



R:BASE Technologies, Inc.
3935 Old William Penn Highway
Murrysville, PA 15668-1854 USA
1+724.733.0053 Phone
1+724.733.0196 Fax
<http://www.rbase.com>

Wednesday, September 26, 2007

What's New in Oterro[®] 3.5

Oterro 3.5 Typical Installation Setup and Files

Directory:

C:\RBTI\Oterro35

C:\Windows or C:\WinNT

C:\Windows\System32 or C:\WinNT\System32

ODBC Data Source Administrator Driver

Files:

BluzVan.RB1
BluzVan.RB2
BluzVan.RB3
BluzVan.RB4
CheckVer.exe
License.rtf
License.txt
Oterro35.ico
ReadMe.txt
Version.txt
WhatsNewInOterro35.pdf

OTERRO35.CFG

OT35IO.DLL
OTERRO35.DLL
OTERRO35_INS.DLL
RBEXT.DLL

Oterro 3.5 Database Driver (*.rb1)

Oterro 3.5 Features and Enhancements

01. ODBC 3.0 Supported Functions:

Core Functions:

SQLAllocConnect
SQLAllocEnv
SQLAllocStmt
SQLBindCol
SQLCancel
SQLColAttributes
SQLConnect
SQLDescribeCol
SQLDisconnect
SQLError
SQLExecDirect
SQLExecute
SQLFetch
SQLFreeConnect
SQLFreeEnv
SQLFreeStmt
SQLGetCursorName
SQLNumResultCols
SQLPrepare
SQLRowCount
SQLSetConnectAttr
SQLSetCursorName
SQLSetParam
SQLTransact

Level I Functions:

SQLColumns
SQLDriverConnect
SQLGetConnectOption
SQLGetData
SQLGetFunctions
SQLGetInfo
SQLGetStmtOption
SQLGetTypeInfo
SQLParamData
SQLPutData
SQLSetConnectOption
SQLSetStmtOption
SQLSpecialColumns
SQLStatistics
SQLTables

Level II Functions:

SQLBrowseConnect
SQLColumnPrivileges
SQLDataSources
SQLDescribeParam
SQLExtendedFetch
SQLForeignKeys
SQLMoreResults
SQLNumParams
SQLNativeSql
SQLParamOptions
SQLPrimaryKeys
SQLProcedureColumns
SQLProcedures
SQLSetPos
SQLSetScrollOptions
SQLTablePrivileges
SQLDrivers
SQLBindParameter

Miscellaneous Functions:

DCDIALOGLOOP
DIIEntryPoint
SQLBindParameter2

R:BASE Functions

RBApiCreateVar
RBApiInfoVar
RBApiGetVar
RBApiConvertVar
RBApiPutVar
RBApiClearVar
SQLGetBlobType
SQLMakeNewRow
SQLGetNewRow
SQLPutNewRow
SQLGetTableID
SQLCount
SQLUnlockRow
SQLSetRow
SQLAllocHandle
SQLBulkOperations
SQLColAttribute
SQLEndTran
SQLGetDiagRec
SQLFetchScroll
SQLFreeHandle
SQLGetConnectAttr
SQLGetStmtAttr
SQLSetConnectAttr
SQLSetStmtAttr
SQLSetEnvAttr
SQLCloseCursor

02. Support for TOP n qualifier for the SELECT commands.

New Syntax:

SELECT TOP n * FROM tablename **WHERE** whereclause **ORDER BY** column(s)

Here is an example:

To show the top 5 bonuses where the bonus is under \$500 from SalesBonus table in ConComp:

SELECT TOP 5 * FROM SalesBonus **WHERE** Bonus < 500 **ORDER BY** Bonus=DESC

The "TOP n" goes between the word SELECT and the column list for the command.

03. Support for new LIMIT syntax. Of course the traditional WHERE LIMIT = ... is also supported as before.

New Syntax:

LIMIT {[offset,] row_count | row_count OFFSET offset}

Examples: (Using Bluzvan Database):

SELECT * FROM Albums **LIMIT** 0,30
Gets rows 1-30 from Albums table

SELECT * FROM Albums **LIMIT** 5,10
Gets rows 6-15 from Albums table

SELECT * FROM Albums **LIMIT** 10
Gets the first 10 rows from Albums table

The LIMIT clause can be used to constrain the number of rows returned by the SELECT statement.

LIMIT takes one or two numeric arguments, which must be integer constants. With two arguments, the first argument specifies the offset of the first row to return, and the second specifies the maximum number of rows to return. The offset of the initial row is 0 (not 1):

```
SELECT * FROM table LIMIT 5,10  
Retrieves rows 6-15
```

To retrieve all rows from a certain offset up to the end of the result set, you can use some large number for the second parameter. This statement retrieves all rows from the 96th row to the last:

```
SELECT * FROM table LIMIT 95,99999999
```

With one argument, the value specifies the number of rows to return from the beginning of the result set:

```
SELECT * FROM table LIMIT 5  
Retrieve first 5 rows
```

In other words, LIMIT n is equivalent to LIMIT 0,n.

- 04. Fully compatible with PHP 5.0.5
- 05. Support for Nested INNER JOINS

Example 01:

```
CONNECT RRBYW10  
SELECT t1.EmpLName,t1.EmpFName,(SUM(t1.Price))=S +  
FROM ((Employee t2 INNER JOIN InvoiceHeader t3 ON t2.EmpID = t3.EmpID) t5 +  
INNER JOIN InvoiceDetail t4 on t4.TransID = t5.TransID) t1 +  
GROUP BY t1.EmpLName,t1.EmpFName
```

Example 02:

```
SELECT (t1.EmpLName+', '&t1.EmpFName)=27 AS Employee_Name,(SUM(t1.Price))=S +  
AS Total_Sale +  
FROM ((Employee t2 INNER JOIN InvoiceHeader t3 ON t2.EmpID = t3.EmpID) t5 +  
INNER JOIN InvoiceDetail t4 on t4.TransID = t5.TransID) t1 +  
GROUP BY t1.EmpLName,t1.EmpFName
```

Example 03:

```
SET ERROR MESSAGE 677 OFF  
DROP VIEW tTotSaleByEmployee  
SET ERROR MESSAGE 677 ON  
CREATE TEMPORARY VIEW `tTotSaleByEmployee` (Employee_Name, Total_Sale) AS +  
SELECT (t1.EmpLName+', '&t1.EmpFName)=27,(SUM(t1.Price))=S +  
FROM ((Employee t2 INNER JOIN InvoiceHeader t3 ON t2.EmpID = t3.EmpID) t5 +  
INNER JOIN InvoiceDetail t4 on t4.TransID = t5.TransID) t1 +  
GROUP BY t1.EmpLName,t1.EmpFName
```

- 06. ADO.NET Compliant.
- 07. Improved speed in ADO.NET environment.
- 08. A new System Variable #NOW has been implemented with data type DATETIME.

Examples:

```
UPDATE tablename SET colname = .#NOW WHERE ...  
  
INSERT INTO tablename (colname1, Colname2, Colname3) +  
VALUES .vColumnValue1, .vColumnValue2, .#NOW
```

09. Enhanced (LISTOF(colname)) command to support WHERE clause.

Example:

```
SELECT (LISTOF(colname)) IN TableViewName WHERE colname = value
```

10. Support for New Date Functions:

-- Integer Day of the Year (IDOY):

Example:

```
SET VARIABLE vIntDayOfYear INTEGER = NULL  
SET VARIABLE vIntDayOfYear = (IDOY(.#DATE))
```

The variable vIntDayOfYear will return the integer day of the year.

-- Integer Week of the Year (IWOY):

Example:

```
SET VARIABLE vIntWeekOfYear INTEGER = NULL  
SET VARIABLE vIntWeekOfYear = (IWOY(.#DATE))
```

The variable vIntWeekOfYear will return the integer week of the year.

-- Integer Days in a Month (IDIM):

Example:

```
SET VARIABLE vIntDaysInMonth INTEGER = NULL  
SET VARIABLE vIntDaysInMonth = (IDIM(.#DATE))
```

The variable vIntDaysInMonth return the integer days in a month.

-- Is Leap Year (ILY): (1 = Yes, 0 = No)

Example:

```
SET VARIABLE vIsLeapYear INTEGER = NULL  
SET VARIABLE vIsLeapYear = (ILY(.#DATE))
```

The variable vIsLeapYear will return 1 if the year is a leap year, and 0 if it is not.

-- Next Business Date (DNW):

Example:

```
SET VARIABLE vNextBusinessDate DATE = NULL  
SET VARIABLE vNextBusinessDate = (DNW(.#DATE))
```

The variable vNextBusinessDate will return date of next business date.

-- Next Weekend Date (DWE):

```
SET VARIABLE vNextWeekEndDate DATE = NULL  
SET VARIABLE vNextWeekEndDate = (DWE(.#DATE))
```

The variable vNextWeekEndDate will return date of next weekend date.

11. New String Manipulation Functions:

```
((TRIM(text))  
((LTRIM(text))  
((RTRIM(text))
```

```
((TRIM(text))
```

Trims leading as well as trailing blanks from text, returning a text string.

In the following example, the value of vStrim is the text string ABCDE without the leading as well as trailing blanks.

```
SET VARIABLE vStrim = (TRIM('   ABCDE   '))
```

(LTRIM(text))

Trims leading blanks from text, returning a text string.

In the following example, the value of vStrim is the text string ABCDE without the leading blanks.

```
SET VARIABLE vStrim = (LTRIM('   ABCDE'))
```

(RTRIM(text))

Trims trailing blanks from text, returning a text string. This is similar to the existing **STRIM** function for string manipulation.

In the following example, the value of vStrim is the text string ABCDE without the trailing blanks.

```
SET VARIABLE vStrim = (RTRIM('ABCDE   '))
```

12. Sting Manipulation and Expression Support for **VARCHAR** Data Type.

Using the new and enhanced VARCHAR data type in Oterro 3.5, now you can:

- A. Concatenate two VARCHAR values into one:

```
SET VAR vMyVarChar = (.varchar1 + .varchar2)
```

- B. Find the offset of the first occurrence of a string:

```
SET VAR vOffset = (SLOCP(.varchar3, 'SOMESTRING', 1))
```

- C. Extract a portion of a VARCHAR:

```
SET VAR vNewVarChar = (SGET(.vVarChar4, 100, .vOffset))
```

- D. Overwrite a portion of a VARCHAR:

```
SET VAR vNewVarChar = (SPUT(.vVarChar4, .vLittleString, 140))
```

- E. Cast VARCHAR data to NOTE or TEXT datatypes (and vice versa)

```
SET VAR vTextVar TEXT = (.vNewVarChar)
```

```
SET VAR vNoteVar NOTE = (.vNewVarChar)
```

```
SET VAR vVarCharVar VARCHAR = (.vTextVar)
```

```
SET VAR vVarCharVar VARCHYAR = (.vNoteVar)
```

13. New String Manipulation Function:

(SLOCP(TextNoteVarCharValue,string,occurrence))

Locates the exact position of a given string and occurrence in a TEXT, NOTE or VARCHAR value. Using **-1** as third parameter will return the LAST occurrence.

Example 01:

```
SET VARIABLE v1 VARCHAR = 'ABCDEABC_AB'
```

```
SET VARIABLE vSloc1 = (SLOCP(.v1,'AB',1))
```

Resulting variable vSloc1 will return the integer value of 1.

Example 02:

```
SET VARIABLE v1 VARCHAR = 'ABCDEABC_AB'
```

```
SET VARIABLE vSloc2 = (SLOCP(.v1,'AB',2))
```

Resulting variable vSloc2 will return the integer value of 6.

14. Support for Foxjet Marksman Pro interface with R:BASE.

15. Support for COPY command with UNC parameters.

16. Support for UNLOAD DATA FOR tableviewname AS ASCII command.

17. Support for UNLOAD DATA FOR tableviewname AS CSV command.

18. Support for COPY command when using UNC path.
19. Support for UPS WorldShip 9.0

Upgrade Policy:

Oterro 3.5 is available as a "Step-Up" for all existing registered users of Oterro 3.1 and as an "Upgrade" for all other versions of Oterro. All subsequent in-line patches will be available as FREE updates to all registered 3.1 users. For complete details, licensing and upgrade prices, please contact RBTI sales team at 1+724.733.0053 or <mailto:sales@rbase.com>

Step-Up or Upgrade to Oterro 3.5 for R:Tango 5.0 is for R:Tango 5.0 Single-Server License ONLY.

Step-Up or Upgrade to Oterro 3.5 for Witango 5.x is for Witango 5.x Single-Server License ONLY.

Step-Up or Upgrade to Oterro 3.5 for R:WEB Suite 3.0 is for R:WEB Suite Single-Server License ONLY.

Acknowledgments:

R:BASE Technologies and its Dream Team would like to thank all those who have been an integral part of making Oterro 3.5, especially those who have been involved first hand to help us fine tune and test all the features implemented for .NET, ColdFusion and CodeCharge Studio 3.0 for R:BASE (Part of R:WEB Suite 3.0, the most productive solution for visually creating database-driven Web applications for R:BASE - without coding), MS Office XP 2003, UPS WorldShip, .NET and Foxjet Marksman Pro.

R:BASE Technologies, Inc., would like to acknowledge the team work of YesSoftware for their dedicated engineers who worked closely with our development team to help Oterro 3.5 work seamlessly with their products.

Copyright:

Copyright © 1982-2007 by R:BASE Technologies, Inc. All rights reserved. No part of this document may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language in any form by any means, without the written permission of R:BASE Technologies, Inc.

Trademarks:

R:BASE®, OTERRO®, R:Scope®, R:WEB Suite®, R:Tango®, R:Accounting®, R:Archive®, R:Admin®, R:Zip®, R:Capture®, R:Chat®, R:Charts®, R:Code®, R:ColumnAnalyzer®, R:Contact®, R:Docs®, R:Documenter®, R:DWF Viewer®, R:BASE Editor®, R:Fax®, R:FormPacker®, R:FTPClient®, R:Java®, R:Linux®, R:Mail®, R:Merge®, R:PDFMerge®, R:PDFSearch®, R:PDFWorks®, R:RemoteControl®, R:Scheduler®, R:Style®, R:Struc®, R:Syntax®, R:WatchDog®, R:PHP®, R:Synchronizer®, R:Magellan®, R:WEB Suite®, R:WEB Reports®, 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.