
МАТЕМАТИЧЕСКИЕ И ИНСТРУМЕНТАЛЬНЫЕ МЕТОДЫ ЭКОНОМИКИ

УДК 330.47

*А.П. Гаврилов, Е.Н. Барышева**

ОБЗОР ГЕНЕРАТОРОВ ОТЧЕТОВ, ИСПОЛЬЗУЕМЫХ ПРИ РАЗРАБОТКЕ ИС

В данной статье будут описаны средства, внедряемые в информационную систему, для возможности создания отчетов и экспорта данных в другие приложения.

Ключевые слова: информационная система, генератор отчетов, TPrinter, FastReport, экспорт данных, бэнд.

В настоящее время даже небольшие компании стремятся автоматизировать свою деятельность. Это может быть автоматизация процесса управления производством, финансами, закупками, персоналом и другими видами деятельности. Задачу автоматизации решает внедрение информационной системы, позволяющей управлять как всей организацией в целом, так и решать отдельные задачи. Функциональность разрабатываемой ИС обсуждается на этапе подготовки технического задания, когда заказчик формулирует требования, предъявляемые к будущей ИС. Одним из основных требований к системе является предоставление отчетности в требуемом формате (экспорт в другие программные средства или вывод на печать). Эта возможность может быть реализована при помощи различных компонентов, входящих в компиляторы Delphi и C++ Builder, которые будут рассмотрены в данной статье.

Опишем одни из самых простых компонентов, позволяющих генерировать отчет. Объект TPrinter – это прямой интерфейс между компьютером и печатающим устройством. Интерфейс включен во все версии Delphi и C++ Builder. Это объект, который позволяет пользователю сначала проконтролировать, что будет выведено в область печати (canvas) и затем отправить это на принтер. Для того, чтобы использовать объект и все его возможности в разрабатываемом приложении достаточно лишь подключить в раздел программы “uses” модуль “printers”. После этого объект TPrinter будет автоматически создаваться при запуске проекта. Пользователь может создавать запросы к нему с помощью функции “Printer”.

В прошлых версиях Delphi/C++ Builder эта функция работала следующим образом: модуль printers использовал глобальную переменную “Printer”, к которой можно

* © Гаврилов А.П., Барышева Е.Н., 2016

Гаврилов Александр Петрович (gavrilov.ap@ya.ru), Барышева Евгения Николаевна (barisheva_zh@hotmail.com), кафедра математики и бизнес-информатики, Самарский университет, 443086, Российская Федерация, г. Самара, Московское шоссе, 34.

было обращаться, включив ее в программу. Однако часто данный модуль включался в программу, но ни разу не применялся в ней. Использование вызова функции, которая возвращает объект TPrinter, уменьшает подобные потери в приложениях.

Объект TPrinter не требует никаких специальных условий для включения в окончателюю версию продукта. Объект использует память при включении в программный код, никак не влияя на само приложение.

Этот объект имеет набор ключевых свойств и методов для отправки информации непосредственно на принтер и является основным средством для выполнения этой задачи [1]. Главной особенностью объекта, которая отличает его от генератора отчетов, выступает полностью невидуальный интерфейс, поэтому функциональность объекта программируется вручную.

Ниже приведен простой пример использования объекта TPrinter для того, чтобы отправить текстовое сообщение на принтер:

```
Uses Printers;  
procedure TForm1.Button1Click(Sender: TObject);  
begin  
  With Printer do Begin  
    BeginDoc; // Начало печати  
    Canvas.Font.Size:=12; //размер шрифта  
    Canvas.Font.Color:=clRed; //цвет шрифта  
    Canvas.Font.Name:='Tahoma'; //название шрифта  
    Canvas.TextOut(100,100,'Пример'); //печатаем текст  
    EndDoc; // Конец печати  
  end;  
end.
```

При печати текста объект TPrinter не производит автоматического переноса строк и текста на новые страницы – эту возможность, также, необходимо программировать вручную. Поэтому для создания сложных отчетов, например, для предоставления в налоговые органы, предпочтительнее использовать компоненты с большим набором инструментов.

Существует другой вид компонентов, позволяющих получить отчет в печатной форме или экспортировать данные в различные популярные форматы файлов для их дальнейшей обработки. Рассмотрим подробнее генераторы отчетов – особый класс программ, реализующий возможность предоставления подробного отчета по данным в нужное время и в требуемой форме.

Особой популярностью среди множества подобных программ-генераторов, пользуется генератор отчетов «FastReports 5.0» – это современное программное решение для внедрения Business Intelligence в программное обеспечение любого уровня. Этот компонент был создан с учетом большинства потребностей разработчиков, которые хотят использовать готовые решения для отчетных средств. «FastReports 5.0» – бэнд-ориентированный генератор отчетов: большой выбор типов бэндов позволяет построить отчет любого уровня сложности [2]. Генератор обладает редактором отчетов, инструментарий которого включает огромное количество средств для визуального оформления, создания, настройки и изменения шаблонов отчетов.

FastReport позволяет осуществлять экспорт построенного отчета в различные форматы для последующего редактирования, архивирования, пересылки по электронной почте и др. На данный момент поддерживается экспорт в 13 форматов: PDF, ODS, ODT, Excel, XML, RTF, HTML, text, CSV, BMP, Jpeg, Tiff, Gif. Существует возможность отправки отчета по электронной почте средствами FastReport в любом из вышеперечисленных форматов. Эту возможность компонента реализуют, используя готовые компоненты, поэтому описывать ее мы не будем.

Рассмотрим пример практического применения генератора: проиллюстрируем подключение к базе данных (БД) и проектирование отчета.

Создание проекта производится при уже подключенной БД и готовой таблице. Для начала из вкладки «FastReports 5.0» установим на форму два компонента: frxDBDataset1 и frxReport1. FrxDBDataset1 – компонент для подключения к отчету источника данных. В нашем случае это таблица с данными. frxReport1 – это собственно сам отчет. При двойном клике на него откроется визуальный дизайнер отчетов. В FrxDBDataset1.DataSource указываем DataSet – источник данных.

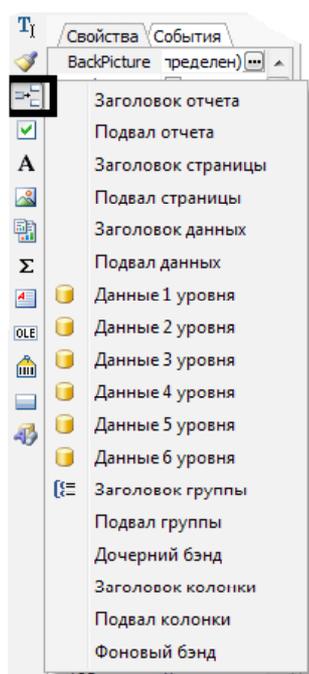


Рисунок 1. Меню бэнда

Перед тем, как начать формировать отчет разберем несколько понятий. Бэнд – это закрепленная область на листе. Внутри бэнда можно разместить дополнительные элементы отчета, чтобы сделать его нагляднее: логотип, название, реквизиты, табличку данных и т.п. Для управления бэндом используют меню (рис. 1).

Рассматривать меню бэндов излишне, достаточно упомянуть, что пункт «подвал страницы» можно использовать для вывода номера страницы. В бэнд «Данные X уровня» выводятся данные из DataSource (что далее будет рассмотрено подробно). «Memo» – поле, используемое для вывода текста. «Picture» – поле, позволяющее вставить в отчет логотип.

Для доступа к визуальному дизайнеру отчетов необходимо кликнуть по компоненту frxReport1. Откроется окно, представленное на рис. 2. Здесь уже добавлен бэнд «Заголовок отчета», в этой области размещают подпись с наименованием отчета и логотип компании. Поместим в бэнд объект «Picture» и в диалоговом окне укажем путь к логотипу.

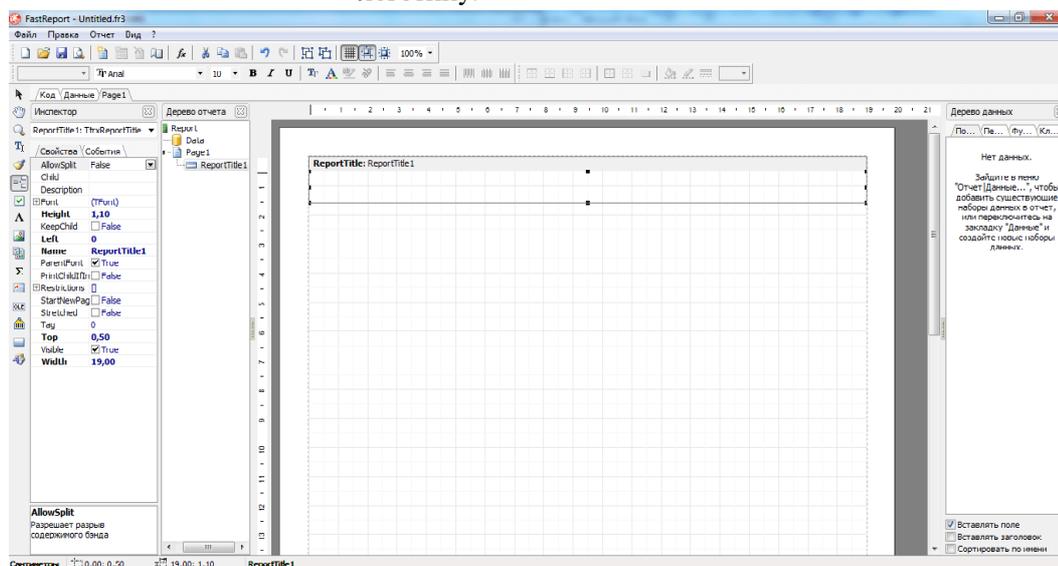


Рисунок 2. Дизайнер отчетов

Рядом с логотипом поместим наименование организации. После подготовки заголовка перейдем к формированию таблицы данных. Предварительно в меню «Отчет» – «Данные...» выберем (в открывшемся окне) источник данных – frxDBDataset1. Fast Report позволяет подключать сразу несколько источников данных к одному отчету. Добавляем в отчет бэнд данных, будет автоматически предложено выбрать источник данных для этого бэнда данных.

Для наглядности таблицы создадим для нее «шапку». Чтобы заголовки столбцов оставались только на первой странице размещаем их в бэнде заголовка отчета. Подписи добавляем в компонент «Мето», ширину компонента приравняем к ширине столбца таблицы.

Затем, во вкладках выберем первую вкладку «Поля БД». Запрос SQL, который необходимо использовать сразу вставим в компонент TQuery. Во вкладке представлен список полей. Далее переносим нужное поле в бэнд данных и выравниваем его по ширине колонки в заголовке отчета [3].

Теперь, чтобы вывести отчет закрываем визуальный дизайнер отчетов Fast Report. Добавляем на форму кнопку «Предварительный просмотр». Ниже приводится код события клика по кнопке.

```
procedure TForm1.ButtonReportClick(Sender: TObject);
begin
    frxReport1.ShowReport;
end;
```

На рис. 3 представлен возможный пример отчета – счет-фактура.

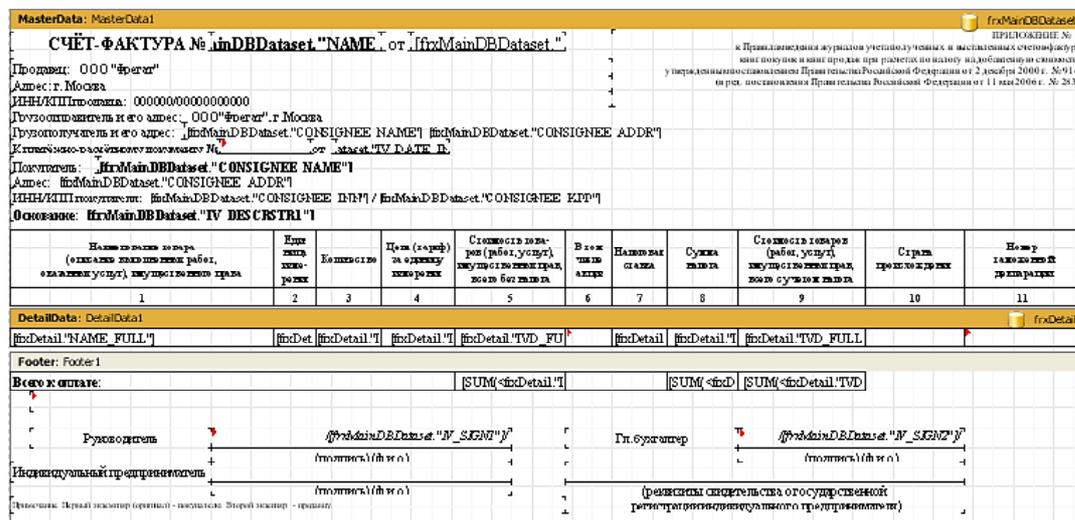


Рисунок 3. Счет-фактура

Конечно, для получения отчета такого вида необходимо использовать большое количество различных компонентов. Обширный набор компонентов дает этому генератору отчетов преимущество, перед остальными программами-генераторами, позволяя адаптировать отчет практически под любые требования заказчика.

Обратим внимание, что отчеты также можно хранить в файлах с расширением fr3 – это собственный формат отчетов в FastReport. Хранение отчета в файле удобно, когда необходимо вызывать отчет из разных мест ИС, это позволит эффективнее расходо-

вать память и избежать большого количества компонентов TfrxReport. Теперь, для каждого отчета необходимо создать отдельную процедуру, которая будет иметь примерно следующий вид:

```
Procedure PrintSomeReport(odsMain, odsDetail: TDataSet);
begin
frxMainData.DataSet:= odsMain;
frxDetailData.DataSet:= odsDetail;
frxRepInv.LoadFromFile("Имя файла отчета.fr3",true);
frxRepInv.ShowReport;
end.
```

После этого, из любой формы достаточно вызвать этот метод, который вы должны были заранее создать в модуле данных с компонентами отчета.

Были описаны два основных компонента, позволяющих отправлять отчеты на печать или экспортировать получившиеся отчеты в другие программы для дальнейшего редактирования [4]. Каждый разработчик решает, какой компонент соответствует поставленным задачам, поскольку не существует такого генератора отчетов, который бы смог сделать и соответствовать абсолютно любым требованиям заказчика. При этом придется потратить значительное время для того, чтобы понять какое из средств дает наибольшую функциональность и гибкость, соответствующую сложности и требованиям отчета.

Библиографический список

1. Барышева Е.Н., Сараев А.Л., Тюкавкин Н.М. Динамическая модель запаздывания эндогенных и экзогенных инвестиций в освоение капиталовложений в производственное предприятие // Вестник Самарского государственного университета. 2015. № 9.1 (131). С. 245–251.
2. Гаврилов А.П., Барышева Е.Н. Разработка и внедрение автоматизированной информационной системы для учета арендуемых помещений // Математика, экономика и управление. 2015. Т. 1. № 3. С. 44–47.
3. Дмитриев Д.С., Балахонов С.Ю. Средства электронного обучения как фактор решения основных проблем бизнес-образования // Математика, экономика и управление. 2015. Т. 1. С. 11–13.
4. Дунаев В.В., Барышева Е.Н. Разработка и внедрение автоматизированной информационной системы для составления расписания ВУЗа. // Математика, экономика и управление. 2015. Т. 1. № 3. С. 99–103.

References

1. Barysheva E.N., Saraev A.L., Tyukavkin N. M. A dynamic model of lag of endogenous and exogenous investment in the development of investment in the industrial enterprise // Vestnik of Samara state University. 2015. No. 9.1 (131). P. 245–251.
2. Gavrilov A.P., Barysheva E.N. Development and implementation of automated information system for accounting of leased premises // Mathematics, Economics and management. 2015. V. 1. No. 3. S. 44–47.
3. Dmitriev D.S., Balakhonov S.J. e-learning Tools as a factor in solving the basic problems of business education // Mathematics, Economics and management. 2015. Vol. 1. P. 11–13.
4. Dunaev V.V., Barysheva E.N. Development and implementation of automated information system for the scheduling of the University. // Mathematics, Economics and management. 2015. Volume 1, No. 3. P. 99–103.

*A.P. Gavrilov, E.N. Barysheva****OVERVIEW OF REPORT GENERATORS USED
IN THE DEVELOPMENT OF INFORMATION SYSTEMS**

This article describes the tools implemented in the information system, to create reports and export data to other applications.

Key words: information system, report generator, TPrinter, FastReport, export date, band.

* *Gavrilov Aleksandr Petrovich* (gavrilov.ap@ya.ru), *Barysheva Evgeniya Nikolaevna* (barisheva_zh@hotmail.com), Department of Mathematics and Business Informatics, Samara University, 34, Moskovskoye shosse, Samara, 443086, Russian Federation.