

RAVE Overview

By Glenn Crouch

ESB Consultancy

<http://www.esbconsult.com>

ADUG Symposium 2003

<http://www.adug.org.au>

1	History of RAVE	2
1.1	RAVE Bundled with Delphi 7	2
1.2	RAVE over the Years	2
1.3	Difference between RAVE 5.08BE and 5.1.0BEX	2
2	RAVE IDE & Page Based Design	3
3	Still got Bands	5
4	Calculations	7
5	Data Mirroring & Scripting	8
6	Code Based Design	10
7	RAVE Renders	12
8	RAVE Server	12
9	End User Design	13
10	Third Party Add-Ons	13
10.1	RANT	13
10.2	Gnostice Rave Renders	13
10.3	ESB Add-Ons	14
10.3.1	AddictTool	14
10.3.2	ESBTools	14
10.3.3	ESBRaveViewer	15
10.4	Other Add-Ons	15
11	Conclusion	16

1 History of RAVE

1.1 *RAVE Bundled with Delphi 7*

With the advent of Borland Delphi 7, RAVE 5 is now included with Professional Edition and above. QuickReports is no longer installed by default, but like when ReportSmith was “removed” it is still there if you wish to use it.

Thus it is an opportune time to have an overview of RAVE, the Reporting Engine from Nevrona Designs.

1.2 *RAVE over the Years*

- RAVE started life as ReportPrinter v1 which was released a couple of months after Delphi 1 was released. It was purely a code-based Report Engine.
- v1.1 came out later in 1995 and added Previewing and Memo support.
- v2 was released as ReportPrinter Pro and came out half way through 1996, and with this release the code base become much more sophisticated.
- v3 came out as ReportPrinter Pro & RAVE – thus adding the Visual IDE for the first time. This was early 1998.
- v4 saw the name change to just RAVE and this was in 2001.
- v5 – our current version – came out in 2002

I bought my copy of Report Printer v1 on 13 June 1995 and paid a whole \$25USD for it ☺ Nevrona Designs decided to start an official volunteer support group at the end of 1996 – TeamND – and I have been a part of that since it began.

1.3 *Difference between RAVE 5.08BE and 5.1.0BEX*

“Borland Edition” is the version of RAVE included with Delphi 7. “Borland Extended Edition” is the commercial version. Borland has recently released an update to v5.08 for registered users, and this is highly recommended.

http://www.borland.com/products/downloads/download_delphi.html

Some of the differences between the bundled and commercial versions:

- Source code – the BEX version comes with full component source code (not the IDE source).
- “Archived” Code Based Components are included with the BEX version for backward compatibility.
- More frequent updates.
- Support for TeeChart Pro – the BE version only supports the bundled TeeChart.
- Improved Scripting in 5.1 BEX

So we are getting a “reasonable” version bundled with Delphi and if any of the above matter to you, then you should consider getting the BEX version.

Other improvements planned for RAVE BEX currently include:

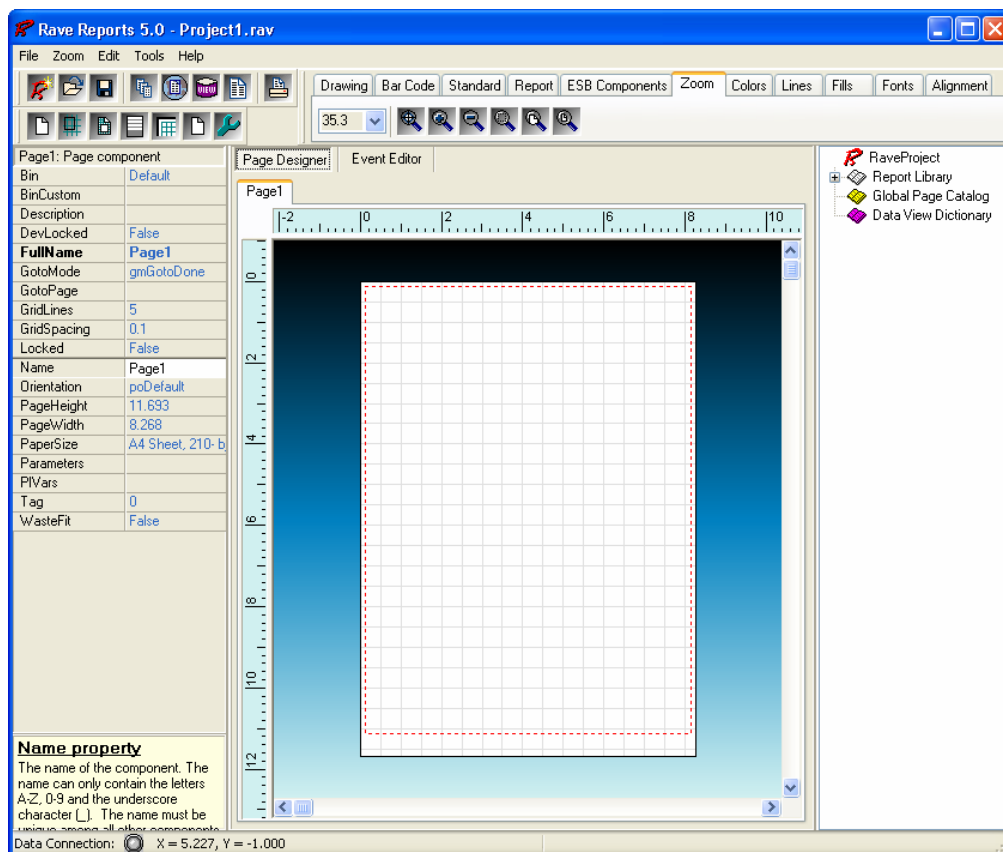
- Native Charting.
- Enhanced Wizards.
- Enhanced Band Logic.

Note: RAVE 5 BEX sells for \$349USD whereas Delphi 7 Owners can “upgrade” the bundled RAVE 5 BE to BEX for \$199USD.

2 RAVE IDE & Page Based Design

The RAVE IDE requires the Delphi IDE to be running for it to be used. Once we have the Delphi IDE running we can simply choose RAVE from the Tool Menu.

One of the first things to Note when you fire RAVE up is that it is a Page based Designer:



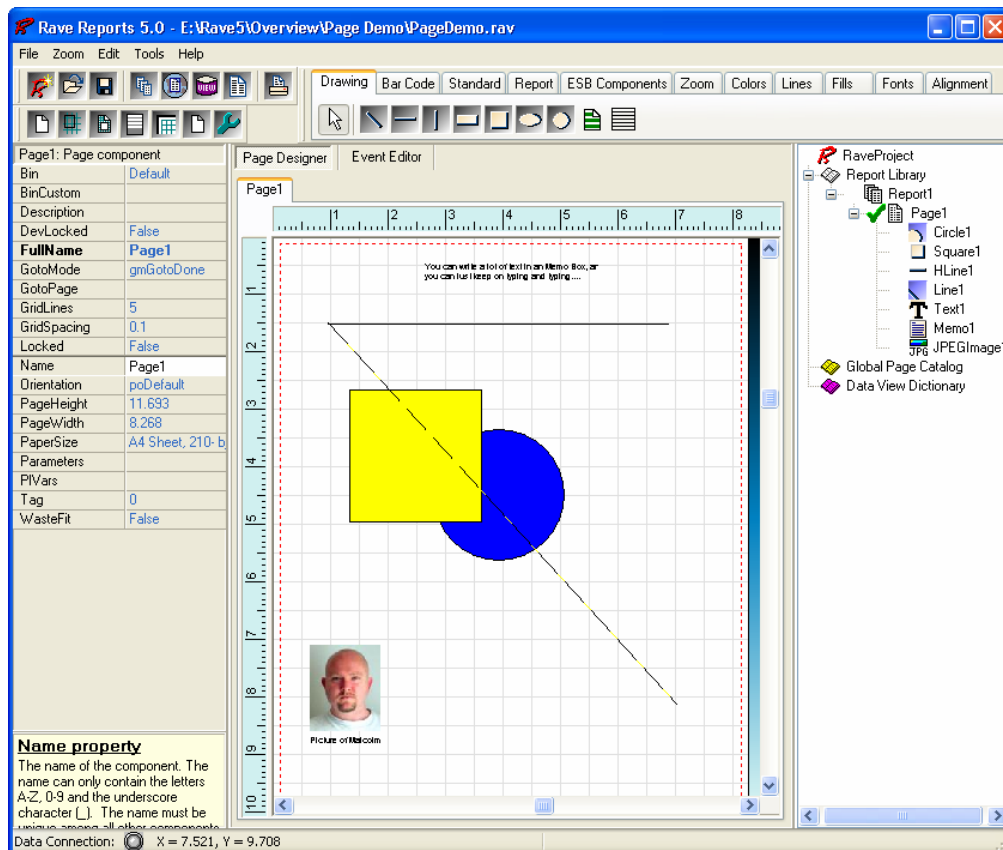
You can see the various parts of the IDE above:

- **ToolBars** are at the Top and include Components, Alignment, Zoom, etc.
- The **Property Panel** is on the Left and has a familiar Delphi feel – though it does have a nice improvement of the Property Help at the bottom.
- The **Project Tree Panel** is on the Right and this also should have a familiar feel about it.

- And in the middle, we have the “Page”. Rave is oriented around this Page Layout.
- The Status Bar (at the bottom) gives us information of Data Connection and Positioning.

First task is to go to configuration and change the Paper to A4 since RAVE comes setup for American Letter. Click on the Spanner in the ToolBar or go to the Menu and choose Edit/Preferences. Whilst in here, ensure that the settings are what you desire.

So let's do some drawing on the Page so we have something like the below:

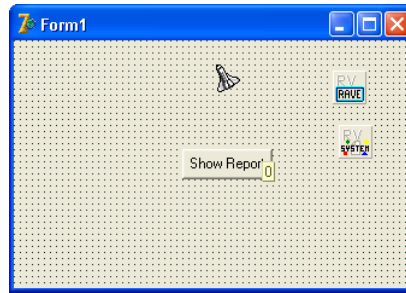


Note that we are using the JPEG Image add-on in the above.

Let's save the above report as PageDemo.rav

Now let's see what is involved in getting the above to work for Delphi. We need an TRVProject to talk to the RAVE Project and a TRVSystem to be able to “adjust” the Printing System. Then Engine Property of the RVProject is set to the RVSystem.

And we'll add a button to generate the Report:



```
uses
    NDCSJPEG; // For JPEG Image Add-On

procedure TForm1.Button1Click(Sender: TObject);
begin
    RVProject1.ProjectFile := 'PageDemo.rav';
    RVProject1.Execute;
end;

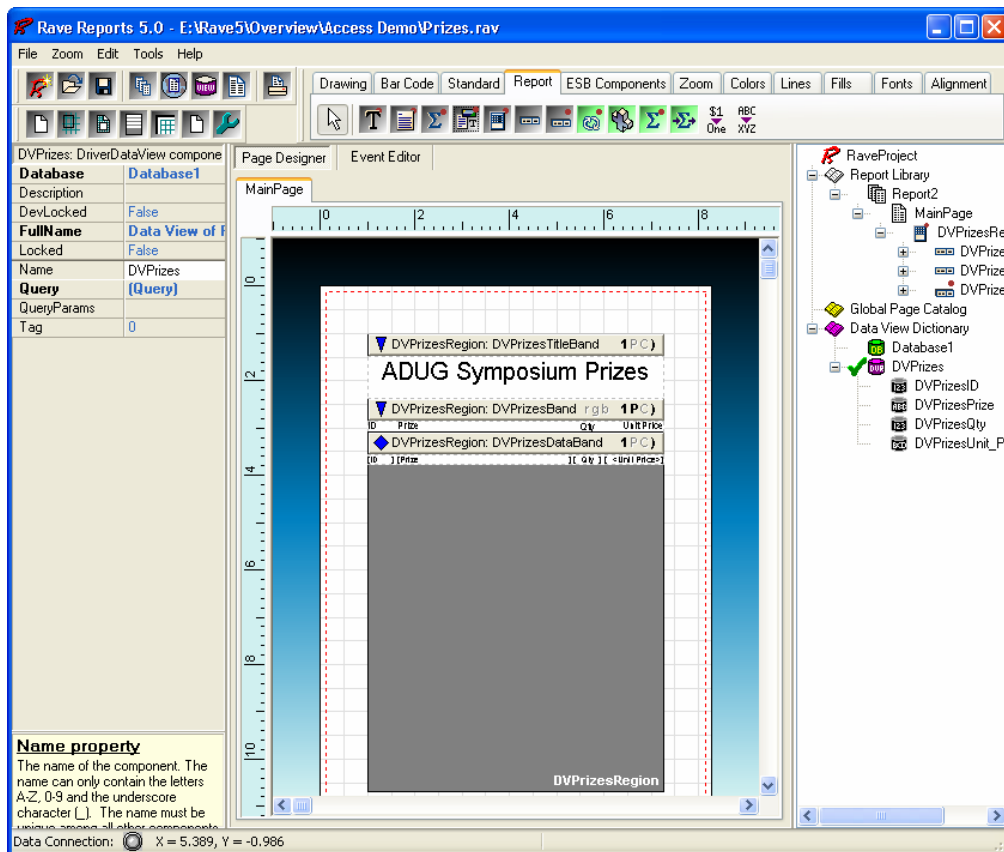
procedure TForm1.FormCreate(Sender: TObject);
begin
    NDCSJPEG.RaveRegister; // For JPEG Image Add-On
end;
```

As can be seen, we've used a RAVE Add-on for the JPEG – more info on how to grab add-ons later – however just need to mention that the above needs NDCSJPEG.PAS in the Library Path.

3 Still got Bands

We've got an MS Access Database that contains the Prize Information for the Symposium. Let's make a report for that.

- Add a Database
- Add a Driver Data View
- Run the Tools/Wizard/Simple Listing and choose all the fields.
- Use the Alignment tools to straighten things up and you should end up with something akin to:



Press the Execute Button and you can see the actual output from the Database.

- Change the Report Name to Prizes1.

Let's save this as Prizes.rav and see what is involved in getting it to work from Delphi. Now remember we have used a Driver DataView, which means all the Database work is in the Report. However to allow Delphi to have access to this, we must add a unit to our uses clause: RvDLADO (since we are accessing the ADO Driver).

So the code looks like:

```
uses
    RvDLADO;

procedure TForm1.Button1Click(Sender: TObject);
begin
    RVProject1.ProjectFile := 'Prizes.rav';
    RVProject1.ExecuteReport ('Prizes1');
end;
```

You can also setup RAVE to use Direct Data Connections from your Delphi Application. Thus the same DataSets being used in your Application can be used from RAVE. In the Sample Files, I show how to do the Prizes Demo using Direct Data Connection but for brevity will continue with the Driver based ones. All the methodology is exactly the same – just a different connection involved.

4 Calculations

RAVE has an extremely powerful set of calculating tools built in. Whilst they do take a bit of getting used to, it is well worth the effort as these are a gold mine when it comes to Report Development.

So let's take our existing project and add Line Totals:

- Adjust Positioning to make room for Line Totals.
- Add a CalcOp (non-visual) component to the DataBand, and set its properties: Src1DataField to Qty, Src2DataField to Unit Price, Operator to Multiply and appropriate Display Format. We also have to assign the output of this calculation so set DestParam to LineTotal. Note: The CalcOp needs to be after the items it is referencing and before the item that references it.
- Add a DataText Component and assign the LineTotal parameter to it.

Note: it is often wise to add your Parameters to the Page, Report or Project, depending on the scope you want them to have.

Okay, let's now add an overall total – this will require us adding a New Band as well:

- Drop a new Band (not a DataBand) on the Report, and set the ControllerBand to the DataBand, and the BandStyle so that it is a Body Footer.
- Add a CalcText Component that will sum up the LineTotal Parameter.
- Add a Text Component to Label the CalcText.
- Add a Horizontal Line

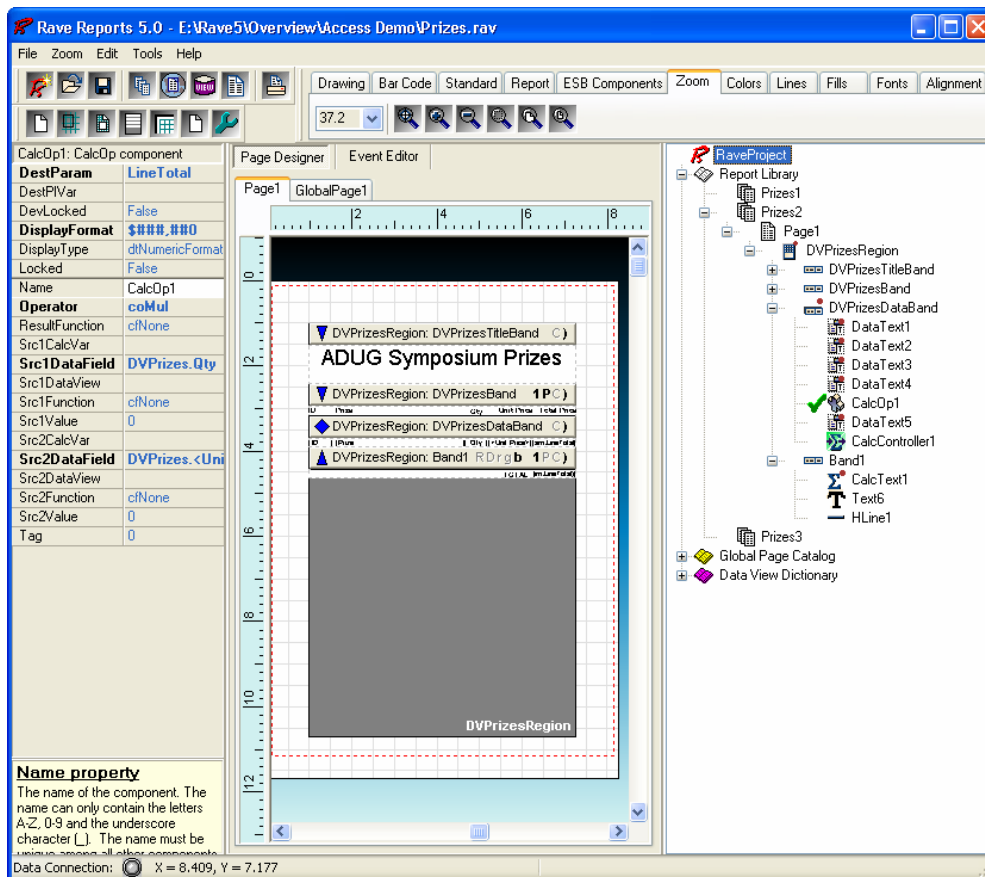
When we run the above the Total remains 0 ☹

We have to tell the CalcText when to sum and to do this:

- Add a CalcController to the DataBand and set the CalcText Controller to this.

Note: The IDE also gives the DataBand as a possible controller. If we were summing up a Data Field then we wouldn't need the CalcController – but we since we are summing up the parameter produced by the CalcOp we do.

The result should be something akin to:

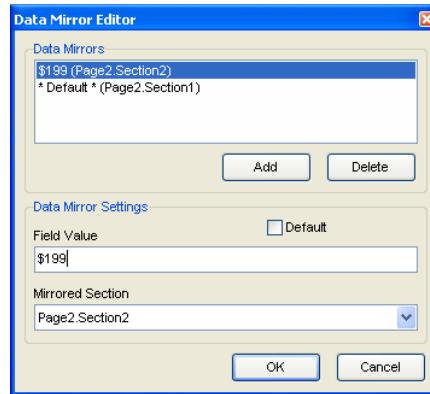


5 Data Mirroring & Scripting

Data Mirroring is another one of those areas in RAVE that take a bit of time to come to terms with but is really worth the effort. Basically, they allow different sections to be used depending on certain conditions.

To start with, let's change the font of the Prize Name for Prizes worth \$199:

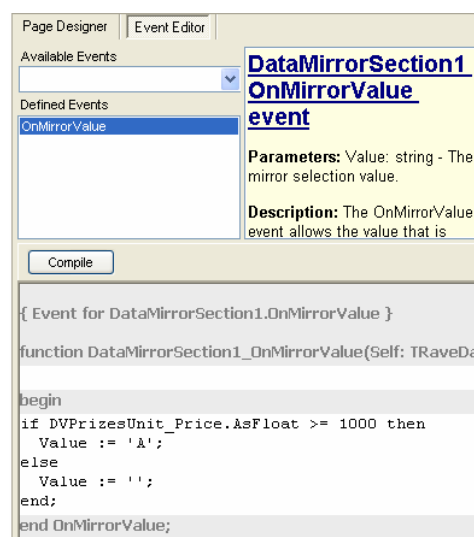
- Add a New Page to the Report.
- Cut the Prize DataText and paste it on this second page.
- Add a DataMirror to the position where the Prize was.
- On Page 2, add 2 Sections and put copies of the Prize DataText into them.
- Change the Font in the second Section.
- In the DataMirror Component set the DataField to Unit Price and assign the two Sections in the DataMirrors Editor. Note how the formatted value is used in the comparison. You can basically think of this like a Case Statement where different Sections are used depending on the field value.



Now equality matches are not always what we are after, and using the Events and Scripting in RAVE, we can do far more sophisticated things. Note that Scripting is still evolving and getting better with each release.

Let's make the font change for Prizes with Unit Value \geq \$1000:

- Click on the DataMirror Component
- Change the view from Page Designer to Event Editor, and add the following to the OnMirror Event:



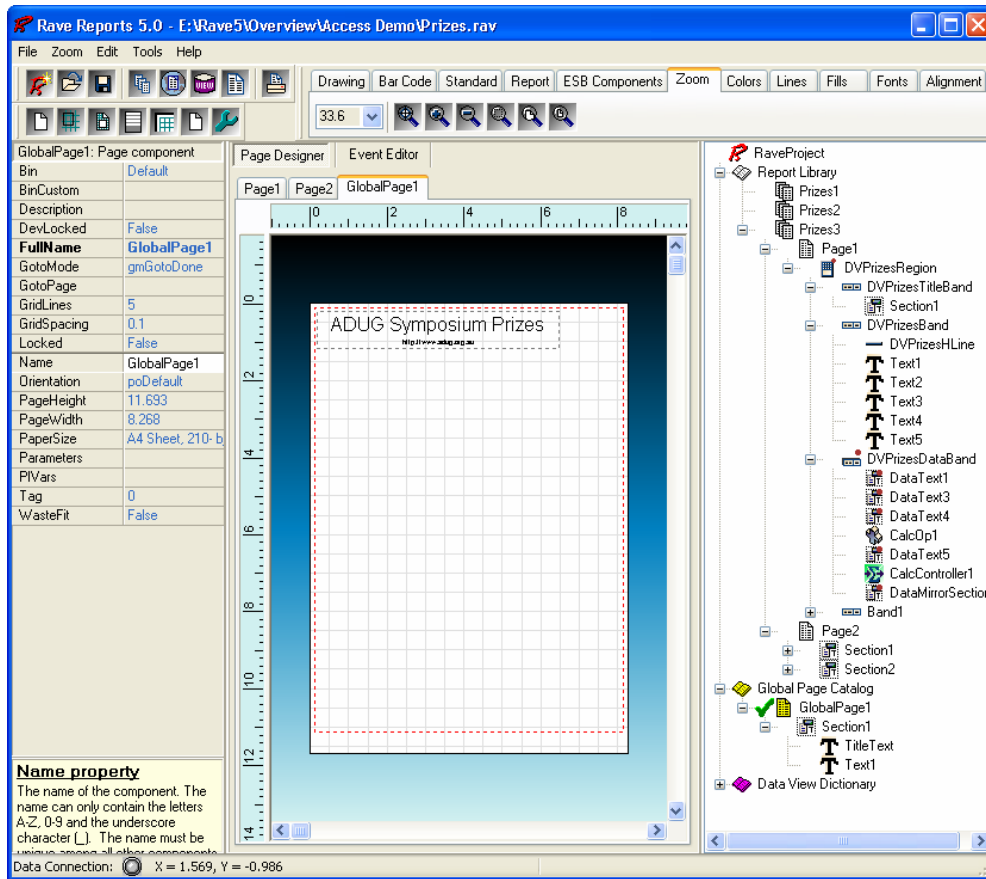
- Remove the DataField from the DataMirror.
- Change the Field for the Second Section from '\$199' to 'A'.

Mirroring can also be used with Global Pages allows us to have items like Letter Heads, Logos, etc – that are in one central place – and thus not only reduce the size of our reports but also make updating easier.

Let's change the Report Header to be use a section from a Global Page:

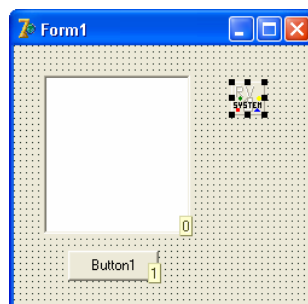
- Add a Global Page.
- Replace the title with a Section.

- On Global Page, create a Section with the Heading Text in it.
- On Page 1, make the Section you add “Mirror” the Global page one.
- Now let’s add the Web Address to the Header by just changing the Global Page.



6 Code Based Design

RAVE had it’s origins as a Code Based Report Engine – and whilst the majority of development work is going into the Visual Side of things, the code based reporting is as feature-rich and stable as ever. Being able to do everything in code does mean that RAVE allows you to do any sort of Report that you can imagine. Given that our time is short, let’s just do a quick example that shows printing a ListBox:



- Drop a ListBox, Button and RVSystem on to a Form.

- In the OnCreate we'll populate the ListBox:

```

procedure TForm1.FormCreate (Sender: TObject);
var
    I: Integer;
    X: Integer;
begin
    Randomize;
    for I := 1 to 100 do
        begin
            X := Trunc (Random (1000)) + 1;
            ListBox1.Items.Add ('Item: ' + IntToStr (X));
        end;
    end;

```

- The in the OnClick of the Button:

```

procedure TForm1.Button1Click (Sender: TObject);
begin
    RVSystem1.Execute
end;

```

- And we do the work in the OnPrint Event of the RVSystem:

```

procedure TForm1.RvSystem1Print (Sender: TObject);
var
    I: Integer;
    RP1: TBaseReport;

    procedure PHeader;
    begin
        with RP1 do
            begin
                Home;
                // Set up the Page
                SectionLeft := 0.5;
                SectionRight := PageWidth - 0.5;
                SectionTop := 1.0;
                SectionBottom := PageHeight - 1.5;

                SetFont ('Arial', 16);
                FontColor := clNavy;
                Bold := True;
                PrintCenter ('List Box Demo', PageWidth / 2);
                Bold := False;
                NewLine;
                NewLine;
                SetFont ('Arial', 12);
                FontColor := clBlack;

                ClearTabs;
                SetTab (SectionLeft, pjCenter, 1.0, 5, BoxLineAll, 0);
                SetTab (NA, pjCenter, PageWidth - 2, 5, BoxLineAll, 0);

                Bold := True;
                PrintTab ('Item No. ');
                PrintTab ('Details');
                Bold := False;
                NewLine;
            end;
    end;

```

```

end;
begin
  RP1 := Sender as TBaseReport;
  with RP1 do
  begin
    PHeader; // Page Header for First page
    for I := 0 to ListBox1.Count - 1 do
    begin
      if LinesLeft < 5 then // New Page
      begin
        NewPage;
        PHeader;
      end;
      PrintTab (IntToStr (I));
      PrintTab (ListBox1.Items [I]);
      NewLine;
    end;
  end;
end;
end;

```

7 RAVE Renders

RAVE supplies Renders that allow you to “save” your reports in different formats. RAVE comes with Renders to do HTML, PDF, RTF and Text. It also supplies the tools needed to develop your own Renders and we will look at other Renders shortly.



By dropping a PDF Render Component on to our Form (or Data Module), we now have the ability to save to PDF when we Preview and click the save as button:



The Save As in the Preview inside the RAVE IDE has all the RAVE Renders activated but in your Delphi Application, you need to choose the one's you want.

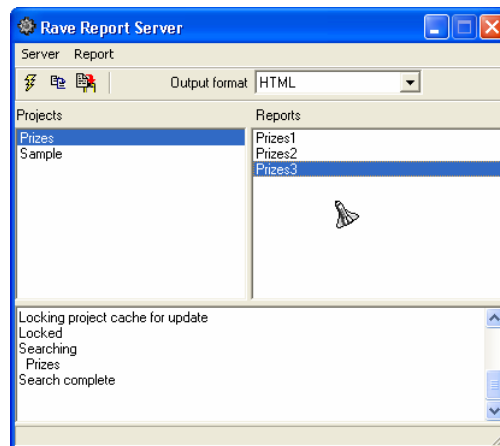
8 RAVE Server

The Rave Server allows you to distribute the processing of Report Generation – especially if you are after generating HTML, PDF or NDR Reports.

To access the Server you need to use commands in the URL – and using this you can easily have Reports on an Intranet or the Internet that are available to people who don't have any Delphi Application running. It does also work as LocalHost.

Now the Server is licensed by concurrent users in a given minute. One of the most overlooked things about the Server, is that the Single User version is FREE!!! When you download the Server you get the Single User version. If you want more concurrent users then you can buy 5-user License and above. But given that Single User is free and that many small businesses actually don't need more than a Single User license, it is well worth checking out.

We'll copy our Prizes RAVE Project over to the server:



Then in our Internet Browser we enter:

http://localhost:4330/ExecuteReport?ND_Project=Prizes&ND_Report=Prizes3&ND_Output=HTML&ND_PageNo=All&Submit=Run+Report

9 End User Design

In addition to purchasing the BEX Version of RAVE, you can also purchase the End User Design, which gives your users a RAVE IDE like ability. This does involve the shipping of a DLL with your Application.

10 Third Party Add-Ons

One of the great strengths of the RAVE Environment – like the Delphi Environment – it lends itself to customisation and improvement, via third party add-ons. You can create New Components, New Wizards, New Property Editors, New Renderers (ie new ways to Export) etc and these can be used just for your Applications, or made Freely Available to other, or even sold Commercially.

10.1 RANT

RAVE supplies RANT (Rave Application iNterface Technology) Specification to allow you to develop your own Add-Ons. Also examining source code to the Add-ons that come with source will help you come to grips with RANT.

Basically, you are programming in Delphi to make a DPK that will produce a BPL.

10.2 Gnostice Rave Renders

If you want to supply sophisticated “export” options to your reports, then I would highly recommend this add-on. We use it in all our client work, as they offer a wider choice of outputs as well as far more configurable:



You not only get improvements on the PDF, HTML, RTF and Text outputs, you also get Excel, JPEG, EMF, WMF, BMP and GIF outputs.

Check it out at: <http://www.gnostice.com>

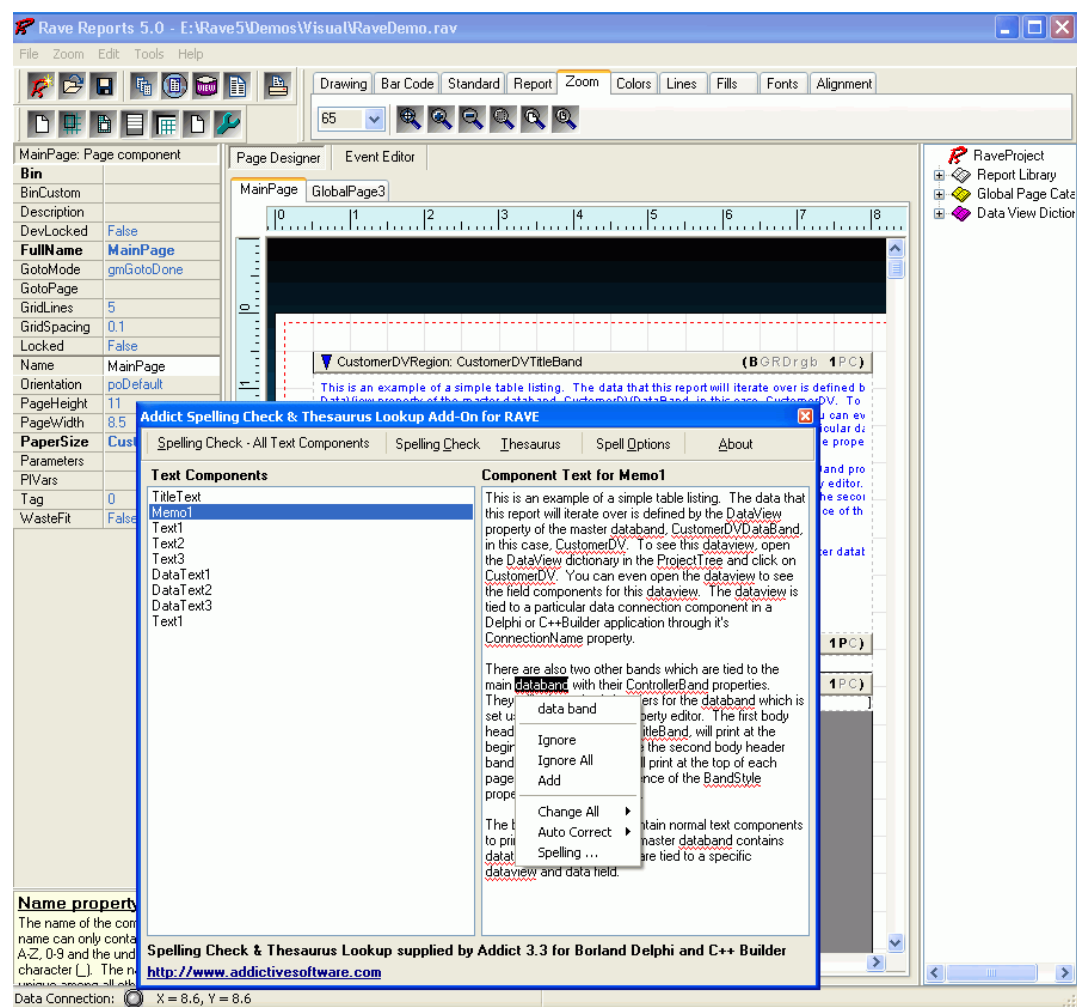
10.3 ESB Add-Ons

I've developed several RAVE Add-ons and more are on the way. The following can all be downloaded and used for free from:

<http://www.esbconsult.com/raveaddons>

10.3.1 AddictTool

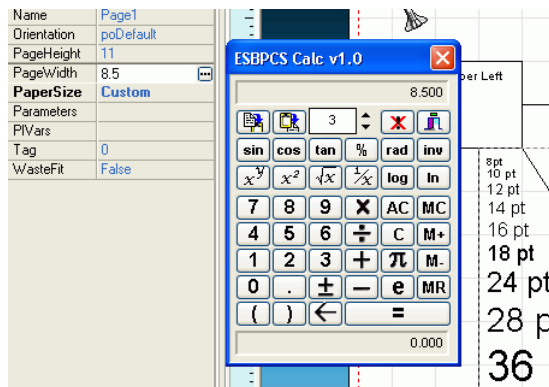
This uses Addict Spell Check & Thesaurus to add a Wizard that checks the spelling and provides Thesaurus Lookup. You don't need to own Addict to use this Wizard.



10.3.2 ESBTools

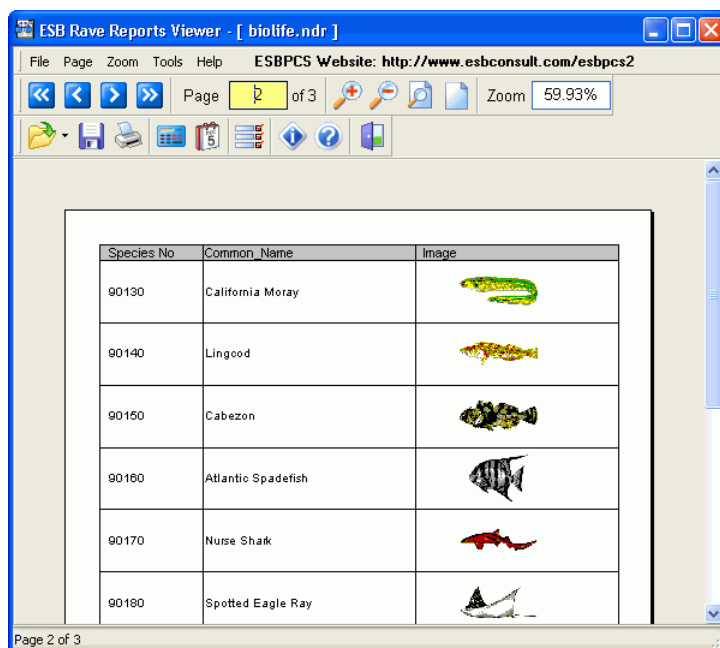
This includes a new Component (with full source) that allows you to have your DataText coloured differently depending on whether it's value is above or below a

given value – eg Positive or Negative. It also includes a Property Editor that adds a Popup Calculator. ESBPCS is not required to use this tool.



10.3.3 ESBReveViewer

This is a handy util that allows you to View NDR Files produced from Applications using RAVE, and then allows you to Print it or to export it to HTML, PDF, RTF or Text. This is really just a “fancy” Preview replacement.



10.4 Other Add-Ons

There are several other free Add-Ons at the Nevrona Site that do things such as add support for JPEG Images, add support for GIF Images, supply “Green Bars”, etc.

You should check the site regularly:

<http://www.nevrona.com/rave/addons.shtml>

11 Conclusion

For more information you may want to check out Nevrona's Website:

<http://www.nevrona.com>

Also the Demos and PowerPoint Presentation used today will be available for download from the ADUG Website, as well as from my company's RAVE Add-Ons sub-web:

<http://www.esbconsult.com/raveaddons>

For RAVE support send your questions to Borland's RAVE Newsgroup or to the appropriate Nevrona Newsgroup:

<http://www.nevrona.com/support>

Kazoo Software have just started to release Online RAVE Training Videos. They are currently offering quarterly and yearly subscription that get you access to everything.

<http://www.kazoosoft.com/videos.aspx>