
Rave Reports

Adaptado do artigo "**Conhecendo o Visual Report Designer do Rave**" de **Sidinei Akira Egashira** disponível no site **delphiBR** em:
<http://delphibr.com.br/artigos/1strave.php>

Índice

Introdução
Características
Componentes
Visual Report Designer
 Invocando a interface visual do Rave Report
 Componentes de Projeto
 Barra de Utilitários e Componentes
 Object Inspector
 Tree View
 Página
 Conexão com Dados

Introdução

O Rave Reports é desenvolvido e mantido pela empresa Nevrona Designs (www.nevrona.com). A partir do Delphi 7 o Rave Reports tornou-se a ferramenta padrão para geração de relatórios, pelos desenvolvedores Delphi. A Nevrona desenvolveu uma edição exclusiva para os produtos Borland, o Rave Reports Borland Edition Version (Rave BEX).

Apesar de ser distribuído junto ao Delphi a partir da versão 7, o Rave não é uma ferramenta nova no mercado. Na verdade, o Rave é uma nova versão do Report Printer, ferramenta que está sendo mantida e desenvolvida desde a primeira versão do Delphi.

Características

1. O Rave possui um ambiente totalmente visual para criação de relatórios. O Rave Visual Designer é semelhante a IDE do Delphi, possuindo paleta de componentes e janelas parecidas com o Object Inspector e Object TreeView;
2. Com o Rave é possível criar relatórios para aplicações VCL, CLX e VCL.NET desenvolvidas em Delphi 6 a 2005 ou C++ Builder 6. Utilizando a versão para Linux podemos criar aplicações multiplataforma utilizando o Kylix;

-
3. Habilita o desenvolvimento de relatórios com acesso direto ao banco de dados, utilizando diversas tecnologias, como: BDE, dbExpress, ADO e IBX. Possui ainda um editor visual para criação de instruções SQL integrado ao ambiente;
 4. Salva relatórios nos formatos RTF, HTML, PDF e texto. Possui ainda um formato proprietário (NDR);
 5. Possui diversas ferramentas para formatação dos relatórios, como alinhamento e posicionamento de objetos;
 6. Todos os relatórios da aplicação podem ser salvos em um único arquivo Rave (RAV). O Rave trabalha com o conceito de projetos, onde um projeto pode conter vários relatórios;
 7. Os relatórios podem ser distribuídos separadamente do executável da aplicação ou embutidos dentro do mesmo;
 8. Os relatórios são baseados em páginas, regiões e bandas;
 9. O Rave possibilita visualizar e testar os relatórios em tempo de projeto a partir do Rave Visual Designer;
 10. Disponibiliza a criação de páginas globais para serem usadas como base para vários relatórios;
 11. Total acesso aos objetos do relatório a partir da aplicação Delphi;
 12. Através da tecnologia de "espelhamento", é possível reutilizar conteúdo e objeto entre os relatórios do projeto Rave;
 13. Componentes para suporte a código de barras;
 14. Linguagem de programação própria (Rave Language) para codificação de eventos do relatório;
 15. Wizards para criação de relatórios simples e mestre / detalhe;
 16. Possui um servidor web (versão Server) para disponibilizar relatórios em browsers;
 17. Disponibiliza um ambiente de desenvolvimento de relatórios para o usuário final, onde o usuário pode alterar e criar seus próprios relatórios. Dependendo do nível de conhecimento do usuário final, o Rave disponibiliza três níveis de acesso: beginner, intermediate e advanced;
 18. Suporte para as versões 4 a 2005 do Delphi e 4 a 6 do C++ Builder.

Componentes Rave Reports

TRvSystem: Responsável pela geração do relatório e envio para Tela ou Impressora, fazendo a ligação com os componentes TRvRender*

TRvProject: Responsável pela ligação entre o projeto Delphi e o projeto Rave Reports

TRvCustomConnection: Usado para acessar dados que não vem de Bancos de Dados como matrizes ou arquivos de registros (records)

TRvDataSetConnection: Usado para acessar componentes de acesso a Bancos de Dados descendentes do TDataSet (Exemplo: Componentes ADO Express e DB Express)

TRvTableConnection: Usado para acessar componentes TTable (BDE) de acesso a Bancos de Dados.

TRvQueryConnection: Usado para acessar componentes TQuery (BDE) de acesso a Bancos de Dados.

TRvNDRWriter: Gera um arquivo em formato binário com o relatório

TRvNDRPrinter: Imprime o arquivo gerado pelo RvNDRWriter

TRvNDRPrinter: Mostra (Preview) o arquivo gerado pelo RvNDRWriter

TRvRenderPDF: Grava relatório em formato PDF

TRvRenderHTML: Grava relatório em formato HTML

TRvRenderRTF: Grava relatório em formato RTF

TRvRenderText: Grava relatório em formato TEXTO

Visual Report Designer

O Rave Reports traz uma interface independente de desenvolvimento de relatórios. Seus projetos de relatórios podem ser gravados em arquivos à parte ou incorporados em seus sistemas. Caso a primeira opção seja utilizada, essa interface visual pode ser instalada em qualquer computador e, tendo acesso aos arquivos de relatórios, você pode alterá-los e numa próxima vez que sua aplicação invocar o relatório as alterações já estarão feitas. Sem necessidade de compilação de seu sistema.

Invocando a interface visual do Rave Reports

A interface visual pode ser invocada de várias formas. Uma delas é colocando um componente RvProject em seu sistema e dando um duplo clique sobre ele. A relação entre seu sistema e o relatório é feito por meio deste componente. Uma segunda forma é acessando a ferramenta a partir do menu Tools do Delphi 7 (veja figura 1). A terceira e última forma é chamando o Rave a partir de seu diretório de instalação (default - c:\Arquivos de Programas\Borland\Delphi7\Rave).

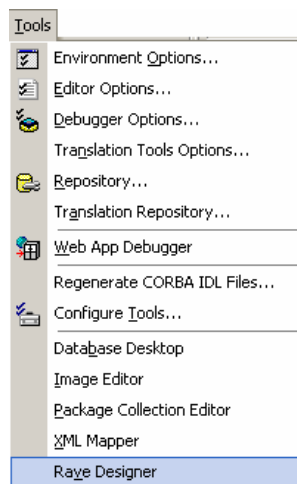


Figura 1 – Invocando o Visual Rave Designer a partir do menu Tools do Delphi7

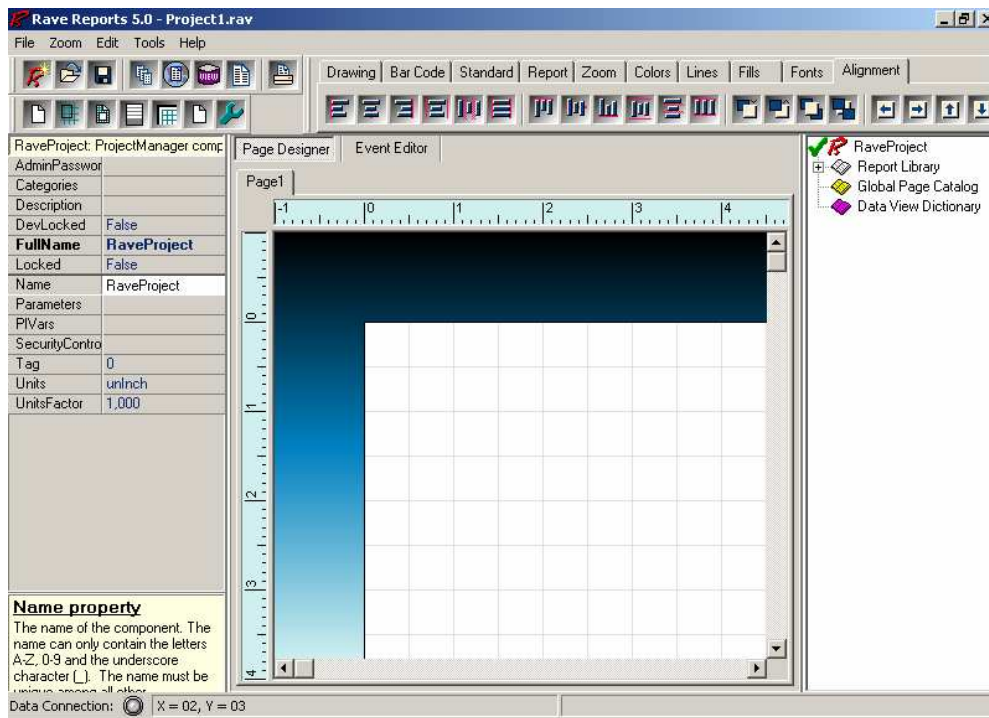


Figura 2 – Visual Designer do Rave Reports

Componentes de Projeto



Figura 3 – Os componentes de projeto

É nesta área que aparecem os componentes para criação de um novo projeto de relatório, fazer as conexões de nossos dados com os relatórios, determinar se serão mostrados os grids e as régua na página de desenvolvimento.

Barras de utilitários e componentes

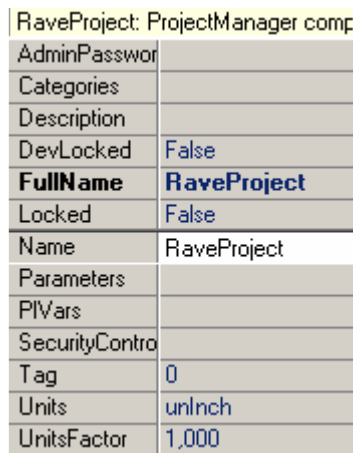


Figura 4 – A barra de componentes e utilitários de relatório

Nesta seção teremos todos os componentes e utilitários que precisamos para criar nossos relatórios. A barra mais importante é a Report Components. É a partir dessa barra que teremos todos os componentes necessários para criar o relatório em si (as bandas, datatexts, datamemos, etc).

Object Inspector

Assim como no Delphi, aqui no Visual Designer você tem um Object Inspector (fica no lado esquerdo da tela). É a partir dele que você trabalha com as propriedades de todos os componentes que você utiliza em seu relatório. Por exemplo, uma banda tem uma série de propriedades, que devem ser alteradas pelo object inspector. Sempre que um componente estiver selecionado na página o Object Inspector estará mostrando seus atributos.



RaveProject: ProjectManager comp	
AdminPasswor	
Categories	
Description	
DevLocked	False
FullName	RaveProject
Locked	False
Name	RaveProject
Parameters	
PIVars	
SecurityContro	
Tag	0
Units	unInch
UnitsFactor	1,000

Figura 5.1 – Object Inspector do Visual Designer

Um fato interessante do Object Inspector é que toda vez que você alterar uma propriedade, a sua descrição fica em negrito. Dessa forma você sabe quais foram as propriedades alteradas e quais ainda estão com o valor default.

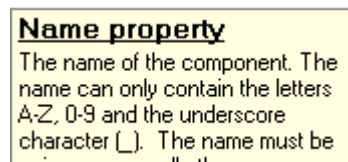


Figura 5.2 – Pequeno descritivo da propriedade selecionada no Object Inspector

Logo abaixo do Object Inspector você pode visualizar uma caixa de texto amarela. Ali é mostrado um hint da propriedade que está selecionada naquele momento. No exemplo mostrado acima, a propriedade Name estava selecionada. Portanto, um pequeno descritivo da propriedade Name é mostrado. Isso facilita muito quando não sabemos exatamente o que a propriedade faz.

Tree View



Figura 6 – O Tree View

Na Tree View (fica no lado direito da tela) você pode visualizar todos os relatórios que contém em seu projeto (um mesmo projeto pode armazenar quantos relatórios forem necessários). Você também poderá visualizar todos os componentes que estão instanciados em seu relatório. Na seção Report Library você tem os relatórios. Na seção Global Page Catalog você tem as partes de relatório que são vistas por todos os relatórios (aqui você aplica o conceito de mirroring) e no Data View Dictionary você visualiza todas as ligações feitas com seus bancos de dados.

Alguns componentes são instanciados no relatório, porém não são visuais. Você só poderá alterar suas propriedades a partir da Tree View.

Página

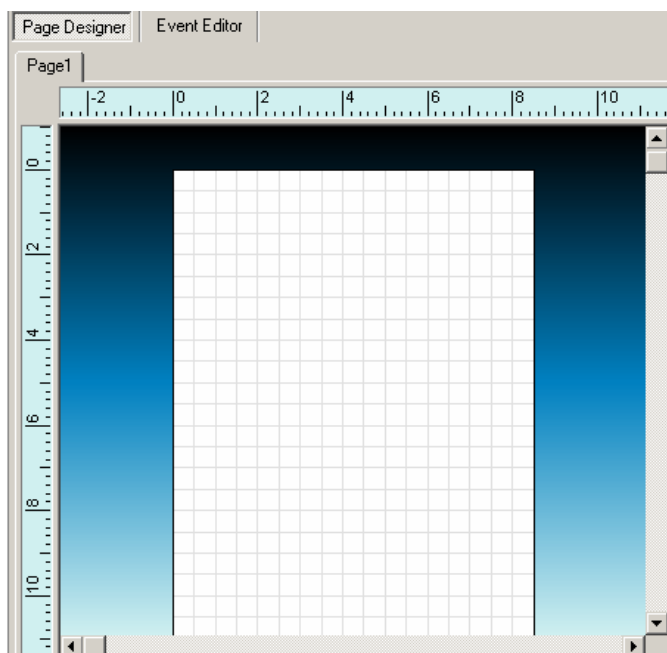


Figura 7.1 – A página – o relatório propriamente dito

A página é considerada a parte mais importante do Visual Designer, pois é nele que tudo acontece. É aqui que seu relatório será desenhado. Note que temos a descrição Page1 em nossa PageControl. Isso quer dizer que estamos editando a página 1 de nosso relatório. Um relatório pode ter mais que uma página. Um pouco acima temos Page Designer, que quer dizer que estamos no modo de desenvolvimento da página. Ao lado temos Event Editor. Ao acessar o Event

Editor, você poderá incluir um código de programa para qualquer componente que estiver estanciado em seu relatório (veja figura 7.2)

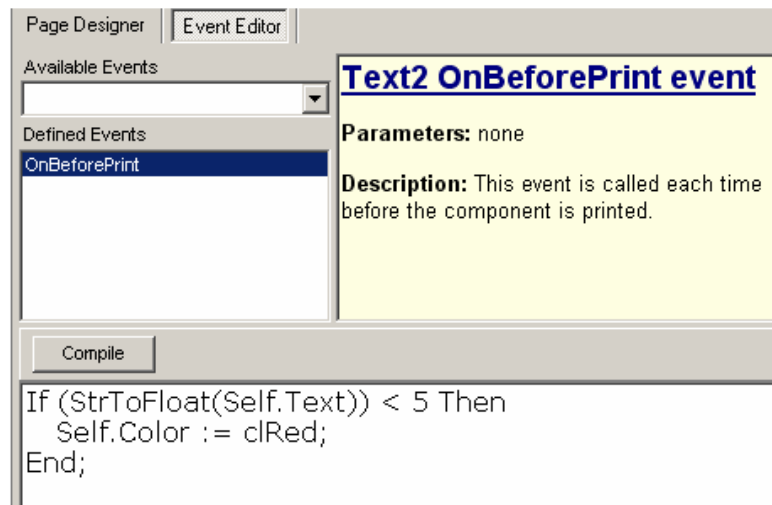






Figura 7.2 – Event Editor – Você pode inserir trechos de código para cada componente de seu relatório

Conexão com Dados



O Data Connection mostra qual é a situação em que seu relatório se encontra em relação as ligações com bancos de dados. Seu LED pode assumir quatro estados (cores) diferentes.

-  Led Cinza – Conexão inativa ou inexistente
-  Led Amarelo – Conexão ativa, porém aguardando resultado
-  Led Verde – Conexão ativa e em uso
-  Led Vermelho – Conexão ativa, porém já excedeu o tempo limite de espera