

# WDRReportGen

## User Manual

**Version 3.9**

2009-08

*Copyright© 2003-2009 LJZsoft Corporation*

All rights reserved

# Contents

|  |           |
|--|-----------|
| <b>CHAPTER 1 INTRODUCTION.....</b>               | <b>1</b>  |
| 1.1 OVERVIEW.....                                | 1         |
| 1.2 FEATURES.....                                | 1         |
| <b>CHAPTER 2 INSTALLATION AND STARTUP .....</b>  | <b>5</b>  |
| 2.1 SOFTWARE REQUIREMENTS.....                   | 5         |
| 2.2 INSTALLING WDRPTGEN.....                     | 5         |
| 2.3 UNINSTALLING WDRPTGEN.....                   | 5         |
| 2.4 COMMAND LINE.....                            | 6         |
| <b>CHAPTER 3 QUICK START.....</b>                | <b>8</b>  |
| 3.1 LEARNING HOW TO USE WDRPTGEN.....            | 8         |
| 3.2 SAMPLE DATABASE.....                         | 8         |
| 3.3 STEPS OF REPORTING.....                      | 9         |
| 3.4 MY FIRST REPORT.....                         | 10        |
| 3.4.1 <i>Creating a report template</i> .....    | 10        |
| 3.4.2 <i>Creating a WRF file</i> .....           | 11        |
| 3.4.3 <i>Configuring the report</i> .....        | 11        |
| 3.4.4 <i>Inputting a function</i> .....          | 11        |
| 3.4.5 <i>Understanding the function</i> .....    | 12        |
| 3.4.6 <i>Running a WRF file</i> .....            | 12        |
| 3.4.7 <i>Opening a report</i> .....              | 13        |
| 3.4.8 <i>Modifying the report template</i> ..... | 13        |
| 3.4.9 <i>Modifying the function</i> .....        | 14        |
| 3.4.10 <i>Generating the report again</i> .....  | 14        |
| 3.5 SAMPLES.....                                 | 15        |
| <b>CHAPTER 4 REPORT TEMPLATES .....</b>          | <b>17</b> |

|   |    |
|---|----|
| 4.1 ABOUT REPORTS .....                                 | 17 |
| 4.2 ABOUT REPORT TEMPLATES.....                         | 17 |
| 4.3 WORD BASIC CONCEPTS.....                            | 17 |
| 4.3.1 Documents.....                                    | 17 |
| 4.3.2 Headers and Footers.....                          | 18 |
| 4.3.3 Tables.....                                       | 18 |
| 4.3.4 Bookmarks.....                                    | 18 |
| 4.3.5 Page Breaks .....                                 | 18 |
| 4.3.6 Graphics and Diagrams.....                        | 18 |
| 4.3.7 Inline Pictures and Floating Pictures .....       | 19 |
| 4.3.8 Charts.....                                       | 19 |
| 4.3.9 Formatting .....                                  | 19 |
| 4.3.10 Fields .....                                     | 21 |
| 4.4 TABLE REPORTS .....                                 | 22 |
| 4.4.1 About Table Reports.....                          | 22 |
| 4.4.2 Creating a Table for a Fixed Table Report.....    | 22 |
| 4.4.3 Creating a Table for a Variable Table Report..... | 23 |
| 4.4.4 Formatting Cells.....                             | 24 |
| 4.4.5 Irregular Tables .....                            | 25 |
| 4.4.6 Referencing Cells.....                            | 26 |
| 4.4.7 Referencing Tables.....                           | 27 |
| 4.4.8 Formatting Cells for Pictures .....               | 28 |
| 4.5 FORM REPORTS.....                                   | 29 |
| 4.5.1 About Form Reports.....                           | 29 |
| 4.5.2 Creating Merge Fields or Quote Fields.....        | 30 |
| 4.5.3 Defining Ranges.....                              | 31 |
| 4.5.4 Formatting Fields for Pictures.....               | 32 |
| 4.6 CHARTS .....  | 33 |

|  |           |
|--|-----------|
| 4.6.1 About Charts .....                                 | 33        |
| 4.6.2 Creating a Blank Chart using Microsoft Graph ..... | 34        |
| 4.6.3 Creating a Blank Chart using Microsoft Excel ..... | 34        |
| 4.6.4 Referencing Charts .....                           | 36        |
| <b>CHAPTER 5 REPORTING WITH WDREPORTGEN .....</b>        | <b>38</b> |
| 5.1 CREATING AND OPENING WRF FILES .....                 | 38        |
| 5.1.1 About WRF files .....                              | 38        |
| 5.1.2 Create a new WRF file .....                        | 38        |
| 5.1.3 Open a WRF file.....                               | 38        |
| 5.1.4 Save a WRF file.....                               | 38        |
| 5.2 CONFIGURING FILES .....                              | 39        |
| 5.2.1 About files .....                                  | 39        |
| 5.2.2 Configuring file information .....                 | 39        |
| 5.2.3 Converting files.....                              | 40        |
| 5.3 CONFIGURING DATA SOURCES .....                       | 42        |
| 5.3.1 About data sources.....                            | 42        |
| 5.3.2 Adding, modifying and deleting a data source.....  | 42        |
| 5.4 CONFIGURING PARAMETERS.....                          | 43        |
| 5.4.1 About parameters.....                              | 43        |
| 5.4.2 Adding, modifying and deleting a parameter.....    | 44        |
| 5.5 INPUTTING FUNCTIONS .....                            | 45        |
| 5.6 RUNNING WRF FILES.....                               | 45        |
| 5.6.1 Windows mode .....                                 | 46        |
| 5.6.2 Command line mode.....                             | 46        |
| 5.7 USING FORMULA FIELDS .....                           | 46        |
| 5.8 SORTING, GROUPING AND TOTALING .....                 | 48        |
| 5.8.1 Sorting data.....                                  | 48        |
| 5.8.2 Totaling.....                                      | 49        |

|  |           |
|--|-----------|
| 5.8.3 Grouping data and subreports.....                | 49        |
| 5.8.4 Subtotaling .....                                | 50        |
| 5.9 PICTURES.....                                      | 51        |
| 5.9.1 Inserting pictures into a report template.....   | 51        |
| 5.9.2 Inserting pictures into a report.....            | 51        |
| 5.10 USING PARAMETERS.....                             | 52        |
| 5.11 PROGRAMMING.....                                  | 55        |
| 5.11.1 Using add-ins, macros .....                     | 55        |
| 5.11.2 Making WRF files programmatically.....          | 56        |
| <b>CHAPTER 6 FUNCTION REFERENCE.....</b>               | <b>57</b> |
| 6.1 REPORT FUNCTION .....                              | 57        |
| 6.1.1 Report Function .....                            | 57        |
| 6.1.2 Fixed Table Report Function .....                | 57        |
| 6.1.3 Non-group Variable Table Report Function.....    | 61        |
| 6.1.4 Group Variable Table Report Function.....        | 65        |
| 6.1.5 Form Report Function .....                       | 70        |
| 6.2 CHART FUNCTION.....                                | 75        |
| 6.2.1 MSGraph Chart Function .....                     | 75        |
| 6.2.2 Excel Chart Function.....                        | 78        |
| 6.3 DOCVARIABLE FUNCTION .....                         | 82        |
| 6.4 EXECSQL FUNCTION .....                             | 83        |
| <b>CHAPTER 7 MENUS, TOOLBAR AND SHORTCUT KEYS.....</b> | <b>85</b> |
| 7.1 FILE MENU .....                                    | 85        |
| 7.2 EDIT MENU.....                                     | 85        |
| 7.3 REPORT MENU.....                                   | 85        |
| 7.4 TOOLS MENU.....                                    | 86        |
| 7.5 HELP MENU.....                                     | 86        |

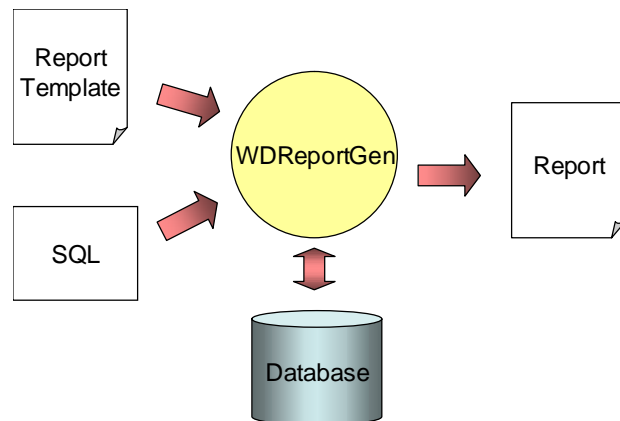
|  |            |
|--|------------|
| 7.6 TOOLBAR.....   | 86         |
| 7.7 SHORTCUT KEYS .....                                  | 87         |
| <b>CHAPTER 8 HINTS AND TIPS.....</b>                     | <b>89</b>  |
| <b>CHAPTER 9 WRF FILE REFERENCE.....</b>                 | <b>95</b>  |
| 9.1 WRF FILE FORMAT .....                                | 95         |
| 9.2 [DATA SOURCE] SECTION.....                           | 96         |
| 9.3 [FILE] SECTION .....                                 | 98         |
| 9.4 [PARAMETER] SECTION.....                             | 100        |
| <b>CHAPTER 10 FORMAT EXPRESSIONS IN DATA CELLS .....</b> | <b>101</b> |
| A.1 FORMATS FOR NUMERIC VALUES .....                     | 101        |
| A.2 FORMATS FOR STRING VALUES .....                      | 105        |
| A.3 FORMATS FOR DATE/TIME VALUES.....                    | 106        |
| <b>CHAPTER 11 LICENSE AND SUPPORT.....</b>               | <b>110</b> |
| 11.1 LICENSE.....  | 110        |
| 11.2 TECHNICAL SUPPORT.....                              | 111        |

# Chapter 1 Introduction

## 1.1 Overview

WDRReportGen is a report generator for Microsoft Word that outputs reports in Microsoft Word document format. If you know how to use Microsoft Word and write SQL statements, you can use WDRReportGen to create all kinds of reports as you need.

To create a report, WDRReportGen need to read a report template file and a WRF file. The report template file is a Microsoft Word document that defines the layouts and formats of a report. The WRF file



contains SQL statements and some information, and tells WDRReportGen how to get data from database and how to put data into a report. First WDRReportGen creates a blank report using the report template file, and then executes SQL statements in the WRF file to get data from database, and fills data into the report to generate the desired report in Microsoft Word document format.

## 1.2 Features

WDRReportGen includes the following features:

- Using Microsoft Word as your reporting tool

Just use Microsoft Word as your reporting tool. You design reports like layouts, formats and styles directly using Microsoft Word. And you will get reports in Microsoft Word document format as a result. Microsoft Word is powerful,

flexible and familiar. You do not need to buy and learn extra reporting tools.

- Making report template directly using Microsoft Word

The main advantage of using WDRReportGen is based on the fact that all formatting is done directly in Microsoft Word. You can take full advantage of Microsoft Word including text formatting, tables, fields, pictures and graphics, drawing, page setup, headers and footers, preview and printing, VBA, macros, and more.

- Accessing to databases using SQL

WDRReportGen executes SQL statements to extract data from database. Supports all type SQL: DML, DDL and DCL. Multiple SQL statements can be executed in one report building process. You can perform queries on databases, insert data into databases, and create database objects like tables. The power of SQL can be harnessed for maximum efficiency in reporting.

- Creating reports without programming experience

You know how to use Microsoft Word and how to write SQL, it is enough. It does not require programming to create reports.

- Connection to all databases using ODBC

WDRReportGen connects to databases using ODBC. Access to all databases which support ODBC such as Oracle, DB2, Sybase, Informix, Microsoft SQL Server, Teradata, MySQL, Microsoft Access, dBase.

- Supporting multi-databases in one report

WDRReportGen supports multi-databases in one report. You can get data from some different databases such as Oracle, DB2 and Microsoft SQL Server, and put these data into one report.

- Generating reports with parameters

WDRReportGen gives you an opportunity to create reports with parameters. You may use parameters in SQL statements. You will be asked to input the values of parameters while generating reports.

- Supporting Windows mode and command line mode

WDRReportGen supports command line mode. So it is possible to call WDRReportGen from other program. For developers, you can integrate WDRReportGen into your application.

- Creating complex reports

You can create complex reports. The complexity might come from report formatting as well as report content.

- Creating reports with charts

WDRReportGen enables you to include sophisticated, colorful charts in your reports. You can use charts any time you want to improve the usefulness of a report.

- Creating reports with pictures

WDRReportGen can insert pictures from the graphics files, and set the inserted way, text wrapping style and size of the pictures according to your instruction.

- Many reports in one Microsoft Word document

One Microsoft Word document may contain many reports. You can generate a book of reports in one generating process.

- Conversion of file formats

WDRReportGen is a converter too. You can convert Word documents to and from other formats, such as HTML, XML, RTF, text, and Works. You also can convert data from database to other file format.

- Generating reports automatically

The process of report generation can be fully automated, periodically or on events. WDRReportGen can be scheduled with Windows Scheduled Tasks or other tools.

- One time configuration

With on time configuration, you can repeatedly generate reports especially periodic reports such as daily, weekly, monthly and annual reports.

- Flexible deployment

WDRReportGen can be run on your desktop or server.

## **Chapter 2 Installation and Startup**

### **2.1 Software Requirements**

Microsoft Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows 2003, Windows Vista or later.

Microsoft Office 97/98, Office 2000, Office XP, Office 2003 or later.

### **2.2 Installing WDRReportGen**

Run installation program, and follow the instructions to complete WDRReportGen installation. For Windows Vista, the data folder should be different from the application folder.

If you don't have Microsoft Office, please install it first.

If your environment is Windows 95/98 and Office 97, and you don't have VB6.0 run-time files installed, please install it. For Windows 2000, Windows XP, Windows 2003 and Office 2000 or later, you do not need to install VB6.0 run-time files because they are included in OS and Office. To install VB6.0 run-time files, just run vbrun60sp5.exe, and follow the instructions.

If you don't have ODBC Driver for the database you want to access installed, please install it.

If your OS is Windows 95/98 and you don't have Microsoft Data Access Components 2.0 (MDAC\_TYP) or later installed, please install it. For Windows 2000, Windows XP and Windows 2003, you do not need to install MDAC\_TYP because it is preinstalled in OS. To install MDAC\_TYP, just run mdac\_typ.exe, and follow the instructions.

### **2.3 Uninstalling WDRReportGen**

1. Quit WDRReportGen.

2. Double-click the Add/Remove Programs icon in the Windows Control Panel.
3. Do one of the following:
  - For Windows 2000, Windows XP and Windows 2003 Edition:  
Click WDRReportGen in the Currently installed programs box, and then click the Change/Remove button.
  - For Windows 98 and Windows NT 4.0:  
Click WDRReportGen on the Install/Uninstall tab, and then click the Add/Remove button.
4. Follow the instructions on the screen to complete uninstalling the program.

## 2.4 Command Line

WDRReportGen can be run in Windows mode or command line mode. The Syntax of command is:

```
wordreport <wrf file name> [-C] [-D] [-I interval] [-U1 user1] [-P1 pwd1] ...
[-U10 user10] [-P10 pwd10] [pa1 pa2 ... pa10]
```

- |               |  |
|---------------|--|
| wrf file name | Specifying a WRF (.wrf) file that tells WDRReportGen how to get data from data sources and how to put data into a report.                                    |
| -C            | Run WDRReportGen in command line mode.   |
| -D            | Display the generated report with Microsoft Word.  |
| -I interval   | Log the processing records message. If interval is greater than 1, it is the interval of records. If interval is less than 1, it is the percent of interval. |
| -U1 user1 ... | Specify the user names. user1 is the user name of the first  |
| -U10 user10   | data source. user2 is the user name of the second data source.....   |
| -P1 pwd1 ...  | Specify the passwords. pwd1 is the password of the first data  |
| -P10 pwd10    | source. pwd2 is the password of the second data source.....  |

pa1 ... pa10    The values of the parameters defined in the WRF file. You can use parameters in SQL statements. WDRReportGen will replace the names of parameters in a SQL statement with the actual values before it executes a SQL statement. You can use no more than 10 parameters in one report.

For example, you have defined two parameters in your WRF file. The first parameter is sales date, and the second is the category of the products. You can run WDRReportGen in command line mode as follows:

```
wordreport c:\WordReport\myreport.wrf -c 1996-05-01 "Dairy Products"
```

## Chapter 3 Quick Start

### 3.1 Learning how to use WDRReportGen

You can teach yourself how to use WDRReportGen by choosing from the methods available in this section:

- You can study the sample reports and sample database included with WDRReportGen.
- You can use the detailed descriptions and instructions in the “My First Report”.

### 3.2 Sample Database

WDRReportGen comes with Sample.mdb, a sample database you can use when learning the program. Sample.mdb is a Microsoft Access database. Virtually all of the examples in this manual are based on Sample.mdb data. The sample reports access the sample database through the ODBC data source name “Report Sample”. When you install WDRReportGen, you can choose to add the ODBC data source name. And you also can add the ODBC data source name manually.

To create the System DSN “Report Sample”, do as follows:

1. Click the Windows **Start** button, choose **Settings**, and then click **Control Panel**.
2. On computers running Microsoft Windows 2000 or later, double-click **Administrative Tools**, and then double-click **Data Sources (ODBC)**. The **ODBC Data Source Administrator** dialog box appears. On computers running previous versions of Microsoft Windows, double-click **32-bit ODBC** or **ODBC**.
3. Select the **System DSN** tab, and then press **Add** button.

4. Choose **Microsoft Access Driver (\*.mdb)**, then press **Finish** button.
5. In the **ODBC Microsoft Access Setup** dialog box, type **Report Sample** in the **Data Source Name** box.
6. Press the **Select** button, and browse to select **Sample.mdb**.
7. Press **OK** button to close the **ODBC Microsoft Access Setup** dialog box.
8. Press **OK** button to close the **ODBC Data Source Administrator** dialog box.

### **3.3 Steps of Reporting**

To create a report with WDRReportGen, you should do as follows:

#### 1. Prepare works

Before you create a report, you should determine the layout of the report, and know where and how to get the data.

You must know how to access the databases you are reporting from. So you need the data source name, user name and password. If you don't have added data sources, please add data sources first. Run ODBC Administrator, you can add a new data source. For detailed information about configuring ODBC, refer to *ODBC Administrator Help*.

#### 2. Make a report template file

Create a report template file using Microsoft Word. The report template file is a Microsoft Word document. For detailed information about report template, refer to "Report Templates" in this document.

#### 3. Create a WRF file

Create a WRF file with a .wrf extension using WDRReportGen. There are two steps to create a WRF file.

##### (1) Configure the report

Define the names of data sources, the name of the report template file, the name of the report file and the name of the log file. If you want to use

parameters in SQL statements, define these parameters.

## (2) Write functions

Write functions and SQL statements that specify how to get data from data sources and how to put data into the report.

For detailed information, refer to “Reporting with WDRReportGen” in this document.

## 4. Run the WRF file

Run the WRF file to generate a report in Microsoft Word document. For detailed information about running report, refer to “Running a WRF File” in this document.

## 3.4 My First Report

The following tutorial has been designed to guide you to create your first report. In this tutorial, you will get an introduction to the program as you create a Customer List report. The Customer List is one of the most basic business reports and typically has information such as Customer Name, City, Country, and Contact Name.

### 3.4.1 Creating a report template

1. Run Microsoft Word, a new document will open.
2. On the **Table** menu, point to **Insert**, and then click **Table**. Under **Table size**, select the number of columns and rows. Press **OK** button.
3. Click the cell A1, type “Customer Name”. In the same way, you input “City”, “Country” and “Contact Name” into the cells B1, C1 and D1.
4. Format the text of A1, B1, C1 and D1 as you like, including font, font size, font colour, bold, background, alignment and border.
5. You can change the width of these columns. The report template you have made is as follows:

| Customer Name↵ | City↵ | Country↵ | Contact Name↵ |
|----------------|-------|----------|---------------|
| ↵              | ↵     | ↵        | ↵             |

6. Click **Save** on the **File** menu, chose a directory such as “C:\Report”, type custlist.doc in the **File name** box and press **Save** button.

7. Click **Close** on the **File** menu.

### 3.4.2 Creating a WRF file

1. Run WDRReportGen.

2. Click **New** on the **File** menu.

3. Click **Save** on the **File** menu, chose the directory to which you have saved the report template, type custlist.wrf in the **File name** box and press **Save** button.

### 3.4.3 Configuring the report

1. On the **Report** menu, click **Configuration**. The **Configuration** dialog box appears.

2. Click the **File** tab.

In the **Template File** box, type custlist.doc; In the **Report File** box, type Report\custlist.doc; In the **Log File** box, type Log\custlist.log.

3. Click the **Data Source** tab.

Press **New** button, the **New Data Source** dialog box appears. In the **Name** box, type Report Sample, press **OK** button.

4. On the **Configuration** dialog box, press **OK** button.

### 3.4.4 Inputting a function

In the editor windows, input a function as follows:

```
@F1=Report(table=1 cell=A2)
```

```
SELECT CompanyName
```

```
,CityName
,CountryName
,ContactName
FROM Customers, Cities, Countries
WHERE Customers.CityCode = Cities.CityCode
AND Customers.CountryCode = Cities.CountryCode
AND Customers.CountryCode = Countries.CountryCode
ORDER BY CompanyName, CityName, CountryName
;
```

You can test the SQL statement in a query tool such as Microsoft Access or Microsoft Query.

### 3.4.5 Understanding the function

Before going any further, let us understand this function.

1. The **Report** function will execute the SQL statement, get data from data source, and put data into the report.
2. The **table** argument identifies a table, and the value 1 is the index number of the table. So it is the first table.
3. The **cell** argument specifies the cells that the first record will be filled into. The value is A2. So WDRReportGen will fetch the first record, put the value of CompanyName field into A2, the value of CityName field into B2, the value of CountryName field into C2, and the value of ContactName field into D2. An then it fetch the next record, put them into A3,B3,C3 and D3.....

### 3.4.6 Running a WRF file

1. On the **Report** menu, click **Run**, the **Run Report** dialog box appears.
2. Press **Start** button to run the WRF file.
3. WDRReportGen will generate a report.

4. After the status is **Done**, click **Close** button.

### 3.4.7 Opening a report

1. On the **File** menu, click **Open Report File** to open the report you have generated.

You can view and check the report.

2. On the **File** menu, click **Open Log File** to open the log file that recorded the log information in the report generating..

You can check the log.

3. Close the report file and the log file.

### 3.4.8 Modifying the report template

1. On the **File** menu, click **Open Template File** to open the report template.

2. Change the width of columns. It is very useful to copy some sample data from the report file into the report template for formatting.

3. Insert text before the table, and type Customer List as the report title. To insert text before a table, click in the upper-left cell in the first row of the table, place the insertion point before the text, and then press **ENTER**.

4. Select the second row, and insert a row to the table.

5. Add a border to the table. Select the table, click **Borders and Shading** on the **Format** menu, and then click the **Borders** tab. Select the options you want, and press **OK** button. The external border can be different from the internal border. The report template you have made is as follows:

**Customer List**

| Customer Name | City | Country | Contact Name |
|---------------|------|---------|--------------|
|               |      |         |              |
|               |      |         |              |

6. Select the first row of the table, and click **Heading Rows Repeat** on the

**Table** menu.

7. Save and close the template file.

### 3.4.9 Modifying the function

In the editor windows, modify the function as follows:

```
@F1=Report(table=1 cell=A2 reserve=2)
SELECT CompanyName
,CityName
,CountryName
,ContactName
FROM Customers, Cities, Countries
WHERE Customers.CityCode = Cities.CityCode
AND Customers.CountryCode = Cities.CountryCode
AND Customers.CountryCode = Countries.CountryCode
ORDER BY CompanyName,CityName,CountryName
;
```

The **reserve** argument specifies the number of records for that you reserve some rows. You have reserve two blank rows in the report template so that the format of the last row/column border may be different from the others.

### 3.4.10 Generating the report again

1. Save the WRF file.
2. Run the WRF file to generate the report.
3. Open the report, view and check the report.

The report should now look similar to the following:

## Customer List

| Customer Name                      | City        | Country | Contact Name       |
|------------------------------------|-------------|---------|--------------------|
| Alfreds Futterkiste                | Berlin      | Germany | Maria Anders       |
| Ana Trujillo Emparedados y helados | México D.F. | Mexico  | Ana Trujillo       |
| Antonio Moreno Taquería            | México D.F. | Mexico  | Antonio Moreno     |
| Around the Horn                    | London      | UK      | Thomas Hardy       |
| Berglunds snabbköp                 | Luleå       | Sweden  | Christina Berglund |
| Blauer See Delikatessen            | Mannheim    | Germany | Hanna Moos         |
| Blondel père et fils               | Strasbourg  | France  | Frédérique Citeaux |

Now you have created a report.

### 3.5 Samples

After WDRReportGen is installed, some sample reports are installed too. Use these reports to learn WDRReportGen. The sample reports can be changed to adapt to your own needs.

The sample reports include a sample database, some report template files (.doc) and WRF files (.wrf). They are located in the Application Data\LJZsoft under All Users or your profile folder. WDRReportGen was tested with Microsoft Office 2007. Please download the sample reports for Microsoft Office 2007 from our website.

| Directory                          | Description   |
|------------------------------------|---|
| {data}\Common\SampleDatabase       | Contains the sample database "Sample.mdb".                          |
| {data}\WDRReportGen\Samples        | Contains the report template files (.doc) and the WRF files (.wrf). |
| {data}\WDRReportGen\Samples\Report | Contains the report files (.doc) generated by WDRReportGen.         |
| {data}\WDRReportGen\Samples\Log    | Contains the log files created by WDRReportGen during generating    |

|  |               |
|--|---------------|
|  | report files. |
|--|---------------|

{data} is the path of the data folder. You can select the data folder when you install WDRReportGen. By default, the data folder is the Application Data\LJZsoft folder under All Users. If you install WDRReportGen without administrative privileges, the data folder is the Application Data\LJZsoft folder under the current user. The data folder is usually at:

Windows 95/98: C:\windows\All Users\Application Data\LJZsoft

Windows NT: C:\WinNT\Profiles\All Users\Application Data\LJZsoft

Windows 2000/XP: C:\Documents and Settings\All Users\Application Data\LJZsoft

Windows Vista: C:\ProgramData\LJZsoft

## **Chapter 4 Report Templates**

### **4.1 About Reports**

The report generated by WDRReportGen is a Microsoft Word document. The layouts, formats and styles of the report are defined by a report template, and the data of the report are got from databases such as Oracle, DB2.

### **4.2 About Report Templates**

To make a report using WDRReportGen, you should create a report template first. The report template is a Microsoft Word document that defines the layouts, formats and styles of the report. In the Microsoft Word report template, you can input static content such as titles, descriptions, comments, a cover, a company logo, format the static content, and define the format of the data you will get from databases.

When generating a report, WDRReportGen will copy the report template file to a blank report file, and then put data into the report. So the layouts, formats and styles defined in the report template file will be brought to the final report file.

### **4.3 Word Basic Concepts**

If you have known these concepts of Microsoft Word, please skip this section. For more detail information about Microsoft Word, refer to *Microsoft Word Help*.

#### **4.3.1 Documents**

A document is a Microsoft Word file with extension .doc. You can open and

save it using Microsoft Word. Microsoft Word documents may contain a combination of text, formatting and graphics.

### **4.3.2 Headers and Footers**

Headers and footers are areas in the top and bottom margins of each page in a document. You can insert text or graphics in headers and footers - for example, page numbers, the date, a company logo, the document's title or file name, or the author's name - that are printed at the top or bottom of each page in a document.

### **4.3.3 Tables**

A table is made up of rows and columns of cells that you can fill with text and graphics. Tables are often used to organize and present information. You can set borders, shading, alignment and fonts in tables.

### **4.3.4 Bookmarks**

A bookmark identifies a location or selection of text that you name and identify for future reference. For example, you might use a bookmark to identify text that you want to revise at a later time.

### **4.3.5 Page Breaks**

When you fill a page with text or graphics, Microsoft Word inserts an automatic page break and starts a new page. To force a page break at a specific location, you can insert a manual page break.

### **4.3.6 Graphics and Diagrams**

There are two basic types of graphics that you can use to enhance your

Microsoft Word documents: drawing objects and pictures.

Drawing objects include AutoShapes, diagrams, curves, lines, and WordArt drawing objects. These objects are part of your Word document. Use the Drawing toolbar to change and enhance these objects with colors, patterns, borders, and other effects.

Pictures are graphics that were created from another file. They include bitmaps, scanned pictures and photographs, and clip art. You can change and enhance pictures by using the options on the Picture toolbar and a limited number of options on the Drawing toolbar. In some cases, you must ungroup and convert a picture to a drawing object before you can use the Drawing toolbar options.

### **4.3.7 Inline Pictures and Floating Pictures**

Inline picture: A graphic or other object that is positioned directly in the text of a Microsoft Word document at the insertion point.

Floating picture: A graphic or other object that is inserted in the drawing layer so that you can position it precisely on the page or in front of or behind text or other objects.

### **4.3.8 Charts**

Charts are visually appealing and make it easy for users to see comparisons, patterns, and trends in data. You can create a chart in a Microsoft Word document using Microsoft Graph or Microsoft Excel. When you create a new chart in Word, Microsoft Graph or Microsoft Excel opens and a chart is displayed with its associated data in a data sheet or worksheet.

### **4.3.9 Formatting**

You can use these formatting features of Microsoft Word to effectively display your data.

## ■ Characters formatting

To make text stand out, you can format the text in selected characters. You can set font, color, size of text, bold and italic formats, animate or highlight the text.

## ■ Paragraphs formatting

You can set text alignment, tab stops, line spacing, spacing before or after paragraphs, and borders.

## ■ Bulleted and numbered lists

Bulleted and numbered lists in Microsoft Word are easy to create. You can quickly add bullets or numbers to existing lines of text, or Microsoft Word can automatically create lists as you type.

## ■ Borders, Shading, and Graphic Fills

Borders, shading, and graphic fills can add interest and emphasis to various parts of your document. You can add borders to pages, text, tables and table cells, graphic objects, pictures, and Web frames. You can shade paragraphs and text. You can apply colored or textured fills to your graphic objects.

## ■ Automatic formatting

By using AutoFormat, you can quickly apply formatting such as headings, bulleted and numbered lists, borders, numbers, symbols, and fractions to your text. You can automatically format a document either as you type or after you've written it. In both cases, you can control which automatic changes Microsoft Word makes. You can also turn off automatic formatting.

## ■ Style

A style is a set of formatting characteristics that you can apply to text, tables, and lists in your document to quickly change their appearance. When you apply a style, you apply a whole group of formats in one simple task.

For example, instead of taking three separate steps to format your title as 16 pt, Arial, and center-aligned, you can achieve the same result in one step by

applying the Title style.

### 4.3.10 Fields

#### ■ Fields

Fields are used as placeholders for data that might change in a document and for creating form letters and labels in mail-merge documents. Microsoft Word inserts fields when you use particular commands, such as the Date and Time command on the Insert menu. You can also manually insert your own fields by using the Field command on the Insert menu. Field codes appear between curly brackets, or braces ( { } ). Fields are somewhat like formulas in Microsoft Excel — the field code is like the formula, and the field result is like the value that the formula produces. You can switch between displaying field codes and results in your document.

#### ■ (Formula) fields

(Formula) field calculates a number by using a mathematical formula. You can insert an (Formula) field in a table or in regular text. Computation in tables can be completed using (Formula) fields, such as add, subtract, multiply, divide, sum. Syntax:

*{ = Formula [Bookmark ] [\# Numeric Picture ] }*

#### ■ DocVariable field

DocVariable field defines a document variable. Each document has a collection of variables, which can be added and referenced by the Microsoft Visual Basic for Applications programming language. This field provides a way to display the contents of the document variables in the document. Syntax:

*{ DOCVARIABLE "Name" }*

Inserts the string assigned to a document variable. “Name” is the name of the document variable.

## **4.4 Table Reports**

### **4.4.1 About Table Reports**

A table is made up of rows and columns of cells that you can fill with text and graphics. Tables are often used to make reports, and organize and present information.

WDRReportGen supports two types of table reports: fixed table report, variable table report.

Fixed table report: The number of rows and columns in the table is fixed. When WDRReportGen executes a SQL statement, directly puts the result data into cells in the table.

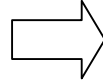
Variable table report: The number of rows or columns in the table is unfixed, and it is variable as the number of result records. When WDRReportGen executes a SQL statement, it repeats the table rows or columns for each record or group, and then puts data into cells of the table.

### **4.4.2 Creating a Table for a Fixed Table Report**

For a fixed table report, you need to create a table in the report template file according to the report. The format of the table is the same as the format in the report, but cells that should be filled data into are blank. When WDRReportGen executes a SQL statement, the data values from data source will be filled into these cells.

|   | <b>A</b> | <b>B</b> |
|---|----------|----------|
| 1 |          |          |
| 2 |          |          |
| 3 |          |          |

The fixed table defined in the report template file



|   | <b>A</b> | <b>B</b> |
|---|----------|----------|
| 1 | 14       | 3.4      |
| 2 | 20       | 5.2      |
| 3 | 8        | 2.7      |

The fixed table filled data by rows in the report file

### 4.4.3 Creating a Table for a Variable Table Report

For a variable table report, you also need to create a table in the report template file according to the report. But you just need to reserve some rows/columns in the table for one or two records. WDRReportGen will add some rows/columns according to the number of the records returned from data source.

| <b>Date</b> | <b>Item Id</b> | <b>Sales</b> |
|-------------|----------------|--------------|
|             |                |              |
|             |                |              |



| <b>Date</b> | <b>Item Id</b> | <b>Sales</b> |
|-------------|----------------|--------------|
| 1998-01-01  | 3              | 150          |
| 1998-01-02  | 3              | 200          |
| 1998-01-03  | 3              | 250          |
| 1998-01-05  | 3              | 350          |
| 1998-01-10  | 3              | 550          |
| 1998-01-21  | 3              | 150          |
| 1998-01-25  | 3              | 200          |
| 1998-01-31  | 3              | 100          |

The variable-rows table defined in the report template file

The variable-rows table filled data by rows in the report file

One record from data source can be put into two or more rows/columns. To do this, you need to create a repeat range that includes two or more rows/columns.

The format of the last row/column border can be different from the others. For

example, the outside borders used double lines, and the inside borders used single lines. To do this, you should reserve the blank rows/columns for 2 records. When WDRReportGen inserts some blank rows/columns, the new rows/columns will inherit the format of the first row/column in the reserved rows/columns.

WDRReportGen will repeat the range for each record. Ranges can be nested. The inside range is for detail record, and the external range is for group. WDRReportGen will repeat the inside range for each record, and repeat the group range for each group.


#### **4.4.4 Formatting Cells**

To format cells that contain static contents, use “**Format**” menu in Microsoft Word. To format cells that contain (Formula) fields, use the switches of fields in Microsoft Word. For more detail information, refer to *Microsoft Word Help*.

For cells in which data are got from database, you can set font, color, alignment using Microsoft Word. But to display values in formatting string, you should use other way.

You should write formatting expressions into data cells in the report template file. WDRReportGen will get the text of the cell as a format expression before it puts a value into a cell, and output the value using the format expression. In fact, WDRReportGen calls the format function in Visual Basic. The text got from a cell is used as the format expression in format function. For a variable table report, WDRReportGen will use the format expressions in the reserved rows/columns. For more information about format expression, refer to “Format Expression in Data Cells”.

| Date       | Quantity | Amount  |
|------------|----------|---------|
| yyyy-mm-dd | ###0     | ###0.00 |
| yyyy-mm-dd | ###0     | ###0.00 |
| yyyy-mm-dd | ###0     | ###0.00 |



| Date       | Quantity | Amount   |
|------------|----------|----------|
| 1999-02-18 | 560      | 827.79   |
| 1999-06-14 | 890      | 1,113.05 |
| 2000-01-21 | 1,240    | 1,552.25 |

The table defined in the report template file

The table generated in the report file

A format expression for numbers can have from one to four sections separated by semicolons. You can define the different formats and colors for positive values, negative values and zeros.

For example, the format "\$#,##0;(\$#,##0)" has two sections: the first defines the format and color (black) for positive values and zeros; the second section defines the format and color (red) for negative values. It displays "2345.12" as "\$2,345", displays "-5432" as "(\$5,432)".

The format "#,##0.00;;;" has three sections: the first defines the format and color (black) for positive values, the second defines the format and color (red) for negative values, the third section defines the format and color (blue) for zeros. Note, the first semicolon ";" is red, the second semicolon ";" is blue. The negative values and zeros are printed using the format of the positive value. But the color for negative values is red, the color for zeros is blue. It displays "8.9" as "8.90", displays "-123" as "-123.00", and displays "0" as "0.00".

#### 4.4.5 Irregular Tables

Tables don't have to consist of simple grids. Not every row has to have the same number of columns. You can merge and split cells to create irregular tables. An irregular table is the table that contains split cells or merge cells, and it does not have the same number of cells for each row or column. While an irregular table provides for an attractive way to display data, but it does make it harder to process the document. You have some difficulty to reference a cell in an irregular table. For example in the following table, for most Office version,

cell1 is in column 3 and row 2, cell2 is in column 3 and row 3. But for some lower Office version, cell2 is in column 2 and row 3. Moreover, an error may occur when you try to work with some rows or columns in an irregular table.

| A |  | B     | C |
|---|--|-------|---|
|   |  | Cell1 |   |
|   |  | Cell2 |   |

Irregular table

To simplify your work and ensure that report function can be executed correctly, you should regularize the irregular tables. Split the merge cells, and remove the border in these cells. For example, the following table is a regularized table, cell1 is in column 3 and row 2, and cell2 is in column 3 and row 3.

| A |  | B     | C |
|---|--|-------|---|
|   |  | Cell1 |   |
|   |  | Cell2 |   |

Regularized table

#### 4.4.6 Referencing Cells

You can reference table cells as A1, A2, B1, B2, and so on, with the letter representing a column and the number representing a row. Cell references in Microsoft Word are always absolute references and are not shown with dollar signs. You can reference an entire row or column in a calculation in the following ways:

- Use a range that includes only the letter or number that represents it - for example, 1:1 to reference the first row in the table. This designation allows the calculation to automatically include all the cells in the row if you decide to add other cells later.
- Use a range that includes the specific cells - for example, a1:a3 to reference a column with three rows. This designation allows the calculation

to include only those particular cells. If you add other cells later and you want the calculation to include them, you need to edit the calculation. Unfortunately, there is no inherent capability of Microsoft Word to inform you of the reference of a cell you have selected. You can get around this problem by using a macro. In the sample file “monthly\_sales.doc”, there is a macro called “CellRef”. The macro can tell you the table number and the cell reference. When you position the insertion pointer in a table cell and then run the macro, it displays a message box that shows the table number and the cell reference of the current cell. You can copy the macro “CellRef” into your documents. It is easy for you to reference a cell.

#### **4.4.7 Referencing Tables**

You can reference a table by an index number or a bookmark. The index number represents the position of the table in a document. The index number starts at 1. So table 1 is the first table in a document, table 2 is the second table, and so on. You can reference a nested table inside a table by an index number like 2-1-2. Table 2-1 is the first table inside table 2, and table 2-1-2 is the second table inside table 2-1. The max nested level WDRReportGen supports is 3. In the sample file “monthly\_sales.doc”, there is a macro called “CellRef”. The macro can tell you the index number of a table. When you position the insertion pointer in a table cell and then run the macro, it displays a message box that shows the table number and the cell reference of the current cell.

You can reference a table by the bookmark too. If you add a bookmark in a table, you can reference the table using the bookmark. If you want to reference a nested table inside a table, you must add the bookmark in the nested table. It is very useful if you do not know the number of tables. To add a bookmark in a table, do as follows:

1. Click in the upper-left cell in the first row of the table, and place the insertion point before the text.
2. On the **Insert** menu, click **Bookmark**.
3. Under **Bookmark name**, type or select a name.

Bookmark names must begin with a letter and can contain numbers. You can't include spaces in a bookmark name.

4. Click **Add**.

Note: Microsoft Word 97 or lower does not support the nested tables.

#### 4.4.8 Formatting Cells for Pictures

To enhance the visual impact of your report, you can insert pictures into your report. WDRReportGen supports many popular graphics file formats: bitmap, JPG, GIF, PNG, TIFF and so on. For the graphics file formats WDRReportGen supports, refer to *Microsoft Word Help*.

You should store the path and name of the graphics files in the database, and identify the image fields in the report function. WDRReportGen will read the graphics files, and insert them into the cells in the report file. You can adjust the position of the pictures by changing the cell margins.

To specify the inserted way, text wrapping style and size, you should write a formatting expression into the cell in the report template file. WDRReportGen will get the text of the cell, and insert a picture into the cell according to the instruction in the format expression. The format expression for pictures as follows:

[wrapstyle] [size]

The **wrapstyle** specifies the inserted way and text wrapping style, and can be one of the following values. The default value is NONE. "NONE" means floating pictures.

| Values | Description |
|--------|-------------|
|--------|-------------|

|           |  |
|-----------|--|
| INLINE    | Inline picture                             |
| NONE      | Floating picture, none text-wrapping style |
| SQUARE    | Square text-wrapping style                 |
| THROUGH   | Through text-wrapping style                |
| TIGHT     | Tight text-wrapping style                  |
| TOPBOTTOM | TopBottom text-wrapping style              |

The **size** specifies the size of a picture. Possible values are STRETCH, Wnnn or / and Hnnn. "STRETCH" means that the picture is resized to fit within the cell. "W100" means that the width of the picture is set to 100 points. "H50" means that the height of the picture is set to 50 points. The default means the original size. If you just specify the width or height of the picture, not both, WDRReportGen will retain the original proportions of the picture when WDRReportGen resize it.

### Example

```
inline w120 h90
```

### Remarks

WDRReportGen will insert an inline picture, and set the width of the picture to 120 points, the height to 90 points.

Note: On Microsoft Word 97 or lower version, it may not work correctly if you insert a floating picture into a table. And it will split the table under some text wrapping style. You should insert inline pictures into a table.

## 4.5 Form Reports

### 4.5.1 About Form Reports

Beside table reports, WDRReportGen supports form reports too. For a form report, you can get data from data sources, and put data as text, list, title and table in the report file. So you can make a form report as follows:

## AUSTRALIA

**Company:** G'day, Mate

**Contact:** Wendy Mackenzie      **Title:** Sales Representative

**Address:** 170 Prince Edward Parade Hunter's Hill

**City:** Sydney      **Country:** Australia

**Postal Code:** 2042

**Phone:** (02) 555-5914      **Fax:** (02) 555-4873

**Home Page:** <http://www.microsoft.com/accessdev/sampleapps/gdaymate.htm>

### 4.5.2 Creating Merge Fields or Quote Fields

The report template file of a form report must contain merge fields or quote fields where data values will be inserted. A field can be created in text, list, title, table or other. When WDRReportGen is run, it will replace the merge fields or quote fields with values from data source.

To create a merge field or quote field:

1. Click where you want to insert a field.
2. On the **Insert** menu, click **Field**.
3. From the **Field names** list, select **MergeField** or **Quote**.
4. In the **Field name** text box, enter a name for the merge field or quote field.

For example, enter ProductName. Remember the field name, you will use it in the report function.

You can show field codes, and edit the field codes. To switch between field codes and results, do one of the following:

- To show or hide the field code for a specific field, click the field or the field results, and then press SHIFT+F9.
- To show or hide field codes for all fields in the document, press ALT+F9.

If the data value is a number, date or time, and you want to display it in a custom format, use the switches of fields like “\#” or “\@”. For more detail information, refer to *Microsoft Word Help*.

### 4.5.3 Defining Ranges

A range represents a contiguous area in a document, and contains text, lists, tables, table rows or paragraphs. When WDRReportGen generates a report, it will repeat the range for each record or group.

A range can be defined by a Word bookmark. Or the entire document will be defined as the default range.

To define a range with a bookmark:

1. Create a document as your report template. For example, you create a template as follows:

| Product Name  | Product ID  | Quantity Per Unit | Unit Price  |
|---------------|-------------|-------------------|-------------|
| «ProductName» | «ProductID» | «QuantityPerUnit» | «UnitPrice» |

2. Select the range that you want to define as a repeat block. For example, you select a table row.

| Product Name  | Product ID  | Quantity Per Unit | Unit Price  |
|---------------|-------------|-------------------|-------------|
| «ProductName» | «ProductID» | «QuantityPerUnit» | «UnitPrice» |

3. On the **Insert** menu, click **Bookmark**.
4. Under **Bookmark name**, enter a bookmark name and click **Add**. For example, enter Product.

In Microsoft Word, bookmarks are hidden by default. To show bookmarks:

1. On the **Tools** menu, click **Options**, and then click the **View** tab.
2. Select the **Bookmarks** check box.
3. The bookmark appears in brackets ([...]) on the screen.

| Product Name    | Product ID  | Quantity Per Unit | Unit Price  |
|-----------------|-------------|-------------------|-------------|
| [«ProductName»] | «ProductID» | «QuantityPerUnit» | «UnitPrice» |

Ranges can be nested. The inside range is for detail record, and the external range is for group. For example, you defines a bookmark Category for the group of product category, and a bookmark Product for the detail record of

product.

«**CategoryName**»

«Description»

| Product Name  | Product ID  | Quantity Per Unit | Unit Price  |
|---------------|-------------|-------------------|-------------|
| «ProductName» | «ProductID» | «QuantityPerUnit» | «UnitPrice» |

#### 4.5.4 Formatting Fields for Pictures

To enhance the visual impact of your report, you can insert pictures into your report. WDRReportGen supports many popular graphics file formats: bitmap, JPG, GIF, PNG, TIFF and so on. For the graphics file formats WDRReportGen supports, refer to *Microsoft Word Help*.

You should store the path and name of the graphics files in the database, and identify the image fields in the report function. WDRReportGen will read the graphics files, and insert them into the fields in the report file.

To specify the inserted way, text wrapping style and the size, you should write a formatting expression in the field switch “\#” in the report template file.

WDRReportGen will get the format string in the field switch “\#”, and insert a picture according to the instruction in the format expression. The format expression for pictures as follows:

[wrapstyle] [size]

The **wrapstyle** specifies the inserted way and text wrapping style, and can be one of the following values. The default value is INLINE. “INLINE” means inline pictures.

| Values | Description                                |
|--------|--|
| INLINE | Inline picture                             |
| NONE   | Floating picture, none text-wrapping style |
| SQUARE | Square text-wrapping style                 |

|           |                               |
|-----------|-------------------------------|
| THROUGH   | Through text-wrapping style   |
| TIGHT     | Tight text-wrapping style     |
| TOPBOTTOM | TopBottom text-wrapping style |

The **size** specifies the size of a picture. Possible values are Wnnn or / and Hnnn. “W100” means that the width of the picture is set to 100 points. “H50” means that the height of the picture is set to 50 points. The default means the original size. If you just specify the width or height of the picture, not both, WDRReportGen will retain the original proportions of the picture when WDRReportGen resize it.

### Example

```
/# “square w84”
```

### Remarks

On the supposition that the original picture is size 144 x 168 points. WDRReportGen will insert a floating picture, apply the square text-wrapping style, set the height of the picture to 72 points, and the width to 84 points.

## 4.6 Charts

### 4.6.1 About Charts

You can create many different types of charts in Microsoft Word. The chart software may be Microsoft Graph or Microsoft Excel. It is depended on the version of your Microsoft Word. For Microsoft Word 2003 or earlier, the default chart software is Microsoft Graph. For Microsoft Word 2007 or later, the default chart software is Microsoft Excel.

WDRReportGen supports two kinds of charts created by Microsoft Graph or Excel. It executes a SQL statement, and puts the result data into the datasheet or worksheet of the chart. To work with charts created in Graph or Excel, you must have Graph or Excel installed.

## 4.6.2 Creating a Blank Chart using Microsoft Graph

To create a Graph chart in the report using WDRReportGen, you need to add a Graph chart in the report template file first. The chart will be brought into the report file with the same chart type, display option, data format, label format and other chart item

If your Microsoft Office is earlier than Office 2007, or Microsoft Excel 2007 is not installed, when you create a new chart in Microsoft Word, Microsoft Graph opens.

To add a Graph chart in the template file:

1. Open the report template file using Microsoft Word.
2. On the **Insert** menu, click **Chart**.
3. Change the sample data on the datasheet as you need.
4. Modify the chart. For example, you want to change the chart type, make the text larger, or change colors, patterns, lines, fills, and borders in charts.
5. After you have finished the modification, delete data from the chart. You should keep a blank chart in the report template file. WDRReportGen will put data into the datasheet of the chart.
6. Change the chart to an inline shape if it is a floating shape. On the **Format** menu, click **Object**, click the **Layout** tab, and then click **In Line** text-wrapping style.

For more detail information, refer to *Microsoft Word Help and Microsoft Graph Help*.

## 4.6.3 Creating a Blank Chart using Microsoft Excel

To create an Excel chart in the report using WDRReportGen, you need to add an Excel chart object in the report template file first. The chart will be brought into the report file with the same chart type, display option, data format, label

format and other chart item.

To add an Excel chart in the template file:

1. Open the report template file using Microsoft Word.
2. Insert a chart with a chart sheet and a worksheet. For more information to insert an Excel chart object in Microsoft Word, please refer to the following part.
3. Change the sample data on the worksheet as you need.
4. Modify the chart. For example, you want to change the chart type, make the text larger, or change colors, patterns, lines, fills, and borders in charts.
  - If the report type is fix, the data range of the chart should be all rows/columns for the returned records.
  - If the report type is var, the data range of the chart should be 2 rows/columns.
5. After you have finished the modification, delete data from the chart. You should keep a blank chart in the report template file, and make the chart sheet active. WDRReportGen will put data into the worksheet of the chart.
6. Change the chart to an inline shape if it is a floating shape. On the **Format** menu, click **Object**, click the **Layout** tab, and then click **In Line** text-wrapping style.

By default, Microsoft Word 2007 uses Microsoft Excel to create charts, but doesn't expose the chart as a normal Excel object. There are two ways to create Excel chart object.

The first way for Microsoft Word 2007: Create a chart in Microsoft Excel, and copy it into Microsoft Word as an Excel chart object.

1. Open Microsoft Word and Microsoft Excel.
2. In Microsoft Excel, create a chart, and move the chart to a new worksheet.
3. In Microsoft Excel, click on the chart, and copy the chart.
4. Switch to Microsoft Word, place the insertion point where you want to create

the chart.

5. On the **Home** tab, in the **Clipboard** group, click **Paste Special** and choose **Microsoft Office Excel Chart Object**. And click **OK** button.

The second way for Microsoft Word 2007: Insert an Excel worksheet in Microsoft Word, and create a chart in the Excel worksheet object.

1. In Microsoft Word, place the insertion point where you want to create the chart.

2. On the **Insert** tab, in the **Tables** group, click **Table**, and then click **Excel Spreadsheet**. You will see an Excel worksheet object.

3. Right-click the object, point to **Worksheet Object** on the shortcut menu, and choose **Open** from the submenu. Microsoft Excel will appear.

4. Create a chart in Microsoft Excel, and move the chart to a new worksheet.

5. When you've finished, choose **Close & Return** from the **File** menu.

For Microsoft Word 2003 or earlier, you can import Excel charts object.

1. In Microsoft Word, place the insertion point where you want to create the chart.

2. Click **Object** on the **Insert** menu, and then select the **Microsoft Excel Chart**.

3. You can work the Excel chart object by right-clicking the object, and pointing to **Worksheet Object** on the shortcut menu, and choosing **Open** from the submenu.

4. When you've finished, choose **Close & Return** from the **File** menu.

For more detail information, refer to *Microsoft Word Help and Microsoft Excel Help*.

#### **4.6.4 Referencing Charts**

You can reference a chart by an index number or a bookmark. The index number represents the position of the chart in a document. The index number

starts at 1. So chart 1 is the first chart in a document, chart 2 is the second chart, and so on.

You can reference a chart by the bookmark too. You might use a bookmark to identify a chart if you assigned the bookmark to the chart. To add a bookmark, do as follows:

1. Create a chart in the report template.
2. Select the chart you want a bookmark assigned to.
3. On the **Insert** menu, click **Bookmark**.
4. Under **Bookmark name**, enter a bookmark name and click **Add**. For example, enter Chart1.

In Microsoft Word, bookmarks are hidden by default. To show bookmarks:

1. On the **Tools** menu, click **Options**, and then click the **View** tab.
2. Select the **Bookmarks** check box.
3. The bookmark appears in brackets ([...]) on the screen.

# Chapter 5 Reporting with WDRReportGen

## 5.1 Creating and Opening WRF Files

### 5.1.1 About WRF files

To generate a report with WDRReportGen, you must create a WRF file with a .wrf extension. The WRF file contains information such as the name of the report template file, the name of the report file, the log file name, data sources, parameters and functions. The WRF file tells WDRReportGen how to get data from data sources and how to put data into a report.

### 5.1.2 Create a new WRF file

On the **File** menu, click **New**.

### 5.1.3 Open a WRF file

1. On the **File** menu, click **Open**.
2. In the **Look in** list, click the drive, folder, or Internet location that contains the file you want to open.
3. In the folder list, locate and open the folder that contains the file.
4. Click the file, and then press **Open** button.

### 5.1.4 Save a WRF file

On the **File** menu, click **Save**. If you're saving the file for the first time, you'll be asked to give it a name.

If you want save a file to another name, do as follows:

1. On the **File** menu, click **Save As**.
2. In the **File name** box, enter a new name for the file.

3. Press **Save** button.

## 5.2 Configuring Files

### 5.2.1 About files

You should specify the report template file, report file, report file type and log file. The report template file defines layouts, formats and styles of the report. The report file is the report you want to generate. The type of the report file can be different from the template file. The log file records the log information in the report generating.

The file path can be a relative path or an absolute path. If it is a relative path, the base path is the path of the WRF file. In the paths and names of report file, template file and log file, you can use parameters. For detailed information about parameters, refer to “Configuring Parameters” in this document.

### 5.2.2 Configuring file information

1. On the **Report** menu, click **Configuration**. The **Configuration** dialog box appears.
2. Click the **File** tab.
3. Input the path and name of the template file, the report file and the log file into their text box.
4. In the **File Type** box, click the file type you want. If the file type of the report is same as the template file, click the **(Default)** in the **File Type** box.  
WDRReportGen will display the converter type.
5. If you want to protect the report, select the **Protect Report** check box. If the check box is selected, the Word report generated is protected, and can not be modified. If you select **Random Password** option button, a random password will be created to protect the report. If you select **Input Password** option

button, you can input a password to protect the report.

6. Press **OK** button to confirm the changes, press **Cancel** button to discard the changes.

### 5.2.3 Converting files

You can convert a file from Microsoft Word document to and from another file format. For example, the template file is a RTF file with a .rtf extension, and the report file is a HTML file with a .htm extension. Microsoft Word uses file format converters to open and save documents in different formats. The most commonly used converters are installed with Microsoft Word by default. If you want to open or save in a format that's not installed by default, you may need to install additional converters.

The file formats WDRReportGen supports can be one of these. What file format WDRReportGen supports is dependent on your Microsoft Word and converters installed. For example, Microsoft Word 2003 supports XML, but Microsoft Word 97/2000 does not support it. For more information about converting files, please refer to *Microsoft Word Help*. The file "wconv.cfg" located in the WDRReportGen directory contains the information of file formats. You can expand it if your Microsoft Word supports more file formats.

| File Format Name          | Value | Description            | Extension | Converter  |
|---------------------------|-------|------------------------|-----------|------------|
| wdFormatDocument          | 0     | Word Document          | doc       | Office97   |
| wdFormatTemplate          | 1     | Word Template          | dot       | Office97   |
| wdFormatRTF               | 6     | Rich Text Format       | rtf       | Office97   |
| wdFormatHTML              | 8     | Web Page               | htm html  | Office2000 |
| wdFormatFilteredHTML      | 10    | Web Page Filtered      | htm html  | Office2003 |
| wdFormatWebArchive        | 9     | Single File Web Page   | mht mhtml | Office2003 |
| wdFormatXML               | 11    | XML Document           | xml       | Office2003 |
| wdFormatText              | 2     | Text                   | txt       | Office97   |
| wdFormatTextLineBreaks    | 3     | Text (Line Breaks)     | txt       | Office97   |
| wdFormatDOSText           | 4     | Text (DOS)             | txt       | Office97   |
| wdFormatDOSTextLineBreaks | 5     | Text (DOS Line Breaks) | txt       | Office97   |
| wdFormatUnicodeText       | 7     | Unicode Text           | txt       | Office97   |

|               |  |                                |         |          |
|---------------|--|--------------------------------|---------|----------|
| MSWordWin2    |  | Word 2.x for Windows           | doc     | External |
| MSWordWin5    |  | Word 5.0 for Windows           | doc     | External |
| MSWord6RTFExp |  | Word 97-2003 & 6.0/95<br>– RTF | doc     | External |
| wks632        |  | Works 6.0 & 7.0                | wps     | External |
| MSWorksWin6   |  | Works 6.0 & 7.0                | wtf     | External |
| Works2001     |  | Works 7.0                      | wps     | External |
| WrdPrfctDos   |  | WordPerfect 5.x                | doc     | External |
| WordPerfect6x |  | WordPerfect 6.x                | wpd doc | External |

For Microsoft Word 2007, please copy “wconv2007.cfg” to “wconv.cfg”. This file contains the information of file formats for Microsoft Word 2007.

| File Format Name                | Value | Description                          | Extension  |
|---------------------------------|-------|--------------------------------------|------------|
| WdFormatXMLDocument             | 12    | Word Document                        | Docx       |
| wdFormatXMLDocumentMacroEnabled | 13    | Word Macro-enabled Document          | Docm       |
| WdFormatDocument                | 0     | Word 97-2003 Document                | Doc        |
| WdFormatXMLTemplate             | 14    | Word Template                        | Dotx       |
| wdFormatXMLTemplateMacroEnabled | 15    | Word Macro-enabled Document Template | Dotm       |
| WdFormatTemplate                | 1     | Word 97-2003 Template                | Dot        |
| WdFormatRTF                     | 6     | Rich Text Format                     | Rtf        |
| WdFormatHTML                    | 8     | Web Page                             | htm html   |
| WdFormatFilteredHTML            | 10    | Web Page Filtered                    | htm html   |
| WdFormatWebArchive              | 9     | Single File Web Page                 | mhtm mhtml |
| WdFormatPDF                     | 17    | PDF                                  | Pdf        |
| WdFormatXPS                     | 18    | XPS Document                         | Xps        |
| WdFormatFlatXML                 | 19    | Word XML Document                    | Xml        |
| WdFormatFlatXMLMacroEnabled     | 20    | Word XML Macro-enabled Document      | Xml        |
| WdFormatXML                     | 11    | Word 2003 XML Document               | Xml        |
| WdFormatText                    | 2     | Text                                 | Txt        |
| WdFormatTextLineBreaks          | 3     | Text (Line Breaks)                   | Txt        |
| WdFormatDOSText                 | 4     | Text (DOS)                           | Txt        |
| WdFormatDOSTextLineBreaks       | 5     | Text (DOS Line Breaks)               | Txt        |
| WdFormatUnicodeText             | 7     | Unicode Text                         | Txt        |
| MSWorksWin6                     |       | Works 6.0 - 9.0                      | Wtf        |
| wks632                          |       | Works 6.0 - 9.0                      | Wps        |
| Works2001                       |       | Works 7.0                            | Wps        |
| WrdPrfctDos                     |       | WordPerfect 5.x                      | Doc        |

|               |  |                 |         |
|---------------|--|-----------------|---------|
| WordPerfect6x |  | WordPerfect 6.x | wpd doc |
|---------------|--|-----------------|---------|

Note: For an external converter, the file format name is the class name of the file converter, and the format value may not be the same on another computer. So do not assign a number to the format value.

## 5.3 Configuring Data Sources

### 5.3.1 About data sources

A data source identifies a database computer you want to access. Because of accessing data through ODBC, WDRReportGen can access a wide range of data sources, such as Oracle, DB2, Sybase, Informix, Microsoft SQL Server, Teradata, MySQL, Microsoft Access, dBase. WDRReportGen supports more than one data sources in one report. You can get data from some different databases such as Oracle, DB2 and Microsoft SQL Server, and put them into one report.

### 5.3.2 Adding, modifying and deleting a data source

1. On the **Report** menu, click **Configuration**. The **Configuration** dialog box appears.
2. Click the **Data Source** tab.
3. If you want to add a data source, press **New** button, the **New Data Source** dialog box appears.
  - To define a connection using an ODBC data source name, click **Using ODBC data source name** option, input data source name, user name and password, press **OK** button.
  - To define a connection using a connection string, click **Using connection string** option, input data source name, and connection string, press **OK** button.

4. If you want to modify a data source, click the data source name in the **Data Source** list box, and press **Edit** button, the **Edit Data Source** dialog box appears.

- To define a connection using an ODBC data source name, click **Using ODBC data source name** option, change data source name, user name and password, press **OK** button.
- To define a connection using a connection string, click **Using connection string** option, change data source name, and connection string, press **OK** button.

5. If you want to delete a data source, click the data source name in the **Data Source** list box, and press **Delete** button, the confirmation dialog box appears. Press **Yes** button to delete the data source.

6. You can test a data source. Click the data source name in the **Data Source** list box, and Press **Test** button to display the information of connection to the data source.

7. Select or clear the **Encrypt Password** check box. If the check box is selected, passwords will be saved in an encrypted format. Or passwords will be saved in plain text.

8. Press **OK** button to confirm the changes, press **Cancel** button to discard the changes.

## 5.4 Configuring Parameters

### 5.4.1 About parameters

You can use parameters in SQL statements. These values need to be provided to WDRReportGen before it executes these SQL statements. To use a parameter, you must declare it first. When WDRReportGen generate a report, it will prompt you to input the value of the parameter. WDRReportGen will replace

the parameter name in the SQL statements with the actual value before it submits the SQL statements to data sources.

A parameter has a name, a title and a default value. The name of a parameter identifies the parameter. You can use the names in SQL statements. The titles will be displayed in the prompt dialog box when WDRReportGen is run.

Note: WDRReportGen will replace all strings that are the same as the names of the parameters. You should be careful to define a unique name for each parameter. It is a good choice a name begins with the “\$” character. For example, you give the name “\$ReportDate” for a parameter. Parameters are case-sensitive.

#### **5.4.2 Adding, modifying and deleting a parameter**

1. On the **Report** menu, click **Configuration**. The **Configuration** dialog box appears.
2. Click the **Parameter** tab.
3. If you want to add a parameter, press **New** button, the **New Parameter** dialog box appears. Input parameter name, parameter title and default value, press **OK** button.
4. If you want to modify a parameter, click the parameter name in the **Parameter** list box, and press **Edit** button, the **Edit Parameter** dialog box appears. Change the name, title and default value of the parameter, press **OK** button.
5. If you want to delete a parameter, click the parameter name in the **Parameter** list box, and press **Delete** button, the confirmation dialog box appears. Press **Yes** button to delete the parameter.
6. Press **OK** button to confirm the changes, press **Cancel** button to discard the changes.

## 5.5 Inputting Functions

You should input functions in the editor window. A function includes a SQL statement and some arguments. WDRReportGen executes the SQL statement, and determines whether or how to add data into the report. WDRReportGen sequentially executes functions.

Each function is begin with the "@" character. Syntax:

```
@functionno=functionname(arguments)  
sqlstatement
```

The *functionno* is a label of the report function.

The *functionname* represents a report function.

The *arguments* for a function define various properties for the function. For example, the "table" argument identifies a table in the Microsoft Word document. An argument takes the form *Name="Value"*. The argument value can be delimited by single or double quotes.

The *sqlstatement* is a SQL statement.

For more detailed information about functions, see "Function Reference" in this document.

You can use comments in text. A comment is the "/\*" characters, followed by any sequence of characters (including new lines), followed by the "\*/" characters. You cannot nest comments.

## 5.6 Running WRF Files

You can run a WRF file to generate a report in Microsoft Word document format. WDRReportGen supports Windows mode and command line mode.

### 5.6.1 Windows mode

1. On the **Report** menu, click **Run**, the **Run Report** dialog box appears.
2. If you want to display the generated report, select the **Display Report with Microsoft Word** check box.
3. Press **Start** button to run the WRF file.
4. If parameters are defined in the WRF file, WDRReportGen will pop up a prompt dialog box. Input the values of the parameters, and press **OK** button.
5. While WDRReportGen is being run, it will display some information such as status, SQL count, error count, function No., records count and log information.
6. You can interrupt the running. Click **End** button to interrupt it. WDRReportGen will immediately save and close the report.
7. Click **Close** button after completion.
8. If you want to open the report, click **Open Report File** on the **File** menu.
9. If you want to check the log, click **Open Log File** on the **File** menu.

### 5.6.2 Command line mode

You can run a WRF file in command line. You have defined two parameters in the WRF file "myreport.wrf". The first parameter is sales date "\$SalesDate", and the second is the category of the products "\$Category". You can run WDRReportGen in command line mode as follows:

```
wordreport c:\WordReport\myreport.wrf -c 1996-05-01 "Dairy Products"
```

WDRReportGen will replace "\$SalesDate" in SQL statements with "1996-05-01", replace "\$Category" with "Dairy Products", and then submit SQL statements to data sources.

## 5.7 Using Formula Fields

In Microsoft Word, you can use (Formula) field to calculates a number. To

insert an (Formula) field in a table or in regular text, you can use the **Formula** command (**Table** menu) or press CTRL+F9. For example, the following formula multiplies 2 by 3 and then adds 5 to the result.

{ = 5+2\*3 \# "#,##0.00"}

In an (Formula) field, an expression that can contain any combination of numbers, bookmarks that refer to numbers, fields resulting in numbers, and the available operators and functions. The expression can refer to values in a table and values returned by functions. For more detail information about (Formula) field, refer to *Microsoft Word Help*.

In a report template file, you can use all kind of (Formula) field. And they will be brought to the final report file.

### Example

Add totals such as Total Quantity, Total Amunt.

You can use (Formula) field of Microsoft Word.

1. Create a template file as follows, define the formula of total quantity as "=SUM(ABOVE)" in cell B3, and the formula of total amount as "=SUM(ABOVE)" in cell C3.

| Category Name | Quantity           | Amount             |
|---------------|--------------------|--------------------|
|               | #,##0              | \$#,##0.00         |
| <b>Total</b>  | { =SUM(ABOVE) \# } | { =SUM(ABOVE) \# } |

2. Write report function as follow.

@F2=REPORT(table=Report2 type=var cell=A2)

SELECT c.CategoryName, SUM(d.Quantity), Sum(d.UnitPrice \* d.Quantity \* (1-d.Discount))

FROM Orders o

,OrderDetails d

,Products p

,Categories c

WHERE o.OrderID = d.OrderID

```

AND d.ProductID = p.ProductID
AND p.CategoryID = c.CategoryID
AND YEAR(o.OrderDate) = YEAR('$ReportMonth-01')
AND MONTH(o.OrderDate) = MONTH('$ReportMonth-01')
GROUP BY c.CategoryName
ORDER BY c.CategoryName
;

```

3. The report generated by WDRReportGen is as follows.

| Category Name  | Quantity     | Amount              |
|----------------|--------------|---------------------|
| Beverages      | 925          | \$27,761.57         |
| Condiments     | 378          | \$10,773.27         |
| Confections    | 880          | \$22,877.18         |
| Dairy Products | 581          | \$13,685.32         |
| Grains/Cereals | 189          | \$3,325.40          |
| Meat/Poultry   | 92           | \$4,083.66          |
| Produce        | 351          | \$13,031.20         |
| Seafood        | 669          | \$9,316.54          |
| <b>Total</b>   | <b>4,065</b> | <b>\$104,854.14</b> |

## 5.8 Sorting, Grouping and Totaling

### 5.8.1 Sorting data

Sorting means placing data in some kind of order to help you find and evaluate it. For example, you may want to have a customer list sorted alphabetically by name or by country.

To sort your data, you can use SQL. Use the **ORDER BY** clause to have your results displayed in a sorted order.

```

SELECT EmployeeID
,LastName
,FirstName
,HireDate
FROM Employees

```

```
ORDER BY HireDate; /* ascending sort */
```

In the example above, results will come back in ascending order by hire date. To explicitly specify ascending or descending order, add ASC or DESC, to the end of your ORDER BY clause. The following is an example of a descending order sort.

```
ORDER BY HireDate DESC; /* descending sort */
```

## **5.8.2 Totaling**

You can sum the values, count all the values or only those values that are distinct from one another, and determine the maximum, minimum, average. To add totals, there are two ways.

1. You can add total using (Formula) fields of Microsoft Word, such as SUM(above). For more detail information, refer to “Using Formula Fields” in this document.
2. You can use aggregate functions in SQL statement, such as COUNT, SUM, AVG, MAX, MIN.
  - (1) In the fixed table report, you can add total directly using a separate SQL.
  - (2) In the variable table report, you must add total first using a Fixed Table report function before you use the Variable Table report function. Because the cell address of the total field will change after you use Variable Table report function.

## **5.8.3 Grouping data and subreports**

Grouped data is data that is sorted and broken up into meaningful groups. In a customer list, for example, a group might consist of all those customers living in the same Region.

To group data in a report, you should use GROUP VARIABLE TABLE REPORT function. For more detail information, refer to “Group Variable Table

Report” in this document.

Using GROUP REPORT function, you can make subreports within a report. A subreport would typically be used to perform one-to-many lookups such as Customer / Order / OrderDetails.

To make sub reports within the main report,

1. Write a JOIN SQL statement to access data from two or more tables. For example, you can join Customers, Orders and OrderDetails tables.
2. Use GROUP VARIABLE TABLE REPORT function.

For more detail information, refer to the samples invoice.wrf, product\_catalog.wrf and sales\_detail.wrf within WDRReportGen.

#### **5.8.4 Subtotaling**

A subtotal is a summary that totals or sums numeric values in a group. You can sum the values in each group, count all the values in each group, and determine the maximum, minimum, average in each group. For example, determine the total sales per sales representative in a sales report.

To add subtotals, you can use aggregate function in SQL statement.

1. Use aggregate function and GROUP BY clause, get summary data for each group, and insert results into a temporary table.
2. If you have different kinds of summaries, repeat the step 1, and insert results into another temporary table.
3. Use group report function, and join the detail data and the summary data using JOIN. The summary fields must be included in the group list.
4. Except for sub-totals, you can add total too using aggregate function in SQL statement. You must add total first using a Fixed Table report function before you use the Variable Table report function. Because the cell address of the total field will change after you use Variable Table report function.

For more information, refer to the samples invoice.wrf and sales\_detail.wrf

within WDRReportGen.

## **5.9 Pictures**

### **5.9.1 Inserting pictures into a report template**

To make eye-catching reports, you can add pictures to your reports. You can insert pictures into the report template directly in Microsoft Word. For example, you want to display a logo in your report. You can insert the logo graphics file into the report template. For more information about adding pictures to documents, refer to *Microsoft Word Help*.

### **5.9.2 Inserting pictures into a report**

Except for inserting the static pictures during report design, you want to insert pictures during report building process. You hope a reporting tool to pull pictures from database into Word report. WDRReportGen can insert pictures from the graphics files, and support all graphics file format that Microsoft Word support.

To insert pictures into a report using WDRReportGen, you should do as follows:

1. Store the path and name of the graphics files in the database

You stored the path and file name of the pictures in database, did not store the pictures. The file path can be a relative path, an absolute path or a URL. For example, you store "images\emp1.jpg" in Photo field.

2. Identify the image fields in the report function

Write a report function in the WRF file, and identify the image fields using IMAGE argument. For example,

```
@F1=Report(table=1 ... image=photo)
```

3. Specify the inserted way, text wrapping style and size in the report template

To specify the inserted way, text wrapping style and size, you should write a

formatting expression in the report template file. For a table report, you write a formatting expression in the cell. For a form report, write a formatting expression in the field switch "\#". WDRReportGen will get the formatting expression, and insert a picture into the report according to the instruction in the format expression.

#### 4. Run WDRReportGen to generate report with pictures

During report generating process, WDRReportGen will read the graphics files, and insert them into the report according to your instruction. If the path and file name of the picture is "", WDRReportGen will return "". WDRReportGen will return "#Error" if it does not find the file of the picture.

For more detail information about pictures, refer to the samples employee\_profile.wrf, product\_catalog.wrf within WDRReportGen.

## 5.10 Using Parameters

To use a parameter, you must define it first. If you have defined a parameter name, you can use it in SQL statements. When WDRReportGen is run, it will replace the parameter name in the SQL statements with the actual value before it submits the SQL statements to data sources. Besides in SQL statements, you can use parameters in the paths and names of report file and log file.

In fact, WDRReportGen will replace all strings that are the same as the names of the parameters. You should be careful to define a unique name for each parameter. It is a good choice a name begins with the "\$" character.

### Example

Input an order id to get the order information. The field OrderID is numeric type.

#### 1. Defining a parameter

Define a parameter as follows:

Name: \$OrderID

Title: Order ID (>=10248)

Default: 10360

## 2. Using a parameter

You can use the parameter "\$OrderID" in SQL statements. For example:

```
SELECT o.OrderID
,o.OrderDate
,SUM(d.UnitPrice * d.Quantity * (1-d.Discount)) AS Amount
FROM Orders o, OrderDetails d
WHERE o.OrderID = d.OrderID
AND o.OrderID = $OrderID
GROUP BY o.OrderID, o.OrderDate
;
```

### **Example**

Define two parameters. The first parameter is sales date, and the second is the category of the products. The field OrderDate is date type, and CategoryName is char type.

## 1. Defining parameters

Define parameters as follows:

Name1: \$SalesDate

Title1: Sales Date

Default1: 1996-05-01

Name2: \$Category

Title2: Category of Products

Default2:

## 2. Using parameters

You can use the parameters "\$SalesDate", "\$Category" in SQL statements.

For example:

```
SELECT .....  
FROM Orders, OrderDetails, Products, Categories  
WHERE .....  
AND OrderDate = '$SalesDate'  
AND CategoryName LIKE '$Category%'  
;  
/* For Microsoft Jet SQL, LIKE '$Category*' */
```

### **Example**

Get the information from the database, table and column that you identify when the report is generated.

#### 1. Defining parameters

Define parameters as follows:

Name1: \$Database

Title1: Database Name

Default1:

Name2: \$Table

Title2: Table Name

Default2:

Name3: \$Column

Title3: Column Name

Default3:

#### 2. Using parameters

You can use the parameters "\$Database", "\$Table" and "\$Column" in SQL statements. For example:

```
USE $Database;
```

or

```
DATABASE $Database;
```

```
SELECT $Column  
FROM $Table  
;
```

### **Example**

Use parameters in the path and name of the report file and log file.

#### 1. Defining a parameter

Define a parameter as follows:

Name: \$CustomerID

Title: Customer ID

Default: C000001

#### 2. Using a parameter

ReportFileName=report\report\_{\$CustomerID}.doc

LogFileName=log\report\_{\$CustomerID}.log

or

ReportFileName=report\{\$CustomerID}report.doc

LogFileName=log\{\$CustomerID}report.log

## **5.11 Programming**

### **5.11.1 Using add-ins, macros**

In Microsoft Word, you can automate a task with a macro. A macro is a series of commands and functions that are stored in a Microsoft Visual Basic module and can be run whenever you need to perform the task.

You can write macros in the report template file, and can use automatic macros, such as AutoOpen, AutoClose to automate a task. For examples, you can use AutoOpen macro to make the template, or use AutoClose to change the report after WDRReportGen puts data into the report.

### 5.11.2 Making WRF files programmatically

Sometimes you want to make a WRF file programmatically. You can do this because the WRF file is a text file. You can write a program to make a WRF file using C, perl or DOS shell, and then run WDRReportGen to generate report.

The two steps can be written into a batch file.

1. Write a program to make the WRF file as you need.
2. Write a batch file to call the program and WDRReportGen in command line mode.

For example, you write a batch file runrpt.bat as follows. changewrf is an executable file that reads template.txt and output template.wrf. First runrpt.bat call changewrf to make the WRF file, and then call WDRReportGen to generate the report.

```
@echo off
if "%1"==" " goto usage
goto process
:usage
echo Usage: runrpt ReportDate
echo ReportDate   Date format 'YYYY-MM-DD'
goto :EOF
:process
changepwrf %1 <"template.txt" >"template.wrf"
WordReport "template.wrf" -C %1
```

# Chapter 6 Function Reference

## 6.1 Report Function

### 6.1.1 Report Function

The REPORT function executes a SQL statement to get data from data source, and put data into a table or range in the report file. The REPORT function can make four types of reports:

- Fixed table report
- Non-group variable table report
- Group variable table report
- Form report.

### 6.1.2 Fixed Table Report Function

In a fixed table report, the number of rows and columns is fixed. WDRReportGen executes a SQL statement to get data from data source, and directly fills data vales into the cells of a table in the report file.

#### Syntax

```
Report (...)  
sqlstatement
```

#### Arguments

```
TYPE = "fix"  
TABLE = table  
FILLORDER = fillorder  
CELL= celllist  
RANGE = range  
IMAGE = fieldlist
```

PAGEBREAK = *pagelength*

CONNECT = *datasource*

The **TYPE** argument specifies the report type. "fix" means a fixed table report. The **TABLE** argument identifies a table in the report template. The *table* is the index number of the table or the bookmark name in the table. The index number starts at 1. For examples, table 2 is the second table in the document. The index number of a nested table likes 2-1-2. For examples, table 2-1 is the first table inside table 2, and table 2-1-2 is the second table inside table 2-1. The max nested level WDRReportGen supports is 3.

The **FILLORDER** argument specifies the order in which WDRReportGen fill data. Possible values are row or col. "row" means to fill data by rows, and "col" means to fill data by columns. Default is row.

The **CELL** argument specifies the positions where data values will be inserted. The *celllist* is the list of cells separated by the “,” character. For example, “A2,B2,B3,D2,D3”. The cells in the *celllist* should correspond to the data source fields in the SQL statement. The value of the first field is put into the first cell, and the value of the second field is put into the second cell ..... WDRReportGen will use the next cell if you omit a cell except the first cell. If FILLORDER=“row”, the next cell is the right cell. If FILLORDER=“col”, the next cell is the below cell.

The **RANGE** argument specifies the range in the table to be used for the records. WDRReportGen will skip the range for each record. You can reference a range of cells like “2:4” or “B:D”. The default range is the area that includes all cells for the records.

The **IMAGE** argument specifies the data source fields are picture files. The *fieldlist* is the list of data source fields separated by the “,” character. You can identify a field using the name of field or the index number of field, but not simultaneously. In data source, you stored the path and file name of the picture,

not the picture. The file path can be a relative path, an absolute path or a URL. If it is a relative path, the base path is the path of the report template file. The **PAGEBREAK** argument specifies the page breaks. The unit of page length is r that means record. For example, "6r" or "6" means that WDRReportGen will insert a page break per 6 records. Default is no page break. The **CONNECT** argument specifies the connection to a data source. The CONNECT can takes a string that expresses a data source name or a number that expresses a data source index. The index number of data source is the sequential number defined in the WRF file, and starts at 1. The default implies the first data source.

The *sqlstatement* is a SQL statement such as a SELECT statement.

### Example

The following function makes the report: Top 5 Employees for Sales.

```
@F1=REPORT(type=fix table=6 cell=B2)
```

```
SELECT TOP 5 e.FirstName + ' ' + e.LastName
      , SUM(d.Quantity)
      , Sum(d.UnitPrice * d.Quantity * (1-d.Discount)) AS SalesAmount
FROM Orders o
      ,OrderDetails d
      ,Products p
      ,Employees e
WHERE o.OrderID = d.OrderID
AND d.ProductID = p.ProductID
AND o.EmployeeID = e.EmployeeID
AND YEAR(o.OrderDate) = 1996
AND MONTH(o.OrderDate) = 04
GROUP BY e.FirstName, e.LastName
```

ORDER BY 3 DESC

;

## Result

The fixed table report defined in the report template:

| Rank | Employee Name | Quantity | Amount     | Percent of Total |
|------|---------------|----------|------------|------------------|
| 1    |               | ###0     | \$#,##0.00 | 0.00%            |
| 2    |               | ###0     | \$#,##0.00 | 0.00%            |
| 3    |               | ###0     | \$#,##0.00 | 0.00%            |
| 4    |               | ###0     | \$#,##0.00 | 0.00%            |
| 5    |               | ###0     | \$#,##0.00 | 0.00%            |

The fixed table report generated in the report:

| Rank | Employee Name    | Quantity | Amount      | Percent of Total |
|------|------------------|----------|-------------|------------------|
| 1    | Nancy Davolio    | 467      | \$24,827.45 | 23.68%           |
| 2    | Laura Callahan   | 912      | \$20,728.13 | 19.77%           |
| 3    | Janet Leverling  | 578      | \$16,360.12 | 15.60%           |
| 4    | Andrew Fuller    | 558      | \$13,937.64 | 13.29%           |
| 5    | Margaret Peacock | 481      | \$8,298.45  | 7.91%            |

## Remarks

1. The SQL statement will get the information of top 5 employees for sales, including employee name, quantity of products, and sales amount.
2. type="fix". It is a fixed table report.
3. table = 6. WDRReportGen will put data into the sixth table in the report file.
4. cell=B2. The cells corresponding to the first record are "B2,C2,D2,E2".
5. The default range is "B2:E2".
6. WDRReportGen executes the SQL statement, and gets data from data source. First, it fetches the first record, puts the value of the first field into cell B2, the value of the second field into cell C2, the value of the third field into D2, and the value of the fourth field into E2. And then it fetches the next record, skips one row, and puts data into cells B3, C3, D3 and E3.....

### 6.1.3 Non-group Variable Table Report Function

In a variable table report, the number of rows or columns in the table is unfixed, and it is variable as the number of the result records. WDRReportGen executes a SQL statement to get data from data source, inserts some blank rows/columns or copy a range for each record, then fills data values into the cells of a table in the report file.

#### Syntax

```
Report (...)  
sqlstatement
```

#### Arguments

```
TYPE = "var"  
TABLE = table  
FILLORDER = fillorder  
CELL= celllist  
RANGE = range  
IMAGE = fieldlist  
RESERVE = reserverecords  
PAGEBREAK = pagelength  
NODATA = nodataoption  
CONNECT = datasource
```

The **TYPE** argument specifies the report type. "var" means a variable table report. Default is var.

The **TABLE** argument identifies a table in the report template. The *table* is the index number of the table or the bookmark name in the table. The index number starts at 1. For examples, table 2 is the second table in the document. The index number of a nested table likes 2-1-2. For examples, table 2-1 is the first table inside table 2, and table 2-1-2 is the second table inside table 2-1.

The max nested level WDRReportGen supports is 3.

The **FILLORDER** argument specifies the order in which WDRReportGen fill data. Possible values are row or col. "row" means to fill data by rows, and "col" means to fill data by columns. Default is row.

The **CELL** argument specifies the positions where data values will be inserted.

The *celllist* is the list of cells separated by the “,” character. For example, “A2,B2,B3,D2,D3”. The cells in the *celllist* should correspond to the data source fields in the SQL statement. The value of the first field is put into the first cell, and the value of the second field is put into the second cell .....

WDRReportGen will use the next cell if you omit a cell except the first cell. If FILLORDER=“row”, the next cell is the right cell. If FILLORDER=“col”, the next cell is the below cell.

The **RANGE** or **COPYRANGE** argument specifies the range in the table to be used for the records. WDRReportGen will skip or repeat the range for each record. You can reference a range of cells like “2:4” or “B:D”. The default range is the area that includes all cells for the records. For Range argument, WDRReportGen will insert the blank rows/columns of the range for each record. For COPYRANGE argument, it will copy the original range and insert the copied range for each record.

The **IMAGE** argument specifies the fields are picture files. The *fieldlist* is the list of data source fields separated by the “,” character. You can identify a field using the name of field or the index number of field, but not simultaneously. In data source, you stored the path and file name of the picture, not the picture. The file path can be a relative path, an absolute path or a URL. If it is a relative path, the base path is the path of the report template file.

The **RESERVE** argument specifies the number of the records for which you reserved some rows/columns in the report template for the report. The *reserverecords* represents the number of the records you reserved in the

report template. Possible values are 1 or 2. One means you reserved some rows/columns for one record, and two means some rows/columns for two records. Default is 1.

The **PAGEBREAK** argument specifies the page breaks. The unit of page length is r that means record. For example, "6r" or "6" means that WDRReportGen will insert a page break per 6 records. Default is no page break.

The **NODATA** argument specifies an option when no data are returned from data source. If the value is "delrange", WDRReportGen will delete the range when no data are returned. "deltable" means to delete the table. Default is to do nothing.

The **CONNECT** argument specifies the connection to a data source. The CONNECT can takes a string that expresses a data source name or a number that expresses a data source index. The index number of data source is the sequential number defined in the WRF file, and starts at 1. The default implies the first data source.

The **sqlstatement** is a SQL statement such as a SELECT statement.

### Example

The following function will makes the report: Mail Label.

```
@F1=Report(type=var table=1 cell=B7,B8,B9,B10 copyrange=1:11 pagebreak  
= 4r)
```

```
SELECT CompanyName
```

```
,Address
```

```
,CityName & ', ' & CountryName
```

```
,PostalCode
```

```
FROM Customers, Cities, Countries
```

```
WHERE Customers.CityCode = Cities.CityCode
```

```
AND Customers.CountryCode = Cities.CountryCode
```


AND Customers.CountryCode = Countries.CountryCode

ORDER BY CompanyName

;

### Result

The non-group variable table report defined in the report template:

|            |  |  |
|------------|--|--|
|            |  |  |
|            | XYZ Limited Co.                            |  |
|            | XYZ Building No.88 AAA Street BBB District |  |
|            | Beijing China, 100123                      |  |
|            |  |  |
| <b>To:</b> |  |  |
|            |  |  |
|            |  |  |
|            |  |  |
|            |  |  |
|            |  |  |

The non-group variable table report generated in the report:

|            |  |  |
|------------|--|--|
|            | XYZ Limited Co.                            |  |
|            | XYZ Building No.88 AAA Street BBB District |  |
|            | Beijing China, 100123                      |  |
|            |  |  |
| <b>To:</b> | <b>Alfreds Futterkiste</b>                 |  |
|            | Obere Str. 57                              |  |
|            | Berlin, Germany                            |  |
|            | 12209                                      |  |
|            |  |  |
|            | XYZ Limited Co.                            |  |
|            | XYZ Building No.88 AAA Street BBB District |  |
|            | Beijing China, 100123                      |  |
|            |  |  |
| <b>To:</b> | <b>Ana Trujillo Emparedados y helados</b>  |  |
|            | Avda. de la Constitución 2222              |  |
|            | México D.F., Mexico                        |  |
|            | 05021                                      |  |
|            |  |  |

### Remarks

1. The SQL statement will get the information of customers including company name, address, city name, country name, and postal code.

2. type="var". It is a variable table report. And there is no GROUP argument, so it is a non-group variable table report.
3. table=1. WDRReportGen will put data into the first table in the report file.
4. cell=B7,B8,B9,B10. These cells correspond to the first record.
5. copyrange=1:11. Because the default range is "B7:B9", you must specify a range explicitly. WDRReportGen will copy the range for each record.
6. pagebreak = 4r. WDRReportGen will add a page break per 4 records.
7. WDRReportGen executes the SQL statement, and gets data from data source. First, it fetches the first record, copy the range, and fill data. And then it fetches the next record..... One page contains 4 mail labels.

## 6.1.4 Group Variable Table Report Function

The Group Variable Table Report function generates a variable table report and group data. In a variable table report, the number of rows or columns in the table is unfixed, and it is variable as the number of the result records. WDRReportGen executes a SQL statement to get data from data source, copy the group range for each group, copy the detail range for each record, then fills data into the table.

### Syntax

```
Report (...)  
sqlstatement
```

### Arguments

```
TYPE = "var"  
TABLE = table  
FILLORDER = fillorder  
CELL= celllist  
RANGE = range  
GROUP= grouplist
```

GROUPRANGE = *grouprange*

IMAGE = *fieldlist*

PAGEBREAK = *pagelength*

NODATA = *nodataoption*

CONNECT = *datasource*

The **TYPE** argument specifies the report type. "var" means a variable table report. Default is var.

The **TABLE** argument identifies a table in the report template. The *table* is the index number of the table or the bookmark name in the table. The index number starts at 1. For examples, table 2 is the second table in the document. The index number of a nested table likes 2-1-2. For examples, table 2-1 is the first table inside table 2, and table 2-1-2 is the second table inside table 2-1. The max nested level WDRReportGen supports is 3.

The **FILLORDER** argument specifies the order in which WDRReportGen fill data. Possible values are row or col. "row" means to fill data by rows, and "col" means to fill data by columns. Default is row.

The **CELL** argument specifies the positions where data values will be inserted. The *celllist* is the list of cells separated by the “,” character. For example, “A2,B2,B3,D2,D3”. The cells in the *celllist* should correspond to the data source fields in the SQL statement. The value of the first field is put into the first cell, and the value of the second field is put into the second cell ..... WDRReportGen will use the next cell if you omit a cell except the first cell. If FILLORDER=“row”, the next cell is the right cell. If FILLORDER=“col”, the next cell is the below cell.

The **RANGE** or **COPYRANGE** argument specifies the range in the table to be used for the details. WDRReportGen will skip or repeat the range for each record. You can reference a range of cells like “2:4” or “B:D”. The default range is the area that includes all cells for the details. For Range argument,

WDRReportGen will insert the blank rows/columns of the range for each record. For COPYRANGE argument, it will copy the original range and insert the copied range for each record. But if the range of any group is not same as the range of the details, RANGE is same as COPYRANGE.

The **GROUP** argument specifies the group of the report. The *grouplist* is the list of data source fields separated by the “,” character. You can identify a field using the name or index number of the field, but not simultaneously. In one report, there may be up to 10 groups. Notes: the order of the groups should be in accordance with the order of the ORDER BY clause in the SQL statement.

The **GROUPRANGE** argument follows the GROUP argument, and specifies the range of the group in the table. For example, the grouprange of level 1 must follow the group of level 1, and the grouprange of level 2 must follow the group of level 2. WDRReportGen will repeat the group range for each group.

The range of the group should contain the range of the details and the area that includes all cells for this group. You reference a group range like “2:4” or “B:D”. For example, there are two groups, the range of the group one contains all cells for the group one and the range of the group two, and the range of the group two contains all cells for the group two and the range of the details. The default range is the area that includes all cells for this group and the range or group range for the lower level group.

The **IMAGE** argument specifies the fields are picture files. The *fieldlist* is the list of data source fields separated by the “,” character. You can identify a field using the name of field or the index number of field, but not simultaneously. In data source, you stored the path and file name of the picture, not the picture. The file path can be a relative path, an absolute path or a URL. If it is a relative path, the base path is the path of the report template file.

The **PAGEBREAK** argument specifies the page breaks. The unit of page length is r or g. "r" means record, "g1" means group one, "g2" means group

two..... For example, "6r" or "6" means that WDRReportGen will insert a page break per 6 records, "1g1" or "1g" means a page break per group one, and "1g1,6r" means a page break per group one or 6 records. Default is "" that means no page break.

The **NODATA** argument specifies an option when no data are returned from data source. If the value is "delrange", WDRReportGen will delete the range when no data are returned. "deltable" means to delete the table. Default is to do nothing.

The **CONNECT** argument specifies the connection to a data source. The CONNECT can takes a string that expresses a data source name or a number that expresses a data source index. The index number of data source is the sequential number defined in the WRF file, and starts at 1. The default implies the first data source.

The **sqlstatement** is a SQL statement such as a SELECT statement.

### Example

The following function will makes the report: Customer Profile.

```
@F1= Report(table=1 cell=A2,B3,C3,D3,D4,E3,E4,E5
```

```
copyrange=2:5 group=1 pagebreak = 5r)
```

```
SELECT LEFT(CompanyName,1)
```

```
,CompanyName
```

```
,ContactName
```

```
, 'Phone: ' & Phone
```

```
, 'Fax: ' & Fax
```

```
,Address
```

```
,CityName & ', ' & CountryName
```

```
,PostalCode
```

```
FROM Customers, Cities, Countries
```

```

WHERE Customers.CityCode = Cities.CityCode
AND Customers.CountryCode = Cities.CountryCode
AND Customers.CountryCode = Countries.CountryCode
ORDER BY CompanyName
;

```

## Result

The group variable table report defined in the report template:

| Customer Name | Contact Name | Phone/Fax | Address |
|---------------|--------------|-----------|---------|
|               |              |           |         |
|               |              |           |         |
|               |              |           |         |
|               |              |           |         |

The group variable table report generated in the report:

| Customer Name                      | Contact Name       | Phone/Fax                                  | Address   |
|------------------------------------|--------------------|--|---|
| <b>A</b>                           |                    |  |   |
| Alfreds Futterkiste                | Maria Anders       | Phone: 030-0074321<br>Fax: 030-0076545     | Obere Str. 57<br>Berlin, Germany<br>12209                     |
| Ana Trujillo Emparedados y helados | Ana Trujillo       | Phone: (5) 555-4729<br>Fax: (5) 555-3745   | Avda. de la Constitución 2222<br>México D.F., Mexico<br>05021 |
| <b>B</b>                           |                    |  |   |
| Berglunds snabbköp                 | Christina Berglund | Phone: 0921-12 34 56<br>Fax: 0921-12 34 57 | Berguvsvägen 8<br>Luleå, Sweden<br>S-951 22                   |
| Blauer See Delikatessen            | Hanna Moos         | Phone: 0621-08460<br>Fax: 0621-08924       | Forsterstr. 57<br>Mannheim, Germany<br>68306                  |

## Remarks

1. The SQL statement will get the information of customers including company name, contact name, phone, fax, address, city name, country name, and postal code.
2. There are TABLE argument and GROUP argument, so it is a group variable table report.
3. table=1. WDRReportGen will put data into the first table in the report file.
4. group=1. WDRReportGen will group data by the first letter of the company name.

5. cell= A2,B3,C3,D3,D4,E3,E4,E5. These cells correspond to the first record.
6. copyrange=2:5. Because the default range is “B3:E5”, you must specify a range explicitly. WDRReportGen will copy the range for each record.
7. There is no grouprange. WDRReportGen will give a default. The default grouprange is “2:5”.
8. pagebreak = 5r. WDRReportGen will add a page break per 5 records.
9. WDRReportGen executes the SQL statement, gets data from data source, and puts data into the table in the report file. Because the range is same as the group range, WDRReportGen will copy the range for each record, fill the value of the first field per group, and fill the values of other fields per record. One page contains the information of 5 records.

### **6.1.5 Form Report Function**

For a form report, you can put data from data source as text, list, title and table in the report file. WDRReportGen executes a SQL statement to get data from data source, copy the group range for each group, copy the detail range for each record, then fills data into the report.

#### **Syntax**

*Report(...)*  
*sqlstatement*

#### **Arguments**

TYPE = “form”  
CELL= *celllist*  
RANGE = *range*  
GROUP= *grouplist*  
GROUPRANGE = *grouprange*  
IMAGE = *fieldlist*  
PAGEBREAK = *pagelength*

NODATA = *nodataoption*

CONNECT = *datasource*

The **TYPE** argument specifies the report type. "form" means a form report.

The **CELL** argument specifies the positions where data values will be inserted.

The *celllist* is the list of merge fields or quote fields separated by the “,” character. For example, “ProductName,ProductID,QuantityPerUnit,UnitPrice”.

The merge fields or quote fields in the *celllist* should correspond to the data source fields in the SQL statement. The value of the first data source field is put into the first merge field or quote field, and the value of the second data source field is put into the second merge field or quote field .....

The **RANGE** argument specifies the range to be used for the records.

WDRReportGen will repeat the range for each record. A range is defined by a bookmark. You reference a range using a bookmark name. The default range is the group range or the entire document.

The **GROUP** argument specifies the group of the report. The *grouplist* is the list of data source fields separated by the “,” character. You can identify a field using the name of field or the index number of field, but not simultaneously. In one report, there may be up to 10 groups. The first GROUP is group one, the second is group two..... Notes: the order of groups should be in accordance with the order of ORDER BY clause in the SQL statement.

The **GROUPRANGE** argument follows the GROUP argument, and specifies the range of the group in the table. For example, the grouprange of level 1 must follow the group of level 1, and the grouprange of level 2 must follow the group of level 2. WDRReportGen will repeat the range for each group. A range is defined by a bookmark. You reference a range using a bookmark name. The range of the group should contain the range of the details and the area that includes all merge fields or quote fields for this group. For example, there are two groups, the range of the group one contains all merge fields or quote fields

for the group one and the range of the group two, and the range of the group two contains all merge fields or quote fields for the group two and the range of the details. The default range is the range of the upper level group or the entire document.

The **IMAGE** argument specifies the fields are picture files. The *fieldlist* is the list of data source fields separated by the “,” character. You can identify a field using the name of field or the index number of field, but not simultaneously. In data source, you stored the path and file name of the picture, not the picture. The file path can be a relative path, an absolute path or a URL. If it is a relative path, the base path is the path of the report template file.

The **PAGEBREAK** argument specifies the page breaks. The unit of page length is r or g. "r" means record, "g1" means group one, "g2" means group two..... For example, “6r” or “6” means that WDRReportGen will insert a page break per 6 records, “1g1” or “1g” means a page break per group one, and “1g1,6r” means a page break per group one or 6 records. Default is “” that means no page break.

The **NODATA** argument specifies an option when no data are returned from data source. If the value is "delrange", WDRReportGen will delete the range when no data are returned. Default is to do nothing.

The **CONNECT** argument specifies the connection to a data source. The CONNECT can takes a string that expresses a data source name or a number that expresses a data source index. The index number of data source is the sequential number defined in the WRF file, and starts at 1. The default implies the first data source.

The **sqlstatement** is a SQL statement such as a SELECT statement.

### **Example**

The following function will makes the report: Product Catalog.

```

@F1=Report(type=form cell=CategoryName,Description
,ProductName,ProductID,QuantityPerUnit,UnitPrice
range=Product group=1,2 grouprange=Category)
SELECT CategoryName
,Description
,ProductName
,ProductID
,QuantityPerUnit
,UnitPrice
FROM Products, Categories
WHERE Products.CategoryID = Categories.CategoryID
ORDER BY 1,3
;

```

### Result

The form report defined in the report template:

| « <b>CategoryName</b> » |             |                   |             |
|-------------------------|-------------|-------------------|-------------|
| «Description»           |             |                   |             |
| Product Name            | Product ID  | Quantity Per Unit | Unit Price  |
| «ProductName»           | «ProductID» | «QuantityPerUnit» | «UnitPrice» |

The form report generated in the report:

## Beverages

Soft drinks, coffees, teas, beers, and ales

| Product Name       | Product ID | Quantity Per Unit  | Unit Price |
|--------------------|------------|--------------------|------------|
| Chai               | 1          | 10 boxes x 20 bags | \$18.00    |
| Chang              | 2          | 24 - 12 oz bottles | \$19.00    |
| Chartreuse verte   | 39         | 750 cc per bottle  | \$18.00    |
| Côte de Blaye      | 38         | 12 - 75 cl bottles | \$263.50   |
| Guaraná Fantástica | 24         | 12 - 355 ml cans   | \$4.50     |
| Ippoh Coffee       | 43         | 16 - 500 g tins    | \$46.00    |

### Remarks

1. The SQL statement will get the information of products including product category, category description, product name, product ID, quantity per unit, unit price.
2. type="form". It is a form report.
3. cell=CategoryName, Description, ProductName, ProductID, QuantityPerUnit, UnitPrice. These merge fields or quote fields correspond to data source fields in the SQL statement.
4. range= Product. The bookmark "Product" defines the range for detail record. WDRReportGen will copy the range for each record.
5. group=1,2. WDRReportGen will group data by CategoryName and Description.
6. grouprange= Category. The bookmark "Category" defines the group range. WDRReportGen will copy the range for each group.
7. WDRReportGen executes the SQL statement, and gets data from data source. It fetches a record, copy the group range for each group, copy the range for each record, and inserts data into the merge fields or quote fields.

## 6.2 Chart Function

The CHART function executes a SELECT statement to get data from data source, and put data into the datasheet of a chart in the report file. The CHART function supports two types of charts:

- MSGraph chart
- Excel chart

### 6.2.1 MSGraph Chart Function

If you use Microsoft Graph to create a chart, you can use MSGraph Chart function.

#### Syntax

```
Chart(...)  
sqlstatement
```

#### Arguments

```
CHART = chart  
FILLORDER = fillorder  
CELL= celllist  
RANGE = range  
CONNECT = datasource
```

The **CHART** argument identifies a chart in the report template. The *chart* is the index number or the bookmark name of the chart. The index number starts at 1. For examples, chart 2 is the second chart in the document. You can reference a chart by a bookmark. For examples, chart="Chart1". "Chart1" is the bookmark of a chart.

The **FILLORDER** argument specifies the order in which WDRReportGen fill data. Possible values are row or col. "row" means to fill data by rows, and "col" means to fill data by columns. Default is col.

The **CELL** argument specifies the positions where data values will be inserted. The *celllist* is the list of cells or fields separated by the “,” character. The *celllist* identifies the cells in a data sheet. For example, “A2,B2,B3,D2,D3”. The cells in the *celllist* should correspond to the data source fields in the SQL statement. The value of the first field is put into the first cell, and the value of the second field is put into the second cell ..... WDRReportGen will use the next cell if you omit a cell except the first cell. If FILLORDER=”row”, the next cell is the right cell. If FILLORDER=”col”, the next cell is the below cell. Note: On the datasheet, the leftmost column and the top row, which are commonly used for legend text or axis labels, are referred to as column 0 (zero) and row 0 (zero). The **RANGE** argument specifies the range in the datasheet of the chart to be used for the records. WDRReportGen will skip the rows/columns of the range for each record. A range is composed of some rows or columns. You can reference a range of cells like “2:4” or “B:D”. The default range is the area that includes all cells for details.

The **CONNECT** argument specifies the connection to a data source. The **CONNECT** can takes a string that expresses a data source name or a number that expresses a data source index. The index number of data source is the sequential number defined in the WRF file, and starts at 1. The default implies the first data source.

The *sqlstatement* is a SQL statement such as a SELECT statement.

### **Example**

The following function makes the chart: Sales by Categories.

```
@F3_2=CHART(chart=Chart3 cell=A0)
SELECT c.CategoryName
      , Sum(d.UnitPrice * d.Quantity * (1-d.Discount))
FROM Orders o
      ,OrderDetails d
```

```

,Products p
,Categories c
WHERE o.OrderID = d.OrderID
AND d.ProductID = p.ProductID
AND p.CategoryID = c.CategoryID
AND YEAR(o.OrderDate) = 1996
AND MONTH(o.OrderDate) = 04
GROUP BY c.CategoryName
ORDER BY c.CategoryName

```

**Result**

The datasheet of the chart defined in the report template:

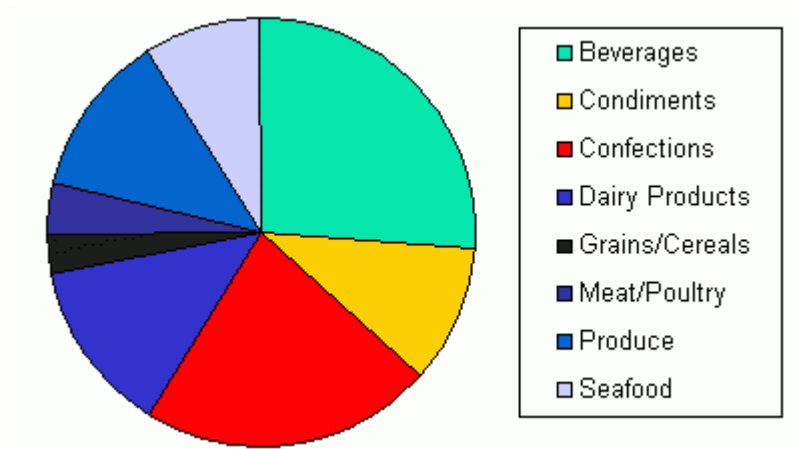
|   |               | A | B | C | D |
|---|---------------|---|---|---|---|
|   | Category Name |   |   |   |   |
| 1 | Amount        |   |   |   |   |
| 2 |               |   |   |   |   |

The chart defined in the report template is a blank chart.

The datasheet of the chart generated in the report:

|   |               | A         | B          | C           | D              |
|---|---------------|-----------|------------|-------------|----------------|
|   | Category Name | Beverages | Condiments | Confections | Dairy Products |
| 1 | Amount        | 27761.57  | 10773.27   | 22877.18    | 13685.32       |
| 2 |               |           |            |             |                |

The chart generated in the report:



**Remarks**

1. The SQL statement will get the information of sales by categories, including category name, and sales amount.
2. chart = Chart3. "Chart3" is the bookmark name of a chart.
3. The default fillorder is col. WDRReportGen will fill data by columns.
4. cell=A0. The cells corresponding to the first record are "A0,A1".
5. The default range is "A:A".
6. WDRReportGen executes the SQL statement, and gets data from data source. First, it fetches the first record, puts the value of the first field into cell A0, the value of the second field into cell A1. And then it fetches the next record, skips one column, and puts data into cells B0, B1.....

## 6.2.2 Excel Chart Function

If you use Microsoft Excel to create a chart, you can use Excel Chart function.

### Syntax

```
Chart(...)  
sqlstatement
```

### Arguments

```
CHART = chart  
TYPE = type  
FILLORDER = fillorder  
CELL= celllist  
RANGE = range  
CONNECT = datasource
```

The **CHART** argument identifies a chart in the report template. The *chart* is the index number or the bookmark name of the chart. The index number starts at 1. For examples, chart 2 is the second chart in the document. You can reference a chart by a bookmark. For examples, chart="Chart1". "Chart1" is the bookmark of a chart.

The **TYPE** argument specifies the report type. Possible values are fix or var. "fix" means that WDRReportGen will directly fill data values into the worksheet of the chart. "var" means that WDRReportGen will add some blank rows/columns before filling data values into the worksheet of the chart. Default is var. When the report type is "var", you should reserve two rows/columns in the worksheet in the report template, and set the data range of the chart to 2 rows/columns. The RESERVE must be 2.

The **FILLORDER** argument specifies the order in which WDRReportGen fill data. Possible values are row or col. "row" means to fill data by rows, and "col" means to fill data by columns. Default is row.

The **CELL** argument specifies the positions where data values will be inserted. The *celllist* is the list of cells or fields separated by the "," character. The *celllist* identifies the cells in a worksheet. For example, "A2,B2,B3,D2,D3". The cells in the *celllist* should correspond to the data source fields in the SQL statement. The value of the first field is put into the first cell, and the value of the second field is put into the second cell ..... WDRReportGen will use the next cell if you omit a cell except the first cell. If FILLORDER="row", the next cell is the right cell. If FILLORDER="col", the next cell is the below cell.

The **RANGE** argument specifies the range in the worksheet of the chart to be used for the records. WDRReportGen will skip the rows/columns of the range for each record. A range is composed of some rows or columns. You can reference a range of cells like "2:4" or "B:D". The default range is the area that includes all cells for details. For a variable table report, WDRReportGen will insert the blank rows/columns of the range for each record.

The **CONNECT** argument specifies the connection to a data source. The CONNECT can takes a string that expresses a data source name or a number that expresses a data source index. The index number of data source is the sequential number defined in the WRF file, and starts at 1. The default implies

the first data source.

The **sqlstatement** is a SQL statement such as a SELECT statement.

### Example

The following function makes the chart: Sales by Categories.

```
@F3_2=CHART(chart=Chart3 cell=A2)
SELECT c.CategoryName
      , Sum(d.UnitPrice * d.Quantity * (1-d.Discount))
FROM Orders o
      ,OrderDetails d
      ,Products p
      ,Categories c
WHERE o.OrderID = d.OrderID
AND d.ProductID = p.ProductID
AND p.CategoryID = c.CategoryID
AND YEAR(o.OrderDate) = 1996
AND MONTH(o.OrderDate) = 04
GROUP BY c.CategoryName
ORDER BY c.CategoryName
```

### Result

The worksheet of the chart defined in the report template:

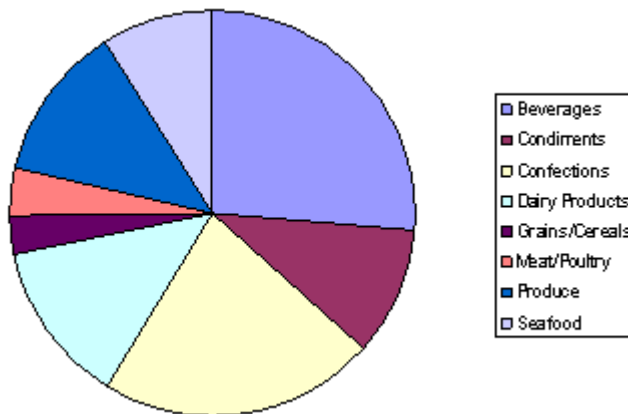
|   | A             | B      | C | D | E | F |
|---|---------------|--------|---|---|---|---|
| 1 | Category Name | Amount |   |   |   |   |
| 2 |               |        |   |   |   |   |
| 3 |               |        |   |   |   |   |
| 4 |               |        |   |   |   |   |

The chart defined in the report template is a blank chart.

The worksheet of the chart generated in the report:

|   | A                    | B             | C | D | E | F |
|---|----------------------|---------------|---|---|---|---|
| 1 | <b>Category Name</b> | <b>Amount</b> |   |   |   |   |
| 2 | Beverages            | 27761.57499   |   |   |   |   |
| 3 | Condiments           | 10773.26999   |   |   |   |   |
| 4 | Confections          | 22877.17998   |   |   |   |   |
| 5 | Dairy Products       | 13685.32499   |   |   |   |   |
| 6 | Grains/Cereals       | 3325.399995   |   |   |   |   |
| 7 | Meat/Poultry         | 4083.659998   |   |   |   |   |
| 8 | Produce              | 13031.2       |   |   |   |   |
| 9 | Seafood              | 9316.544988   |   |   |   |   |

The chart generated in the report:



### Remarks

1. The SQL statement will get the information of sales by categories, including category name, and sales amount.
2. chart = Chart3. "Chart3" is the bookmark name of a chart.
3. The default type is var. You should reserve 2 blank rows in the worksheet, and select the 2 rows as the data range of the chart.
4. The default fillorder is row. WDRReportGen will fill data by rows.
5. cell=A2. The cells corresponding to the first record are "A2,B2".
6. The default range is "2:2".
7. WDRReportGen executes the SQL statement, and gets data from data source. First, it will add some blank rows in the worksheet according to the number of the records. And then it will put data into the worksheet.

## 6.3 DocVariable Function

The DOCVARIABLE function executes a SQL statement, and assigns the results to the document variables defined in the Microsoft Word document. The document variable is defined using DocVariable field.

### Syntax

DocVariable(...)

*sqlstatement*

### Arguments

NAME= *namelist*

CONNECT= *datasource*

The **NAME** argument specifies the name of the document variables you want assign values to. The *namelist* is the list of variable names separated by the “,” character. For example, “BeginDate, EndDate” means two document variables: BeginDate and EndDate that should be defined in the report template. The variables in the *namelist* should correspond to the fields in the SQL statement. The value of the first field is put into the first variable, and the value of the second field is put into the second variable ...

The **CONNECT** argument specifies the connection to a data source. The CONNECT can takes a string that expresses a data source name or a number that expresses a data source index. The index number of data source is the sequential number defined in the WRF file, and starts at 1. The default implies the first data source.

The *sqlstatement* is a SQL statement such as a SELECT statement.

WDRReportGen will just fetch the first record, no matter how many records are returned from data source. DocVariable function supports headers and footers. You can use it to put data into headers or footers.

### Example

The following function executes a SQL statement, assigns the values of fields to document variables.

```
@F1=DOCVARIABLE(NAME=BeginDate,EndDate)
SELECT min_date, max_date
FROM tmp0
;
```

### Remarks

WDRReportGen executes the SQL statement, get data from data source. It assigns the value of field “min\_date” to the document variable “BeginDate” and the value of field “max\_date” to the document variable “EndDate”.

You should define the document variables in the report template first. The document variables are defined as follows:

```
{ DOCVARIABLE BeginDate \@ "YYYY-MM-DD" \* MERGEFORMAT}
{ DOCVARIABLE EndDate \@ "YYYY-MM-DD" \* MERGEFORMAT}
```

To define a document variable, do as follows:

1. Run Microsoft Word program, and open the report template file.
2. Click where you want to insert the information.
3. On the **Insert** menu, click **Field**.
4. Click **DocVariable**, and then input a field name.
5. Press **OK** button.

## 6.4 ExecSQL Function

The EXECSQL function executes a SQL statement, but does not return result to report.

### Syntax

```
ExecSQL(...)
    sqlstatement
```

### Arguments

CONNECT= *datasource*

The **CONNECT** argument specifies the connection to a data source. The CONNECT can takes a string that expresses a data source name or a number that expresses a data source index. The index number of data source is the sequential number defined in the WRF file, and starts at 1. The default implies the first data source.

The **sqlstatement** is a SQL statement that can be DDL (Data Definition Language), DML (Data Manipulation Language) and even DCL (Data Control Language).

Using EXECSQL function, you can open a database, create a temporary table, insert data into a temporary table, update data, execute a stored procedure, and drop a table. It is very useful to create a temporary table, and prepare data for REPORT function.

### **Example**

The following functions will create a table tmp0, and add some records into table. No result is returned to the report file.

```
@F1=EXECSQL()  
CREATE TABLE tmp0 (  
min_date DATE,  
max_date DATE)  
;  
@F2=EXECSQL()  
INSERT INTO tmp0  
SELECT ...  
;
```

# Chapter 7 Menus, Toolbar and Shortcut Keys

## 7.1 File Menu

The File menu offers the following commands:

|                      |  |
|----------------------|--|
| New                  | Creates a new WRF file.                            |
| Open                 | Opens an existing WRF file.                        |
| Close                | Closes an opened WRF file.                         |
| Save                 | Saves an opened WRF file using the same filename.  |
| Save As              | Saves an opened WRF file to a specified file name. |
| Open Report Template | Opens an existing report template file.            |
| Open Report File     | Opens an existing report file.                     |
| Open Log File        | Opens an existing log file.                        |
| Recent Files         | Opens last WRF files you closed.                   |
| Exit                 | Exits WDRReportGen.                                |

## 7.2 Edit Menu

The Edit menu offers the following commands:

|            |   |
|------------|---|
| Undo       | Reverse previous editing operation.                           |
| Cut        | Deletes text from the document and moves it to the clipboard. |
| Copy       | Copies text from the document to the clipboard.               |
| Paste      | Pastes text from the clipboard into the document.             |
| Delete     | Deletes the selection.  |
| Select All | Selects the entire text.                                      |
| Find       | Finds the specified text.                                     |
| Find Next  | Finds the next matching text.                                 |
| Replace    | Replaces specific text with different text.                   |
| Go to      | Goes to specified line or function in the document.           |

## 7.3 Report Menu

The Report menu offers the following commands:

|               |   |
|---------------|---|
| Configuration | Configures the file names, data sources and parameters. |
|---------------|---|

|     |   |
|-----|---|
| Run | Runs the WRF file to generate a report. |
|-----|---|

## 7.4 Tools Menu

The Tools menu offers the following commands:

|        |               |
|--------|---------------|
| Option | Sets options. |
|--------|---------------|





## 7.5 Help Menu











The Help menu offers the following commands:

|                |  |
|----------------|--|
| Help Context   | Starts the online help system.   |
| Tutorial       | Starts a brief step-by-step tutorial.  |
| Tip of the Day | Displays a dialog containing a useful tip about WDRReportGen.                  |
| Hints and Tips | Displays miscellaneous hints and tips on how to use WDRReportGen productively. |
| Shortcut Keys  | Shows the keyboard map.  |
| Home Page      | Takes you to the home page of WDRReportGen web site.                           |
| Support        | Takes you to the support page of WDRReportGen web site.                        |
| Buy Now        | Buy WDRReportGen immediately.  |
| About          | Displays the version number of WDRReportGen.                                   |

## 7.6 Toolbar

The toolbar provides quick access to many features. The buttons on the toolbar perform the following commands:

| Buttons   | Commands  |
|---|---|
|  | Creates a new WRF file.                           |
|  | Opens an existing WRF file.                       |
|  | Saves an opened WRF file using the same filename. |
|  | Open the report template file.                    |

|   |   |
|---|---|
|    | Open the report file.   |
|    | Deletes text from the document and moves it to the clipboard. |
|    | Copies text from the document to the clipboard.               |
|    | Pastes text from the clipboard into the document.             |
|    | Reverse previous editing operation.                           |
|    | Finds the specified text.                                     |
|    | Goes to specified line or function in the document.           |
|    | Runs the WRF file to generate a report.                       |
|   | Starts the online help system.                                |
|  | Buy WDRReportGen immediately.                                 |

## 7.7 Shortcut Keys

| Shortcut Keys | Commands  |
|---------------|---|
| Ctrl+N        | Creates a new WRF file.                                       |
| Ctrl+O        | Opens an existing WRF file.                                   |
| Ctrl+S        | Saves an opened WRF file using the same filename.             |
| Ctrl+U        | Reverse previous editing operation.                           |
| Ctrl+X        | Deletes text from the document and moves it to the clipboard. |
| Ctrl+C        | Copies text from the document to the clipboard.               |
| Ctrl+V        | Pastes text from the clipboard into the document.             |
| Delete        | Deletes the selection.  |
| Ctrl+A        | Selects the entire text.                                      |
| Ctrl+F        | Finds the specified text.                                     |

|        |   |
|--------|---|
| F3     | Finds the next matching text.                           |
| Ctrl+H | Replaces specific text with different text.             |
| Ctrl+G | Goes to specified line or function in the document.     |
| F2     | Configures the file names, data sources and parameters. |
| F5     | Runs the WRF file to generate a report.                 |
| F1     | Starts the online help system.                          |

## Chapter 8 Hints and Tips

You can run WDRReportGen from the command line. The format is:

```
wordreport <wrf file name> [-c] [-d] [-u1 user1] [-p1 pwd1] ... [pa1 pa2 ...]
```

For example:

```
wordreport c:\wordreport\monthlysales.wrf -c 199605
```

WDRReportGen can be scheduled with Windows Scheduled Tasks or other tools. The process of generating reports can be fully automated, periodically or on events.

WDRReportGen comes with a sample database Sample.mdb and some sample reports. You can use them when learning the program. To use the sample reports, you must add a data source named "Report Sample" to specify the sample database.

To make a report template, you can use some sample data. It is very useful especially for formatting. After you have made the report template, you delete the sample data.

For a table report, you can format the value from data sources with a format expression. You should write a format expression into a data cell in the report template file first. WDRReportGen will get the text of the cell as a format expression before it puts a value into a cell, and output the value using the format expression.

You can define the different formats and colors for positive values, negative values and zeros.

For a form report, you can format the value from data sources with the switch of a merge field or quote field like “\#” or “\@”.

In the sample file “monthly\_sales.doc”, there is a macro called “CellRef”. The macro can tell you the table number and the cell reference.

An irregular table does not have the same number of cells for each row or column. It does make it harder to process the document. In an irregular table, you have some difficulty to reference a cell, and an error may occur when you try to work with some rows or columns.

You can set the width of cells in the same column to be different, and keep the table have the same number of cells for each row. Select two or more cells in one row, merge them into one cell, and split the cell into two or more cells. You can drag the boundary and change the cell width.

To create a chart in the report template file, you can use some sample data. Using sample data, you can set the various chart options. After you have made the report template, you delete the sample data.

For MSGraph chart, on the datasheet, the leftmost column and the top row, which are commonly used for legend text or axis labels, are referred to as column 0 (zero) and row 0 (zero).

By default, Microsoft Word 2007 uses Microsoft Excel to create charts, but doesn't expose the chart as a normal Excel object. WDRReportGen can not access the charts. You must insert an Excel chart object that WDRReportGen

can access.

You can use (formula) fields to perform calculations in a report template file.

WDRReportGen is a converter too. Besides Microsoft Word document, you can generate a report in other file format such as HTML, XML, RTF, text, Works.

You also can convert data from database to other file format.

You can protect the generated report so that it can not be modified. To protect the report, select the **Protect Report** check box in the **Configuration** dialog box.

You can edit a WRF file (.wrf) with a text editor such as Notepad.

If you associate WDRReportGen with the file extension “.wrf”, a WRF file with the extension “.wrf” will open in WDRReportGen when you double-click the file.

The information:

File Extension: .wrf

Action: open

Application: "C:\Program Files\LJZsoft\wordreport.exe" "%1"

For the report template file, report file and log file, it is possible to give a relative path. If it is a relative path, the base path is the path of the WRF file.

In the SQL statements, you can use parameters. To use parameters, you must define them first.

In the paths and names of the report file, template file and log file, you can use

parameters. To use parameters, you must define them first.

The default log file is wordreport.log under the WDRReportGen program directory. If you do not define the log file name, or can not create the log file defined, you can find log information in the wordreport.log under the WDRReportGen program directory.

You should be careful to define a unique name for each parameter, because WDRReportGen will replace all strings that are the same as the names of the parameters. It is a good choice a name begins with the “\$” character such as “\$ReportDate”.

In the text editor window, you can use comments. A comment is the “/” characters, followed by any sequence of characters (including new lines), followed by the “\*/” characters. You cannot nest comments.

To add totals or subtotals, you can use the aggregate functions in SQL statement.

To group data in a report, you should use GROUP VARIABLE TABLE REPORT function.

In REPORT function, the order of groups should be in accordance with the order of ORDER BY clause in the SQL statement.

If you add a bookmark in a table, you can reference the table using the bookmark.

WDRReportGen supports nested tables. You can reference a nested table by a table index like 2-1-2 or a bookmark.

In a HTML file, you can add a bookmark using the <a> tag. For example, The following example defines a bookmark named "salesreport".

```
<a name=salesreport>Sales Report</a>
```

You can create reports with pictures using WDRReportGen. You should store the path and name of the graphics file in the database, identify the image fields in the report function, and specify the inserted way, text wrapping style and size in the report template file.

If you insert pictures into the cells using table report function, you can adjust the position of the pictures by changing the cell margins.

To convert from pixels to points, it is depend on the screen resolution (DPI). If you have a 96 dpi screen (Windows PC), 4 pixels are equal to 3 points.

DocVariable function supports headers and footers. You can use it to put data into headers or footers.

It is very useful to create a temporary table. You can prepare data using INSERT/UPDATE/DELETE/INSERT SELECT, and then make a report using REPORT function.

You can write a program to make a WRF file using C, perl or DOS shell, and then run WDRReportGen to generate report. The two steps can be written into a batch file.

If you do not save a password in the WRF file, a login dialog box will appear when you run the WRF file in WDRReportGen. You can input password interactively to log on to the data source.

It will take more time using COPYRANGE argument than using RANGE argument. Fixed table report is slower than non-group variable table report with RANGE argument, and faster than non-group variable table report with COPYRANGE argument. In general, group variable table report is slower than non-group variable table report. But if the ranges of all groups and details are same and you do not use COPYRANGE argument, it is faster.

WDRReportGen will use clipboard. You can not copy and paste during report generating.

WDRReportGen supports Microsoft Word 2007. You can use docx file as report file and template file. Please copy "wconv2007.cfg" to "wconv.cfg".

## Chapter 9 WRF File Reference

### 9.1 WRF File Format

The layout of a WRF file is as the following:

```
WordReport Version 2.0
```

```
[Data Source]
```

```
.....
```

```
[File]
```

```
.....
```

```
[Parameter]
```

```
.....
```

```
[SQL]
```

```
.....
```

“WordReport” is the flag of WRF file. “Version 2.0” is the version of the WRF file.

A WRF file contains several sections. The sections of [Data Source], [File], and [Parameter] consist of a group of related settings. The sections and settings are listed in the WRF file in the following format:

```
[section name]
```

```
keyname=value
```

In this example, [section name] is the name of a section. The enclosing brackets ([]) are required, and the left bracket must be in the leftmost column on the screen.

The keyname=value statement defines the value of each setting. A keyname is the name of a setting. It can consist of any combination of letters and digits, and must be followed immediately by an equal sign (=). The value can be an integer, a string, or a quoted string, depending on the setting.

You can include comments in these sections. You must begin each line of a comment with a semicolon (;).

The [SQL] section consists of functions. Each function is begin with the "@" character. Syntax:

```
@functionno=functionname(arguments)
sqlstatement
```

The *functionno* is a label of the report function.

The *functionname* represents a report function.

The *arguments* for a function define various properties for the function. An argument takes the form *Name="Value"*. The argument value can be delimited by single or double quotes.

The *sqlstatement* is a SQL statement.

You can use comments in [SQL] section. A comment is the "/\*" characters, followed by any sequence of characters (including new lines), followed by the "\*" characters. You cannot nest comments.

## 9.2 [Data Source] Section

The [Data Source] section contains information how to connect to data sources.

```
Name1=<name1>
Name2=<name2>
.....
Name10=<name10>
```

These settings specify the names of data sources you want to connect to.

Name1 specifies the name of the first data source. Name2 specifies the name of the second data source..... You can define up to 10 data sources in one WRF file. You can make a connection to a data source using an ODBC data

source name or a connection string. Even if you use a connection string to make a connection, you should define a name that you can reference in functions.

*User1=<username1>*

*User2=<username2>*

.....

*User10=<username10>*

These settings specify the user names. If you use an ODBC data source name to make a connection, you should define user name and password. If you use a connection string to make a connection, WDRReportGen will ignore the setting. User1 specifies the user name of the first data source. User2 specifies the user name of the second data source..... They are optional settings. If defined default user and password in ODBC data source, you may not define them.

*Password1=<password1>*

*Password2=<password2>*

.....

*Password10=<password10>*

These settings specify the user passwords. If you use an ODBC data source name to make a connection, you should define user name and password. If you use a connection string to make a connection, WDRReportGen will ignore the setting. Password1 specifies the password of the first data source. Password2 specifies the password of the second data source..... They are optional settings. If defined default user and password in ODBC data source, you may not define them.

*ConnectionString1=<connectionstring1>*

*ConnectionString2=<connectionstring2>*

.....

*ConnectionString10=<connectionstring10>*

These settings specify the connection strings. If you defined a connection string, WDRReportGen will make a connection to the data source using the connection string, and ignore the settings of the name, user and password. But you must define a name that you can reference in functions.

ConnectionString1 specifies the connection string of the first data source.

ConnectionString2 specifies the connection string of the second data source..... They are optional settings. If no connection string, WDRReportGen will make a connection to data source using the ODBC data source name.

*EncryptPassword =Y/N*

This setting specifies how to save the passwords of data sources. If the value is Y, the passwords will be saved in an encrypted format. If the value is N, the passwords will be saved in plain text.

### **9.3 [FILE] Section**

[FILE] section contains information about files.

*ReportTemplateFileName=<templatefilename>*

This setting specifies the name of the report template file. <templatefilename> value is the name and path of the report template file. The file path can be a relative path or an absolute path. If it is a relative path, the base path is the path of the WRF file.

*ReportFileName=<reportfilename>*

This setting specifies the name of the report file. <reportfilename> value is the

name and path of the report file. The file path can be a relative path or an absolute path. If it is a relative path, the base path is the path of the WRF file. In <reportfilename>, you can use parameters.

*ReportFileType=<reportfiletype>*

This setting specifies the type of the report file. <reportfiletype> value is the name or value of the file format. For example, wdFormatRTF or 6. What file format WDRReportGen supports is dependent on your Microsoft Word and converters installed.

*ProtectReport=Y/N*

This setting specifies whether the report generated is protected. If the value is Y, the report is protected, and can not be modified. If the value is N, the report is not protected. Default is N.

*ProtectionPassword=<protectionpassword>*

This setting specifies the password that is used to protect the report. <protectionpassword> value is the password. This setting is valid when ProtectReport is Y. If there is not this setting and ProtectReport is Y, a random password will be created.

*LogFileName=<logfilename>*

This setting specifies the name of the log file. <logfilename> value is the name and path of the log file. The file path can be a relative path or an absolute path. If it is a relative path, the base path is the path of the WRF file. In <logfilename>, you can use parameters.

## 9.4 [PARAMETER] Section

[PARAMETER] section contains information about parameters.

*Name1=<name1>*

*Name2=<name2>*

.....

*Name10=<name10>*

These settings specify the names of the parameters. Name1 specify the name of the first parameter, Name2 specifies the name of the second parameter.....

You can define up to 10 parameters in one WRF file.

*Title1=<title1>*

*Title2=<title2>*

.....

*Title10=<title10>*

These settings specify the titles of the parameters. Title1 specifies the title of the first parameter. Title2 specifies the title of the second parameter.....

*Default1=<default1>*

*Default2=<default2>*

.....

*Default10=<default10>*

These settings specify the default values of the parameters. Default1 specifies the default value of the first parameter. Default2 specifies the default value of the second parameter.....

## Chapter 10 Format Expressions in Data Cells

For a cell in which data are got from data source, you can set the format using a format expression. WDRReportGen gets the text from the cell, and outputs the result using it as the format expression. In fact, WDRReportGen calls the format function in Visual Basic. For more information about format, refer to Format Function in *Visual Basic for Applications Reference*.

### A.1 Formats for Numeric Values

A user-defined format expression for numbers can have from one to four sections separated by semicolons. If the format argument contains one of the named numeric formats, only one section is allowed.

| If you use       | The result is   |
|------------------|---|
| One section only | The format expression applies to all values.  |
| Two sections     | The first section applies to positive values and zeros, the second to negative values.  |
| Three sections   | The first section applies to positive values, the second to negative values, and the third to zeros.                            |
| Four sections    | The first section applies to positive values, the second to negative values, the third to zeros, and the fourth to Null values. |

The following example has two sections: the first defines the format for positive values and zeros; the second section defines the format for negative values.

```
"$#,##0;($#,##0)"
```

If you include semicolons with nothing between them, the missing section is printed using the format of the positive value. For example, the following format displays positive and negative values using the format in the first section and

displays "Zero" if the value is zero.

"\$#,##0;;\Z\e\r\o"

The following table identifies characters you can use to create user-defined number formats:

| Character | Description   |
|-----------|---|
| None      | Display the number with no formatting.  |
| (0)       | Digit placeholder. Display a digit or a zero. If the expression has a digit in the position where the 0 appears in the format string, display it; otherwise, display a zero in that position.<br><br>If the number has fewer digits than there are zeros (on either side of the decimal) in the format expression, display leading or trailing zeros. If the number has more digits to the right of the decimal separator than there are zeros to the right of the decimal separator in the format expression, round the number to as many decimal places as there are zeros. If the number has more digits to the left of the decimal separator than there are zeros to the left of the decimal separator in the format expression, display the extra digits without modification. |
| (#)       | Digit placeholder. Display a digit or nothing. If the expression has a digit in the position where the # appears in the format string, display it; otherwise, display nothing in that position.<br><br>This symbol works like the 0 digit placeholder, except that leading and trailing zeros aren't displayed if the number has the same or fewer digits than there are # characters on either side of the decimal separator in the format expression.   |
| (.)       | Decimal placeholder. In some locales, a comma is used as the decimal separator. The decimal placeholder determines how many digits are displayed to the left and right of the   |

|     |   |
|-----|---|
|     | <p>decimal separator. If the format expression contains only number signs to the left of this symbol, numbers smaller than 1 begin with a decimal separator. To display a leading zero displayed with fractional numbers, use 0 as the first digit placeholder to the left of the decimal separator. The actual character used as a decimal placeholder in the formatted output depends on the Number Format recognized by your system.</p>   |
| (%) | <p>Percentage placeholder. The expression is multiplied by 100. The percent character (%) is inserted in the position where it appears in the format string.</p>  |
| (,) | <p>Thousand separator. In some locales, a period is used as a thousand separator. The thousand separator separates thousands from hundreds within a number that has four or more places to the left of the decimal separator. Standard use of the thousand separator is specified if the format contains a thousand separator surrounded by digit placeholders (0 or #). Two adjacent thousand separators or a thousand separator immediately to the left of the decimal separator (whether or not a decimal is specified) means "scale the number by dividing it by 1000, rounding as needed." For example, you can use the format string "##0,," to represent 100 million as 100. Numbers smaller than 1 million are displayed as 0. Two adjacent thousand separators in any position other than immediately to the left of the decimal separator are treated simply as specifying the use of a thousand separator. The actual character used as the thousand separator in the formatted output depends on the Number Format recognized</p> |

|               |  |
|---------------|--|
|               | by your system.  |
| (:)           | Time separator. In some locales, other characters may be used to represent the time separator. The time separator separates hours, minutes, and seconds when time values are formatted. The actual character used as the time separator in formatted output is determined by your system settings.   |
| (/)           | Date separator. In some locales, other characters may be used to represent the date separator. The date separator separates the day, month, and year when date values are formatted. The actual character used as the date separator in formatted output is determined by your system settings.  |
| (E- E+ e- e+) | Scientific format. If the format expression contains at least one digit placeholder (0 or #) to the right of E-, E+, e-, or e+, the number is displayed in scientific format and E or e is inserted between the number and its exponent. The number of digit placeholders to the right determines the number of digits in the exponent. Use E- or e- to place a minus sign next to negative exponents. Use E+ or e+ to place a plus sign next to positive exponents. |
| - + \$ ( )    | Display a literal character. To display a character other than one of those listed, precede it with a backslash (\) or enclose it in double quotation marks (" ").   |
| (\)           | Display the next character in the format string. To display a character that has special meaning as a literal character, precede it with a backslash (\). The backslash itself isn't displayed. Using a backslash is the same as enclosing the next character in double quotation marks. To display a  |

|         |  |
|---------|--|
|         | backslash, use two backslashes (\\).<br>Examples of characters that can't be displayed as literal characters are the date-formatting and time-formatting characters (a, c, d, h, m, n, p, q, s, t, w, y, / and :), the numeric-formatting characters (#, 0, %, E, e, comma, and period), and the string-formatting characters (@, &, <, >, and !). |
| ("ABC") | Display the string inside the double quotation marks (" "). To include a string in format from within code, you must use Chr(34) to enclose the text (34 is the character code for a quotation mark (")).  |

## A.2 Formats for String Values

A format expression for strings can have one section or two sections separated by a semicolon (;).

|                  |   |
|------------------|---|
| If you use       | The result is   |
| One section only | The format applies to all string data.  |
| Two sections     | The first section applies to string data, the second to Null values and zero-length strings (""). |

You can use any of the following characters to create a format expression for strings:

| Character | Description   |
|-----------|---|
| @         | Character placeholder. Display a character or a space. If the string has a character in the position where the at symbol (@) appears in the format string, display it; otherwise, display a space in that position. Placeholders are filled from right to left unless there is an exclamation point character (!) in the format |

|   |   |
|---|---|
|   | string.   |
| & | Character placeholder. Display a character or nothing. If the string has a character in the position where the ampersand (&) appears, display it; otherwise, display nothing. Placeholders are filled from right to left unless there is an exclamation point character (!) in the format string. |
| < | Force lowercase. Display all characters in lowercase format.  |
| > | Force uppercase. Display all characters in uppercase format.  |
| ! | Force left to right fill of placeholders. The default is to fill placeholders from right to left.   |

### A.3 Formats for Date/Time Values

The following table identifies characters you can use to create user-defined date/time formats:

| Character | Description  |
|-----------|--|
| (:)       | Time separator. In some locales, other characters may be used to represent the time separator. The time separator separates hours, minutes, and seconds when time values are formatted. The actual character used as the time separator in formatted output is determined by your system settings. |
| (/)       | Date separator. In some locales, other characters may be used to represent the date separator. The date separator separates the day, month, and year when date values are formatted. The actual character used as the date separator in formatted output is determined by your system settings.    |
| c         | Display the date as ddddd and display the time as tttt, in that order. Display only date information if there is no fractional part  |

|        |   |
|--------|---|
|        | to the date serial number; display only time information if there is no integer portion.  |
| d      | Display the day as a number without a leading zero (1 – 31).  |
| dd     | Display the day as a number with a leading zero (01 – 31).  |
| ddd    | Display the day as an abbreviation (Sun – Sat).   |
| dddd   | Display the day as a full name (Sunday – Saturday).   |
| dddddd | Display the date as a complete date (including day, month, and year), formatted according to your system's short date format setting. The default short date format is m/d/yy.                          |
| dddddd | Display a date serial number as a complete date (including day, month, and year) formatted according to the long date setting recognized by your system. The default long date format is mmmm dd, yyyy. |
| aaaa   | The same as dddd, only it's the localized version of the string.  |
| w      | Display the day of the week as a number (1 for Sunday through 7 for Saturday).  |
| ww     | Display the week of the year as a number (1 – 54).  |
| m      | Display the month as a number without a leading zero (1 – 12). If m immediately follows h or hh, the minute rather than the month is displayed.   |
| mm     | Display the month as a number with a leading zero (01 – 12). If m immediately follows h or hh, the minute rather than the month is displayed.   |
| mmm    | Display the month as an abbreviation (Jan – Dec).   |
| mmmm   | Display the month as a full month name (January – December).  |
| oooo   | The same as mmmm, only it's the localized version of the string.  |
| q      | Display the quarter of the year as a number (1 – 4).  |

|           |   |
|-----------|---|
| y         | Display the day of the year as a number (1 – 366).  |
| yy        | Display the year as a 2-digit number (00 – 99).   |
| yyyy      | Display the year as a 4-digit number (100 – 9999).  |
| h         | Display the hour as a number without leading zeros (0 – 23).  |
| Hh        | Display the hour as a number with leading zeros (00 – 23).  |
| N         | Display the minute as a number without leading zeros (0 – 59).  |
| Nn        | Display the minute as a number with leading zeros (00 – 59).  |
| S         | Display the second as a number without leading zeros (0 – 59).  |
| Ss        | Display the second as a number with leading zeros (00 – 59).  |
| t t t t t | Display a time as a complete time (including hour, minute, and second), formatted using the time separator defined by the time format recognized by your system. A leading zero is displayed if the leading zero option is selected and the time is before 10:00 A.M. or P.M. The default time format is h:mm:ss. |
| AM/PM     | Use the 12-hour clock and display an uppercase AM with any hour before noon; display an uppercase PM with any hour between noon and 11:59 P.M.  |
| am/pm     | Use the 12-hour clock and display a lowercase AM with any hour before noon; display a lowercase PM with any hour between noon and 11:59 P.M.  |
| A/P       | Use the 12-hour clock and display an uppercase A with any hour before noon; display an uppercase P with any hour between noon and 11:59 P.M.  |
| a/p       | Use the 12-hour clock and display a lowercase A with any hour before noon; display a lowercase P with any hour between noon and 11:59 P.M.  |
| AMPM      | Use the 12-hour clock and display the AM string literal as defined  |

|  |  |
|--|--|
|  | <p>by your system with any hour before noon; display the PM string literal as defined by your system with any hour between noon and 11:59 P.M. AMPM can be either uppercase or lowercase, but the case of the string displayed matches the string as defined by your system settings. The default format is AM/PM.</p> |
|--|--|

# Chapter 11 License and Support

## 11.1 License

### Your Agreement to This License

You should carefully read the following terms and conditions before using, installing, copying, or distributing this software. Your use, installation, copying, or distribution of WDRReportGen indicates your acceptance of this agreement ("License").

### NO WARRANTY

WDREPORTGEN IS DISTRIBUTED "AS IS". NO WARRANTY OF ANY KIND IS EXPRESSED OR IMPLIED. THE AUTHOR WILL NOT BE LIABLE FOR DATA LOSS, DAMAGES, LOSS OF PROFITS OR ANY OTHER KIND OF LOSS WHILE USING OR MISUSING THIS SOFTWARE.

### Evaluation

WDRReportGen is not free software. You may use this software for evaluation purposes without charge for a period of 30 days. If you use this software after the 30 day evaluation period, you must purchase it.

### Use

Each Licensed Copy of this Software may either be used by a user who uses the software personally on one or more computers, or installed on a single computer used nonsimultaneously by multiple people, but not both. This is not a concurrent use license. If you install the Software onto a multi-user platform, server or network, each and every individual user of the Software must be licensed separately.

## **Distribution**

You may copy the evaluation version of this software and documentation as you wish, and give exact copies of the original evaluation version to anyone, and distribute the evaluation version of the software and documentation in its unmodified form via electronic means. You are specifically prohibited from charging, or requesting donations without permission from the author.

You may not modify, copy, share, distribute, re-sell, transfer or sub-license the full version of this software except that you may make copies for archive purposes only. Any such unauthorized use shall result in immediate and automatic termination of this license.

## **Edition**

There are 2 types of editions issued for WDRReportGen.

### 1) WDRReportGen Standard Edition

User can execute no more than 50 SQL statements in one report processing.

### 2) WDRReportGen Professional Edition

User can execute SQL statements unlimitedly in one report processing.

## **11.2 Technical Support**

If you encounter any problems in usage of WDRReportGen, and need the technical support:

- Go to our support web site at:

<http://www.ljzsoft.com/support.htm>

- Send email to [support@ljzsoft.com](mailto:support@ljzsoft.com)