

Review: Fastreport Server 2.0

Michaël Van Canneyt

March 11, 2008

Abstract

Fastreport server is a Windows service application which serves database reports to clients. Clients can connect using a specialized client application or using a web browser.

1 Introduction

Most - if not all - database applications need some kind of reports: specialized printouts, based on data in the database. These printouts can be hard-coded in the application, or they are created using some reporting tool. Many reporting tools exist, and Delphi programmers almost certainly know the FastReport set of reporting components: If a lot of printouts must be created, then designing them using a reporting tool is definitely recommended.

Originally designed as a set of powerful reporting components that can be purchased for use with Delphi (They must be purchased because Delphi ships by default with another, less versatile, set of reporting components), they have grown to a reporting tool that can be used independent of Delphi: the FastReport Server. The new version 2.0 will be examined here, based on a demo made available for the purpose of the review.

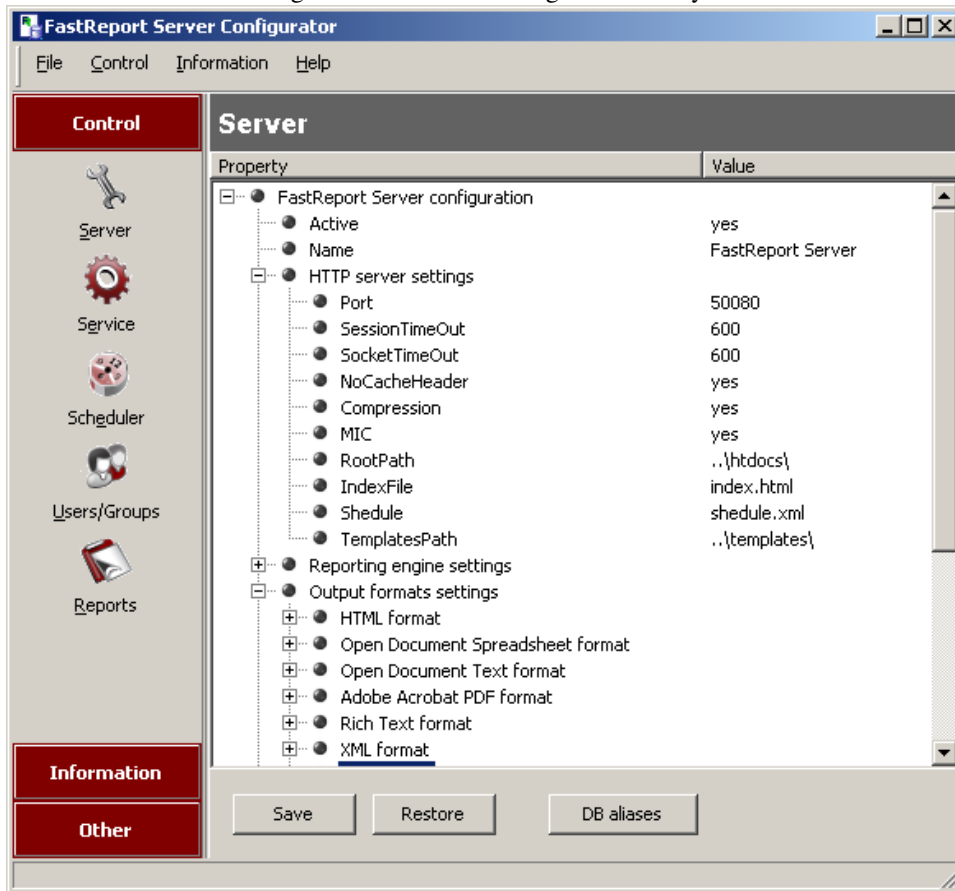
2 architecture

How does Fastreport server work ? It consists of 4 applications:

1. A server application, an invisible service application which sits in the background and creates the reports.
2. A server configuration tool. It is needed only to configure the server application: it can be used to set the location of the reports and databases, and set various options for the various output engines, set the port on which the service should listen and examine logs etc.
3. A report designer. The banded report designer is the same designer as used in the Delphi Fastreport component set.
4. A client report viewing application. It will connect to the service application and display a list of available reports. It allows to select and preview one of the available reports.

In fact, the client application is not necessary for use: any web browser can be used to connect to the service application and request a report: the service is actually a small

Figure 1: The server configuration utility



webservice that handles web requests. All that must be done is enter the correct starting point in the address bar of the browser.

A typical setup would be a central server machine, configured to connect to the corporate database, with all needed report designs on the local harddisk. Applications that need a printout can simply point the browser to the report server, and the report can be either saved to disk or printed from there.

3 The server application

As said, the server application is actually a small webservice that sits in the background and waits for a request for a report to arrive. As soon as a request arrives, the service creates and sends the report to the client. It operates completely in the background.

To configure it, a small configuration application is provided, it can be seen in figure 1 on page 2: The 2 most important items to be configured are

1. The port on which the service should listen: by default this is port 80, so if port 80 is in use by for instance the webservice, then another port must be selected (note that the port may need to be made accessible by adjusting the firewall rules).
2. The location of the report files: the report server will look for reports in this location only.

The location can be any directory in the system. Folders below this location are presented as 'categories' of reports when the list of reports is presented to the user.

The demo installation has 2 sets of reports: the reports are the same in both sets, but the first set connects to an MS-Access database using ADO or ODBC, the second set connects directly to a firebird database, bypassing any ODBC or ADO mechanisms. Since the installation did not setup (or failed to set up) the ODBC connection for the MS-Access demo database, one of the first things to do was to point the server to the set of reports that connect to the firebird database server.

Many other things can be configured in the service. Fastreport can export reports in many different formats: HTML, PDF, RTF, XML, Jpeg, Bmp, Gif, Tiff, Text, CSV, Open Office Document formats. For each of these formats, various settings can be adjusted. For instance, embedding fonts in the generated PDF (off by default) is a good idea if exotic fonts will be used: Not all PC's may have the needed fonts to display your report.

After all settings have been set to ones' satisfaction, they can be saved, and the service application must be restarted for the new settings to take effect- this be done from the configuration application or from the windows services.

The server application can do more than just send reports to clients: it can also generate reports on scheduled moments, and the result can for instance be mailed to an e-mail account as an attachment (obviously, reports containing dialogs cannot be used for this).

The server also implements a security mechanism of users and groups, and can control access to the reports based on username, so not everybody can simply display all available reports.

A CGI application and ISAPI module also exist, they can be used to forward requests from a "real" webserver (e.g. the corporate intranet server) to the report server.

4 The report designer

Delphi programmers will recognize the report designer that comes with FastReport server: indeed, it is the same designer used for the FastReport reporting components.

It is a powerful banded report designer, supporting all features one would expect from a modern report designer: multiple levels of grouping, subreports, cross-tab reports, templates, with a visual query builder and a powerful scripting language and even support for dialogs in the report. 4 scripting languages are supported: Pascalscript, BasicScript, C++Script and JScript, with obvious references to the programming languages. Debugging reports and their scripts is possible !

The use of this designer (shown in figure 2 on page 4) is straightforward and people should feel right at home; The manuals (not installed by the demo application, but downloadable separately) are very detailed and cover all aspects of the creation of reports: too much to cover here.

To display a report, the client can be used: it displays a list of reports (shown in figure 3 on page 4). One of the reports can be selected, and it will be previewed on screen: during preview, it can be printed, exported, mailed, or even - to a certain degree - modified (a unique feature of FastReport).

However, the client is not needed: it is possible to access the FastReport server directly using a web browser. The URL

`http://localhost:50080/`

will result in a list of reports being displayed, provided the server was configured to listen

Figure 2: Fastreport designer

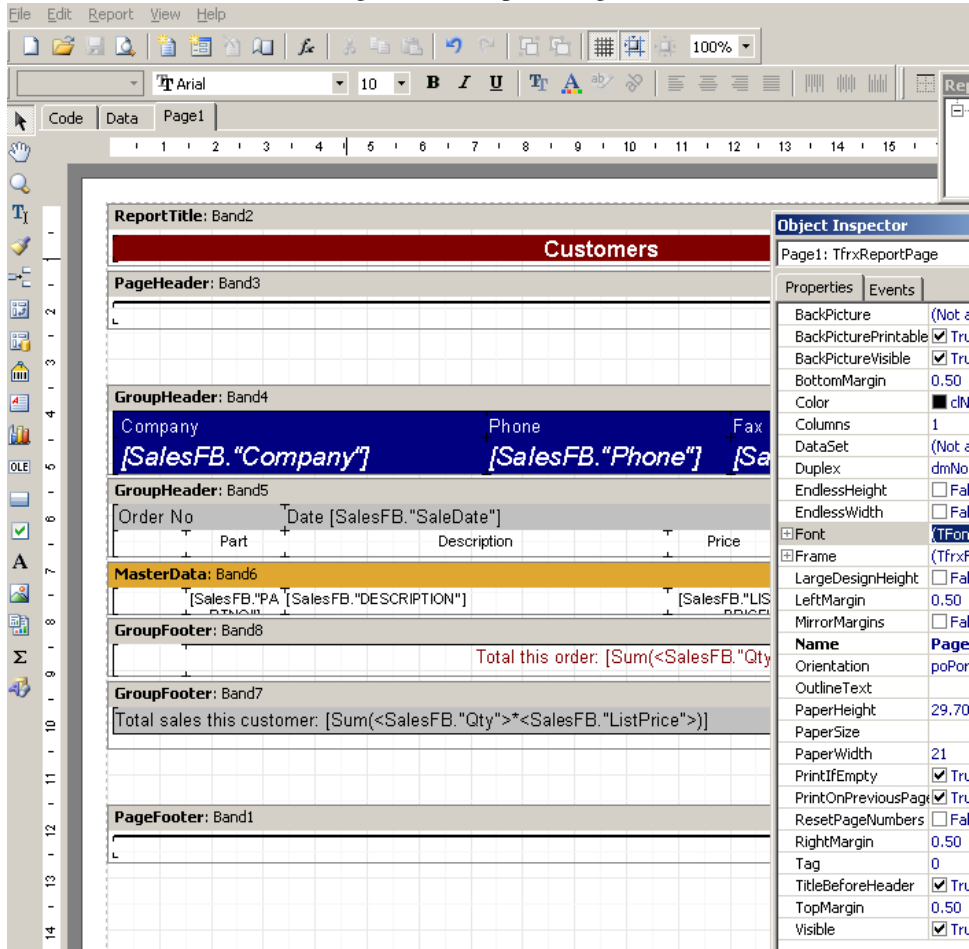


Figure 3: Fastreport client

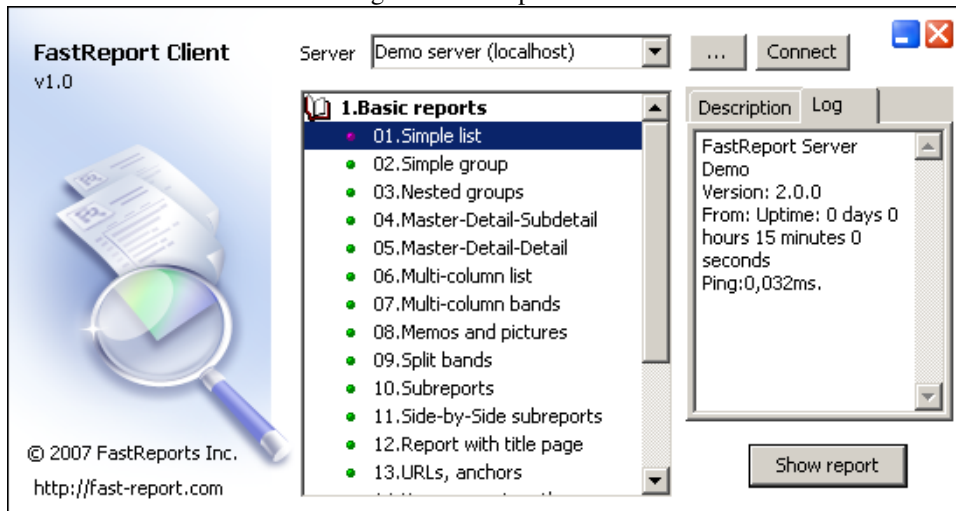
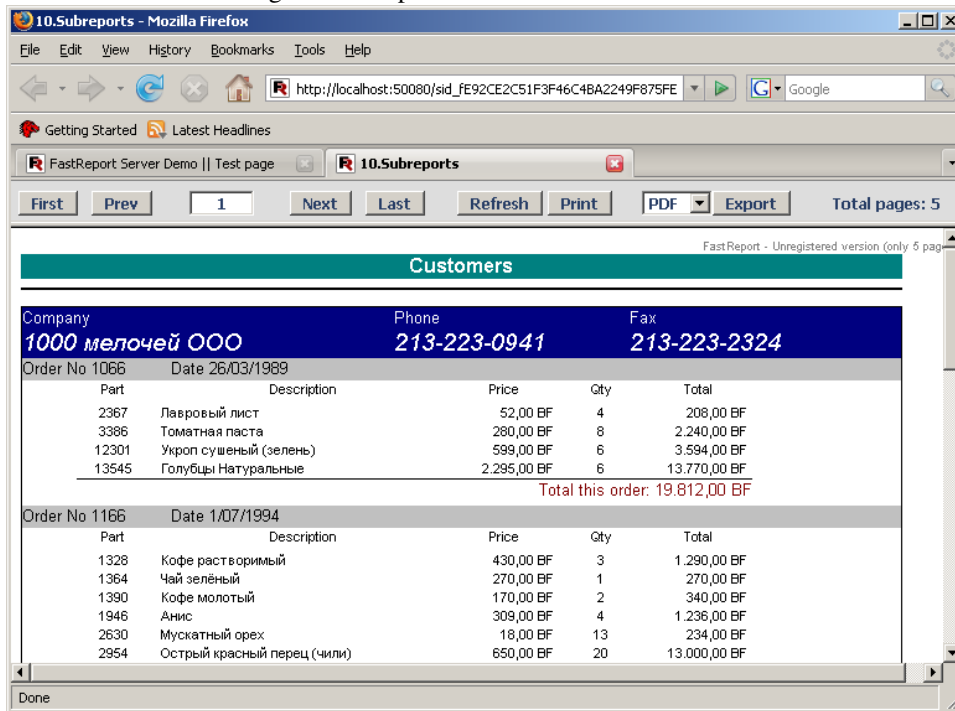


Figure 4: A report as shown in a webbrowser



on port 50080. Clicking one of the reports will display the report in the browser, by default in HTML, as can be seen in figure 4 on page 5. It is possible to browse through the various pages of the report, or a different format can be requested.

5 Conclusion

Users of the Fastreport component suite for Delphi do not need to be convinced that FastReport Server is a very good option for corporate reporting: the quality of the component suite is very good, and FastReport server seems to deliver the same quality for a more wide audience: With the exception of the ODBC DSN not being configured for the demo database and some minor glitches in the user interface of the configurator the demo installation performed flawlessly.

The only missing things - in the opinion of the author - are some kind of MDI interface for the designer: users with a lot of reports would surely appreciate the ability to view 2 reports in design side by side and being able to drag and drop elements from one report to another. The other missing thing is a Linux version of (at least) the server: if a company is running a linux-only shop server-wise, they are left in the cold by this otherwise very powerful reporting solution. Maybe something for version 3.0 ?