

Tuesday, September 27, 2005

DbFile Language

Team Members:

Erwin Polio (ep3)

Amrita Rajagopal (ar2358)

Anton Ushakov (au2101)

Howie Vegter (hrv2101)

Purpose: DBFile is a new language that will facilitate HTML report creation from any delimited text file. DBFile ease of use will allow non-programmers to quickly create reports via a line command, while it's rich language will allow programmers to create more robust customized reports.

DBFile has a convenient Java-like language style with several built-in functions allowing for easy text file manipulation. DBfile also provides programmers with a rich set of attributes for styling and transforming reports. Final DBFile report can be outputted to an XML file or an HTML page. If outputted to HTML, DBFile will produce 100% HTML and Javascript code making it compatible with most, if not all , available browsers.

Background: Users utilize cumbersome applications to produce reports from delimited text file - Microsoft Access, Crystal Reports, Report Writer to name a few, requiring programming experience to produce a favorable output. Using these applications users must first "convert" the text files, sometimes via a Wizard, prior to creating SQL like queries to filter the data for subsequent report "programming".

The regular user does not keep important data in expensive RDBMS (Oracle, MS SQL Server, Sybase), instead many maintain data in delimited text files. Besides the high retail price, RDBMS require programming experience to properly maintain and manage data making this a difficult storage solution for most users. As such, regular users are forced to maintain their own data in multiple delimited text files.

Goals of DBFile: DbFile ease of use is targeted to both non-programmers and programmers. It will parse, filter, and sort a delimited text file via built-in commands. This data manipulation of delimited text files, along with the rich set of report styling and layout, allow for generation of reports without the need to modify or convert existing files..

Example Quick Usage: Reports can be created by simply invoking a single command at the prompt.

```
(DbFile) > Report(<datasource>, <delimiter>, <where clause>, <output: HTML, XML file>)
```

<datasource>	- Full path and name to text file.
<delimiter>	- Delimiter used in file. Leave blank if tab delimited.
<where clause>	- Clause to filter data much as in the SQL Where clause.
<output>	- Output HTML or XML file. If XML file then give full path and

name.

```
(DbFile) > Report("C:\RevenueStatsJulyAugust2005.txt", "", "Quantity > 2000 AND  
Revenue > 5000", "HTML")
```

REVENUE STATISTICS FOR COMPUTER ACCESSORIES JULY-AUGUST 2005			
PRODUCT	QUANTITY	REVENUE	NET REVENUE
Flash Memory Keys (PID: 24852)	12,345	USD 25,378	USD -1,000
DVD R/W - 25 pk (PID: 07425)	2,345	USD 7,123	USD 987
USB Mouse (PID: 23924)	3,789	USD 9,123	USD 125
CD R - 100 pk (PID: 32852)	11,890	USD 30,456	USD -563
Total Profit: USD 1,112			
Total Loss: USD 1,563			
Copyright @ ABCD Computers Inc.			

Fig. 1. Report created:

Example Programming Usage: For a more robust report the user can use the language to code the report.

```
datafile MyData;  
report MyReport;  
// Set location of file  
MyData.Source = "C:\RevenueStatsJulyAugust2005.txt";  
// Set character delimited  
MyData.Delimiter = Tab;  
  
// Unless specified, the field names will be taken from the file header.  
MyData.Field(0).Name = "Product";  
MyData.Field(1).Name = "Quantity";  
MyData.Field(6).Name = "Revenue";  
MyData.Field(7).Name = "Net Revenue";  
MyData.ColumnHeader.BGColor = Purple;  
  
// Criteria for selecting data  
MyData.Where = "Quantity > 2000 AND Revenue > 5000";  
// Sort by  
MyData.Sort = "PartId";  
  
//Set the origin of the report  
MyReport.Source = MyData;  
  
// change column color  
If MyData.("NetRevenue") < 0 Then
```

```

        MyReport("NetRevenue").ForeColor = Red;
End If

// Provide alernating row colors
If MyReport.NumberofLines > 20 then
    MyReport.AlternateRowColor;
End If

//Robust styling options
MyReport.MainColor = White;
MyReport.Font = "Arial";
MyReport.AlternatingColor = Yellow;

//Output
MyReport.Output = HTML;

MyReport.Header = "Revenue Statistics For Computer Accessories July-August 2005";

Dim Profit as integer;
Dim Loss as integer;
// Sum the revenue (Profit and Loss)
Profit = SumRevenue();
Loss = SumLoss()

//Set footer using variables resturned from functions
MyReport.Footer(1)= "Total Profit: USD" & Profit
MyReport.Footer(2)= "Total Loss: USD " & Loss
MyReport.Footer(3)= "Copyrights ABCD Computers Inc.";

//Create report
MyReport.Generate();

Function SumRevenue()
    Dim tmp as integer;
    // Sum Revenue
    For( i = 0; i < MyReport.Rows.Count -1; i++)
        // Only do positive
        If MyReport.Field(7) > 0 Then
            tmp += MyReport.Field(7);
        End If
    Next i
    Return tmp;
End Function

Function SumRevenue()
    Dim tmp as integer;

```

```
// Sum Revenue
For( i = 0; i < MyReport.Rows.Count - 1; i++)
    // Only do negative
    If MyReport.Field(7) < 0 then
        tmp += MyReport.Field(7);
    End If
Next i
Return tmp
End Function
```