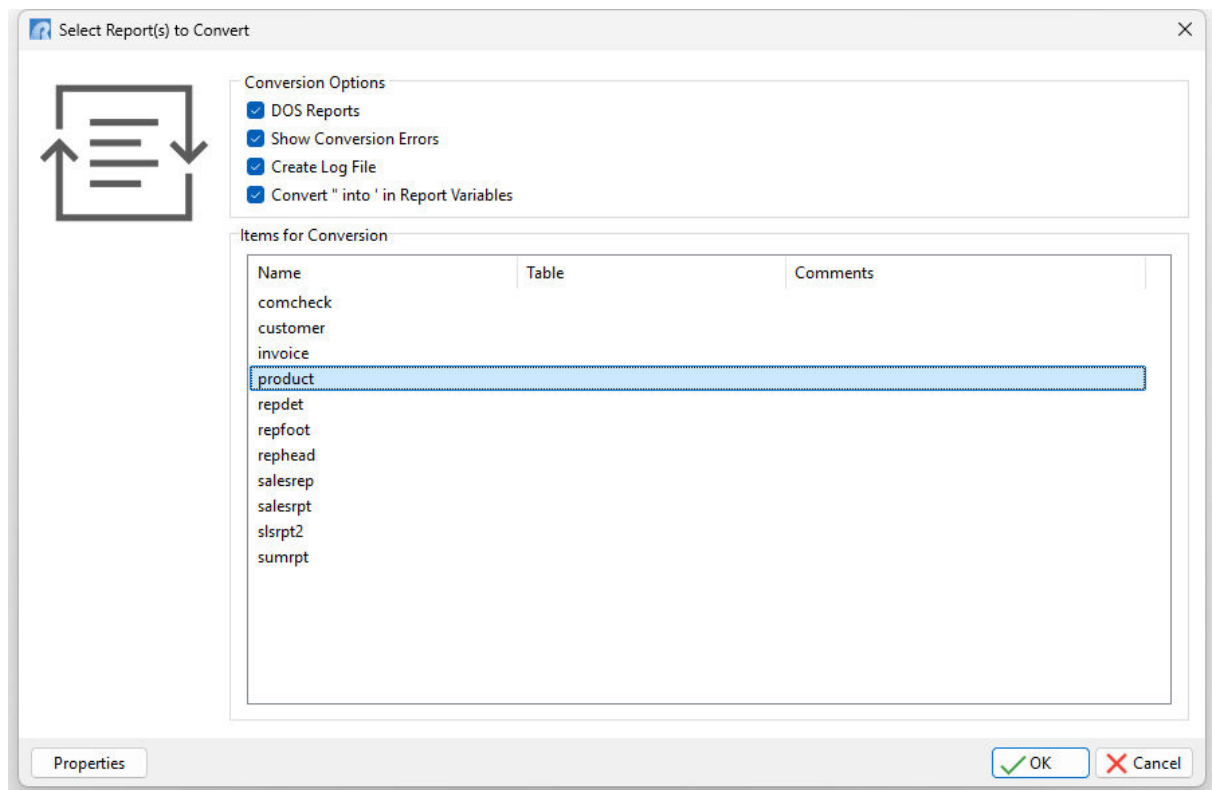


(figure 3)

- Individual reports can be selected and converted. Multiple reports can be converted by holding the [Ctrl] key and selecting individual report names. Or pressing the [Ctrl]+[A] keys will select all listed reports. Press the "OK" button to convert the selected reports. Conversion options include the ability to show errors, create a log file, and convert " into ' for report variables as check boxes. As the earlier steps advised to change the QUOTES setting from double quotes to an apostrophe, the last option will make legacy migrations much easier.

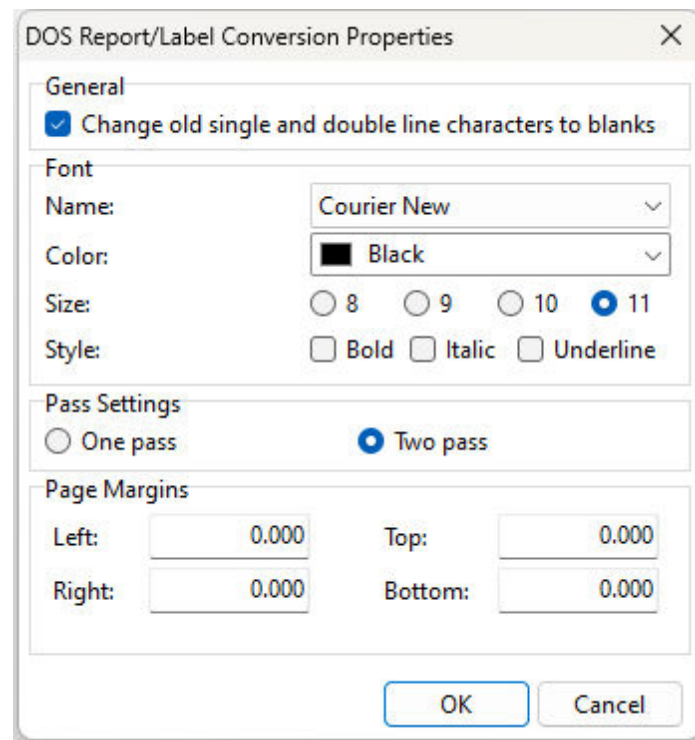
The report(s) will convert and will be displayed in the Database Explorer.

- The database may contain DOS reports. Select the "DOS Reports" check box. The dialog may now contain a list of the available DOS reports to convert (figure 4). Any reports listed are DOS-based reports that were created in a DOS version of R:BASE.



(figure 4)

4. The "Properties" button provides a form conversion option to change single and double line characters to blank spaces, alter the Font (name, color, size, and style), Form Border (style and width), Form Background Color, Text (transparent), and Edit Objects (text color and background) (figure 5).



(figure 5)

Maintaining single and double lines in the conversion process may help see how a report was divided into sections. The pass setting processes system variables once or twice are the report is generated. For DOS reports, the default is one pass. For Windows reports, the default pass setting is two. Margins in DOS reports begin at zero. The margin can be changed for any custom reports.

5. Like with the Windows-based reports, the DOS reports can be individually selected and converted. Multiple reports can be converted by holding the [Ctrl] key and selecting individual report names. Or pressing the [Ctrl]+[A] keys will select all listed reports. Press the "OK" button to convert the selected reports.

The report(s) will convert and will be displayed within the Database Explorer.

At this point, you can now enter the Report Designer to view and make changes to the reports. There are [Issues and Suggestions](#) to review for the converted reports. However, since many of the instructions below refer to specific control/object names and require a basic understanding of the Report Designer, it is recommended that you perform the conversions of the remaining modules (e.g. forms, labels), and then complete the R:BASE Tutorial before continuing.

From the Main Menu, choose "Help" > "R:BASE Tutorial". This tutorial covers step-by-step instructions on building a complete database and application from scratch providing the R:BASE relational logic as it adhere's to Dr. Codd's relational model. This is where one would learn how to quickly adapt to the R:BASE interface!

3.1.2.1 Issues and Suggestions

NOTE: From inside the Report Designer, you can launch the Report Designer help file by pressing the [Shift]+[F1] hot keys.

- **Sub-Totals and Totals**

To calculate sub-totals and totals of table columns, Aggregate Variables have always been used to perform this job. Now, the R:BASE 11 for Windows Report Designer has a built-in object called DB Calc, which performs these same functions on table columns without creating the extra overhead of

variables. You can replace your Variable Label objects with DB Calc objects in instances where the Variables Object results will not be used further down in the report.

- **#DATE, #TIME, #PAGE, Document, and Page Information**

The R:BASE 11 for Windows Report Designer now has a System Variable control, which allows an object to be placed for the current DATE, TIME, Document Name, Print DATE and TIME, or one of many Page Numbering options. All Variable Label objects whose expression value consists of a system variable (#DATE, #TIME) should be deleted and replaced with a System Variable control. Any Variable Label objects whose expression value consists of the #PAGE system variable, you must delete the variable object and replace that object with a System Variable control.

- **Report Variables**

Report variables that perform calculations or combine a number of values should be enclosed in parenthesis. If error messages are displayed for variables performing calculations, like the one below,

```
2 : CURRENCY      vTotal = price * quantity
```

add parenthesis as follows:

```
2 : CURRENCY      vTotal = (price * quantity)
```

- **Report Totals**

To be sure report totals are calculating correctly on all NULL and not NULL records, make sure the **ZERO** setting is ON. Check the setting under "Settings" > "Configuration Settings".

- **Printer Control Codes**

If DOS or machine printer control codes are used in R:BASE Report variables, they can no longer be stored in the Report Variables. A control code would appear in the variable list with an expression as follows: <27 40 115 49 83>

If the control code were used to alter the text on the report (e.g. bold, underline, etc.), the control code must be deleted. There are many new features in the Windows interface that replace control codes for altering report text objects. The R:BASE 11 for Windows version allows you to perform these font changes must easier.

If control codes listed in the report control hardware for paper feeds or to open a cash drawer, then document down the exact code and review the "PCC Label" Control in the Report Designer. This control will support the same functionality used previously.

- **Picture Format**

For a report object, [] may be displayed in the report preview. In R:BASE for DOS and R:BASE for Windows versions prior to 7.x, a "Picture Format" option was available. The option allowed the justification of the value and several formatting options based upon the data type of the object. For example, to add the justification a [>], or [<] was used to right/left justify.

In R:BASE 11, this feature is called "Display Format", which still offers several formatting options for report objects based upon the data type of the object. While the formatting option is still supported, the justification option is no longer supported in the previous context, within the "Display Format" settings. To right/left/center justify text of an object, use the "Format" tool bar which supports many text formatting options.

If you see [] listed in your report preview for a report object, you must right click on the object, select "Display Format", and remove the [] characters.

- **Page Setup**

For DOS to Windows conversions, the default page size setting is set to it's largest values in order to make sure no objects are lost in the conversion process. You must make sure you alter your page settings back to your desired values in order for your reports to print correctly.

To alter the page size settings, follow the below instructions:

1. Select "File" > "Page Setup..." from the main menu bar.
2. Select the "Paper Size" tab
3. For Portrait orientation reports, the Width should be set to "8.5".
4. For Landscape orientation reports, the Height should be set to "8.5".

From this dialog, you can also adjust your report margins from the "Margins" tab.

After making any changes, make sure that all of your report objects are located within the defined page size settings and margins. Otherwise, R:BASE will display a warning that "Controls beyond the new page width will be deleted." In this case, select "Cancel" to avoid losing any controls. Then move your controls within your desired page settings and then make your page size setting changes again.

- **Page Footer Content Moved to the Report Footer**

The Page Footer band behaves differently in 7.x and higher versions, where the band is intentionally positioned across the footer of the page. In converted reports, it may be required to move the Page Footer content to the Report Footer. To do so, select one item in the Page Footer, then press [Ctrl+A] which selects all of the item in the Page Footer. Then, using the mouse, click and drag the selected content from the Page Footer to the Report Footer.

- **Report Preview Tip**

The "Report Preview" tab will display a preview of your report as it may appear when you send the report to the printer or screen. Before selecting the tab, you must make sure that there are no errors listed in your Report variables, otherwise, you will receive an error message for every error listed multiplied by the number of records displayed in the Report Preview window.

What you can do to limit the number of rows displayed in the Report Preview window, thus limiting the number error messages. To do so, alter the Report Default settings for how many rows are displayed when previewing a report.

1. Select "Settings" > "Report/Label Designer" > "Default Settings..." from the main menu bar.
2. Alter the value for "Row Count for Preview:" to a smaller value other than 100; perhaps to 10.

- **Lookup Variable Tip**

Within the R:BASE 11 Report Designer there is an option to run a series of commands (or Action) for a report before the Report Designer is launched. In the Report Designer, choose "Report" from the Menu bar, the select "Actions" > "On Before Design...".

This will run a series of commands, which can include the creation of variables that Lookup Variables are based upon, to avoid annoying error messages when the form is opened in the designer. An example of a lookup variable based upon another variable would be...

```
1 : TEXT      vCompany = Company IN Customer WHERE CustID =
.vCustID
```

The error message appears because the variable vCustID is not defined. Adding the following SET VAR command to the "On Before Design Action..." will alleviate this error message:

```
SET VAR vCustID INTEGER
```

The "On Before Design..." action is located under "Report" > "Actions" on the main menu bar.

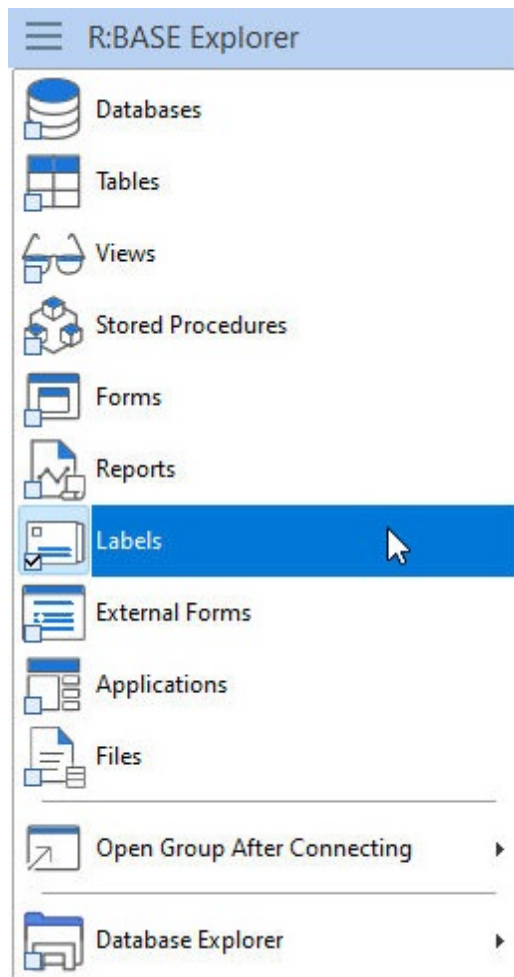
3.1.3 Converting Labels

You should now be looking at the R:BASE window with the Database Explorer displayed. To confirm, you will see the main R:BASE caption read the "R:BASE 11", then your database name listed in square brackets, then "R:BASE Database Explorer" listed in square brackets.

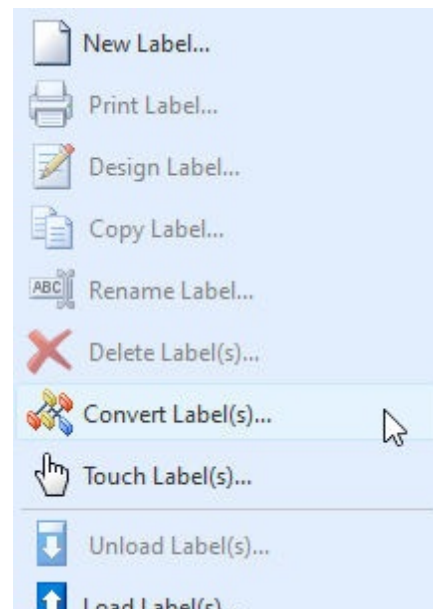
- If you do not see your database listed in brackets, then connect to it now by selecting "Database" > "Connect to Database..." from the main menu bar. You will be prompted to locate your database files where they are stored on your computer. Once connected to your database, you will see the database name listed in the R:BASE 11 program caption, in brackets.
- If you do not see "R:BASE Database Explorer" listed in square brackets, then select the "Database Explorer" button on the main Tool Bar. It is the first button on the left and the hint displayed will read "Database Explorer".

At the Database Explorer window, you will see the "Navigator" listed on the left side of the window containing various menu options of R:BASE database modules (e.g. Tables, Views, Forms, etc.)

1. Click on the Database Explorer button and choose "Labels" (figure 1), then select the "Convert Label(s)..." sub menu option (figure 2).

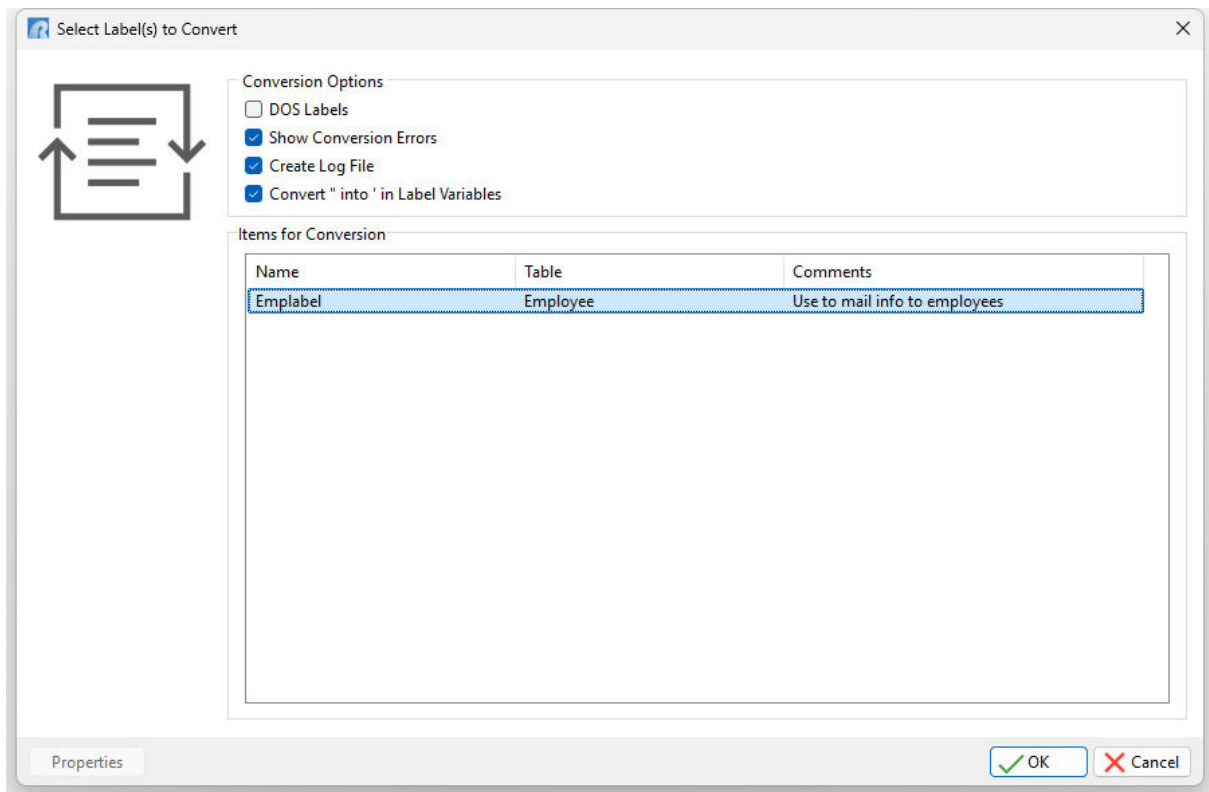


(figure 1)



(figure 2)

Upon selecting this option, a dialog will appear that may contain a list of the available labels to convert (figure 3). The labels listed are Windows-based labels that were created in a previous Windows version of R:BASE.



(figure 3)

2. Individual labels can be selected and converted. Multiple labels can be converted by holding the [Ctrl] key and selecting individual label names. Or pressing the [Ctrl]+[A] keys will select all listed labels. Press the "OK" button to convert the selected labels. Conversion options include the ability to show errors, create a log file, and convert " into ' for label variables as check boxes. As the earlier steps advised to change the QUOTES setting from double quotes to an apostrophe, the last option will make legacy migrations much easier.

The label(s) will convert and will be displayed in the Database Explorer.

3. The database may contain DOS labels. Select the "DOS Labels" check box. The dialog may now contain a list of the available DOS labels to convert. Any labels listed are DOS-based labels that were created in a DOS version of R:BASE. The "Properties" button provides label conversion options to change single and double line characters to blanks, a pass setting, and margins (figure 5).

Maintaining single and double lines in the conversion process may help see how a label was divided into sections. The pass setting processes system variables once or twice are the label is generated. For DOS labels, the default is one pass. For Windows labels, the default pass setting is two. Margins in DOS labels begin at zero. The margin can be changed for any custom reports.

4. Like with the Windows-based labels, the DOS labels can be individually selected and converted. Multiple labels can be converted by holding the [Ctrl] key and selecting individual label names. Or pressing the [Ctrl]+[A] keys will select all listed labels. Press the "OK" button to convert the selected labels.

The label(s) will convert and will be displayed within the Database Explorer.

At this point, you can now enter the Label Designer to view and make changes to the labels. There are [Issues and Suggestions](#) to review for the converted labels. However, since many of the instructions below refer to specific control/object names and require a basic understanding of the Label Designer, it is recommended that you perform the conversions of the remaining modules (e.g. forms, reports), and then complete the R:BASE Tutorial before continuing.

From the Main Menu, choose "Help" > "R:BASE Tutorial". This tutorial covers step-by-step instructions on building a complete database and application from scratch providing the R:BASE relational logic as it adheres to Dr. Codd's relational model. This is where one would learn how to quickly adapt to the R:BASE interface!

3.1.3.1 Issues and Suggestions

NOTE: From inside the Label Designer, you can launch the Label Designer help file by pressing the [Shift] + [F1] hot keys.

- **Sub-Totals and Totals**

To calculate sub-totals and totals of table columns, Aggregate Variables have always been used to perform this job. Now, the R:BASE 11 for Windows Label Designer has a built-in object called DB Calc, which performs these same functions on table columns without creating the extra overhead of variables. You can replace your Variable Label objects with DB Calc objects in instances where the Variables Object results will not be used further down in the label.

- **#DATE, #TIME, #PAGE, Document, and Page Information**

The R:BASE 11 for Windows Label Designer now has a System Variable control, which allows an object to be placed for the current DATE, TIME, Document Name, Print DATE and TIME, or one of many Page Numbering options. All Variable Label objects whose expression value consists of a system variable (#DATE, #TIME) should be deleted and replaced with a System Variable control. Any Variable Label objects whose expression value consists of the #PAGE system variable, you must delete the variable object and replace that object with a System Variable control.

- **Label Variables**

Label variables that perform calculations or combine a number of values should be enclosed in parenthesis. If error messages are displayed for variables performing calculations, like the one below,

```
2 : CURRENCY      vTotal = price * quantity
```

add parenthesis as follows:

```
2 : CURRENCY      vTotal = (price * quantity)
```

- **Label Totals**

To be sure label totals are calculating correctly on all NULL and not NULL records, make sure the **ZERO** setting is ON. Check the setting under "Settings" > "Configuration Settings".

- **Printer Control Codes**

If DOS or machine printer control codes are used in R:BASE Label variables, they can no longer be stored in the Label Variables. A control code would appear in the variable list with an expression as follows: <27 40 115 49 83>

If the control code were used to alter the text on the label (e.g. bold, underline, etc.), the control code must be deleted. There are many new features in the Windows interface that replace control codes for altering label text objects. The R:BASE 11 for Windows version allows you to perform these font changes much easier.

If control codes listed in the label control hardware for paper feeds or to open a cash drawer, then document down the exact code and review the "PCC Label" Control in the Label Designer. This control will support the same functionality used previously.

- **Picture Format**

For a label object, [] may be displayed in the label preview. In R:BASE for DOS and R:BASE for Windows versions prior to 7.x, a "Picture Format" option was available. The option allowed the justification of the value and several formatting options based upon the data type of the object. For example, to add the justification a [>], or [<] was used to right/left justify.

In R:BASE 11, this feature is called "Display Format", which still offers several formatting options for report objects based upon the data type of the object. While the formatting option is still supported, the justification option is no longer supported in the previous context, within the "Display Format" settings. To right/left/center justify text of an object, use the "Format" tool bar which supports many text formatting options.

If you see [] listed in your label preview for a label object, you must right click on the object, select "Display Format", and remove the [] characters.

- **Page Setup**

For DOS to Windows conversions, the default page size setting is set to it's largest values in order to make sure no objects are lost in the conversion process. You must make sure you alter your page settings back to your desired values in order for your labels to print correctly.

To alter the page size settings, follow the below instructions:

1. Select "File" > "Page Setup..." from the main menu bar.
2. Select the "Paper Size" tab
3. For Portrait orientation reports, the Width should be set to "8.5".
4. For Landscape orientation reports, the Height should be set to "8.5".

From this dialog, you can also adjust your labels margins from the "Margins" tab.

After making any changes, make sure that all of your labels objects are located within the defined page size settings and margins. Otherwise, R:BASE will display a warning that "Controls beyond the new page width will be deleted." In this case, select "Cancel" to avoid losing any controls. Then move your controls within your desired page settings and then make your page size setting changes again.

- **Label Preview Tip**

The "Label Preview" tab will display a preview of your label as it may appear when you send the label to the printer or screen. Before selecting the tab, you must make sure that there are no errors listed in your label variables, otherwise, you will receive an error message for every error listed multiplied by the number of records displayed in the Label Preview window.

What you can do to limit the number of rows displayed in the Label Preview window, thus limiting the number error messages. To do so, alter the Label Default settings for how many rows are displayed when previewing a label.

1. Select "Settings" > "Report/Label Designer" > "Default Settings..." from the main menu bar.
2. Alter the value for "Row Count for Preview:" to a smaller value other than 100; perhaps to 10.

- **Lookup Variable Tip**

Recently added to the R:BASE 11 Label Designer is the option to run a command file (or Action) for a form before the Label Designer is launched. In the Label Designer, choose "Label" from the Menu bar, the select "Actions" > "On Before Design...".

This allows you to create any variables that Lookup Variables are based upon to avoid annoying error messages when the label is opened in the designer. An example of a lookup variable based upon another variable would be...


```
1 : TEXT      vCompany = Company IN Customer WHERE CustID =
.vCustID
```

The error message appears because the variable vCustID is not defined. Adding the following SET VAR command to the "On Before Design..." action will alleviate this error message:

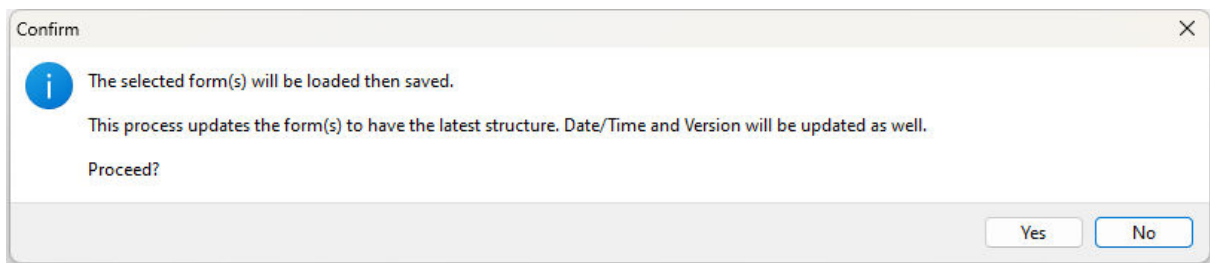
```
SET VAR vCustID INTEGER
```

The "On Before Design..." action is located under "Label" > "Actions" on the main menu bar.

3.2 R:BASE 7.x/Turbo V-8 Forms, Reports, & Labels

R:BASE 7.x/Turbo V-8 forms, reports, and labels will automatically populate the Forms, Reports, and Labels menus of the R:BASE 11 Database Explorer. The items will run in R:BASE 11 with little or no changes required on your behalf. However, it is recommended that you review each item to ensure the modules behave as expected.

To assist with the database migration use the "Touch" utility to quickly update and save forms, reports, and labels when migrating between current versions. The Database Explorer menu works like opening the selected object in the designer, making the content dirty, and saving it. The feature allows a user to save hundreds of forms, reports, and labels when migrating between R:BASE versions.



Corrections have been made in the form table settings for the "Update Existing Rows" option which was corrected to recognized the setting value. The following command can be used to scans and set all form's "Update Existing Rows" option to checked/enabled. Note that you only need to run the command once.

```
PROPERTY APPLICATION RESET_FORMS_ROW_UPDATE_FLAG ' * '
```

A form list may be specified to scan and set the specified form's "Update Existing Rows" option to checked/enabled:

```
PROPERTY APPLICATION RESET_FORMS_ROW_UPDATE_FLAG ' form1 , form2 , form3 '
```

Leaving the last parameter blank scans and sets all form's "Update Existing Rows" option to checked/enabled, where the form modified date is less than November 7, 2017, which is when the "Update Existing Rows" option has been corrected to recognized the setting value.

```
PROPERTY APPLICATION RESET_FORMS_ROW_UPDATE_FLAG ' '
```

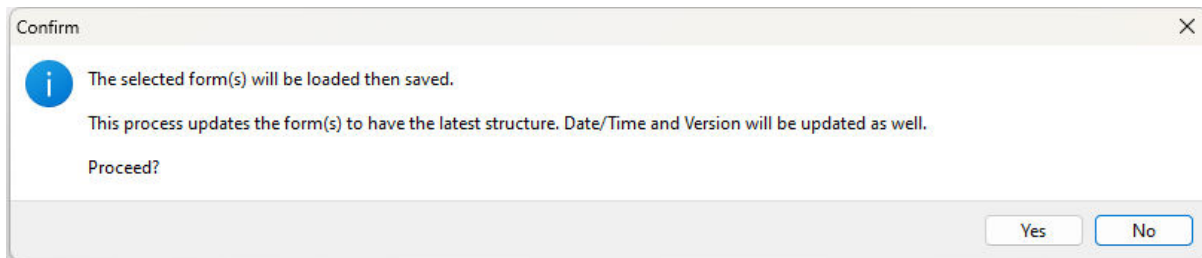
To assist in the migration to R:BASE 11, [new conversion features](#) have been built in for your convenience.

Please review the "What's New in R:BASE 11 for Windows" PDF document for additional details.

3.3 R:BASE eXtreme 9.x (32/64) Forms, Reports, & Labels

Your R:BASE eXtreme 9.x forms, reports, and labels will automatically populate the Forms, Reports, and Labels menus of the R:BASE 11 Database Explorer. The items will run in R:BASE 11 with no changes required on your behalf. It is recommended that you review each item to ensure the modules behave as expected.

To assist with the database migration, use the "Touch" utility to quickly update and save forms, reports, and labels when migrating between current versions. The Database Explorer menu works like opening the selected object in the designer, making the content dirty, and saving it. The feature allows a user to save hundreds of forms, reports, and labels when migrating between R:BASE versions.



Corrections have been made in the form table settings for the "Update Existing Rows" option which was corrected to recognized the setting value. The following command can be used to scans and set all form's "Update Existing Rows" option to checked/enabled. Note that you only need to run the command once.

```
PROPERTY APPLICATION RESET_FORMS_ROW_UPDATE_FLAG '*'
```

A form list may be specified to scan and set the specified form's "Update Existing Rows" option to checked/enabled:

```
PROPERTY APPLICATION RESET_FORMS_ROW_UPDATE_FLAG 'form1,form2,form3'
```

Leaving the last parameter blank scans and sets all form's "Update Existing Rows" option to checked/enabled, where the form modified date is less than November 7, 2017, which is when the "Update Existing Rows" option has been corrected to recognized the setting value.

```
PROPERTY APPLICATION RESET_FORMS_ROW_UPDATE_FLAG ''
```

To assist in the migration to R:BASE 11, [new conversion features](#) have been built in for your convenience.

Please review the "What's New in R:BASE 11 for Windows" PDF document for additional details.

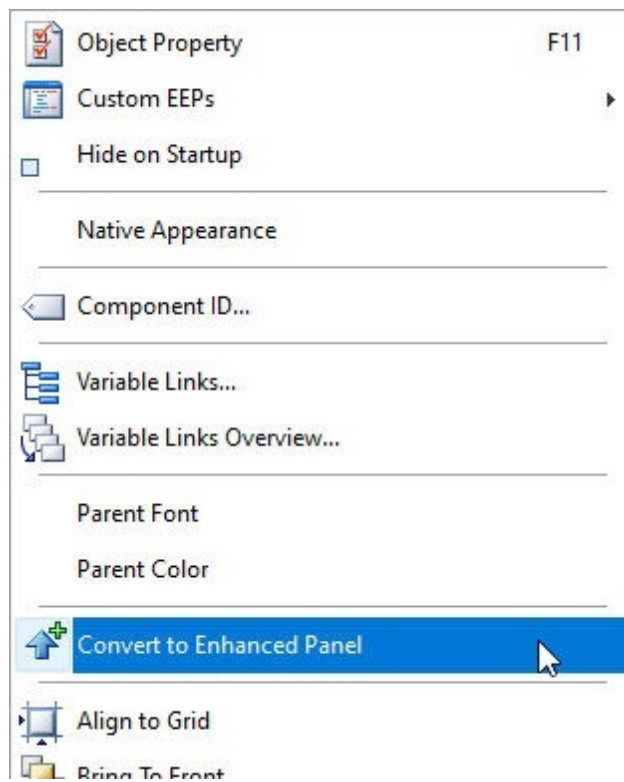
3.4 New Conversion Enhancements

To assist in the migration to R:BASE 11, new conversion features have been built in for your convenience.

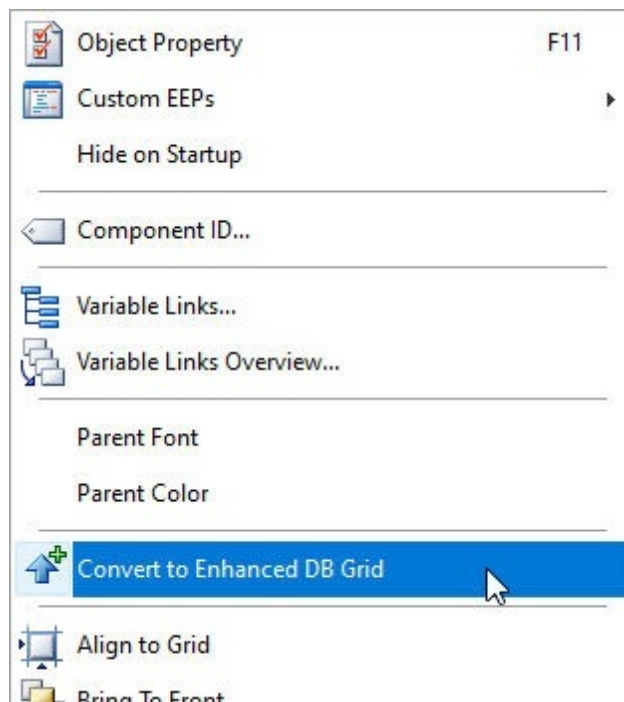
"Single Click" Conversion to Enhanced Form Controls

In the earlier versions of R:BASE, there are several introductory controls that have been enhanced considerably in later R:BASE releases. For example, the Tab Control was provided in R:BASE 7.0 for Windows. Later, in R:BASE 7.1 the Enhanced Tab Control was introduced to offer many additional features. The same comparison can be made for the popular DB Grid control whose "enhanced" counterpart was introduced in R:BASE eXtreme for Windows. Migrating from the introductory control to the enhanced counterpart meant recreating the object on the form, while referencing the properties of the original.

Now, the conversion of the introductory control can be made with a single click. For example, by right clicking on a Panel control, the option to "Convert to Enhanced Panel" is available.



By right clicking on a DB Grid control, the option to "Convert to Enhanced DB Grid" is available.

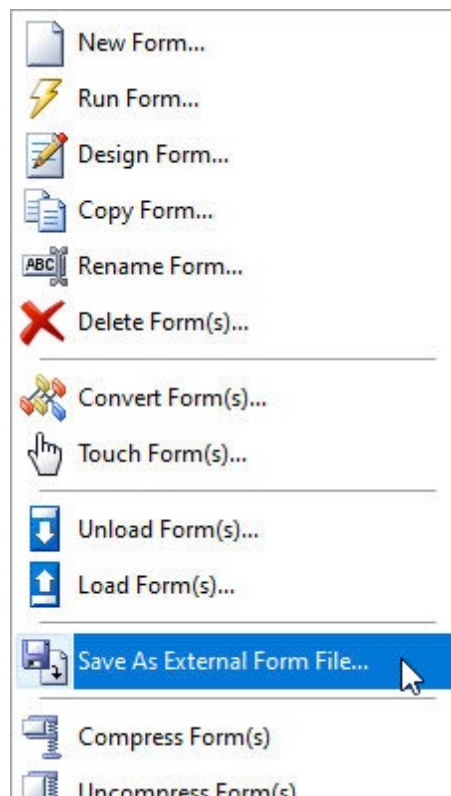


The following single-click conversions to "enhanced" form controls are available:

- Panel -> Convert to Enhanced Panel
- Group Box -> Convert to Enhanced Group Box
- Speed Button -> Convert to Enhanced Speed Button
- Tab Control -> Convert to Enhanced Tab Control
- Calendar -> Convert to Enhanced Calendar
- Variable Calendar -> Convert to Enhanced Variable Calendar
- DB Calendar -> Convert to Enhanced DB Calendar
- DB Grid -> Convert to Enhanced DB Grid
- DB Navigator -> Convert to Enhanced DB Navigator
- DB Rich Edit -> Convert to Advanced DB Rich Edit
- Speed Button -> Convert to Enhanced Speed Button

Variable Form Conversion to External Form File

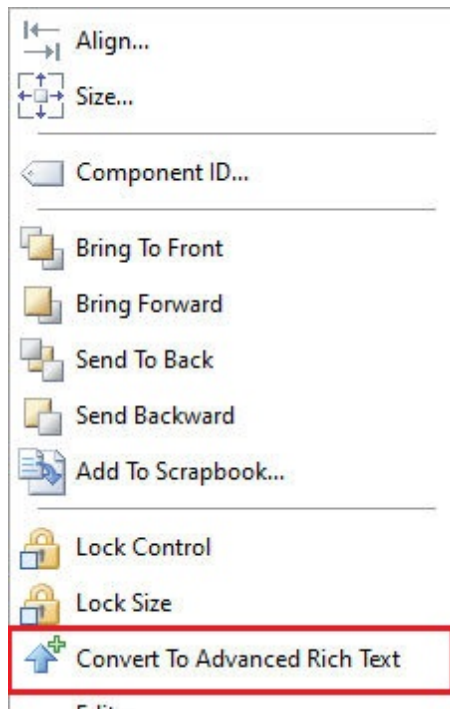
Another new time-saving conversion feature is located in the Database Explorer within the "Forms" options where a new selection allows for a Variable Form to be saved as an External Form File. The option is only enabled when a Variable Form is selected.



"Single Click" Conversion to Advanced Report Controls

The Report/Label Designer includes added context menu options to easily convert rich text objects to advanced rich text objects.

- Rich Text > Convert to Advanced Rich Text
- DB Rich Text > Convert to Advanced DB Rich Text
- Variable Rich Text > Convert to Advanced Variable Rich Text



Please review the "What's New in R:BASE 11 for Windows" PDF document for additional features.

Part

IV

4 Converting Applications

As R:BASE can be customized for any type of business model or company schema, there have always been several ways that developers and users could create an application. Actually, if you think of what an application consists of, you're really looking at a stored series of commands in what are called command files, which are then combined with other command files to form applications.

R:BASE provides all of the tools you need to produce command files and applications, it is just up to you to decide which method serves your needs best.

4.1 R:BASE 6.5++ and Lower Applications

4.1.1 Application Express Users (.APP)

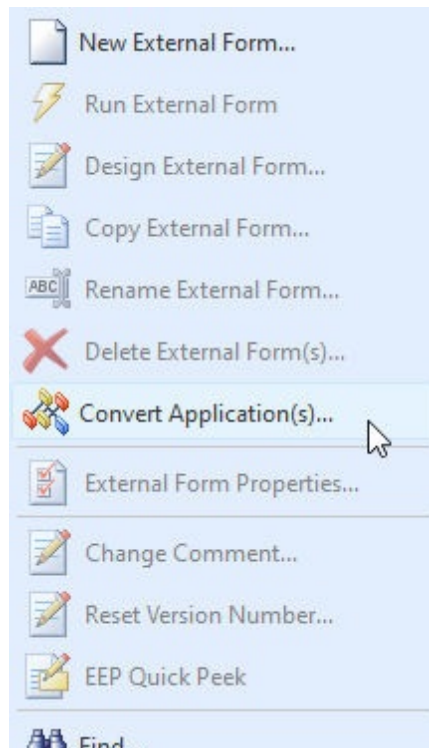
If you are converting an application that was designed using the Application Express module from versions prior to R:BASE 7.x, the actual application file consisting of the R:BASE command language (.APP) can be converted within R:BASE 11. In addition to the .APP file, there will be an .APX and either .API or .APW files. You will only need the .APP or the .API files.

The application may be one large .APP file or several smaller command files; usually with the .CMD file extension. The standard command file extension was .CMD in early R:BASE versions. It may also be possible that no file extension is used for the command files. In this case, it is recommended that the .RMD file extension be added to the files so it may be displayed within the "Command Files" menu of the Database Explorer's Navigator.

In R:BASE for Windows, the new file extension standard is .RMD, as more file servers and mail servers flag the .CMD file extension as a possible virus. Or, if you are using a custom file extension, you can add the extension to the "File Mask:" field under the "Command Files" menu, so the files will then appear in this window.

The source code command syntax within the .APP file contains the menus and menu actions of your custom application in the form of command blocks, menu blocks, and possibly screen blocks. The .APP file is likely located with your database files. So, when you connect to your database using R:BASE 11, you will see the .APP file listed under the "Command Files" menu within the Database Explorer's Navigator. The files will not be listed under the "Applications" menu as the R:BASE 11 Application Designer will not open legacy Application Express projects for editing.

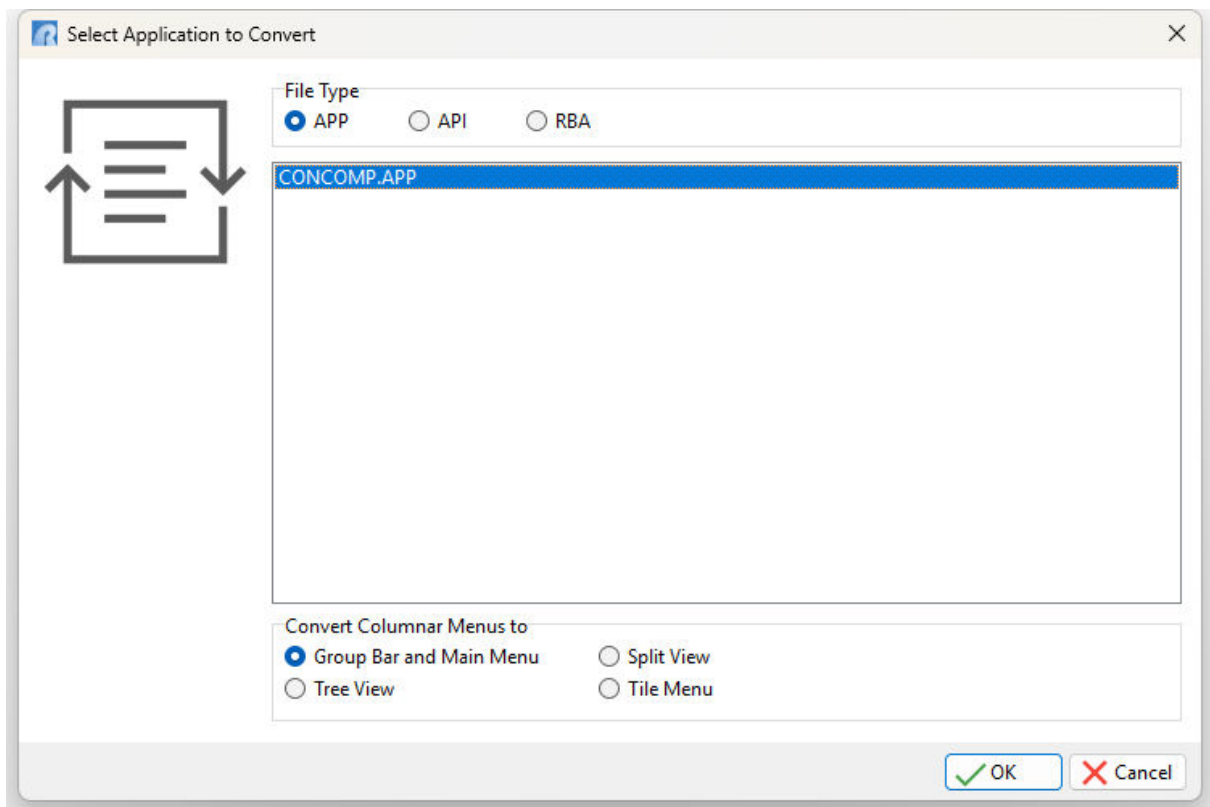
In order to use the .APP file contents in R:BASE 11, a new "**Convert Application(s)...**" option has been made available within the "External Forms" portion of the Database Explorer. This feature will parse through the legacy Windows and DOS legacy application files and converts it to an External Form File. If the file does not contain \$COMMAND blocks or \$MENU Blocks, then no conversion will take place.



Selecting the "Convert Application(s)..." option will display a dialog to select an application file (.APP/.API). If no files are displayed in the dialog, press the "Cancel" button and select "Utilities" > "Set Working Directory..." to navigate to the folder where the application files reside. Then, revisit the "Convert Application(s)..." option.

4.1.1.1 Menu System Types

The converted external form will vary based upon the menu type (POPUP, COLUMN, or PULLDOWN) of the .APP application file. With the Tree View, Split View, and Tile Menu application conversions, the COLUMN and PULLDOWN items are merged, and POPUP items become form commands. With the "Group Bar and Menu Bar" conversion, COLUMN items becomes a Group Bar, PULLDOWN items becomes a Main Menu, and POPUP items become form commands. All menu types will create a collection of user interface groups that contains a list of items to select that can invoke commands.



After selecting the "OK" button, the converted external form will appear in the Database Explorer window. An external form file, with the .RFF file extension, will reside in the folder. If the database and application is moved to another directory, be sure to move the external form file as well. To design the external form and see the converted command syntax, select "Design External Form..." from Navigator options. You will then enter the External Form Designer module to alter the external form.

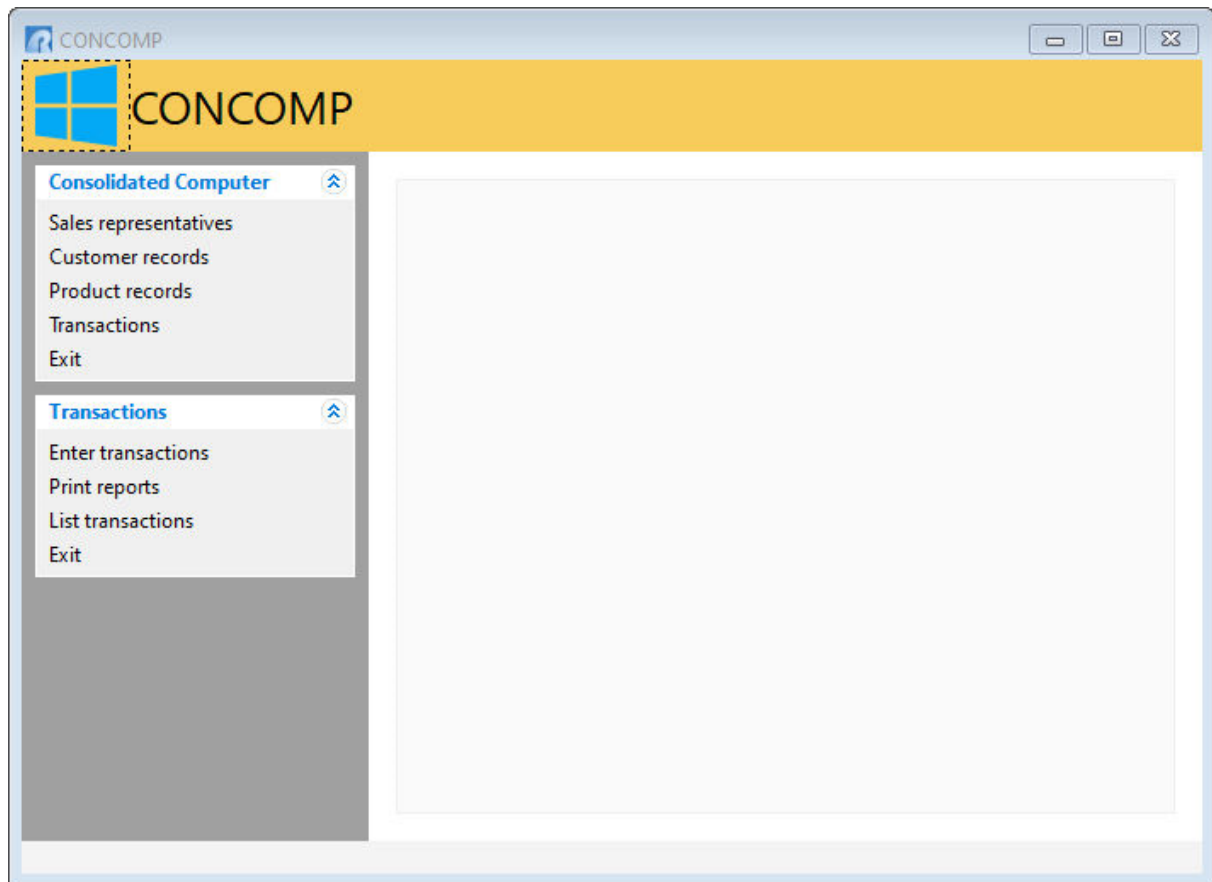
Based upon the menu system type of the original .APP application file, R:BASE will create the following menu system types into the external form:

- [Group Bar](#)
- [Menu Bar](#)
- [Tree View](#)
- [Split View](#)
- [Tile Menu](#)

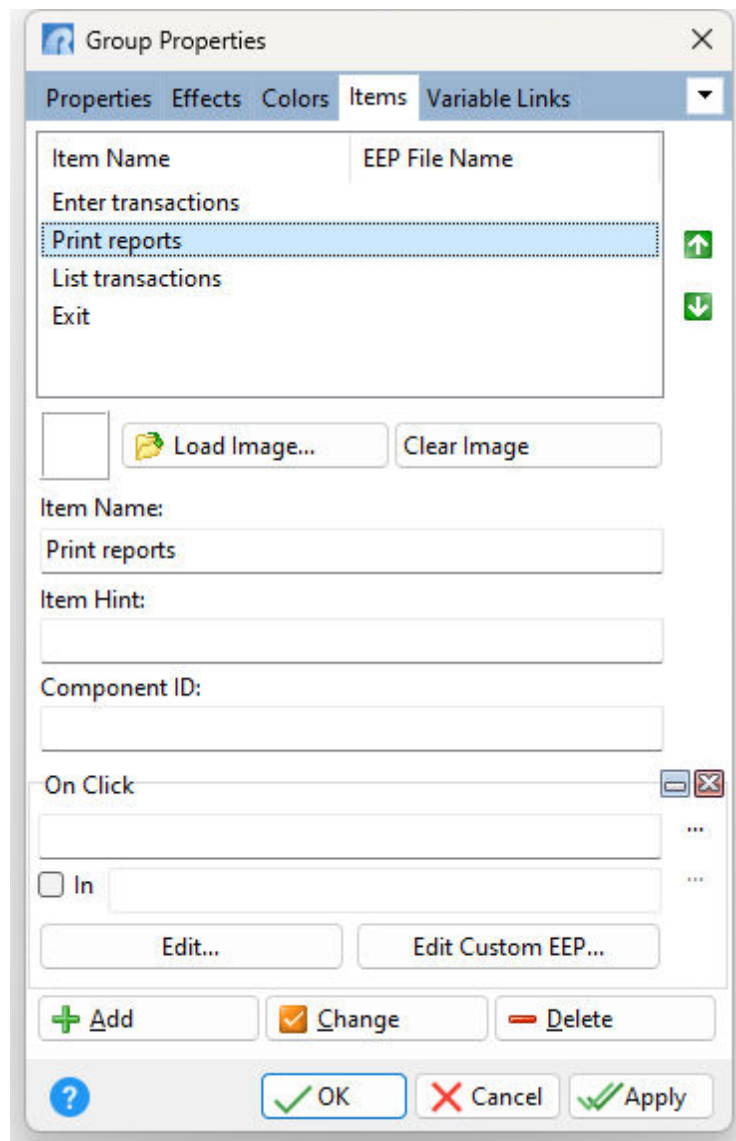
4.1.1.1.1 Group Bar

After the conversion, the Group Bar will be located on the left side of the form with menu options placed accordingly. The Group Bar items displayed should match the \$MENU blocks from the originating .APP application file. There may be some menu options that are no longer used. Make a note of the items as this will save time in the process of reviewing and updating the command syntax.

A Group Bar control is a collection of "groups", that can be opened and closed by using the arrows (looks like two carat characters ^). Each group contains a list of items that are used to run a series of R:BASE commands or an external file. The "Groups" are displayed below with a blue font. The Group Items are located under the Group with a black font. In the below example image, "Transactions" is the second Group, where "Enter transactions", "Print reports", "List transactions", and "Exit" are the Items.



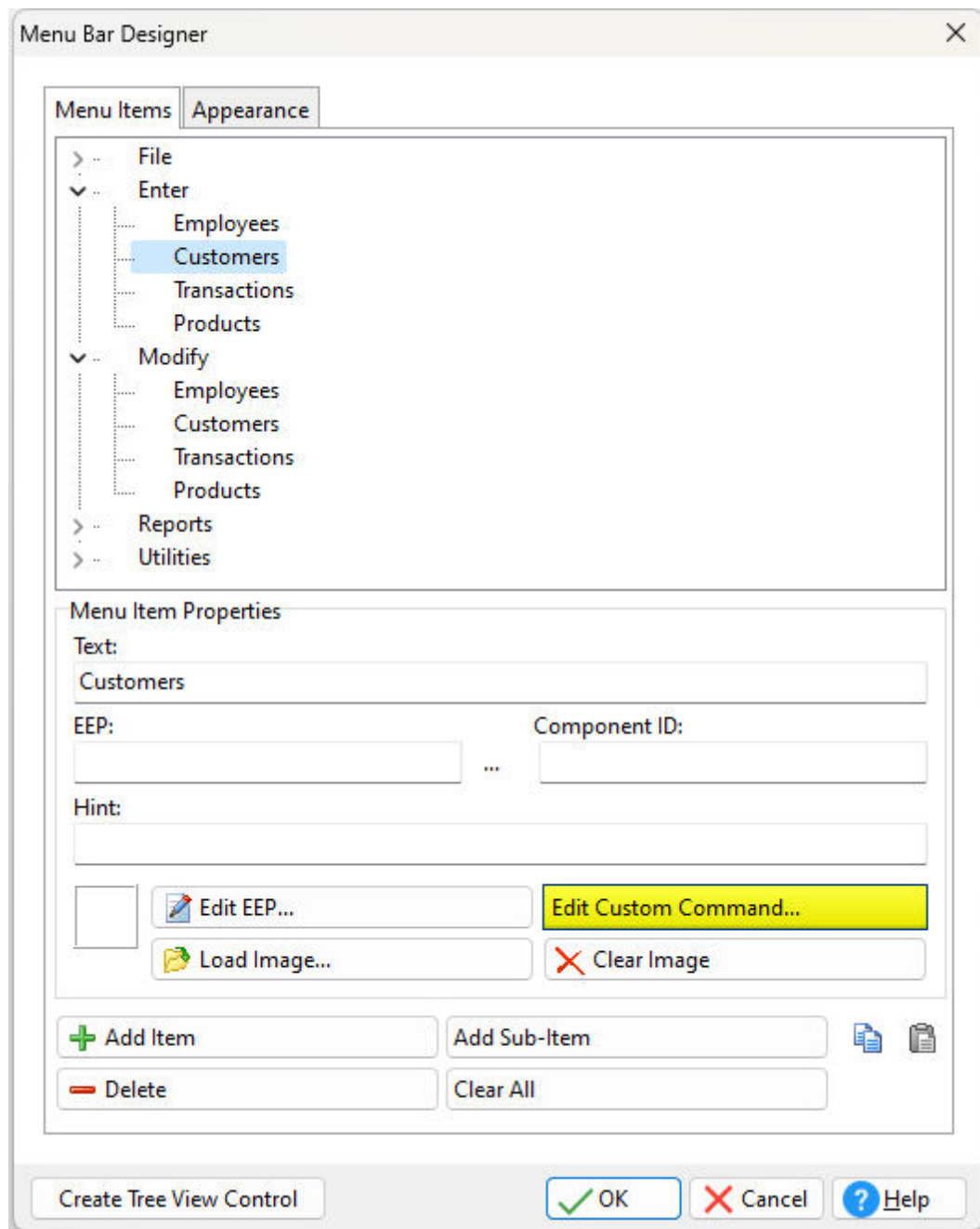
By right clicking on the Group, and selecting "Object Property [F11]", the Group Properties dialog is displayed. Select the "Items" tab to review the list of item names. Any [command syntax](#) loaded in an Item will highlight the "Edit Custom EEP..." button with a yellow background. Selecting the "Edit Custom EEP..." button will display the stored command syntax. Before running the external form and selecting any menu options, the [command syntax must first be checked](#). If no syntax was loaded into the EEP, then the button background will not be yellow. The command syntax would then be loaded into the Custom Form Actions. Select "Layout" > "Custom Form Actions" from the main menu bar to review the command syntax.



4.1.1.1.2 Menu Bar

After the conversion, a blank gray canvas will be displayed in the External Form Designer. The menu system and command syntax for each menu item from the application are stored within the external form file. From the main menu bar, select "Layout" > "Design Menu Bar". The Menu Bar Designer will launch with the application file menus placed accordingly. The Menu Items displayed should match the \$MENU blocks from the originating .APP application file. There may be some menu options that are no longer used. Make a note of the items as this will save time in the process of reviewing and updating the command syntax.

A Menu Bar is a horizontal strip that contains lists of available menus for a certain program. In Windows programs, the menu bar resides at the top of a window. The Menu Bar uses main "Items" which display across the top of a window, and "Sub-Items" which drop down when the main Item is selected. Items and Sub-Items can be used to run a series of R:BASE commands or an external file. In the below example, "Employees" is the main Item, where "Enter New Employee" and "Edit Existing Employee" are the sub-items.



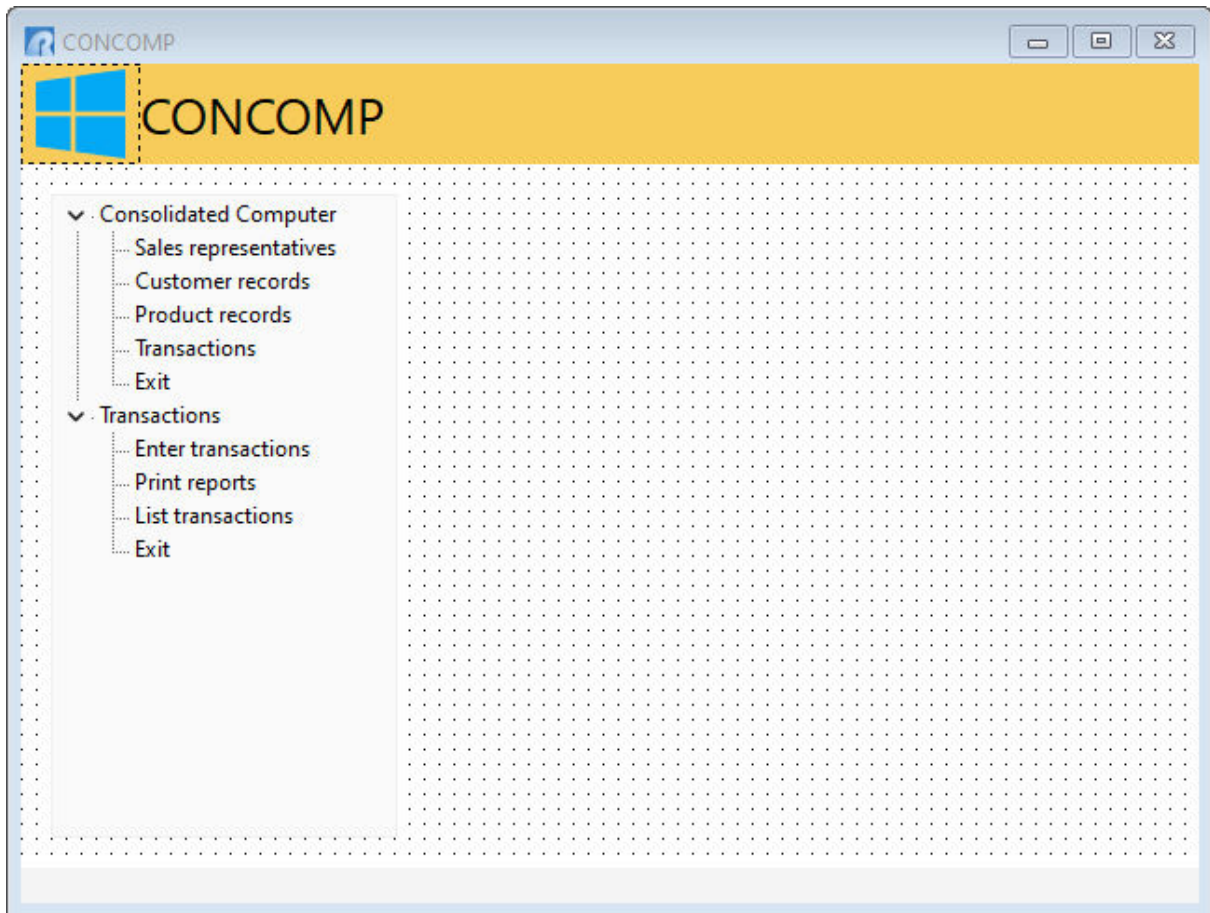
During the conversion, R:BASE loaded the command syntax portions of the .APP application file into the "Custom Command" portions of the Menu Bar. Any [command syntax](#) loaded in a Sub-Item will highlight the "Edit Custom Command..." button with a yellow background. Selecting the "Edit Custom Command..." button will display the command syntax stored within the Item or Sub-Item. Before running the external form and selecting any menu options, the [command syntax must first be checked](#).

If no syntax was loaded into the menu item or sub-item, then the button background will not be yellow. The command syntax would then be loaded into the Custom Form Actions. Select "Layout" > "Custom Form Actions" from the main menu bar to review the command syntax.

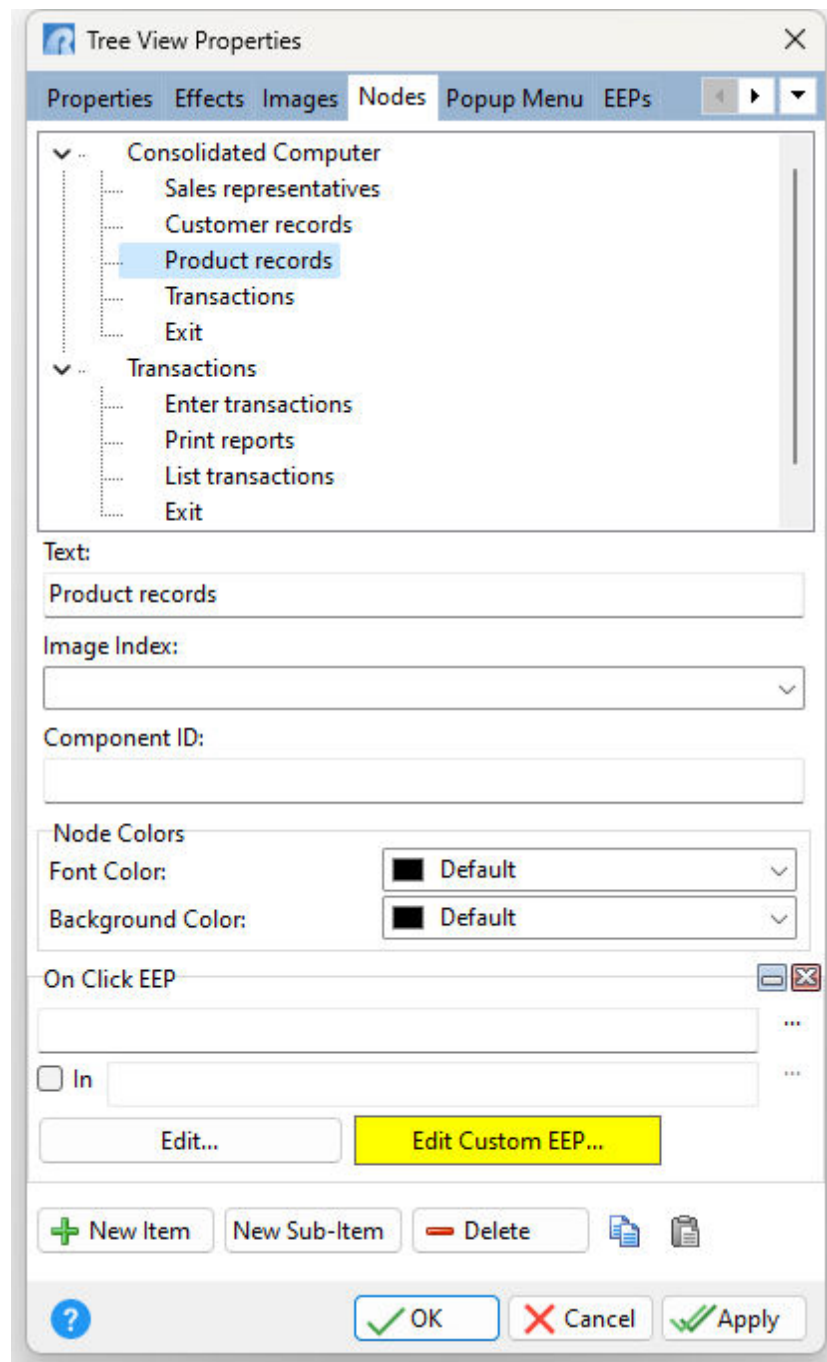
4.1.1.1.3 Tree View

After the conversion, the Tree View will be located on the left side of the form with menu options placed accordingly. The Tree View items displayed should match the \$MENU blocks from the originating .APP application file. There may be some menu options that are no longer used. Make a note of the items as this will save time in the process of reviewing and updating the command syntax.

A Tree View control displays an object containing a hierarchical list of static items (nodes), such as the headings in a document, the entries in an index, or the files and directories on a disk. Each node consists of a label and an optional image, where each node can have a list of subitems associated with it. By selecting the tree structure, the user can expand or collapse the associated list of subitems. The main "Nodes" are displayed below next to the minus characters. The Sub-Items are located within the Nodes linked by the dotted lines. In the below example image, "Transactions" is the second node, where "Enter transactions", "Print reports", "List transactions", and "Exit" are the sub-items.



By right clicking on the Tree View, and selecting "Object Property [F11]", the Tree View Properties dialog is displayed. Select the "Nodes" tab to review the list of nodes and sub-items. Any [command syntax](#) loaded in an sub-item will highlight the "Edit Custom EEP..." button with a yellow background. If no syntax was loaded into the sub-item, then the button background will not be yellow. Selecting the "Edit Custom EEP..." button will display the stored command syntax. Before running the external form and selecting any menu options, the [command syntax must first be checked](#). If no syntax was loaded into the EEP, then the button background will not be yellow. The command syntax would then be loaded into the Custom Form Actions. Select "Layout" > "Custom Form Actions" from the main menu bar to review the command syntax.

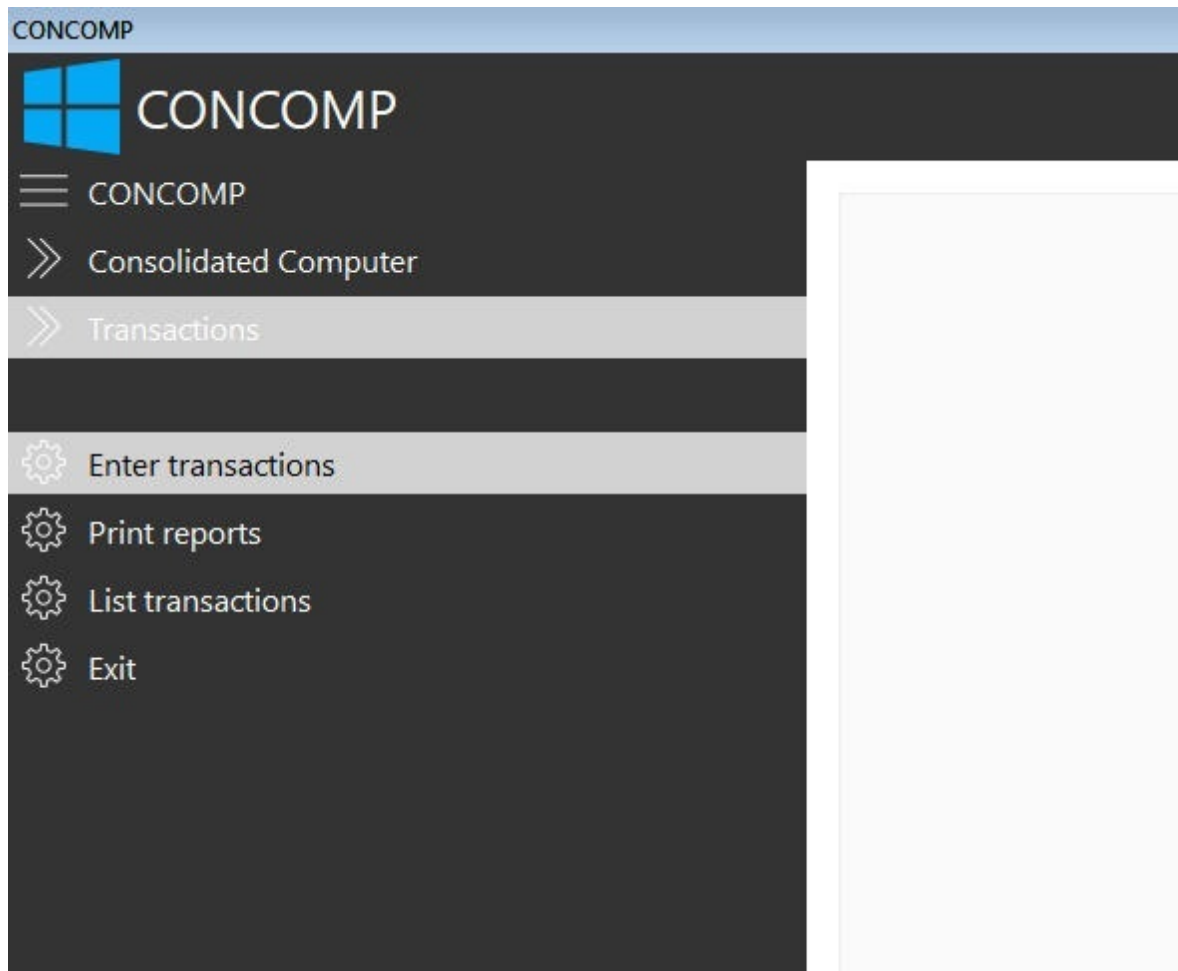


4.1.1.1.4 Split View

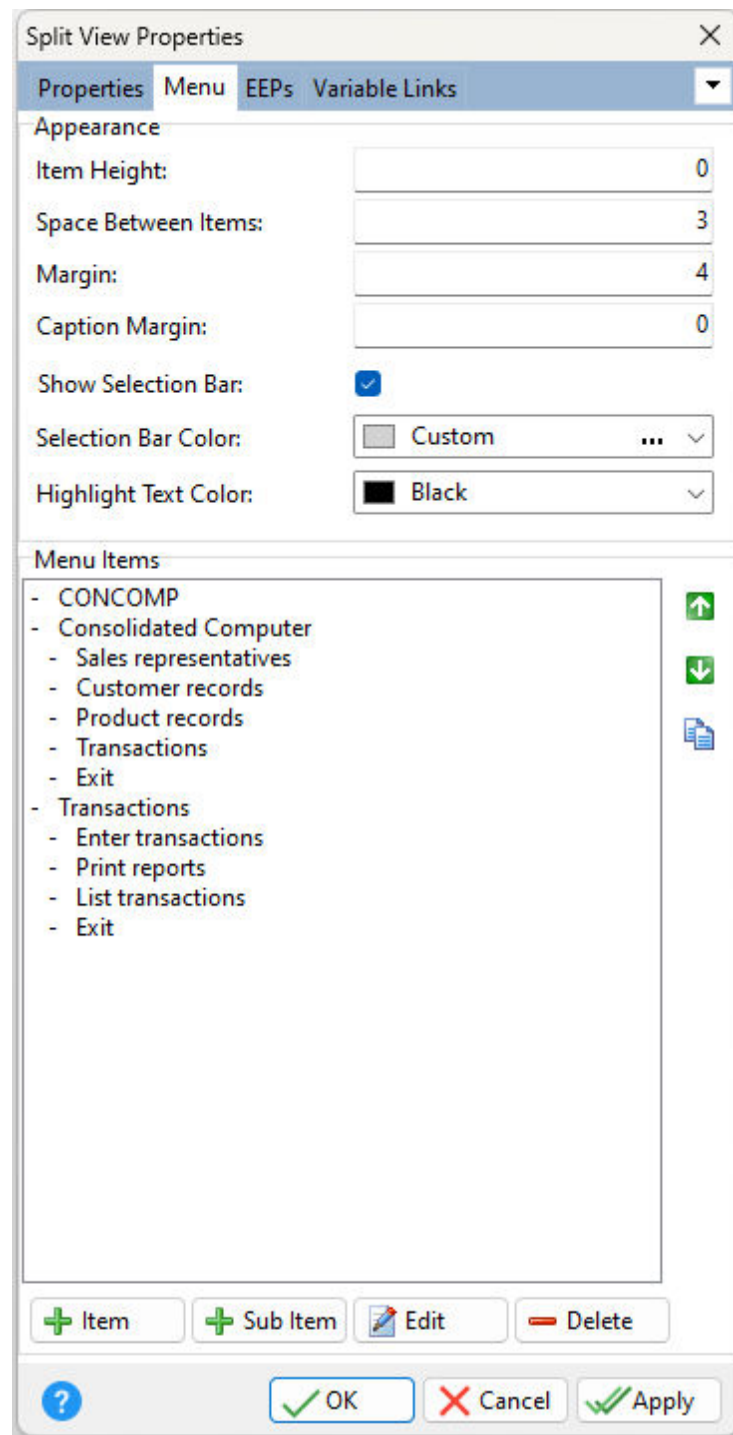
After the conversion, the Split View will be located on the left side of the form with menu options placed accordingly. The Split View items displayed should match the \$MENU blocks from the originating .APP application file. There may be some menu options that are no longer used. Make a note of the items as this will save time in the process of reviewing and updating the command syntax.

The Split View is a control offering a collection of user interface command groups. A command group typically contains a list of clickable items that invoke commands. The Split View presents a list of initial menu options, which is then expanded when the mouse cursor hovers over the list. In the below example

image, "Transactions" is the main menu item, where "Enter transactions", "Print reports", "List transactions", and "Exit" are the sub items.



By right clicking on the Split View, and selecting "Object Property [F11]", the Split View Properties dialog is displayed. Select the "Menu" tab to review the list of menu items.



Select a menu item and then the "Edit" button to review the menu item's properties. Any [command syntax](#) loaded in a menu Item will highlight the "Edit Custom EEP..." button with a yellow background. Selecting the "Edit Custom EEP..." button will display the stored command syntax. Before running the external form and selecting any menu options, the [command syntax must first be checked](#). If no syntax was loaded into the EEP, then the button background will not be yellow. The command syntax would then be loaded into the Custom Form Actions. Select "Layout" > "Custom Form Actions" from the main menu bar to review the command syntax.

Menu Item

Caption and Image

Caption:
Sales representatives

☒ Enabled

Image:
[gear icon] Load... Clear

Component ID
[empty field] ...

On Click
[empty field] ...

☐ In [empty field] ...

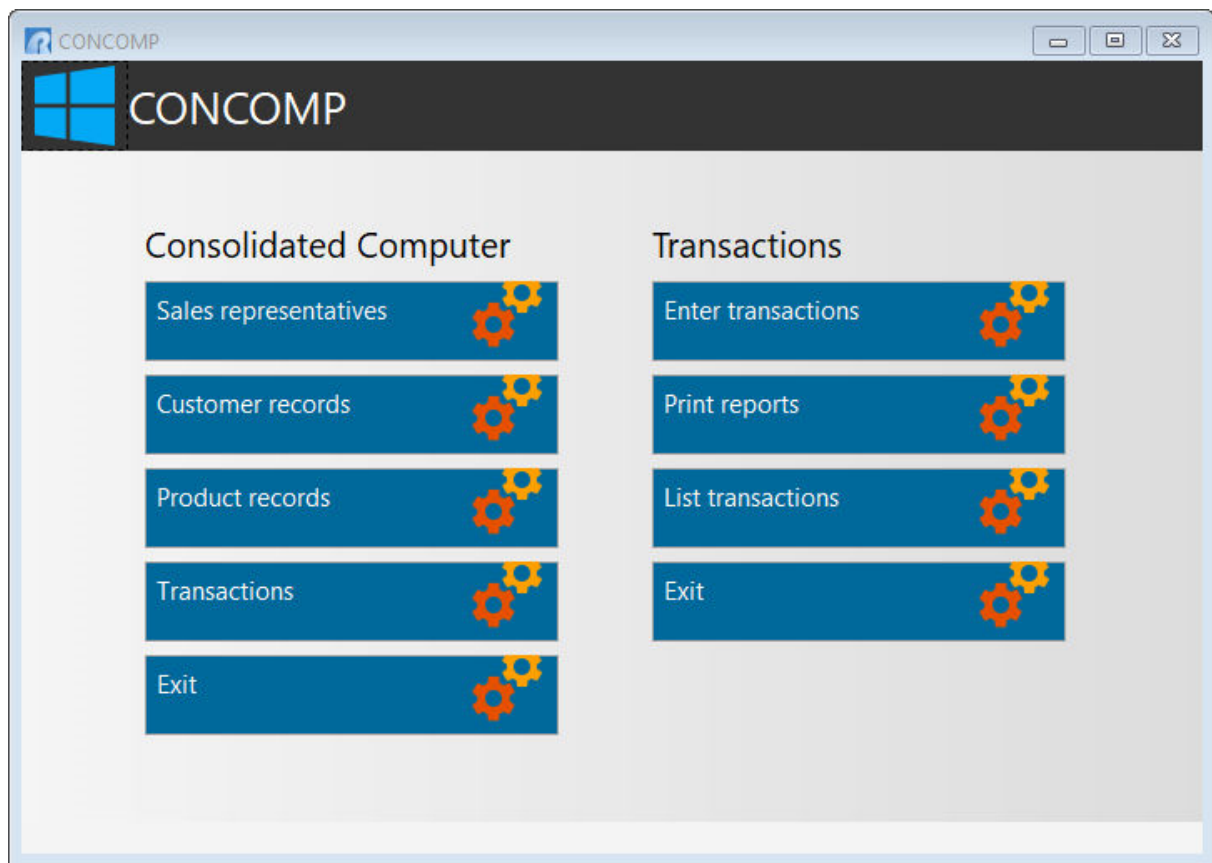
Edit... Edit Custom EEP...

OK Cancel

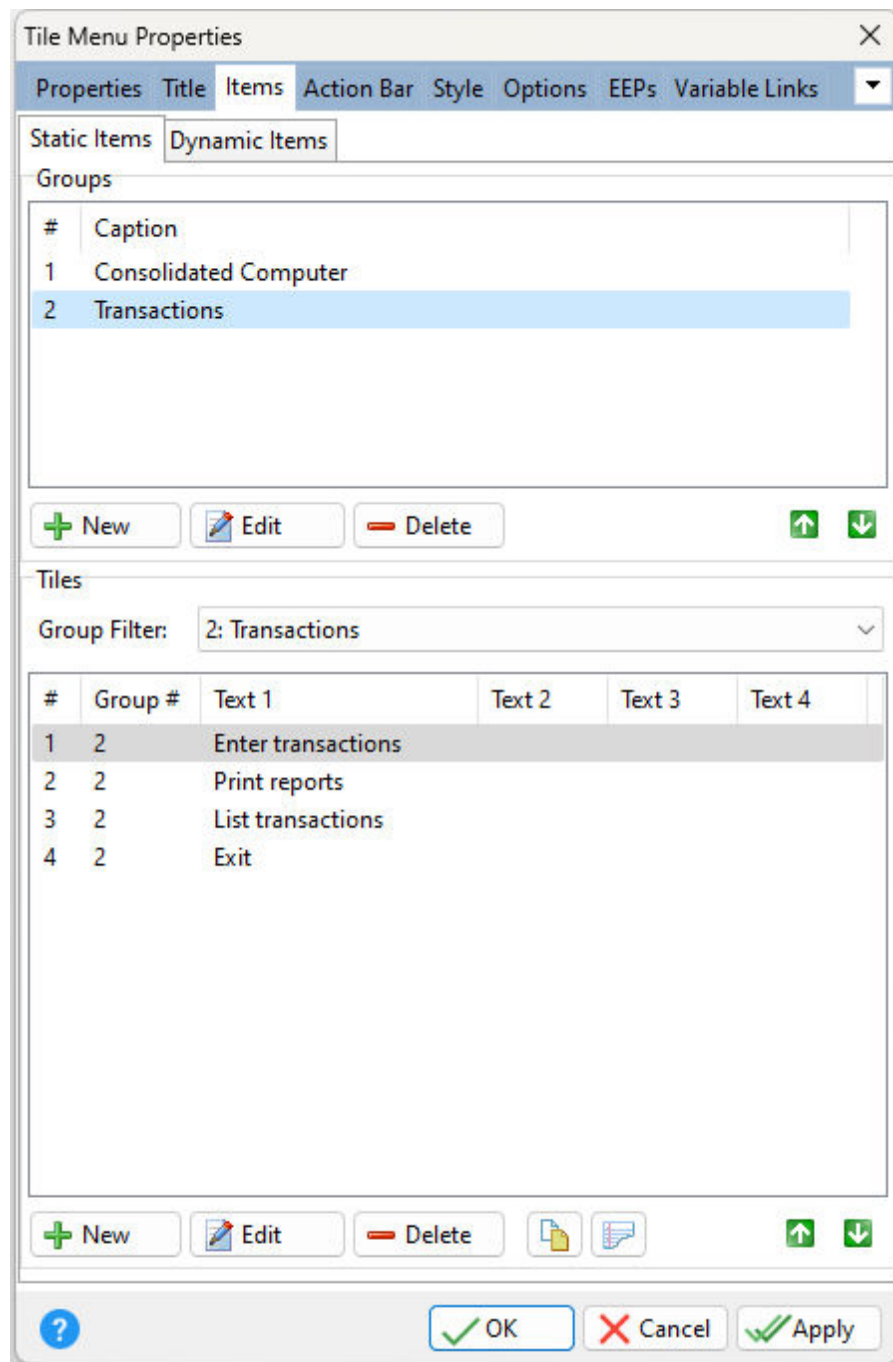
4.1.1.1.5 Tile Menu

After the conversion, the Tile Menu will be located on the center of the form with menu options placed accordingly. The Tile Menu items displayed should match the \$MENU blocks from the originating .APP application file. There may be some menu options that are no longer used. Make a note of the items as this will save time in the process of reviewing and updating the command syntax.

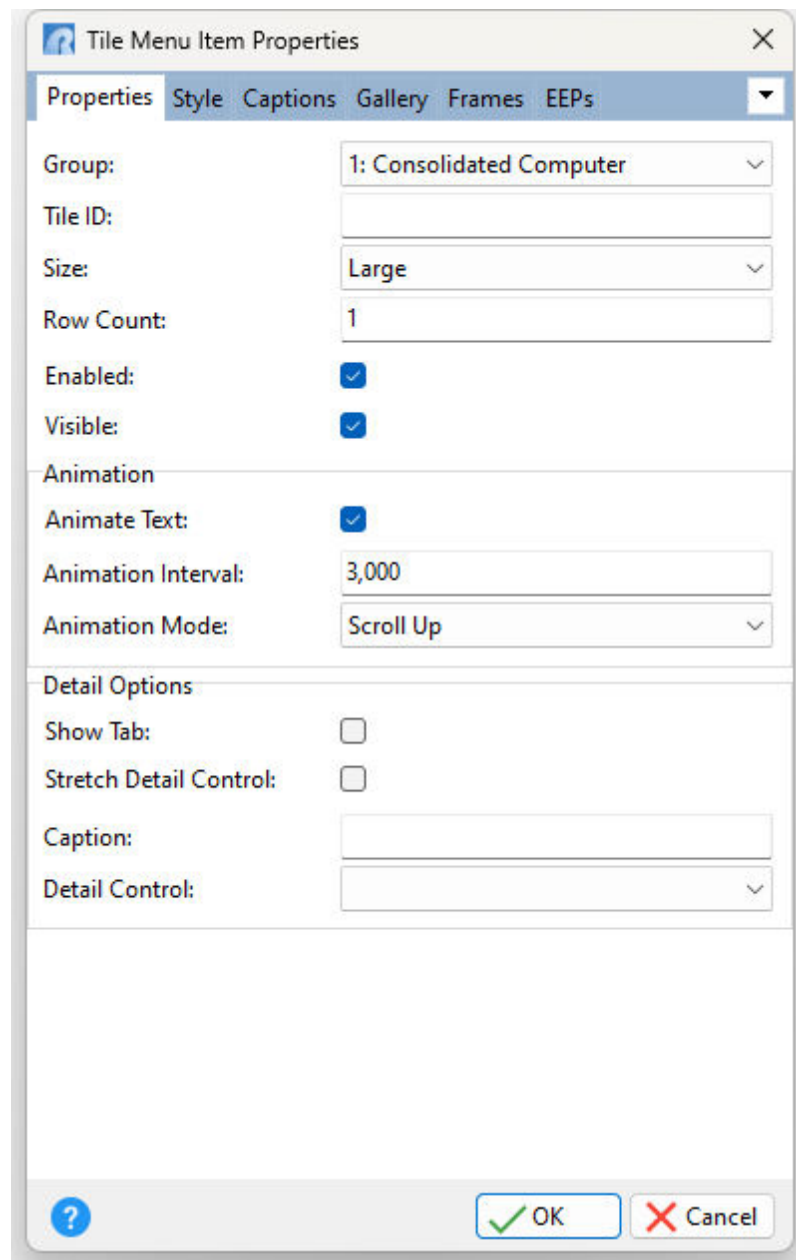
The Tile Menu is a control offering a tile (colored rectangles or squares) interface to create a collection of user interface command groups. A command group typically contains a list of clickable items that invoke commands. The Tile Menu is both a menu control and a container, where the tile and tile group functionality can be used to create a powerful dashboard for an R:BASE application. In the below example image, "Transactions" is a main group, where "Enter transactions", "Print reports", "List transactions", and "Exit" are the tiles.



By right clicking on the Tile Menu, and selecting "Object Property [F11]", the Tile Menu Properties dialog is displayed. Select the "Items" tab to review the list of group and tile items.



Select a tile item and then the "Edit" button to review the tile item's properties. Under the "EEPs" tab, any [command syntax](#) loaded in a tile menu item will highlight the "Edit Custom EEP..." button with a yellow background. Selecting the "Edit Custom EEP..." button will display the stored command syntax. Before running the external form and selecting any menu options, the [command syntax must first be checked](#). If no syntax was loaded into the EEP, then the button background will not be yellow. The command syntax would then be loaded into the Custom Form Actions. Select "Layout" > "Custom Form Actions" from the main menu bar to review the command syntax.

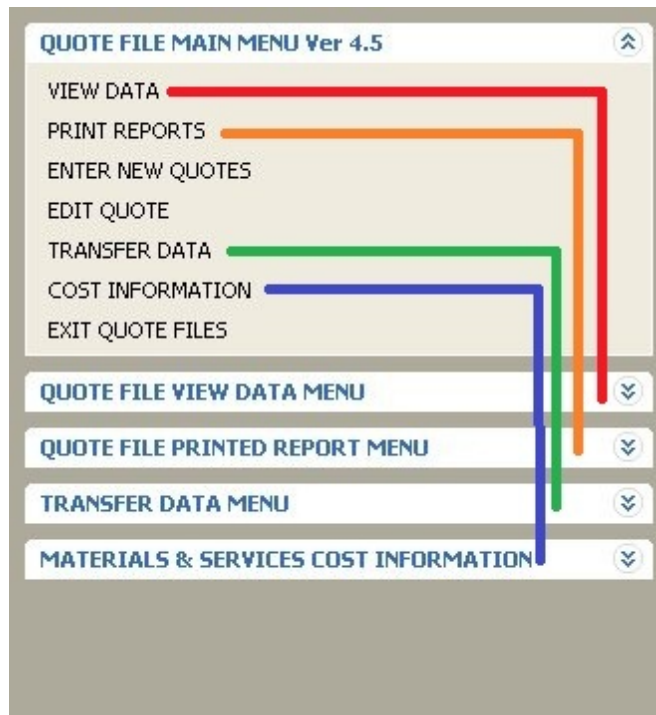


4.1.1.2 Menu Action Types

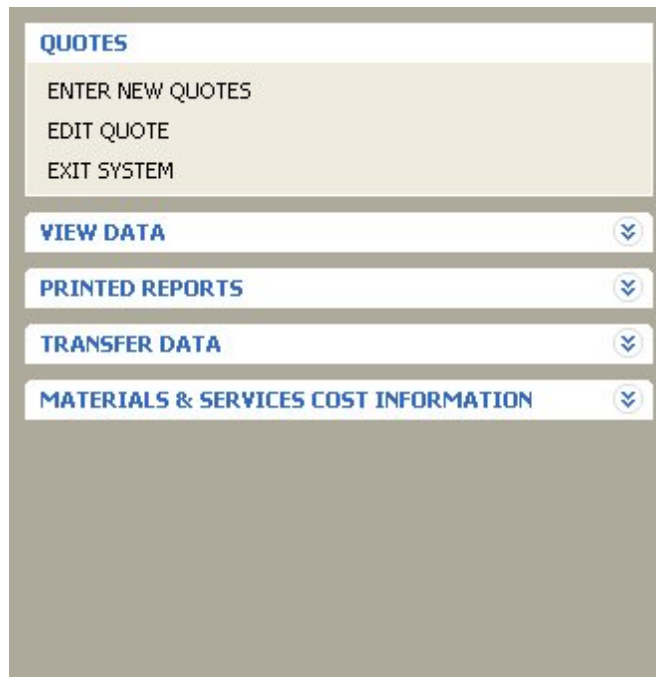
When a menu option is selected in a legacy application, R:BASE would have provided another menu (e.g. sub menu) to the user, or would have performed an action. The two type of menus that can be used include a "menu to a menu" or a "menu to an action."

Menu to a Menu

In some cases, a legacy menu selection was used to jump to another menu. This usually occurs in the first or second tier of a menu system. In the menu system below, the menu options with the red, orange, green, and blue lines were used in a DOS menu system to jump to another screen with additional options. The other three options "ENTER NEW QUOTES", "EDIT QUOTE", and "EXIT QUOTE FILES" do not.



Since the Windows menu system can display more information, these initial tier options with the red, orange, green, and blue lines can be removed, and the displayed Group Bar main group captions can be altered. The end result may look something like this.



Menu to an Action

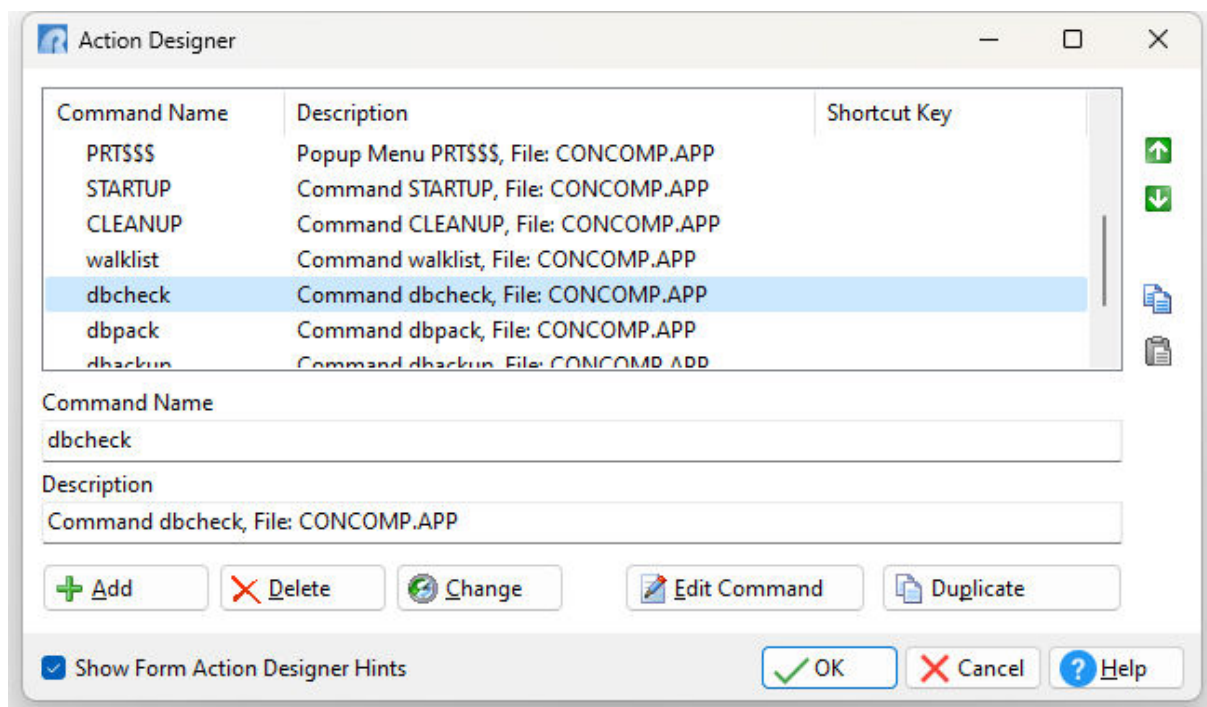
The menu to an action will run a series of commands, a Command Block, or an external file. The action can contain as little as one or two commands, or as much as hundreds of lines of code.

After the conversion, R:BASE loaded the command syntax portions of the .APP application file into the "Custom Command" or "Custom EEP" portions of the new menu system (e.g. Menu Bar, Group Bar, Tree View). Any [command syntax](#) loaded will highlight the "Edit Custom Command..." or "Edit Custom EEP" button with a yellow background. Selecting the button will display the stored command syntax. Before running the external form and selecting any menu options, the command syntax must first be checked.

4.1.1.3 Command Syntax Reference

After the conversion, R:BASE will also load the command block portions of the .APP application file into the external form as separate "Custom Form Actions". From the main menu bar, select "Layout" > "Custom Form Actions". The placement of the command syntax within the Custom Form Actions serves as a secondary reference for the command syntax that was used in the legacy application file. If the individual command blocks were already loaded into the appropriate menu system (Menu Bar, Group Bar, Tree View, Split View, Tile Menu) during the initial conversion process, the syntax here does not need to be updated in addition to the command syntax in the menu system.

The Action Designer, which is used to build and store a library of command blocks, will launch with the application file \$COMMAND blocks loaded separately as individual "Command Name" actions. The "Description" column will list the \$COMMAND block and originating application file. Within each of the listed items is a block of code. Select or hover the mouse cursor over an item will display a hint of the stored command syntax.



Select the "Edit Command" button to review the command syntax stored within the Custom Form Action.

4.1.2 Command Files

The application may be one large .APP file or several smaller files; usually with the .CMD file extension. The standard command file extension was .CMD in early R:BASE versions. It may also be possible that no file extension is used for the command files. In this case, it is recommended that the .RMD file extension be added to the file so it may be displayed within the "Command Files" menu within the Database Explorer's Navigator. Now, in R:BASE 11 for Windows, the new standard is .RMD as more file servers and mail servers flag the .CMD file extension as a possible virus. If you are using a custom file

extension, you can add the extension to the "File Mask:" field under the "Command Files" option and they will then appear in this window.

In order to use your command files in R:BASE 11, you will need to review and [update the command syntax](#) used in the files. Based on the R:BASE version you are upgrading from, there are additional steps needed to upgrade your command syntax. The older your command files are, then the more steps will be needed. And, if you are upgrading from a DOS version, there are additional changes required as well.

With command files as the primary storage format for the code, it is recommended that you open the R:BASE Editor Help file and review the many features available to add more productivity to the conversion process.

Also, in R:BASE 11, there are several different ways to run the menus contained within the files. Please review the several [Application Formats](#) available.

4.1.3 Updating the Command Syntax

At this point, it is necessary to review and update the command syntax so it can be used within R:BASE 11. If the syntax is used to RUN another file, then that file must also be checked and updated, if needed. Based upon the R:BASE version you are upgrading from, there are additional steps needed to update your application code. The older the application is, then the more steps will be needed. And, if you are upgrading from a DOS version, there are additional changes required as well.

You may encounter reserved words (PROJECT, TIME), obsolete commands (SET POINTER), and/or commands whose syntax has been enhanced for your benefit (PRINT, SELECT, CHOOSE). The time invested to perform these changes depends on the size of your command files.

There are two important tools included with R:BASE 11 that will make your command syntax changes more pleasant than you think.

The new R:BASE Editor & R:Style!

The R:BASE Editor is an editor specifically designed for creation and editing of R:BASE program and application files, command files, recorded scripts, and text files. It offers these special R:BASE features:

- You can specify settings such as the font, tab stops, and text wrapping. You can also search and replace text, and print files.
- Any number of program files can be open simultaneously.
- Command files are shown with syntax highlighting with the option of displaying each syntax element in a different color text.
- Safe tabs -- the Tab key inserts spaces.
- Block commands include block indent, block outdent, block comment out, and block comment in.
- Seamless integration with the R:Style, R:BASE code styling and checker.
- And more...

What is R:Style?

R:Style provides structural error checking, and spelling assistance to help debug your code prior to running it. R:Style will tell you where in the file to look, and what error to look for! R:Style is not a full syntax checker, but it is an excellent structure checker, with some syntax checking thrown in as a bonus. R:Style will find:

- Checks illegal or ambiguous commands
- Checks missing, unmatched, or incorrect quotes
- Checks unmatched parenthesis
- Finds missing RETURNS & ENDS, adds them
- Finds missing or out of place structures, IF, THEN, ELSE, ENDIF, SWITCH, ENDSW, WHILE, THEN, ENDWHILE, BREAK
- Duplicate block names in APP files
- Checks illegal use of command words in code or block names
- Checks ill advised RETURN from within WHILE or SWITCH
- Checks illegal dotted variables
- Checks dash comments extending beyond column 79 (breaks codelock)
- Support for finding misspelled or illegal words

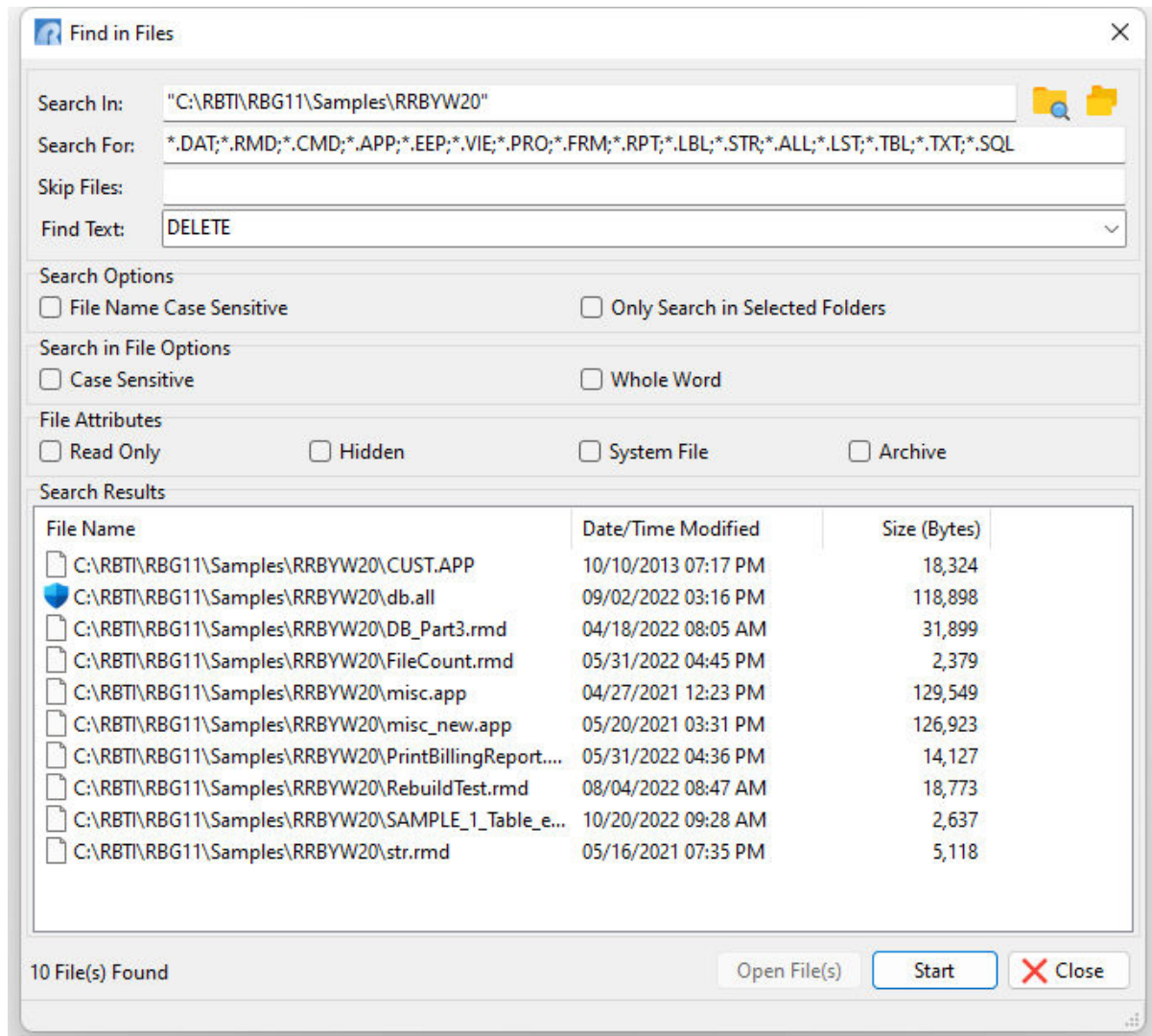
- Integration of RStyle* .PRE, .EXP, .CAS, .VAR, .NEW keyword formatting files into a built-in control panel
- Mismatched lines (= instead of +)
- And more...

4.1.3.1 Table and Column Name Changes

During the initial Preparation stages of the conversion process you were instructed to review change your table and column names which match any invalid names and [reserved words](#), and to record these changes on paper.

If you made any table and/or column name changes, the first step in your command syntax updates is to make these necessary name changes in the application/command files.

For command files, one time-saving suggestion for this task is to use the R:BASE built-in "Find Files" utility. To launch this command file utility, choose "Utilities" > "Find Files" from the main Menu bar. This interface will allow you to search your command files for obsolete commands and other words, so you are not forced to manually search the individual files.

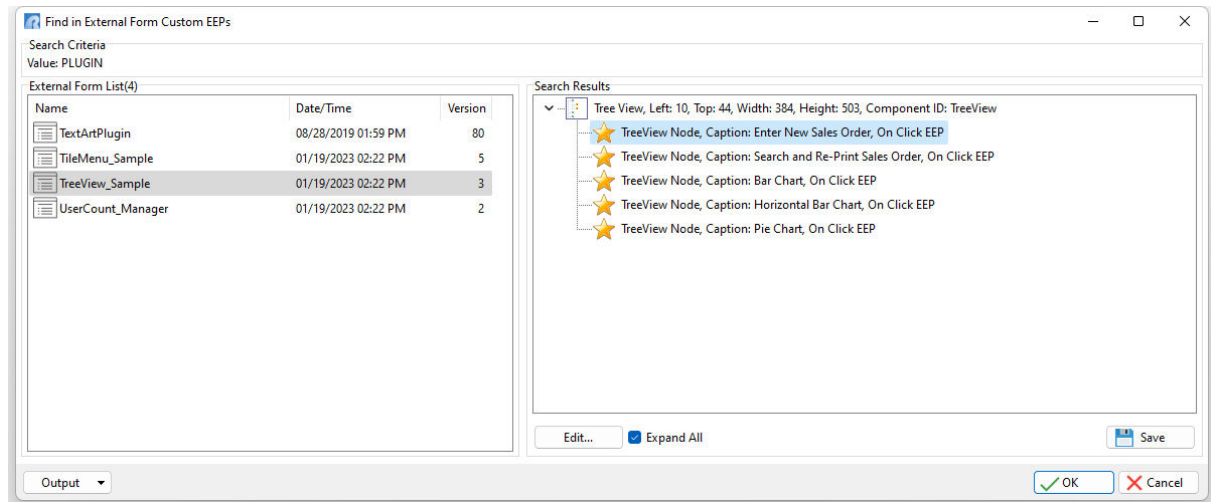


Keep in mind that whether you are using the R:BASE Editor or another text editor to make these command syntax updates, it is wise to take advantage of all of the utilities available in the text editor. In this case where you are replacing keywords, be sure to use any built-in word Search and Replace utilities to make this task easier.

For Custom Form Actions, a built-in search utility is available to search all of the code within the Actions. From within the External Form Designer, select "Edit" > "Find in Custom EEPs" from the main menu bar to search for text within Custom EEPs for the external form. The key combination [Ctrl]+[F] can also be used used to launch the "Find in Custom EEPs" dialog.



After the search is complete, a list of external form objects and custom form actions that match the search will be displayed. The external form object(s) will contain the exact location where the Custom EEP is in use for the object. The command syntax can be launched in the R:BASE Editor by double clicking on it, or by selecting the "Edit..." button. The search results can be saved to several output formats. The "Expand All" check box allows the list of EEPs in the search results to expand for full viewing.



4.1.3.2 Obsolete Commands

The following table contains a list of commands that are considered to be obsolete. Most of these commands have been made obsolete by their SQL Compliant equivalent, and there is no guarantee that these commands will be supported in future versions of R:BASE. Any code which relies on any of these commands should be considered dangerous as their use may have unintended and unpredictable results.

While most items have a direct replacement there are some items that are listed as having no direct replacement. However it may be possible to replace those commands with other commands that emulate that functionality. For more information on the "Replacement" commands, refer to the R:BASE Help "Command Index", or press [F1] from any location in R:BASE.

One suggestion for task this task is to use the R:BASE built-in "Find in Files" utility. To launch this command file utility, choose "Utilities" > "Find in Files" from the main Menu bar. This interface will allow you to search your command files for obsolete commands and other words, so you are not forced to manually search the individual files.

Obsolete Command	Replacement	Notes
#FORM_COLUMNNAME	RBTI_FORM_COLNAME	See also RBTI System Variables
#FORM_FORMNAME	RBTI_FORM_FORMNAME	See also RBTI System Variables
#FORM_TABLENAME	RBTI_FORM_TBLNAME	See also RBTI System Variables
AVE()	LAVG	
BUILD KEY	CREATE INDEX	
CHANGE	UPDATE	
CHANGE COLUMN	ALTER TABLE	
CLOSE	DISCONNECT	
COLUMNS	CREATE TABLE	
(CVAL('FORM_ALIAS'))	RBTI_FORM_ALIAS	See also RBTI System Variables
(CVAL('FORM_CAPTION'))	GETPROPERTY RBASE_FORM CAPTION vCaption	
(CVAL('FORM_DATA'))	RBTI_FORM_COLVALUE	See also RBTI System Variables
(CVAL('FORM_DATA_TYPE'))	RBTI_FORM_DATATYPE	See also RBTI System Variables
(CVAL('FORM_FIELD_NAME'))	RBTI_FORM_COLNAME	See also RBTI System Variables
(CVAL('FORM_NAME'))	RBTI_FORM_FORMNAME	See also RBTI System Variables
(CVAL('FORM_TABLE'))	RBTI_FORM_TABLENAME	See also RBTI System Variables
DEFINE	CREATE SCHEMA	
DELETE KEY	DROP INDEX	
DELETE FROM #n	DELETE FROM TableName WHERE CURRENT OF CURSOR	See also DECLARE CURSOR
DISPLAY	EDIT USING	Used with Variable Forms.
DRAW	NONE	Used with Variable Forms. No direct replacement.
ECHO	SET ECHO ON	
EDIT VAR ...	NONE	Used with Variable Forms. No direct replacement.
ENTER VAR ...	NONE	Used with Variable Forms. No direct replacement.
ESC	[ESC]	Used in IF and WHERE clauses to compare to the Esc key.
Value1 EQ Value2	Value1 = Value2	Value1 equals Value2
EXISTS	IS NOT NULL	
EXPAND	ALTER TABLE	
EXPRESS	RBAPP	
FAILS	IS NULL	

FILLIN	DIALOG	In windows consider using DIALOG, ENTER or EDIT.
Value1 GE Value2	Value1 >= Value2	Value1 greater than or equal to Value2.
Value1 GT Value2	Value1 > Value2	Value1 greater than Value2.
IN #n	WHERE CURRENT OF CURSOR	Used with UPDATE and SET VAR commands.
Value1 LE Value2	Value1 <= Value2	Value1 less than or equal to Value2.
Value1 LT Value2	Value1 < Value2	Value1 less than or equal to Value2.
MAX()	LMAX()	
MICRORIM_RETURN	STP_RETURN	
MICRORIM_WALLPAPER	NONE	This variable was only used with Forms and Reports created for R:WEB.
MIN()	LMIN()	
Value1 NE Value2	Value1 <> Value2	Value1 does not equal Value2.
NEXT	FETCH	See also DECLARE CURSOR
NOECHO	SET ECHO OFF	
OPEN	CONNECT	
OWNER	CREATE SCHEMA	
PASSWORDS	GRANT	
PLAYBACK	NONE	No Replacement.
POPUP/PULLDOWN Menus	NONE	Use #LIST option of CHOOSE command. Use LISTOF command option to create list of items.
PROMPTS		Now DOS Only.
RECORD	NONE	
REDEFINE	ALTER TABLE	
REMOVE	DROP	
RESET	NONE	No Replacement.
#RETURN	NONE	Used with Variable Forms. No Direct Replacement.
SCREEN RESTORE	RECALC VARIABLES	Now DOS Only.
SELECT FROM #n	SELECT FROM TableName	See also DECLARE CURSOR WHERE CURRENT OF cursor
SET CGA	NONE	No Replacement.
SET EXTENDED	NONE	R:BASE always operates in Extended Mode now.
SET INTENSITY	NONE	Use "Selected Row" color option in Scrolling Region Properties.
SET OLDLINE	NONE	Use the Page Style option in Reports.
SET POINTER	DECLARE CURSOR	

SET RBGSIZE		No replacement necessary due to imbedded form properties. Command is ignored.
SET USERAPP	NONE	No replacement.
SNAP	NONE	No replacement.
SORTED BY	ORDER BY	
USER	CONNECT IDENTIFIED BY ...	
VIEW	CREATE VIEW	
WRITE		In Windows consider using DIALOG, PAUSE, ENTER or EDIT.
ZIP RBEDIT	RBEDIT	RBEDIT is no longer a separate executable.
ZIP RBSYNC	RBSYNC	RBSYNC is no longer a separate executable.

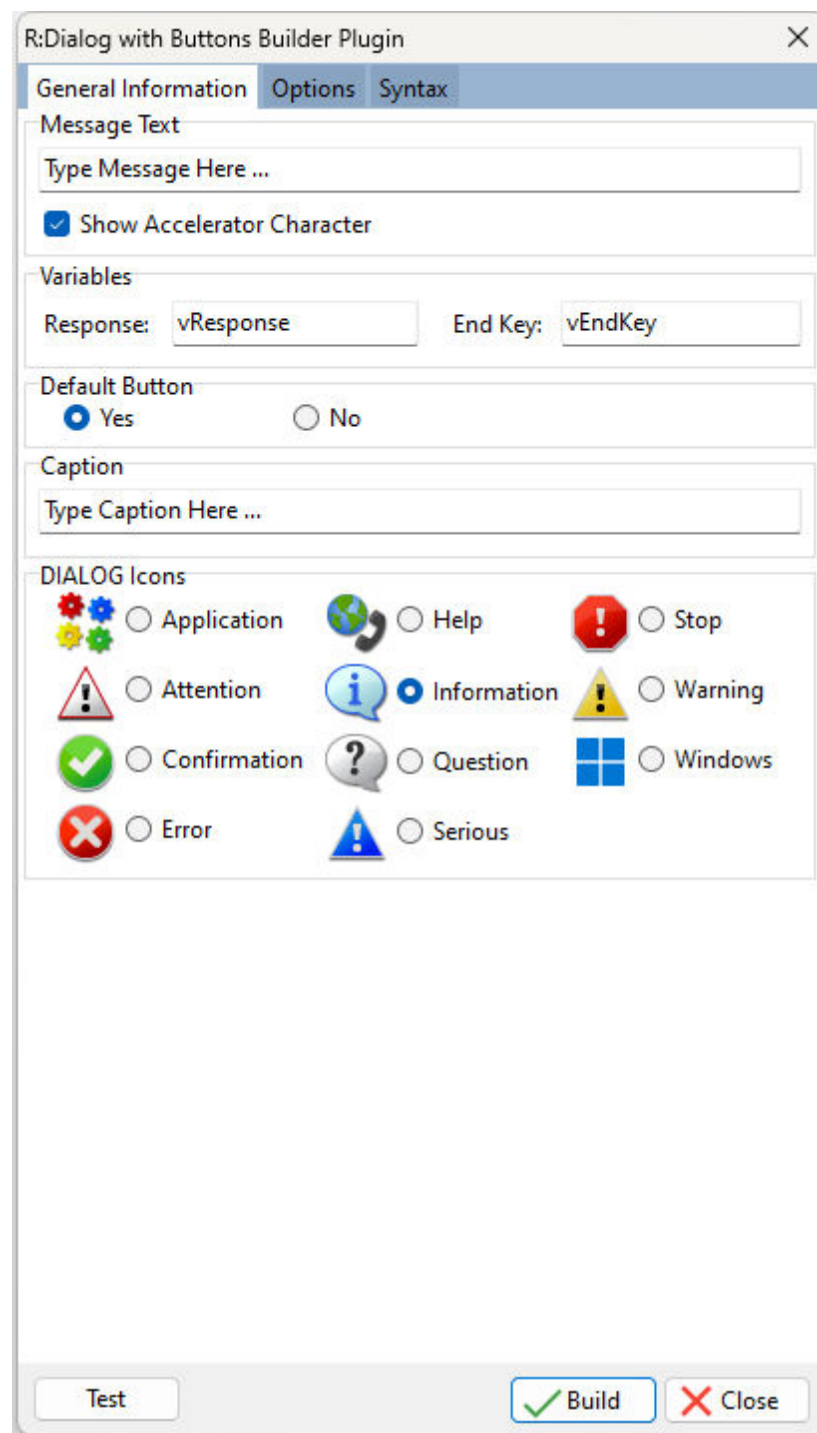
4.1.3.3 DOS to Windows Conversion

When converting from DOS to Windows, there are command syntax changes that may or may not require your attention.

Instances where you will positively update the command syntax are any locations where DOS message commands (WRITE) or information prompts (FILLIN) which should be replaced by their Windows counterpart (PAUSE and DIALOG).

These commands send simple messages or prompts for data entry to the end user. Your time involved in this step depends on how much user interaction is in the code.

To assist you with these replacements, R:BASE 11 for Windows includes several syntax builders for the DIALOG, PAUSE, and CHOOSE commands. They can be accessed from the menu bar under "Utilities" > "Plugins".



For **WRITE**, you may have this;

```
WRITE 'THE PROCESS IS COMPLETE!' AT 10 05
```

and will replace it with this:

```
PAUSE 2 USING 'THE PROCESS IS COMPLETE!' CAPTION ' ' ICON INFO
```

For **FILLIN**, you may have this;

```
FILLIN vdte USING 'ENTER DEADLINE DATE: ' AT 12 10
```

and will replace it with this:

```
DIALOG 'ENTER DEADLINE DATE:' vdte venke 1 CAPTION 'Question'
ICON QUESTION
```

The PAUSE and DIALOG samples provided above do not include the many options that are available for each command to professionally display a message on the screen. Please review the "All About The Dialog Command" and "All About The Pause Command" PDF documents for the complete syntax available for each command. The PDF documents are located within the default program directory of R:BASE 11 (C:\RBTI\RBG11).

4.1.3.4 Enhanced Commands

There will be commands whose syntax has been enhanced for your benefit. Most of these commands will not require mandatory changes other than for an enhanced display (CHOOSE, DIALOG, PAUSE), but the PRINT command is an example of a command that has been enhanced 20 times over, and will require you to alter your syntax.

You may have this:

```
OUT PRINTER
PRINT bulletin SORTED BY poscl
OUT screen
```

and will replace it with this:

```
PRINT bulletin ORDER BY poscl ASC OPTION PRINTER
```

or even this:

```
PRINT bulletin ORDER BY poscl ASC OPTION PRINTER +
|SHOW_CANCEL_DIALOG OFF |TRAY 2 |COLLATION ON +
|COPIES 2 |ORIENTATION LANDSCAPE |PRINTER_NAME HPLaser1345
```

4.1.3.5 Unsupported Command Parameters

The application file may have \$MENU blocks where the CHOOSE command is used to display a menu. The CHOOSE command "IN MENU.APX" is no longer supported for "PULLDOWN" menu blocks.

If the approach of the application file conversion is to run the .APP file as a command file, then the following method can be used to move the "menu list" into the CHOOSE command with the #LIST parameter.

Instead of the following:

```
CHOOSE PICK1 FROM MENU0000 IN Menu.APX
--
-- rest of file
--
$MENU
MENU0000
PULLDOWN | |
```

```
|Entry|
|Product Quality Plan|
|Inquiry|
|Reports|
|Maintenance|
ENDC
```

You would use:

```
CHOOSE PICK1 FROM #LIST 'Entry,Product Quality Plan,Inquiry,Reports,Maintenance' +
TITLE 'SELECT OPTION...' CAPTION 'Main Menu Options' LINES 5 +
OPTION List_Font_Color NAVY|List_Back_Color WHITE|List_Font_Size 12 +
|Title_Font_Color NAVY|Title_Back_Color LIGHT GRAY +
|Window_Back_Color LIGHT GRAY|Title_Font_Size 14
```

The above will display a choose pop-up window to select the desired option.

Although this will allow the application file to work, we recommend using one of the current [application formats](#) to display a Windows driven menu system.

4.1.4 New Application Formats

When building R:BASE databases and applications, R:BASE users have several options as to how they want to store their "application".

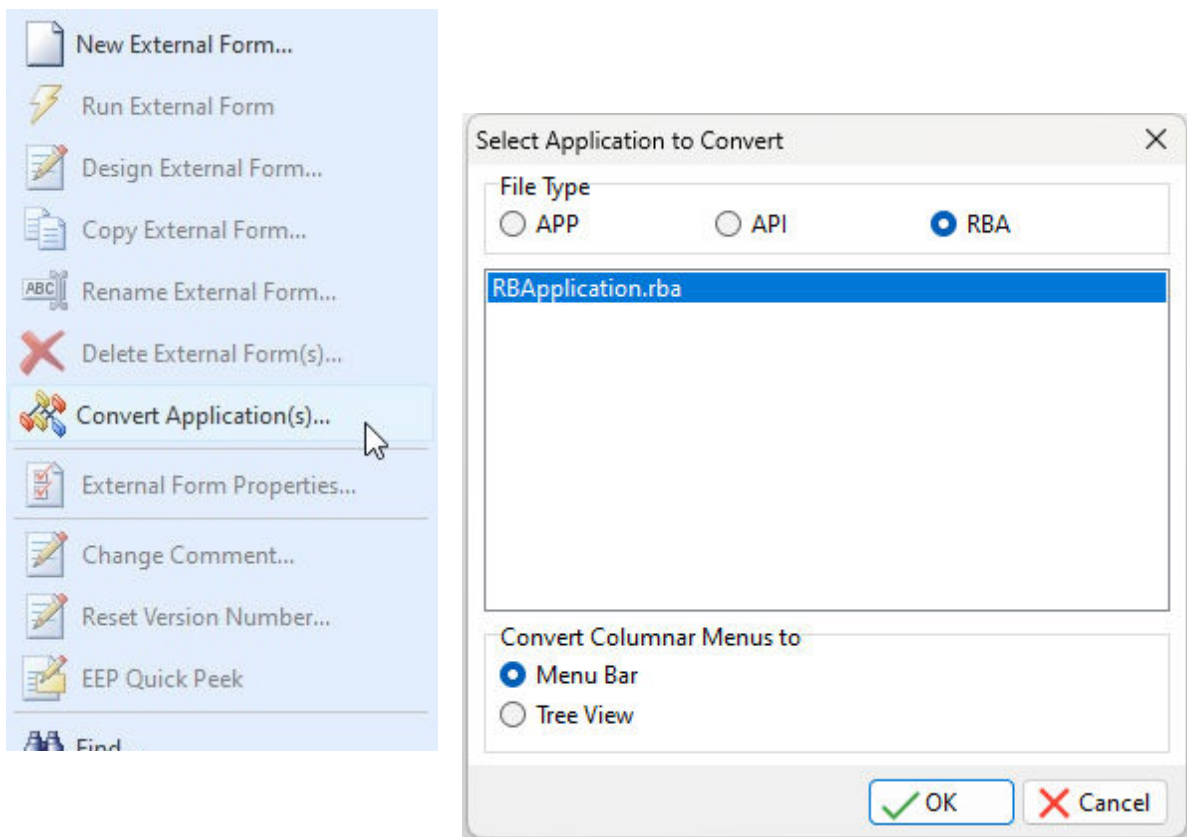
1. **Forms** - an application can be completely Form driven by storing all of the code in form EEPs.
2. **Application Builder** - using the Application Builder interface, an application can be stored in a .RBA file, separate from the database files. This offers additional security over using just command files.
3. **Command Files** - using only code (e.g. CHOOSE, DIALOG, PAUSE, etc), an application can be entirely written in one or many RMD, APP, CMD, and DAT files.
4. **External Form Files** - an application can be driven using External Forms, by storing all of the code in form EEPs. Like an R:BASE Application file (.RBA), External Form Files are stored outside of the database.
5. **All, or some, of the above** - an application can consist of all of a few of the above methods to store R:BASE menu and command files.

4.2 R:BASE 7.x/Turbo V-8 Applications (.RBA)

Any .RBA Application files, Form-Driven Applications, External Form Files, or command files running in R:BASE 7.x or Turbo V-8 will run in R:BASE 11 with little or no changes. The R:BASE 7.x versions include 7.0, 7.1, 7.5 and 7.6.

It is recommended that you review your application menus and actions to ensure the system is 100% operational.

Also, within the External Form File portion of the Database Explorer, the new "Convert Application(s)..." option is available to convert R:BASE application files (.RBA) to an External Form File.



Selecting this option will create an external form that matches the same properties used within the RBA application file. For the "Menu Bar" layout, the "Main Menu" section is converted to a form "Menu Bar", the "Actions" are converted to "Custom Form Actions", and the "Toolbar" is converted to a toolbar in the "Toolbar Designer" when the external form file is created. A "Tree View" control is also available as a menu display method, where menu options are loaded into a hierarchical list with "Actions" placed and referenced into Custom EEPs and Custom Form Actions.

Plugins

When upgrading from 7.6/V-8 to R:BASE 11, you will also need to upgrade any Plugins, if used. The older .RBL Plugin files cannot be used in R:BASE 11 as a different plugin type and file extension (.RBM) is used.

The free Plugins that are shipped with R:BASE are contained within the R:BASE 11 executable. Initially, you will not see any Plugins in the R:BASE program folder. The Plugins contained within the R:BASE executable include:

- LoadDirectoryName
- LoadFileName
- LoadFileNamePlus
- RCalculator

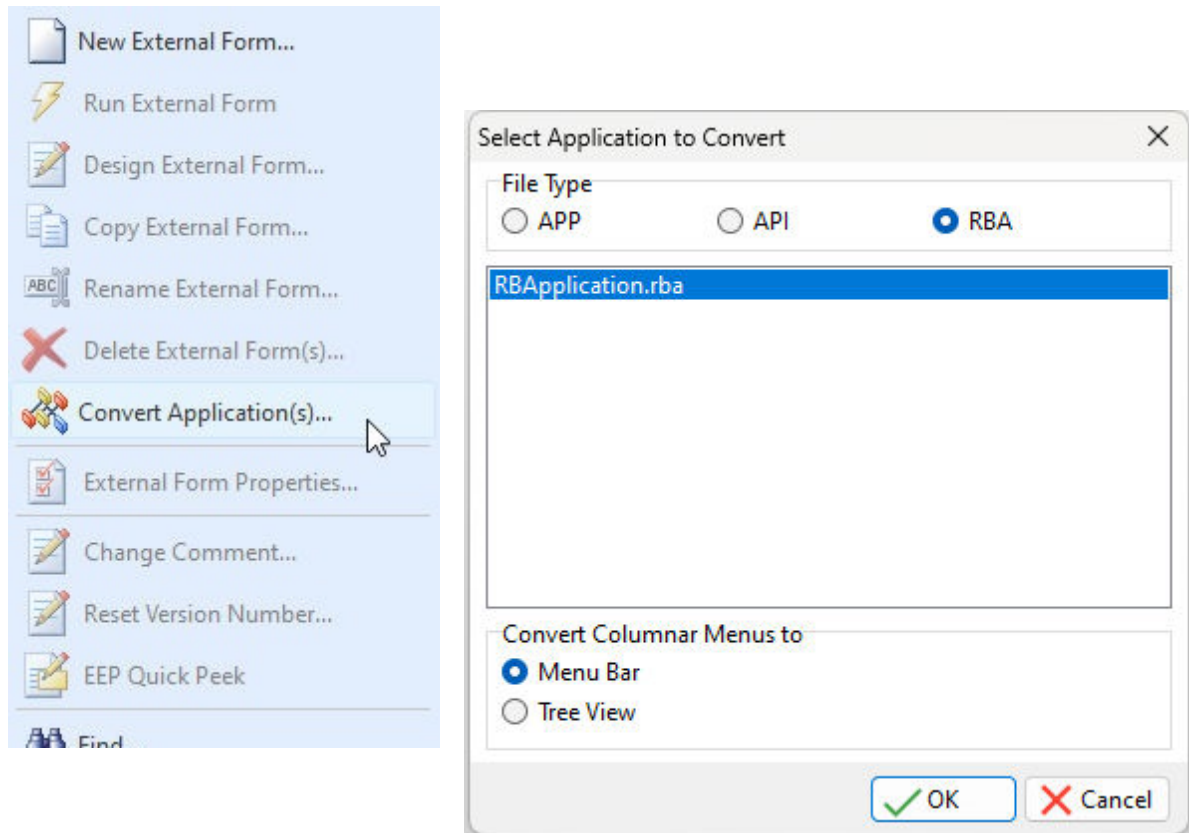
When you run the PLUGIN command with your R:BASE program, the window will display appropriately.

Please refer to the PLUGIN section within the Command Index to verify the supported syntax.

4.3 R:BASE eXtreme 9.x Applications

Any .RBA Application files, Form-Driven Applications, External Form Files, or command files running in R:BASE eXtreme will run in R:BASE 11 with no changes required on your behalf. It is recommended that you review your application menus and actions to ensure the system is 100% operational.

Within the External Form File portion of the Database Explorer, the new "Convert Application(s)..." option is available to convert R:BASE application files (.RBA) to an External Form File.



Selecting this option will create an external form that matches the same properties used within the RBA application file. For the "Menu Bar" layout, the "Main Menu" section is converted to a form "Menu Bar", the "Actions" are converted to "Custom Form Actions", and the "Toolbar" is converted to a toolbar in the "Toolbar Designer" when the external form file is created. A "Tree View" control is also available as a menu display method, where menu options are loaded into a hierarchical list with "Actions" placed and referenced into Custom EEPs and Custom Form Actions.

Plugins

When upgrading from R:BASE eXtreme 9.x to R:BASE 11, you will also need to upgrade any Plugins, if used. The 9.x specific Plugin files cannot be used in R:BASE 11.

4.4 PAGELOCK Setting

R:BASE 10.5 and higher versions include a new (actually separate) PAGELOCK operating condition, which specifies how R:BASE locks data in a multi-user environment, based on the position of the row in the data file?

Previously, the R:BASE operating condition was controlled by QUALCOLS as a dual-functioning setting, specifying unique columns for ODBC tables and for page locking. Now page locking (PAGELOCK) is separate from ODBC unique columns (QUALCOLS).

What does this mean for R:BASE users?

The application code must be updated to add PAGELOCK and set the condition to ON/OFF appropriately where QUALCOLS was used for page locking. Examples below demonstrate the setting could also be used dynamically in case-by-case instances to increase multi-user performance.

To summarize the code changes, if you used SET QUALCOLS 10 for faster page locking, add SET PAGELOCK OFF to your code. If you used SET QUALCOLS 2 to enforce a page lock for a faster UPDATE/DELETE process, add SET PAGELOCK ON in your code.

PAGELOCK specifies how R:BASE locks data when updating and deleting rows.

The settings for PAGELOCK are:

ON - R:BASE uses page locking or row locking as appropriate. When PAGELOCK is ON and two or more users are updating rows within the same page of data, R:BASE only lets the first user update rows--the other users are locked out until the first user's update has been completed.

OFF - R:BASE uses a fast row-locking method where only row locking is applied with no page locking. When PAGELOCK is OFF, you can lock rows of data instead of locking a page of data. You increase multi-user performance when PAGELOCK is OFF. And even more so when STATICDB and FASTLOCK are on.

If you know that your application mainly updates or deletes data a row at a time, rather than many rows, set PAGELOCK to OFF for row locking. In this case, R:BASE locks a row, reads the row, makes the change, and then releases the row. Otherwise, set PAGELOCK ON for page locking when you are doing an UPDATE and/or DELETE affecting many rows in a table.

Keep in mind that the PAGELOCK setting can be changed dynamically and can be different for different users using the same database.

Technically, the most efficient and fastest method for updating data in multi-user environment is to SET STATICDB ON, SET FASTLOCK ON, and SET PAGELOCK OFF. This particular combination will result in the fewest contentions between users.

Notes:

- FASTLOCK and PAGELOCK can be set on at the same time.
- Setting STATICDB and FASTLOCK to ON (in that order), with PAGELOCK set to OFF will significantly increase multi-user performance with individual row changes.
- PAGELOCK is not the same as SET ROWLOCKS.
- Setting the value of PAGELOCK does not change the setting of ROWLOCKS.
- The PAGELOCK setting can be changed dynamically and can be different for different users using the same database.

Example:

```
-- The UPDATE must alter the values for many rows
SET FEEDBACK ON
SET PAGELOCK ON -- use page locking
UPDATE <tablename> SET <columnname> = value -- no WHERE Clause
SET PAGELOCK OFF -- use row locking
SET FEEDBACK OFF
CLS
```

Part

V

5 ODBC Compliance

If you are planning on using ODBC with R:BASE, you must make sure that your character settings are set correctly by performing the following:

1. Go to the R> Prompt
2. Connect to your database
3. Type: `SHOW CHAR`
4. Compare your values to the chart below

<u>Character</u>	<u>Set To</u>	<u>Description</u>
MANY	%	Percent Sign
SINGLE	_	Underscore
QUOTES	'	Single Quote

5. If they do not match, change them by typing `SET TYPE=CHAR`
(where TYPE is from the "Character" column and CHAR from the "Set To" column)

For example, `SET MANY=%`

Continue below to ensure your database files are [Y2K](#) compliant.

Part

VI

6 Y2K Compliance

For more information on Y2K Compliance please visit our Y2K Page at <http://www.rbase.com/support/y2k/>

To check the Y2K compliance of your database do the following.

1. Go to an R> Prompt
2. Connect to your Database
3. Type SHOW DATE

Most likely the settings will be as follows... (This is the default for upgraded databases).

```
DATE FORMAT MM/DD/YY
DATE SEQUENCE MDDYY
DATE CENTURY 19
DATE YEAR 0
```

If so, then your database is NOT Y2K compliant. To activate the Y2K compliancy features of your database we recommend the following:

Set your Format to 4 Y's

When they are set that way then 1/1/1999 will be seen as a valid year and 1/1/99 will also be a valid year. If you are set to 2 Y's then only 1/1/99 will be a valid year.

Set the Date Year to a Convenient Year

The Date Century and Date Year determine how 2 digit years such as 1/1/99 are handled. To determine your settings mentally combine the century and the year (in this case 19 and 0 make 1900) and then add 99 (in this case 1999) all two digit years will be interpreted as being between these dates. Setting Date Year to 80 will result in 19 + 80 giving 1980 and then adding 99 making 2079. Thus all two digit dates will be between 1980 and 2079. For example a date entered as 1/1/60 would be seen as 2060.

Older versions set to four Y's would see 2 digit dates as being complete four digit years. For example, 1/1/99 would be January First in the year 99. This is known as First Century data. For this reason if your database is already set to four Y's on the format we recommend using the following command to check for invalid dates:

```
TALLY datecol FROM tablename WHERE datecol < 1/1/1950
```

This should build a list of all dates (and how often they occur) that are possibly incorrectly entered and that you can then deal with. Be careful not to go overboard. For example: if you have your family tree stored in an R:BASE Database then be careful that you leave Great Grandmother's birthday in the 1800's where it is supposed to be even though the Tally command above would flag that as an "invalid" date.

To resolve data issues where date values are listed as 1900, 1901, etc., where the expected value is 2000, 2001, etc., please continue.

1. Disconnect and make a backup copy of your database.
2. Start R:BASE and connect to the database.
3. At the R> prompt, type:

```
UPDATE TableName SET DateColumnName = (ADDYR(DateColumnName,100)) +
WHERE (IYR(DateColumnName)) IN (1900, 1901, 1902, 1903, 1904, 1905, 1906, 1907,
1908, 1909, 1910)
```

Where:

"TableName" is the name of table

"DateColumnName" is the name of date column in that table

Part

VII

7 Database Integrity Routine

Follow the steps below and to validate the integrity of your database.

1. Make a backup copy of the database.
2. Copy the entire database to your "local drive" in a separate folder with at least twice as much available free disk space.
3. Start R:BASE using the latest version/update and switch the current folder to the appropriate database folder on your local drive.
4. At the R>, connect to the database:

```
DISCONNECT
SET MESSAGES ON
SET ERROR MESSAGES ON
SET MULTI OFF

CONNECT dbname
or connect to the database with OWNER password, if any
CONNECT dbname IDENTIFIED BY owner password
```

5. Check the connected database CHARacter settings, especially the IDQUOTES. At the R> prompt:

```
CLS
SHOW CHAR
```

Notice the setting for IDQUOTES (the last item on list).

If this setting is blank/null, make sure to set the IDQUOTES settings to ` (that is a single reversed quote, on the same key as the ~ tilde). At the R>, type:

```
SET IDQUOTES=`
```

6. Now create the unload file with NULL set to -0-. At the R> prompt:

```
CLS
SET NULL -0-
OUTPUT newdb.ALL
UNLOAD ALL
OUTPUT SCREEN
```

This step will create two files (newdb.ALL and newdb.LOB)

7. DISCONNECT, and then rename this database to some other name. At the R> prompt:

```
DISCONNECT

RENAME dbname.RX? dbnameBK.RX?
or for RB1 through RB4 files, use:
RENAME dbname.RB? dbnameBK.RB?
```

8. Now, let's rebuild the fresh database and see if it passes the integrity check. At the R> prompt:

```
CLS
RUN newdb.ALL
```

Watch the activity and all messages on the screen. Don't fall asleep. You might miss an important warning. Completion time may vary based on size of the database (number of tables/records/indexes)

If there were no warnings or error messages, you've got a fresh/healthy database.

Errors?

1. If there are any warning or -ERROR- message(s), take them seriously. In that case, disconnect and delete the bad database (built using RUN newdb.ALL). At the R> prompt:

```
DISCONNECT
```

```
DELETE dbname.RX?
```

or for RB1 through RB4 files, use:

```
DELETE dbname.RB?
```

```
DELETE newdb.ALL
```

```
DELETE newdb.LOB
```

2. Rename and connect to the previously saved database (Step 7).

```
RENAME dbnameBK.RX? dbname.RX?
```

or for RB1 through RB4 files, use:

```
RENAME dbname.RB? dbnameBK.RB?
```

```
CONNECT dbname
```

3. CORRECT all -ERROR-s accordingly and then repeat Step 6.
4. Do not quit or give up until you see a completely fresh database without any warnings or errors.

If there is a considerable amount of errors with both rebuilding the structure and/or data, please refer to the "Database Repair Routine" within the Database Maintenance PDF document in the R:BASE program directory.

Part

VIII

8 Useful Resources

- . R:BASE Home Page: <https://www.rbase.com>
- . Up-to-Date R:BASE Updates: <https://www.rbaseupdates.com>
- . Current Product Details and Documentation: <https://www.rbase.com/rbg11>
- . Support Home Page: <https://www.rbase.com/support>
- . Product Registration: <https://www.rbase.com/register>
- . Official R:BASE Facebook Page: <https://www.facebook.com/rbase>
- . Sample Applications: <https://www.razzak.com/sampleapplications>
- . Technical Documents (From the Edge): <https://www.razzak.com/fte>
- . Education and Training: <https://www.rbase.com/training>
- . Product News: <https://www.rbase.com/news>
- . Upcoming Events: <https://www.rbase.com/events>
- . R:BASE Online Help Manual: <https://www.rbase.com/support/rsyntax>
- . Form Properties Documentation: <https://www.rbase.com/support/FormProperties.pdf>
- . R:BASE Beginners Tutorial: <https://www.rbase.com/support/rtutorial>
- . R:BASE Solutions (Vertical Market Applications): <https://www.rbase.com/products/rbasesolutions>

Part

IX

9 Feedback

Suggestions and Enhancement Requests:

From time to time, everyone comes up with an idea for something they'd like a software product to do differently.

If you come across an idea that you think might make a nice enhancement, your input is always welcome.

Please submit your suggestion and/or enhancement request to the R:BASE Developers' Corner Crew (R:DCC) and describe what you think might make an ideal enhancement. In R:BASE, the R:DCC Client is fully integrated to communicate with the R:BASE development team. From the main menu bar, choose "Help" > "R:DCC Client". If you do not have a login profile, select "New User" to create one.

If you have a sample you wish to provide, have the files prepared within a zip archive prior to initiating the request. You will be prompted to upload any attachments during the submission process.

Unless additional information is needed, you will not receive a direct response. You can periodically check the status of your submitted enhancement request.

If you are experiencing any difficulties with the R:DCC Client, please send an e-mail to rdcc@rbase.com.

Reporting Bugs:

If you experience something you think might be a bug, please report it to the R:BASE Developers' Corner Crew. In R:BASE, the R:DCC Client is fully integrated to communicate with the R:BASE development team. From the main menu bar, choose "Help" > "R:DCC Client". If you do not have a login profile, select "New User" to create one.

You will need to describe:

- What you did, what happened, and what you expected to happen
- The product version and build
- Any error message displayed
- The operating system in use
- Anything else you think might be relevant

If you have a sample you wish to provide, have the files prepared within a zip archive prior to initiating the bug report. You will be prompted to upload any attachments during the submission process.

Unless additional information is needed, you will not receive a direct response. You can periodically check the status of your submitted bug.

If you are experiencing any difficulties with the R:DCC Client, please send an e-mail to rdcc@rbase.com.

Index

- # -

#DATE 34, 39
#PAGE 34, 39
#TIME 34, 39

- \$ -

\$COMMAND 62

- A -

Action Designer 62
Advanced DB Rich Edit 42
aggregate variables 34, 39
API 47
APP 47, 48, 71, 72
application builder 72
application express 47
application file 48
applications 47
APW 47
APX 47

- B -

backup 7

- C -

Calendar 42
CHOOSE 63, 71
CMD 47, 62, 72
column 7
command block 47
command blocks 62
command files 47
Command Name 62
Command Syntax 62, 63
commands 65
connect 18, 24
control codes 34, 39

Conversion 42
convert 12, 24, 31, 37, 42, 47
Convert to 42
copy 7
Custom Command 51, 60
Custom EEP 48, 49, 53, 54, 57, 60
Custom Form Actions 62

- D -

DAT 47, 72
DATE 79
DB Calendar 42
DB Grid 42
DB Navigator 42
DB Rich Edit 42
DBCcalc 34, 39
DIALOG 69, 71
DOS 24, 31, 37

- E -

EEP 47, 62
Enhanced Calendar 42
Enhanced DB Calendar 42
Enhanced DB Grid 42
Enhanced DB Navigator 42
Enhanced Group Box 42
Enhanced Panel 42
Enhanced Speed Button 42
Enhanced Tab Control 42
Enhanced Variable Calendar 42
Express 47
External Form 72
External Form File 42
external form files 47

- F -

feedback 87
FILLIN 69
Find Files 64
Find in Files 65
flow chart 11
Form 72
Form Designer 29
forms 24

- G -

Group 49
Group Bar 48, 49
Group Box 42

- H -

hint 62

- I -

IDQUOTES 18, 77
integrity 82
Items 49

- L -

labels 24, 37
landscape 34, 39
lookup variables 29, 34, 39

- M -

MANY 77
margins 34, 39
Menu Action Types 60
Menu Bar 48, 51
menu block 47, 51, 71
Menu Item 54
Menu System Types 48
Menu to a Menu 60
Menu to an Action 60
MULTI 18
Multi-User 74

- N -

New Conversion Enhancements 42
NEWDB 13
Nodes 53
NULL 82

- O -

obsolete 65
ODBC 7, 77
On Before Design Action 29
Owner 18

- P -

Page Locking 74
page setup 34, 39
page size 34, 39
PAGELOCK 74
Panel 42
password 18
PAUSE 69, 71
PCC Label 34, 39
pop-up 48
portrait 34, 39
prepare 24
preview 34, 39
PRINT 63, 71
printer codes 34, 39
PROPERTY 62
pull down 48
push button 29

- Q -

QUOTES 77

- R -

R:BASE Editor 63
R:Style 63
RB1 11
RB2 11
RB3 11
RB4 11
RBA 47, 72
RBEDIT 82
RBF 11
RBS 11
RELOAD 18
RENAME 82
reports 24, 31

reserved 7
reserved words 7, 64
RMD 47, 62, 72
RUN 82
RX1 11
RX2 11
RX3 11
RX4 11

- S -

Save as External Form File 42
screen block 47
SELECT 63
SET VAR 29
SINGLE 77
Speed Button 42
Split View 54
STATICDB 18
Sub-Items 53
system tables 7

- T -

Tab Control 29, 42
table 7
Tile Menu 57
Tree View 48, 53
TURBO 18

- U -

UNLOAD 82
unsupported 65
update the command syntax 62
Updating the Command Syntax 63
upgrade 24
user 18
utility 64, 65

- V -

Variable Calendar 42
Variable Form 42
variables 29, 34, 39

- W -

width 34, 39
WRITE 69

- Y -

Y2K 79

- Z -

ZERO 34, 39

Notes