PASVirtualDBScroll

This manual provides information on using, extending, and contributing toward the development of the PASVirtualDBScroll Component Packages.

Background

Most data-aware components fall into two variants:

1. Components that display a single field of a single record, and can easily perform any applicable scrollbar calculations
2. Components that display multiple fields of multiple records in a grid. These components display each record in rows of equal height. They only display the records currently in the record buffer, so extremely large datasets can be scrolled. Since every row is the same height, and a constant number of records is displayed at all times, the scrollbar calculations for these components are also trivial.

I was developing a software application for displaying the contents of logbooks, and found that I could either use a data grid, in which case displaying records that had significant variation in length would be cumbersome and unaesthetic, or write the records to a plain memo component which would eventually result in memory issues as the number of records increased.

In order to get around these constraints, my goal was to create a set of Lazarus components that meet the following requirements:

- Scroll through DataSets of up to approximately 8 million records without knowing beforehand how large each record is
- Display each record in its entirety, with hundreds or thousands of lines allowed per record
- Allow programmers to determine how the data will be formatted in the display

I am still currently in the process of developing and improving these components, but I welcome everyone to view, use, or contribute toward this project.

The repository includes a test project that is being developed along with the components, as well as a set of SQLite Databases with 100, 200, 1000, 5000, 10000, and 20000 dummy records of approximately 1 to 20 lines of text each.

Using the Components

The project consists of three visual components:
- PASVirtualDBScrollMemo
- PASVirtualDBScrollRichMemo

- PASVirtualDBScrollSynEdit

There are also non-visual base classes upon which the visual components are built.
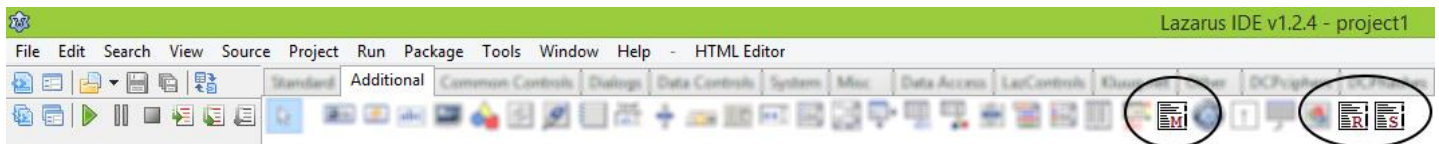
The basic DataSet scrolling functionality of all three components is the same, but the embedded Memo, RichMemo, or SynEdit subcomponents and their applicable properties are available to the programmer.

Installing the Component Packages

The components come in three different packages:

Package	Description
PASVirtualDBScroll.lpk	This is the base package for this set of components and contains the TMemo-based version of the component. This package must be installed prior to attempting installation of the other packages.
PASVirtualDBScrollRichMemoPkg.lpk	This package contains the TRichMemo-based version of the component. This package requires both the PASVirtualDBScroll and RichMemo packages.
PASVirtualDBScrollSynEditPkg.lpk	This package contains the TSynEdit-based version of the component. This package requires both the PASVirtualDBScroll and SynEdit packages.

Assuming all three packages are installed, the following icons will appear on the "Additional" Components Tab

General Operation Concept

These components are Data-Aware, and update automatically based upon the current state of the DataSet. In order to use these components, the DataSource property (and the DataSource.DataSet property) must be set.

Whenever the DataSet is opened, the Record Count is determined, a series of calculations is performed, and a portion of the DataSet is displayed in the component based upon the Format specified by the programmer. When the DataSet is closed, the component is cleared.

*NOTE: In order to use PASVirtualDBScrollRichMemo, you may need to add the richmemopackage Packageas a requirement in your project via the Project Inspector
*NOTE: In order to use PASVirtualDBScrollSynEdit, you may need to add the SynEdit Package as a requirement in your project via the Project Inspector

Properties Specific to These Components

DataSource	Should be set to a valid TDataSource Component or Object
DataSource.DataSet	Should be set to a valid TDataSet Component or Object, or one inherited from TDataSet
EScrollBar	The ScrollBar used for scrolling the component.
EMemo ERichMemo ESynEdit	The subcomponent which actually displays the DataSet contents.
EPopupInfo	
EPopupInfo.IsDisplayed	Allows the programmer to determine whether the info popup should be displayed as a user scrolls through the DataSet
Format	Used to specify how the data should be formatted. Can be designed at designtime, or specified at runtime.
LineResolution	The number of positions on the scrollbar allocated per record. This property is automatically calculated based upon the number of Records in the DataSet. (read-only)
RecordChunkSize	Used to set the number of records per chunk. Three RecordChunks will be loaded into the component at any given time. Therefore, if the value is set to 50, then 150 records will be loaded at a time. Allowable range is 1 to 500, and default is 50.
RecordCount	The number of records in the dataset. For instance, if the DataSet is an SQLQuery, this value is the number of records returned from a query. This property is automatically calculated when the dataset is opened. (read-only)

Specifying the Format

Assume a DataSet that returns records from a logfile of some sort with the following fields:

Field Name	Field Type
id	Integer
Entry	String
Name	String

Then the following are all acceptable formats for displaying these records:

```
------------------------------------------------------------
id: <<<*id:Integer*>>>              Name: <<<*Name:String*>>>
------------------------------------------------------------
Entry:
<<<*Entry:String*>>>
```

```
Name of Person Making Entry: <<<*Name:String*>>>
Entry: <<<*Entry:String*>>>
```

```
Record Number:   <<<*id:Integer*>>>
Log Entry:
<<<*Entry:String*>>>
```

PASVirtualDBScroll Development

PASVirtualDBScroll is currently under development. The following sections will assist you in helping with its development.

Development Environment

PASVirtualDBScroll is currently being developed with Lazarus 1.2.4 (a rapid application development IDE using the Free Pascal Compiler FPC 2.6.4)

Additional Required Packages:
RichMemo - Part of the Code and Component Repository (CCR)
SynEdit – A syntax highlighting edit/memo control

Source Code and Binaries

PASVirtualDBScroll is currently developed by Jack D Linke of Projects and Stuff, but everyone is encouraged to contribute toward development if interested. The source code repository for PASVirtualDBScroll is hosted on GitHub at https://github.com/Projects-and-Stuff/VirtualDBScroll

Donating to PASVirtualDBScroll

If you are interested in contributing to PASVirtualDBScroll but do not have the time to assist with code development, please consider donating toward the project.