

СОЗДАНИЕ МНОГООКОННЫХ ПРИЛОЖЕНИЙ SDI СТИЛЯ В СРЕДЕ ПРОГРАММИРОВАНИЯ DELPHI

CREATING MULTI-WINDOW SDI STYLE APPLICATIONS IN THE DELPHI

D. Zdor
E. Savelyeva

Summary. Object. Reveal the meaningful elements of the design technology for a multi-window SDI style application in the Delphi programming environment, illustrate the main stages and technological methods with a specific example.

Methods. Analysis and synthesis of special literature on the research problem, generalization and systematization, modeling.

Findings. The windowed graphical interface is predominant. When developing a windowed application in Delphi, the main building block of an application window is the form. Multi-window applications can be MDI (Multi Document Interface) style or SDI (Single Document Interface) style. The article reveals the content elements of the design technology for a multi-window SDI style application. The main elements of the described technology include: 1) designing the start window interface; 2) adding a new form (several forms) to the project, designing their interface, taking into account the relationship with other forms of the project; 3) implementation of the interconnection of modules corresponding to the interconnected forms of the project; 4) writing event handlers to implement window functions, including software implementation of the transition between windows of a multi-window application.

Conclusions. The article describes the technology for designing a multi-window SDI-style application in the Delphi programming environment. The technology is described using an example of a specific application with an illustration of fragments of program code. The content elements of the presented technology can serve as a practical basis for building a multi-window SDI-style application in the Delphi programming environment. The results obtained can be used in the further development of issues of creating multi-window applications.

Keywords: multi-window application; the form; form style; form interface.

Здор Дмитрий Валерьевич

*К.п.н., доцент, ФГБОУ ВО «Приморская
государственная сельскохозяйственная академия»
dmitriy.dv@inbox.ru*

Савельева Екатерина Владимировна

*К.т.н., доцент, ФГБОУ ВО «Приморская
государственная сельскохозяйственная академия»
savva.6969@mail.ru*

Аннотация. Цель. Выявить содержательные элементы технологии проектирования многооконного приложения SDI стиля в среде программирования Delphi, проиллюстрировать на конкретном примере основные этапы и технологические приемы.

Методы. Анализ и синтез специальной литературы по проблеме исследования, обобщение и систематизация, моделирование.

Результаты. Оконный графический интерфейс является преобладающим. При разработке оконного приложения в Delphi основным строительным блоком окна приложения является форма. Многооконные приложения могут иметь стиль MDI (Multi Document Interface) или стиль SDI (Single Document Interface). В статье выявлены содержательные элементы технологии проектирования многооконного приложения SDI стиля. К основным элементам описанной технологии относятся: 1) проектирование интерфейса стартового окна; 2) добавление к проекту новой формы (нескольких форм), проектирование их интерфейса с учетом взаимосвязи с другими формами проекта; 3) осуществление взаимосвязи модулей, соответствующих взаимосвязанным по смыслу формам проекта; 4) написание обработчиков событий для реализации функций окон, в том числе, осуществление программной реализации перехода между окнами многооконного приложения.

Выводы. В статье представлено описание технологии проектирования многооконного приложения SDI стиля в среде программирования Delphi. Технология описана на примере конкретного приложения с иллюстрацией фрагментов программного кода. Содержательные элементы представленной технологии могут служить практической основой для построения многооконного приложения SDI стиля в среде программирования Delphi. Полученные результаты могут быть использованы в дальнейшей разработке вопросов создания многооконных приложений.

Ключевые слова: многооконное приложение; форма; стиль формы; интерфейс формы.

Введение

В настоящее время оконный графический интерфейс является преобладающим видом интерфейса пользователя в современных операционных системах. В связи с этим, в процессе разработки прикладного программного обеспечения для графических операционных систем одной из актуальных задач является создание либо использование графических объектов [7, с. 40]. При этом большинство программ имеет многооконную структуру. В рамках данной статьи рассматривается проблема создания многооконных приложений в среде программирования Delphi.

Delphi является системой программирования очень высокого уровня. Она берет на себя значительную часть работы по управлению компьютером, что делает возможным в простых случаях обходиться без особых знаний о деталях ее работы [12, с. 33].

Различным аспектам программирования в среде Delphi посвящены многие работы.

Общетеоретические аспекты объектно-ориентированного программирования в среде Delphi и основные конструкции языка Object Pascal (Delphi) рассмотрены в работах С. Тейксейра [1], А.Я. Архангельского [2], готовые алгоритмы решения стандартных задач обработки информации приведены в работе Р. Стивине [13]. Обзор компонентов разработки приложений в среде Delphi проведен в работах А.Я. Архангельского [2], В.Э. Гофмана [3], П.Г. Дарахвелидзе [5], В.М. Пестрикова [11], при этом вопросу создания новых компонентов уделяет внимание в своей работе Р. Конопка [8].

Форма как основной компонент создания оконного приложения в среде Delphi рассмотрен получил специальное рассмотрение в работах В.Э. Гофмана [3], П.Г. Дарахвелидзе [5]. Авторами проведена характеристика форм, вопросы создания и уничтожения форм, визуализация форм, организация взаимодействия форм. Рассмотрение данных вопросов носит общий характер, выполнен без конкретизации применения технологических приемов работы с формами при создании многооконных приложений.

Технология создания оконного приложения рассмотрена в работах Н.Б. Культина [9], В.М. Пестрикова [11], А.Г. Федорова [14]. При этом в данных работах акцент авторами сделан на графические возможности Delphi при создании приложений, работу с мультимедиа. А.А. Ремнев [12] приводит примеры готовых приложений в среде Delphi, эти примеры многообразны в контексте демонстрации технологических приемов

создания приложений. Однако вопрос создания многооконных приложений остался без внимания.

Принципиальное значение для нас имеет вопрос создания дочерних окон в приложениях Delphi, рассмотренный в работе М.Е. Фленова [15]. Ученый анализирует методы работы с окнами, модальные и немодальные окна, многодокументные MDI-окна, приведены отдельные операторы и фрагменты программных кодов по управлению окнами. При этом вопрос о создании многооконных приложений SDI стиля не получил детального рассмотрения в данной работе.

Анализ литературы показал, что вопрос проектирования многооконных приложений не получил достаточного внимания авторов. В большинстве работ не исследованы вопросы технологии создания многооконного приложения с использованием форм.

Полученные результаты послужили основой для рассмотрения вопроса о поиске содержательных элементов технологии проектирования многооконного приложения в среде Delphi.

При разработке оконного приложения в Delphi форма является основным строительным блоком окна приложения. Любая программа имеет как минимум одну связанную с ней форму, которая называется главной, — эта форма появляется на экране в момент старта программы. Однако программа может иметь сколько угодно форм, каждая из которых решает какую-то локальную задачу и появляется на экране по мере необходимости [2, с. 439].

Многооконные приложения могут иметь стиль MDI (Multi Document Interface) или стиль SDI (Single Document Interface). В стиле MDI создается главное окно, которое при этом является родительским, а внутри главного окна создаются дочерние окна. Дочерние окна не могут выходить за пределы родительского окна, все элементы управления располагаются в родительском окне, имеется механизм обеспечения взаимодействия родительского окна с дочерними. Данный стиль многооконных приложений не нашел большого распространения в практике программирования, хотя можно привести примеры достаточно популярных приложений Windows, выполненных в стиле MDI.

В большинстве случаев многооконные приложения составляются как SDI. В этом случае все окна работают независимо друг от друга, отсутствуют ограничения на их размеры и расположение, любое из окон может быть объявлено формально главным в рамках приложения, каждое из окон имеет, как правило, свои интерфейсные элементы управления.

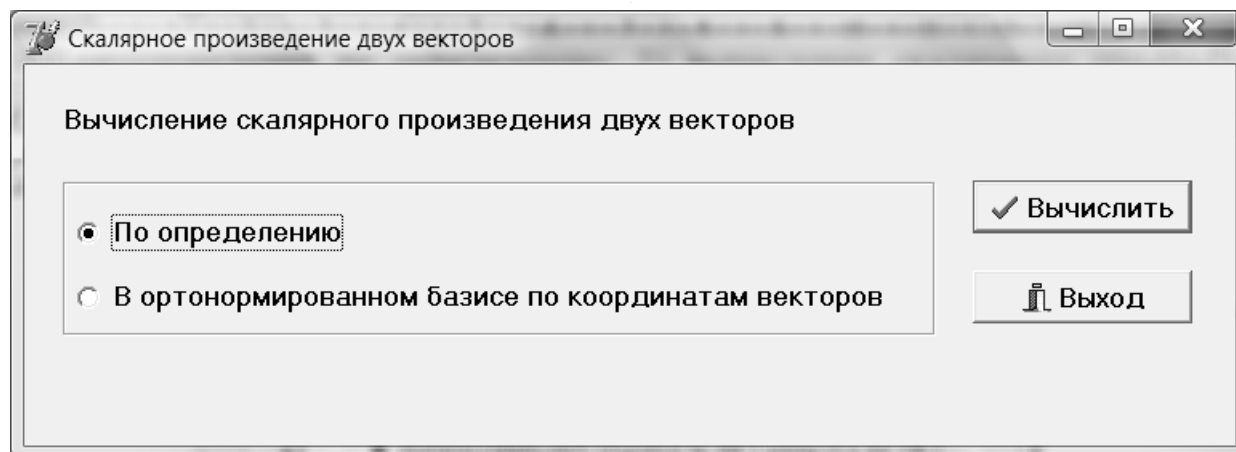


Рис. 1. Примерный вид главного окна приложения

Стандартные однодокументные (SDI) формы подразделяются в свою очередь на немодальные и модальные. Немодальная форма реализует стандартное окно Windows, которое имеет заголовок, стандартный набор оконных кнопок, допускает наличие главного меню, и может в любое время терять фокус, то есть становится пассивным. Модальные формы реализуют в основном диалоговые окна Windows, которые при активизации запрещают переключение фокуса на другие формы приложения (но не других приложений), пока оно не будет закрыто. Модальные окна, как правило, не имеют полного набора оконных кнопок, главного меню и т.п.

Зафиксированный выше результат позволил сфокусироваться на основном направлении нашего исследования — технологии проектирования многооконного приложения SDI стиля.

Методы и методология

Методологической базой служила научная литература по проблеме исследования. Теоретической базой служили пособия и рекомендации по программированию в среде программирования Delphi.

Методы исследования: анализ и синтез специальной литературы по проблеме исследования, обобщение и систематизация, моделирование, тестирование программного обеспечения.

Обсуждение

По умолчанию каждый новый проект Delphi состоит только из одной формы. Чтобы в проект добавить новую форму, следует обратиться к хранилищу объектов, в котором выбрать форму необходимой разновидности.

Менеджер проекта автоматически подключает новую форму, вносит соответствующие изменения в файл проекта. Первая из подключенных к проекту форм является главной. Главная форма служит основой прорисовки стартового окна приложения. Следует заметить, что в приложениях SDI стиля любая из добавленных к проекту форм может являться главной. Таким образом, программист может назначить любую форму в качестве главной, обратившись в окно настройки параметров проекта.

Заметим, что автоматически при запуске приложения на экране появляется только главная форма. Чтобы отобразить на экране окно какой-либо другой форме необходимо в программном коде предусмотреть обращение к методам Show или ShowModal. Метод Show вызывает окно соответствующей формы в качестве немодального, а ShowModal, соответственно в качестве модального. Метод Hide скрывает окно. Метод Close закрывает окно. Рассмотренные методы являются основой технологии проектирования многооконного приложения SDI стиля.

Рассмотрим содержательно элементы технологии проектирования многооконного приложения на конкретном примере.

Приложение «Скалярное произведение двух векторов» будет состоять из главной формы (рис. 1), в которой пользователю будет предложен выбор способа вычисления скалярного произведения: 1) вычисление скалярного произведения по определению; 2) вычисление скалярного произведения в ортонормированном базисе по координатам векторов. В зависимости от выбора пользователя должно открываться вспомогательное окно, в котором осуществляется решение данной задачи выбранным пользователем способом.

Рис. 2. Примерный вид вспомогательного окна (вычисление скалярного произведения по определению)

Рис. 3. Примерный вид вспомогательного окна (вычисление скалярного произведения по координатам векторов)

Работа начинается с создания интерфейса окна. Свойству Name главной формы зададим значение MainForm. Добавим в проект две новые формы, задав свойство Name для них: Determination (для вспомогательной формы вычисления скалярного произведения по определению), и Coordinates (для вычисления скалярного произведения по координатам векторов). Установим в инспекторе объектов свойство FormStyle

для всех форм в fsNormal поскольку разрабатываемое многооконное приложение будет иметь SDI стиль. Теперь необходимо спроектировать интерфейс форм Determination (рис. 2) и Coordinates (рис. 3), используя компоненты Label, Edit, BitBtn, Panel.

Осуществим связь модуля, соответствующего главной форме приложения, с двумя другими модулями, ко-

торые, в свою очередь, свяжем с модулем, соответствующим главной форме. Для этого используется команда главного меню *File — Use Unit...*

Напишем обработчики события OnClick для командных кнопок формы MainForm.

Кнопка *Вычислить*

```
procedure TMainForm.BitBtn1Click(Sender: TObject);
begin
if RadioGroup1.ItemIndex=0 then
begin
Determination.Show; // отобразить форму
MainForm.Hide; // скрыть форму
end
else begin
Coordinates.Show; // отобразить форму
MainForm.Hide; // скрыть форму
end;
end;
```

Кнопка *Выход*

```
procedure TMainForm.BitBtn2Click(Sender: TObject);
begin
Mainform.Close; // закрыть форму
end;
```

Напишем обработчики события OnClick для командных кнопок формы Determination.

Кнопка *Вычислить*

```
procedure TDetermination.BitBtn3Click(Sender:
TObject);
var s: real;
begin
s:=StrToFloatDef(Edit1.Text,0)*StrToFloatDef(Edit2.
Text,0)*
cos(StrToFloatDef(Edit3.Text,0)*pi/180);
Panel1.Caption:='Скалярное произведение = '+
FloatToStrF(s, ffFixed,10,2);
end;
```

Кнопка *Очистить*

```
procedure TDetermination.BitBtn4Click(Sender:
TObject);
begin
Edit1.Text:='';
Edit2.Text:='';
Edit3.Text:='';
Panel1.Caption:='Скалярное произведение = ';
end;
```

Кнопка *Вернуться на главную*

```
procedure TDetermination.BitBtn1Click(Sender:
TObject);
begin
Mainform.Show; // показать главную форму
Determination.Hide; // скрыть вспомогательную форму
end;
```

Кнопка *Выход*

```
procedure TDetermination.BitBtn2Click(Sender:
TObject);
begin
Application.Terminate;
end;
```

Результаты

Обобщение на теоретическом уровне приведенной технологии проектирования многооконного приложения, представленной на конкретном примере, позволяет сформулировать содержательные элементы технологии проектирования многооконного приложения SDI стиля в среде программирования Delphi. К основным элементам описанной технологии относятся: 1) проектирование интерфейса стартового окна; 2) добавление к проекту новой формы (нескольких форм), проектирование их интерфейса с учетом взаимосвязи с другими формами проекта; 3) осуществление взаимосвязи модулей, соответствующих взаимосвязанным по смыслу формам проекта; 4) написание обработчиков событий для реализации функций окон, в том числе, осуществление программной реализации перехода между окнами многооконного приложения.

Выводы

Таким образом, на примере приложения была реализована технология проектирования многооконного приложения SDI стиля в среде программирования Delphi. На каждом этапе реализации технологии определено содержание элементов технологии.

Комментарии к программному коду поясняют назначение операторов, необходимость в использовании которых продиктована программной реализацией элементов рассматриваемой технологии.

Жанр работы не позволяет привести ряд других примеров и подробнее описать выполнение технологических операций. Вместе с тем, в работе достаточно полно представлено описание технологии проектирования многооконного приложения SDI стиля в среде программирования Delphi.

Содержательные элементы представленной технологии могут служить практической основой для построе-

ния многооконного приложения SDI стиля в среде программирования Delphi. Опора на описанную технологию позволяет разработку многооконных приложений пере-

вести в разряд шаблонных задач. Полученные результаты могут быть использованы в дальнейшей разработке вопросов создания многооконных приложений.

ЛИТЕРАТУРА

1. Borland Delphi 6: руководство разработчика: Пер. с англ. [Текст] / Стив Тейксейра, Ксавье Пачеко. — М.: Вильямс, 2002. — 1112 с.
2. Архангельский А.Я. Delphi 7 [Текст] / А.Я. Архангельский. — М.: Изд-во БИНОМ, 2003. — 1023 с.
3. Гофман В.Э. Delphi. Быстрый старт [Текст] / В.Э. Гофман, А.Д. Хомоненко. — СПб.: БХВ-Петербург, 2003. — 288 с.
4. Дантемманн Джефф. Программирование в среде Delphi: Пер. с англ. [Текст] / Джефф Дантемманн, Джим Мишел, Дон Тейлор. — Киев: НИПФ «ДиаСофт Лтд», 1995. — 608 с.
5. Дарахвелидзе П.Г. Delphi 2005 для Win32 [Текст] / П.Г. Дарахвелидзе, Е.П. Марков. — СПб.: БХВ-Петербург, 2005. — 1112 с.
6. Здор Д.В. Информационные технологии обработки текста: программная реализация средствами Delphi [Текст] / Д.В. Здор // Информационные технологии в жизни современного человека: Материалы международной научно-практической конференции. — Саратов: Изд-во ЦИМ «Академия бизнеса», 2013. — С. 44–49.
7. Здор Д.В. Технология создания и использования графики в приложениях Delphi [Текст] / Д.В. Здор // Научная мысль. — 2017. — № 4. — С. 40–44.
8. Конопка Р. Создание оригинальных компонент в среде Delphi: Пер. с англ. [Текст] / Р. Конопка. — Киев.: НИПФ «ДиаСофт Лтд», 1996. — 512 с.
9. Культин Н.Б. Основы программирования в Delphi XE [Текст] / Н.Б. Культин. — СПб.: БХВ-Петербург, 2011. — 416 с.
10. Мизрохи С.В. Turbo Pascal и объектно-ориентированное программирование [Текст] / С.В. Мизрохи. — М.: Финансы и статистика, 1992. — 193 с.
11. Пестриков В.М. Delphi на примерах [Текст] / В.М. Пестриков, А.Н. Маслобоев. — СПб.: БХВ-Петербург, 2005. — 496 с.
12. Ремнев А.А. Курс Delphi для начинающих. Полигон нестандартных задач [Текст] / А.А. Ремнев, С.В. Федотова. — М.: Солон-Пресс, 2015. — 360 с.
13. Стивене Р. Delphi. Готовые алгоритмы: Пер. с англ. [Текст] / Род Стивене — М.: ДМК Пресс; СПб.: Питер, 2004. — 384 с.
14. Федоров А.Г. Создание Windows-приложений в среде Delphi [Текст] / А.Г. Федоров. — М.: ТОО «Компьютер Пресс», 1999. — 347 с.
15. Фленов М.Е. Библия Delphi [Текст] / М.Е. Фленов. — 2-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2008. — 800 с.

© Здор Дмитрий Валерьевич (dmitriy.dv@inbox.ru), Савельева Екатерина Владимировна (savva.6969@mail.ru).

Журнал «Современная наука: актуальные проблемы теории и практики»