

## **Разработка автоматизированной системы по интеграции частично гомогенных баз данных**

*Винокуров Анатолий Станиславович*

*Приамурский государственный университет имени Шолом-Алейхема*

*Магистрант*

### **Аннотация**

В статье рассматривается разработка автоматизированной системы для решения задачи приведения таблиц двух частично гомогенных баз данных к общему знаменателю, интеграции таблиц внутри одной базы данных, а также интеграции двух частично гомогенных баз данных. Такая программа должна автоматизировать процесс расчета коэффициентов подобия с последующей интеграцией баз данных на основе этих оценок.

**Ключевые слова:** информационные технологии, автоматизированная система, интеграция баз данных, слияние баз данных, консолидация баз данных, однородные базы данных, частичная гомогенность, ИТ-проект, интегрированная среда разработки, IDE, Lazarus, Delphi.

## **Development of an automated system for the integration of partially homogeneous databases**

*Vinokurov Anatoly Stanislavovich*

*Sholom-Aleichem Priamursky State University*

*Undergraduate*

### **Abstract**

The article deals with the development of an automated system to solve the problem of bringing the tables of two partially homogeneous databases to a common denominator, the integration of tables within one database, as well as the integration of two partially homogeneous databases. Such a program should automate the process of calculating similarity coefficients and then integrate the databases based on these estimates.

**Keywords:** information technology, automated system, database integration, database merging, database consolidation, homogeneous databases, partial homogeneity, it project, integrated development environment, IDE, Lazarus, Delphi.

На основе предложенных в рамках магистерской диссертации «Исследование вопросов и разработка методов интеграции частично гомогенных баз данных» методов интеграции частично однородных баз данных, было принято решение разработать собственную автоматизированную систему для решения задачи приведения таблиц двух

частично гомогенных баз данных к общему знаменателю, интеграции таблиц внутри одной базы данных, а также интеграции двух частично гомогенных баз данных. Такая программа должна автоматизировать процесс расчета коэффициентов подобия с последующей интеграцией баз данных на основе этих оценок. Оценку коэффициентов подобия производит пользователь. Именно за ним остается право подтвердить либо опровергнуть результаты расчета программы.

Разработке приложений для интеграции баз данных посвятили свои работы многие исследователи. M.R.Genesereth, A.M.Keller, O.M.Duschka [1] описали разработку собственного программного обеспечения под коммерческим названием «Infomaster», выполняющего интеграцию баз данных. L.L.Yan, M.T.Otsu, L.Liu [2] предложили модель, выполняющую создание интегрированного доступа к разнородным источникам данных, а также разработали на основе данной модели медиаторную программную систему «Aurora», автоматизирующую процесс интеграции баз данных. C.Li, R.Yerneni, V.Vassalos, H.Garcia-molina, Y.Papakonstantinou, J.Ullman, M.Valiveti [3] разработали программную систему «T.S.I.M.M.I.S.» в виде медиатора, который, на основе пользовательских запросов, выстраивает таблицу, содержащую ссылки на атрибуты исходной базы данных. W.Klas, P.Fankhauser, P.Muth, T.C.Rakow, E.J.Neuhold [4] рассмотрели разработку собственной программной системы «V.O.D.A.K.», осуществляющей интеграцию нескольких баз данных с использованием открытых объектно-ориентированных систем. J.Lee, S.E.Madnick, M.D.Siegel [5] описали требования к разработке программного обеспечения по интеграции реляционных баз данных. Ph.Thiran, J.-L.Hainaut, S.Bodart, A.Deflorenne, J.M.Nick [6] провели разработку программного продукта под коммерческим названием «dbMain», предназначенного для обслуживания и разработки схем интеграции баз данных. P.Ziegler и другие [7–9] описали разработку и результаты тестирования программного обеспечения «S.I.R.U.P.», реализующего семантическую интеграцию гетерогенных данных. C.Reynaud и другие [10–12] провели исследование вопросов построения масштабируемых медиаторных систем. В результате исследования разработали собственную информационную систему «P.I.C.S.E.L.» для решения поставленных вопросов. A.Levy, A.Rajaraman, J.Ordille [13] разработали программную систему «Information Manifold», оптимизирующую запросы с разнородных источников данных на основе онтологии исходных баз данных. M.Boyd [14–15] описали разработку системы «AutoMed», решающую задачи интеграции данных для гетерогенных источников данных.

Разработка программы производится в свободнораспространяемой среде разработки Lazarus 1.8.2 [16] на языке программирования Object Pascal (рис. 1), а затем, после завершения разработки и тестирования нашего программного обеспечения, проект был перенесен в условно-бесплатную кроссплатформенную среду разработки IDE Embarcadero Delphi [17]. Данная среда разработки была выбрана из-за наличия пробной лицензии, поддержки

кроссплатформенной разработки, а также Delphi-подобного окружения (рис. 2), а перенос проекта с одной среды в другую, более мощную, позволил внести в проект дополнительные преимущества более мощной среды разработки.

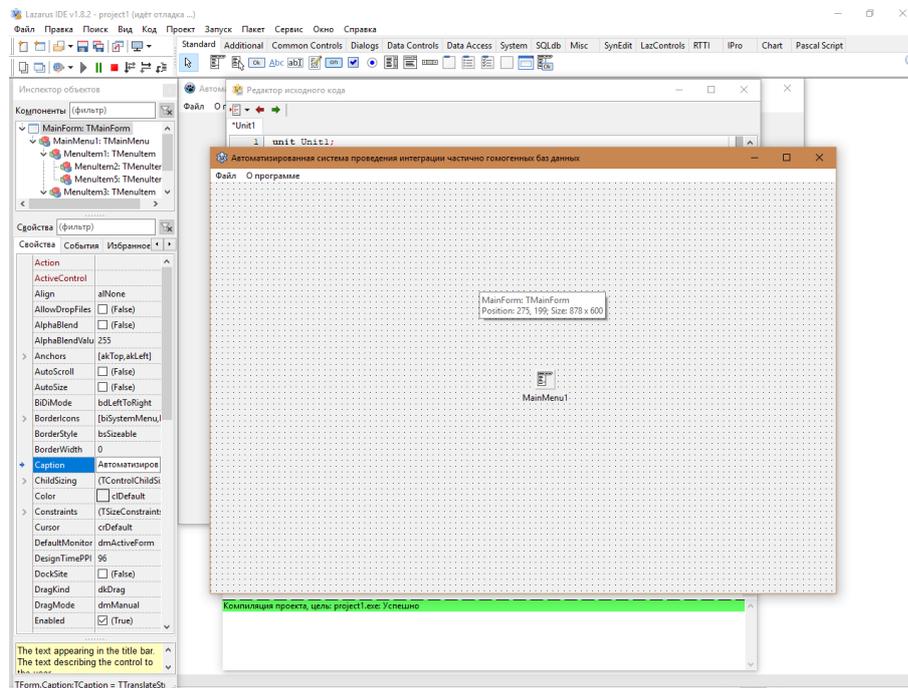


Рисунок 1 – Среда разработки Lazarus

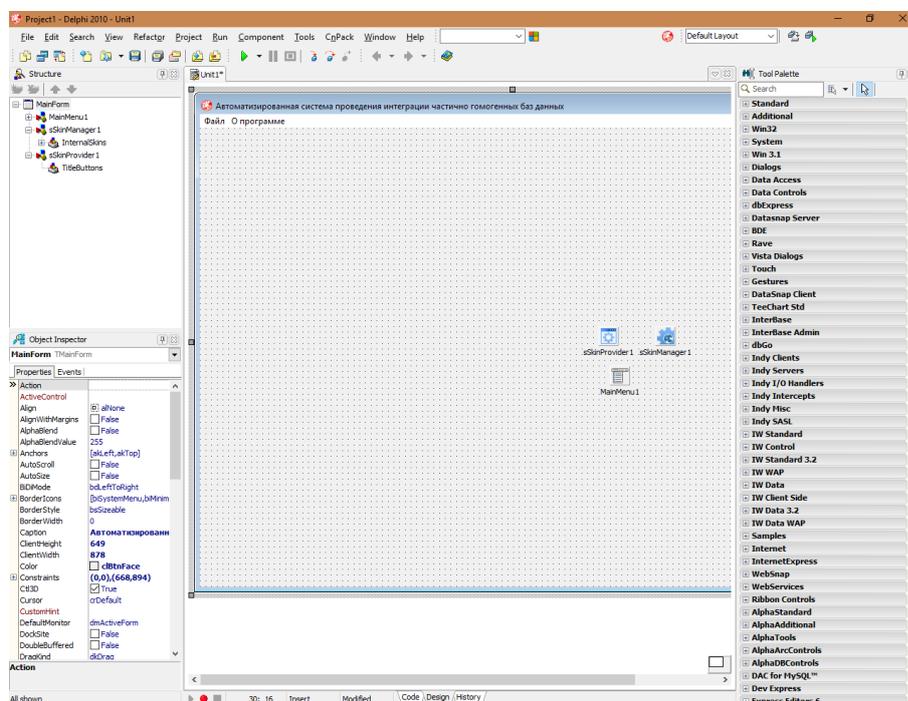


Рисунок 2 – Среда разработки Embarcadero Delphi

В проекте были использованы стандартные компоненты Delphi, а также пакет свободно распространяемых компонентов AlphaControls v13.16 [18],

благодаря которым появляется возможность добавить уникальный стиль оформления и визуализации пользовательскому интерфейсу программы.

Перейдем к рассмотрению проекта. Нами была создана главная форма программы под названием «MainForm». Главной форме программы в свойстве Formstyle было установлено значения fsMDIForm, а также задан минимальный и максимальные размеры формы – свойства MaxHeight, MaxWidth, MinHeight и MinWidth группы свойств Constraints. На главной форме располагается компонент «MainMenu», отвечающий за создание главного меню, а также компоненты «sSkinProvider» и «sSkinManager» пакета AlphaControls (рис. 3).

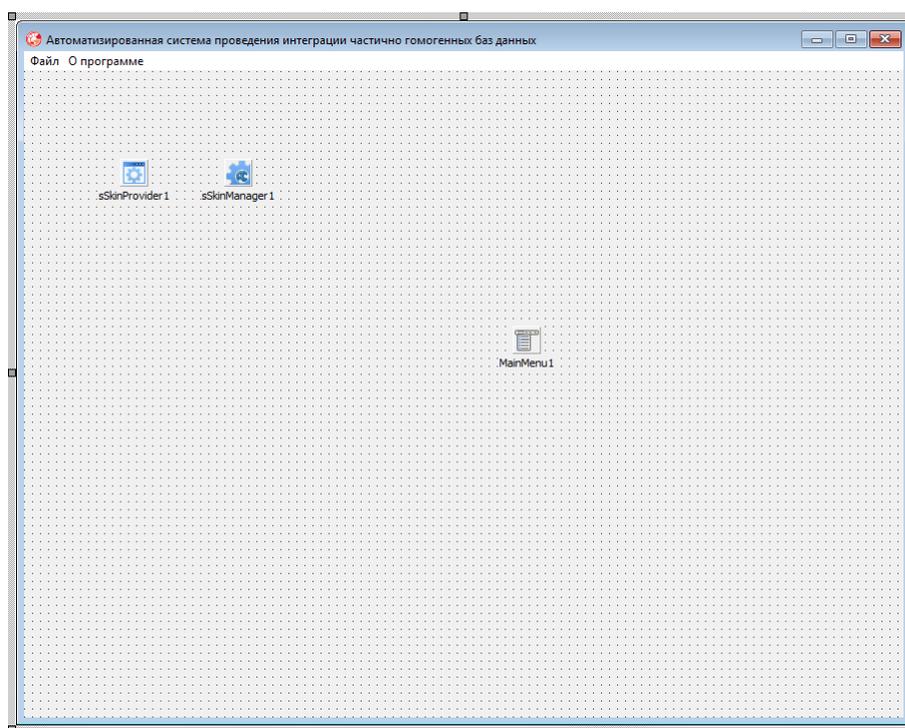


Рисунок 3 – Главная форма программы

Далее в проект были добавлены функциональные формы, которые будут отображаться внутри главной формы «MainForm», благодаря значению fsMDIChild свойства formstyle каждой из формы.

Структура программы представлена на рисунке 4.

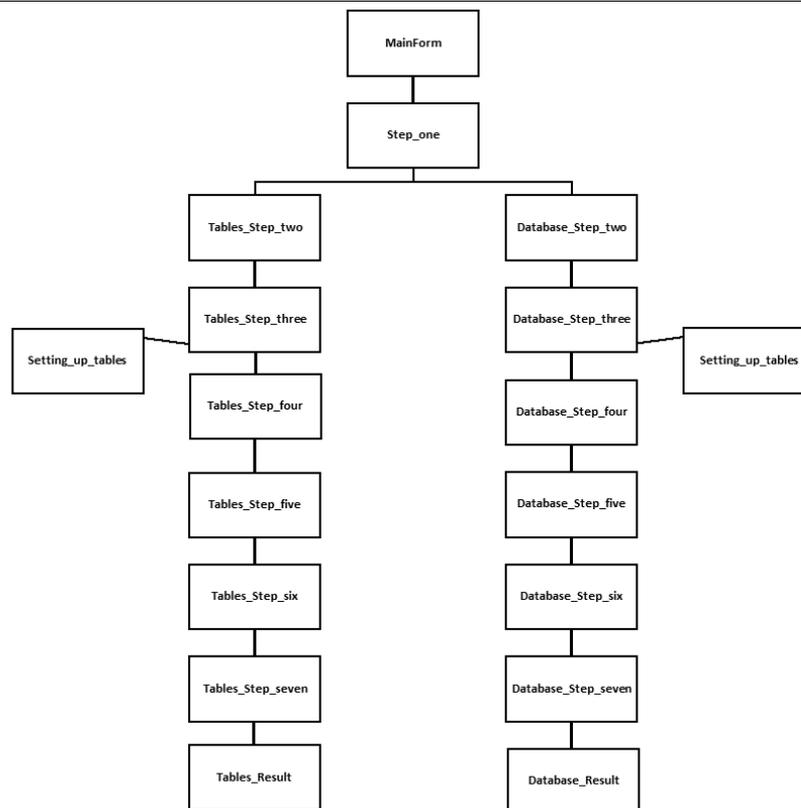


Рисунок 4 – Структура программы

На форме «Step\_one» расположены следующие компоненты sLabel, sRadioGroup, sButton (рис. 5).

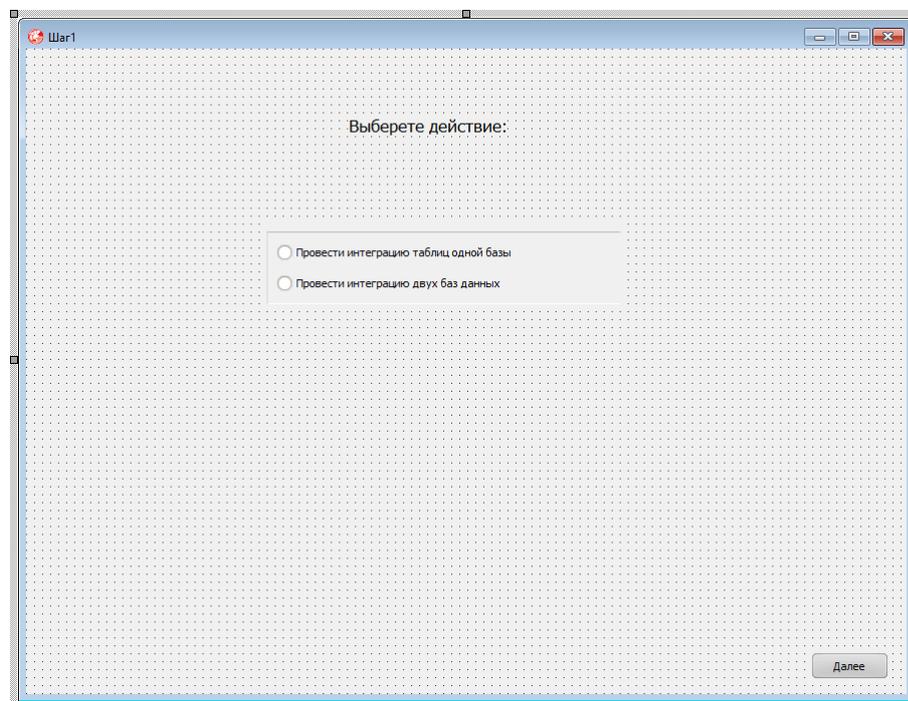


Рисунок 5 – Форма «Step\_one»

Компонент sRadioGroup предназначен для предоставления пользователю выбора варианта интеграции: при выборе «Провести

интеграцию таблиц одной базы» происходит скрытие формы «Step\_one» и создание формы «Tables\_Step\_two»; при выборе «Провести интеграцию двух баз данных» происходит скрытие формы «Step\_one» и создание формы «Database\_Step\_two».

Рассмотрим форму «Tables\_Step\_two», предназначенной для подключения к одной базе данных и последующей интеграции ее таблиц.

На форме «Tables\_Step\_two» расположены следующие компоненты sLabel, sEdit, sButton, sGroupBox, MySQLDatabase, sStatusBar, PopupMenu (рис. 6).

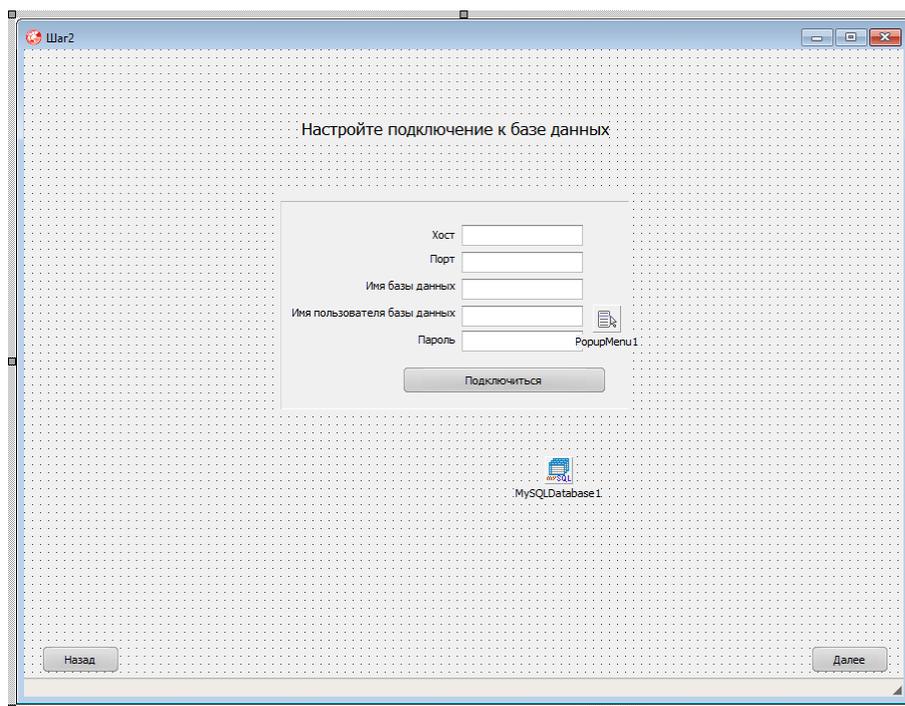


Рисунок 6 – Форма «Tables\_Step\_two»

Компоненты sEdit предназначены для ввода параметров подключения к базе. Для скрытия/отображения вводимого пользователем в поле пароля используется компонент PopupMenu. Кнопка «Подключить» (компонент sButton), производит подключение к базе данных на основе параметров подключения, введенных пользователем, а также осуществляет контроль ошибок подключения. Кнопка «Далее» осуществляет скрытие текущей формы и переход к следующей форме «Tables\_Step\_three». Кнопка «Назад» осуществляет закрытие текущей формы и освобождение выделенной под нее памяти.

На форме «Tables\_Step\_three» расположены следующие компоненты sLabel, sButton, sGroupBox, sDBGrid, MySQLQuery, DataSource (рис. 7).

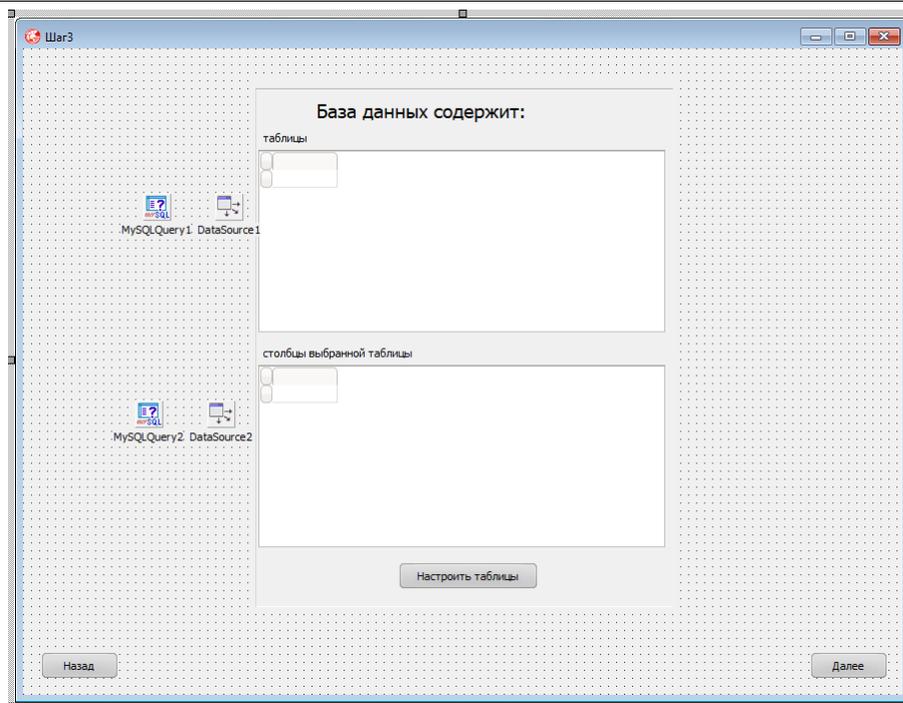


Рисунок 7 – форма «Tables\_Step\_three»

Компонент sDBGrid предназначен для отображения списка таблиц базы данных. При выборе таблицы в sDBGrid1, список, содержащий название столбцов выбранной таблицы, а также полный перечень параметров выводится в sDBGrid2.

Кнопка «Настроить таблицы», производит скрытие текущей формы и переход к специальной форме «Setting\_up\_tables».

Рассмотрим подробнее форму «Setting\_up\_tables».

На форме «Setting\_up\_tables» расположены следующие компоненты sLabel, sButton, sRadioGroup, sPageControl, sCheckBox, sEdit, sComboBox, sSpinEdit, MySQLTable, MySQLQuery, DataSource (рис. 8).

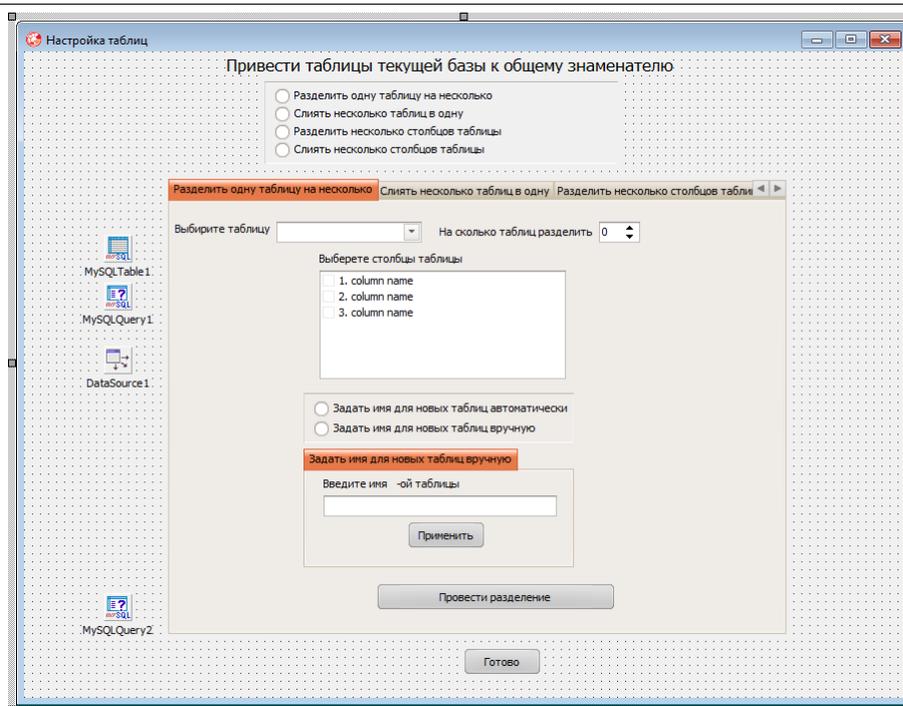


Рисунок 8 – Форма «Setting\_up\_tables» программы

Компонент `sRadioGroup1`, расположенный в верхней части формы предназначен для предоставления пользователю выбора дальнейшего действия. При выборе, пользователю отображается соответствующая вкладка компонента `sPageControl1` с расположенными на ней функциями.

Рассмотрим подробнее все возможности функционального выбора.

При выборе пункта «Разделить одну таблицу на несколько» `sRadioGroup1`, происходит отображение вкладки «Разделить одну таблицу на несколько» `sPageControl1`. На данной вкладке пользователь может выбрать таблицу в `sComboBox`, выбрать на какое количество таблиц дублировать данную таблицу, а также выбрать какие именно столбцы данной таблицы необходимо задействовать.

Далее пользователю необходимо выбрать нужный пункт `sRadioGroup2`. При выборе пункта «Задать имя для новых таблиц автоматически», имя для каждой новой таблицы будет задано программой и будет содержать префикс, а вкладка «Задать имя для новых таблиц вручную» `sPageControl2` будет скрыта. При выборе пункта «Задать имя для новых таблиц вручную», отображается вкладка «Задать имя для новых таблиц вручную» `sPageControl2` на которой пользователю самому необходимо ввести имя для каждой новой таблицы в `sEdit`, и при этом, каждый раз нажимать кнопку «Применить».

По окончании работы пользователю необходимо нажать на кнопку «Провести разделение», и новые таблицы будут добавлены в базу при помощи SQL запроса компонента `MySQLQuery` (рис. 9).

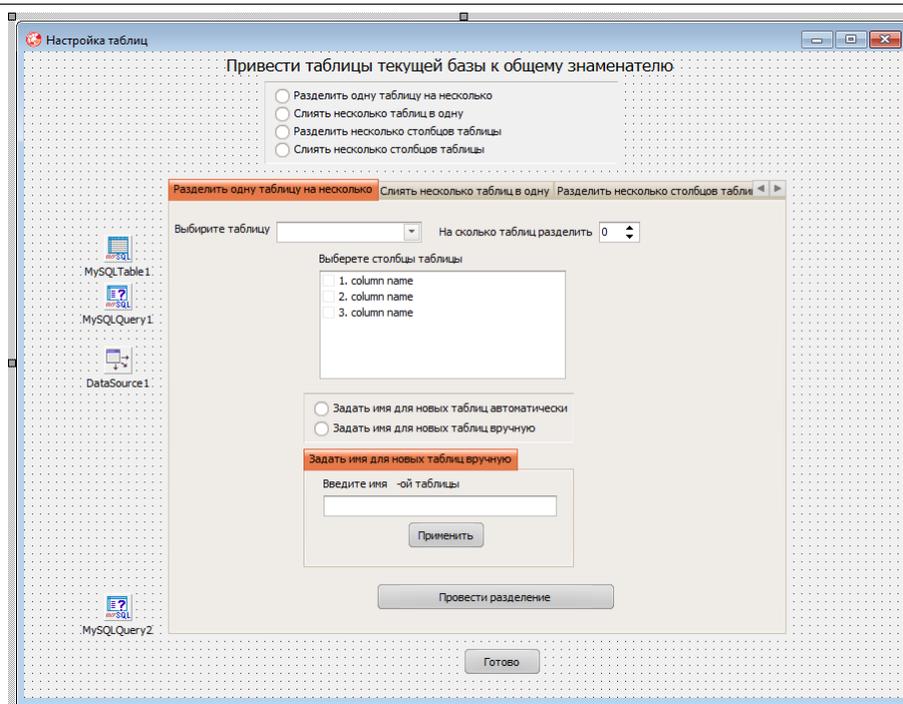


Рисунок 9 – Функция «Разделить одну таблицу на несколько» формы «Setting\_up\_tables» программы

При выборе пункта «Слить несколько таблиц в одну» sRadioGroup1, происходит отображение вкладки «Слить несколько таблиц в одну» sPageControl1. На данной вкладке пользователь может выбрать сразу несколько таблиц, а также столбцы для каждой выбранной таблицы в sCheckListBox.

Далее пользователю необходимо выбрать нужный пункт sRadioGroup3. При выборе пункта «Слить в существующую таблицу», откроется вкладка «Слить в существующую таблицу» sPageControl3, на которой пользователю будет предложено выбрать таблицу в sComboBox, в которую будет произведена перекачка информации при помощи SQL запроса компонента MySQLQuery (рис. 10).

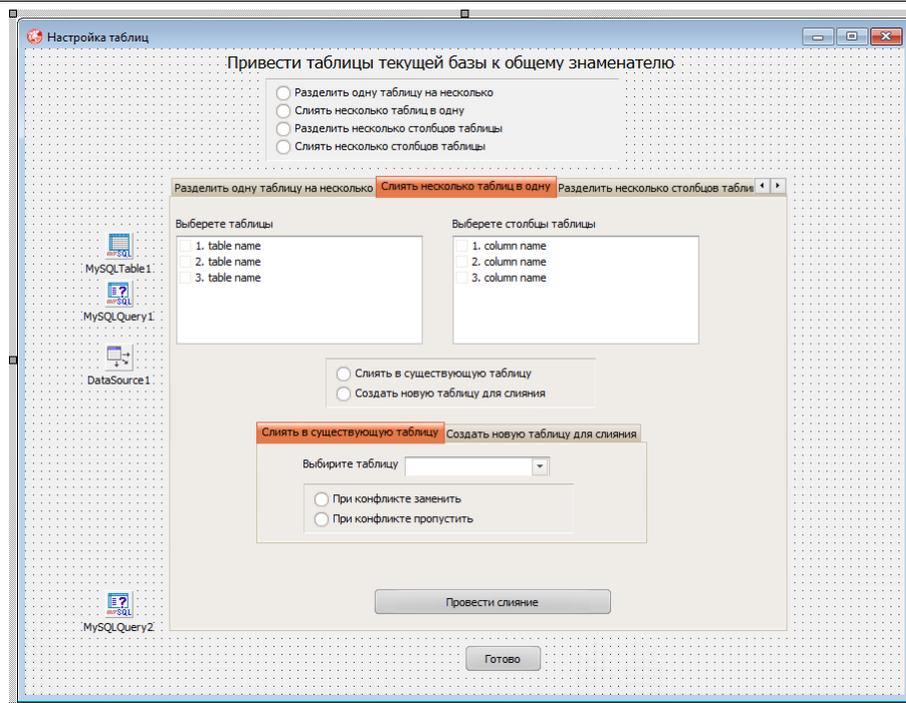


Рисунок 10 – Пункт «Слить в существующую таблицу» функции «Слить несколько таблиц в одну» формы «Setting\_up\_tables»

При выборе пункта «Создать новую таблицу для слияния», откроется вкладка «Создать новую таблицу для слияния» sPageControl3, на которой пользователю необходимо ввести имя для новой таблицы в sEdit, в которую будет произведена перекачка информации при помощи SQL запроса компонента MySQLQuery (рис. 11).

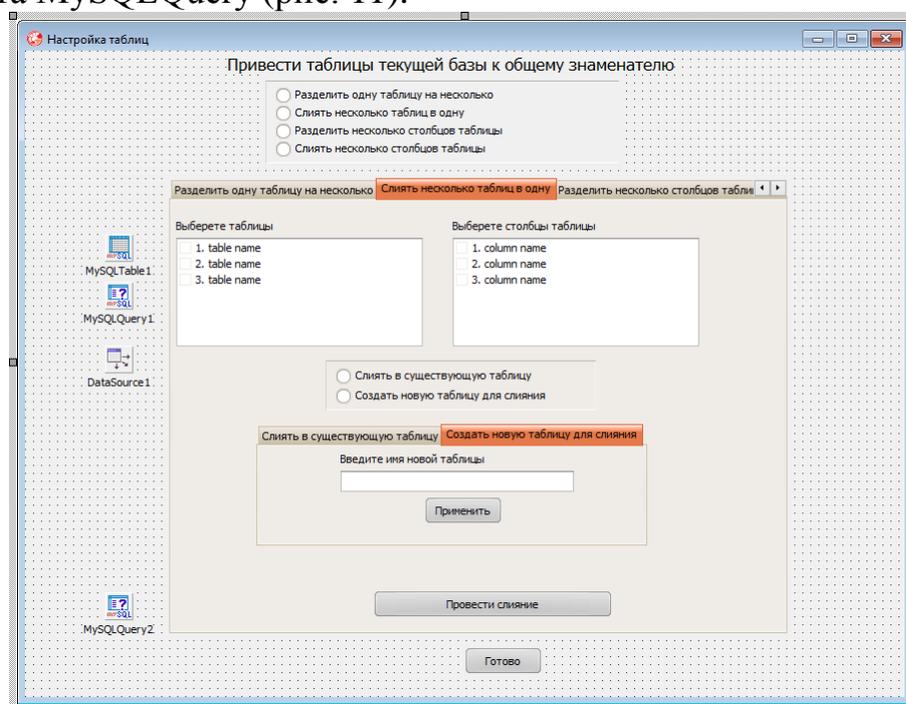


Рисунок 11 – Пункт «Создать новую таблицу для слияния» функции «Слить несколько таблиц в одну» формы «Setting\_up\_tables»

При выборе пункта «Разделить несколько столбцов таблицы» sRadioGroup1, происходит отображение вкладки «Разделить несколько столбцов таблицы» sPageControl1. На данной вкладке пользователю необходимо выбрать таблицу и ее столбец в соответствующих sComboBox, выбрать на какое количество столбцов разделить данный столбец таблицы, а также выбрать «тип разделителя» в sComboBox. Предполагается несколько стандартных видов разделителей: «Пробел», «Точка», «Точка с запятой», а также «Свой», позволяющий ввести символы непосредственно в само поле sComboBox. Благодаря выбранному типу разделителя, программа будет знать, как отделить информацию для каждого нового столбца из исходного. Данная функция обеспечивает лишь небольшую автоматизацию трудоемкого процесса отделения слов в предложении, хранящемся в столбце таблицы, и не всегда может гарантировать стопроцентный результат. Ответственность за корректное отделение таблиц целиком и полностью остается за пользователем (Рис. 12).

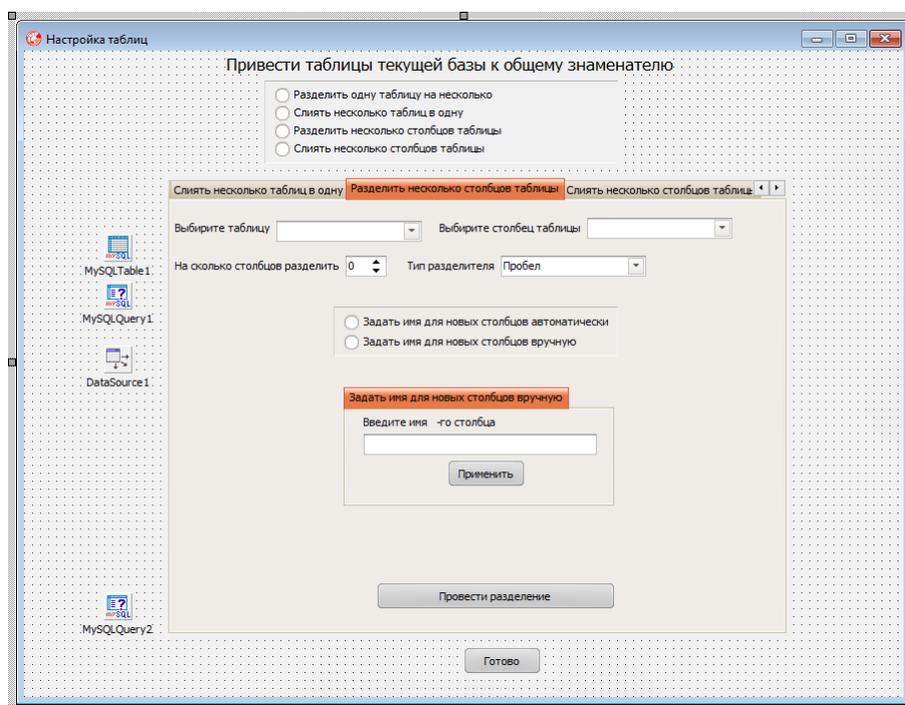


Рисунок 12 – Функция «Разделить несколько столбцов таблицы» формы «Setting\_up\_tables»

При выборе пункта «Слить несколько столбцов таблицы» sRadioGroup1, происходит отображение вкладки «Слить несколько столбцов таблицы» sPageControl1. На данной вкладке пользователь может выбрать таблицу в sComboBox, а также выбрать столбцы данной таблицы, которые необходимо слить, в sCheckBoxList.

Далее пользователю необходимо выбрать нужный пункт sRadioGroup4. При выборе пункта «Слить в существующий столбец», откроется вкладка «Слить в существующий столбец» sPageControl4, на которой пользователю будет предложено выбрать столбец таблицы в sComboBox. При выборе

пункта «Создать новый столбец для слияния», откроется вкладка «Создать новый столбец для слияния» sPageControl4, на которой пользователю необходимо ввести имя нового столбца в sEdit, после чего будет произведена перекачка информации при помощи SQL запроса компонента MySQLQuery (рис. 13 – 14).

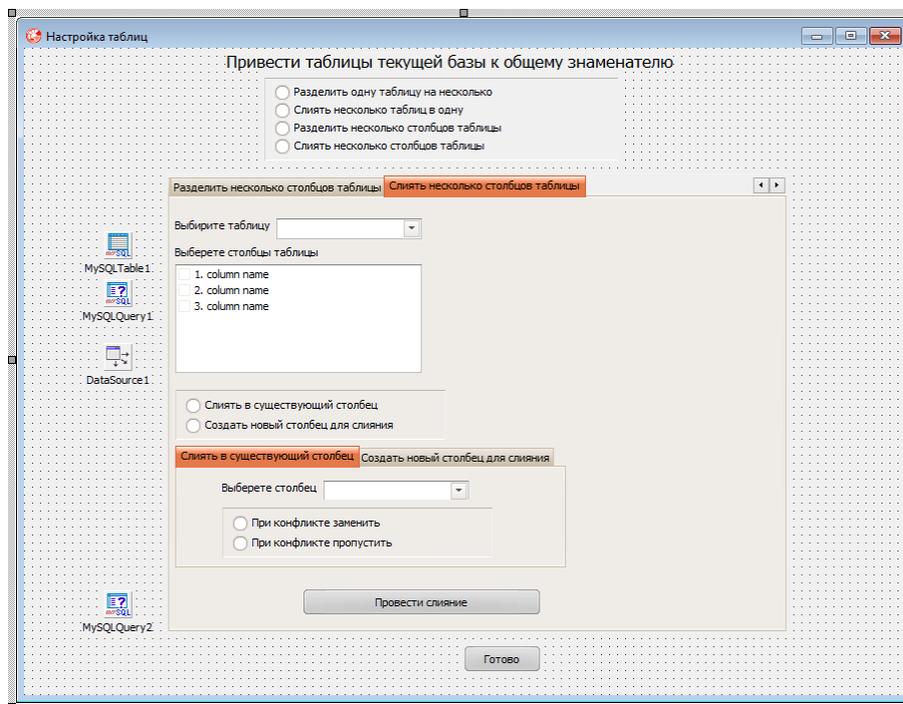


Рисунок 13 – Пункт «Слить в существующий столбец» функции «Слить несколько столбцов таблицы» формы «Setting\_up\_tables»

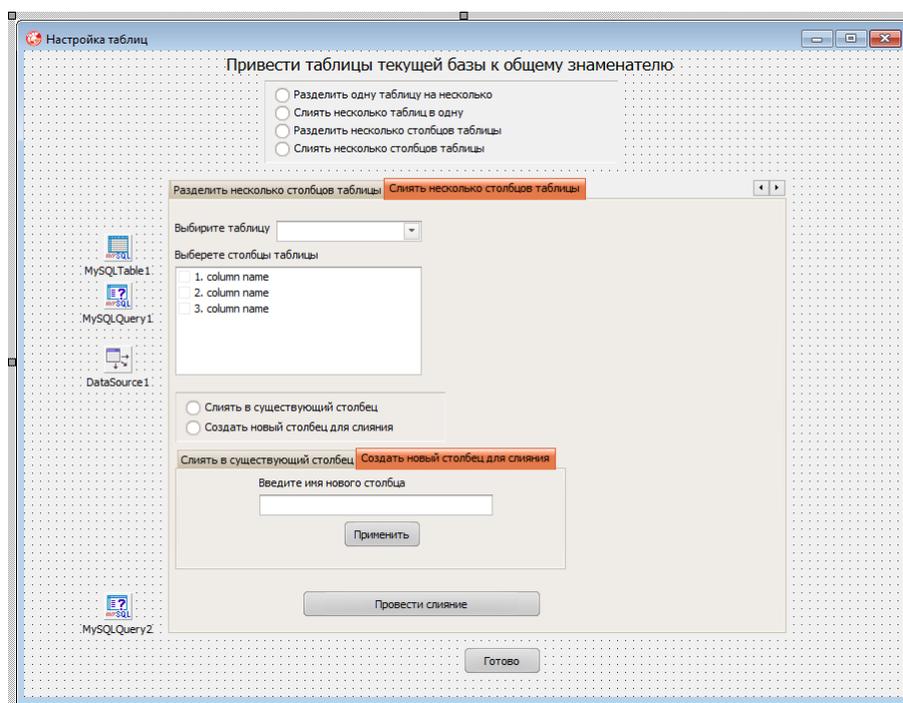


Рисунок 14 – Пункт «Создать новый столбец для слияния» функции «Слить несколько столбцов таблицы» формы «Setting\_up\_tables»

По завершению работы с настройкой таблиц (формы «Setting\_up\_tables») необходимо нажать кнопку «Готово». Произойдет скрытие формы «Setting\_up\_tables» и переход, обратно, к форме «Tables\_Step\_three».

Кнопка «Далее» осуществляет скрытие текущей формы (формы «Tables\_Step\_three») и переход к следующей форме «Tables\_Step\_four».

На форме «Tables\_Step\_four» расположены следующие компоненты sLabel, sButton, sGroupBox, sComboBox, StringGrid, MySQLTable, MySQLQuery, DataSource (рис. 15).

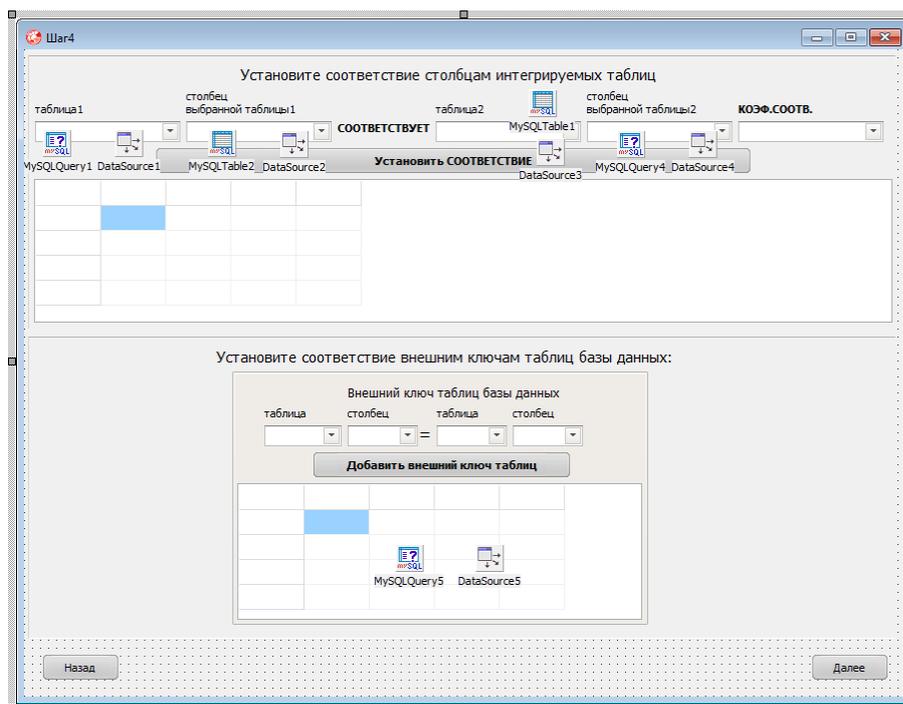


Рисунок 15 – Форма «Tables\_Step\_four»

Благодаря компонентам sComboBox, расположенным в верхней части формы, пользователь может выбрать первую таблицу и ее столбец, а также вторую таблицу и ее столбец, которые содержат схожие данные, а также указать коэффициент соответствия. Данный коэффициент отвечает за веса критериев соответствия и участвует в формуле расчета подобия на следующем этапе. Коэффициент может быть следующих видов: «Семантические атрибуты», «Атрибуты времени», «Универсальный идентификатор», «Атрибуты внешних ключей», «Факты». Нажатие на кнопку «Установить соответствие» производит вывод выбранных параметров в StringGrid1. На основе установленного пользователем соответствия, программа в дальнейшем начнет производить сравнение по записям именно выбранных столбцов каждой из таблиц базы данных.

Ниже расположена еще одна группа компонентов sComboBox, отвечающих за добавление внешних ключей. Данный пункт необходим, если выбранные на предыдущем этапе таблицы содержат внешние ключи других таблиц, хранящихся в базе данных. Данный пункт не является обязательным

и настраивается по усмотрению пользователя. Нажатие на кнопку «Добавить внешний ключ таблиц» производит вывод выбранных параметров в StringGrid2.

Кнопка «Далее» осуществляет скрытие текущей формы и переход к следующей форме «Tables\_Step\_five». Кнопка «Назад» осуществляет закрытие текущей формы и освобождение выделенной под нее памяти.

На форме «Tables\_Step\_five» расположены следующие компоненты sLabel, sButton, sDBGrid, DataSource, MySQLQuery, StringGrid (рис. 16).

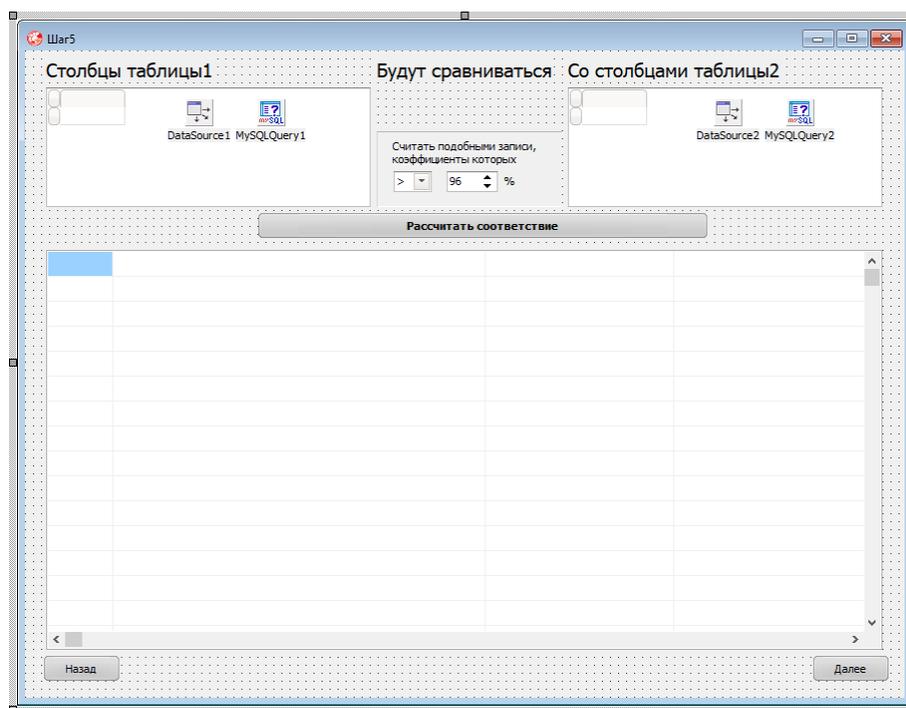


Рисунок 16 – Форма «Tables\_Step\_five»

Компонент MySQLQuery содержит SQL запрос, представленный перечислением таблиц, выбранных на предыдущем этапе при расстановке соответствия. Именно в таком порядке информация будет отображена в компонентах sDBGrid, расположенных в верхней части формы.

Кнопка «Рассчитать соответствие» производит извлечение записи из n-го столбца первой таблицы, и сравнивает ее со всеми записями n-го столбца второй таблицы. Затем берет следующую запись того-же столбца первой таблицы и сравнивает со всеми записями столбца второй таблицы. Цикл продолжается до тех пор, пока не закончатся все столбцы обеих таблиц. По окончании расчета, пользователю будет выведена в StringGrid следующая информация: название сравниваемой записи n-го столбца первой таблицы, напротив нее будет выведена название сравниваемой записи n-го столбца второй таблицы, рядом с ней коэффициент подобия; а также предусмотрен вывод в предпоследний столбец StringGrid информации, содержащей среднеарифметическое значение коэффициентов подобия и надпись «подобны» на против каждой сравниваемой записи. Данная надпись ставится на основе значения, выбранного пользователем в пункте «Считать

подобными записи, коэффициенты которых». В последнем столбце записям, чье среднее арифметическое значение коэффициентов больше этого значения, автоматически присваивается значение «подобны». Теперь пользователь сам может изменить это значение или присвоить его другим записям, просто кликнув на нужную ячейку последнего столбца StringGrid. Для удобства было добавлено программно создание компонента ComboBox при клике на ячейку StringGrid.

Кнопка «Далее» осуществляет скрытие текущей формы и переход к следующей форме «Tables\_Step\_six».

На форме «Tables\_Step\_six» расположены следующие компоненты sLabel, sButton, StringGrid (рис. 17).

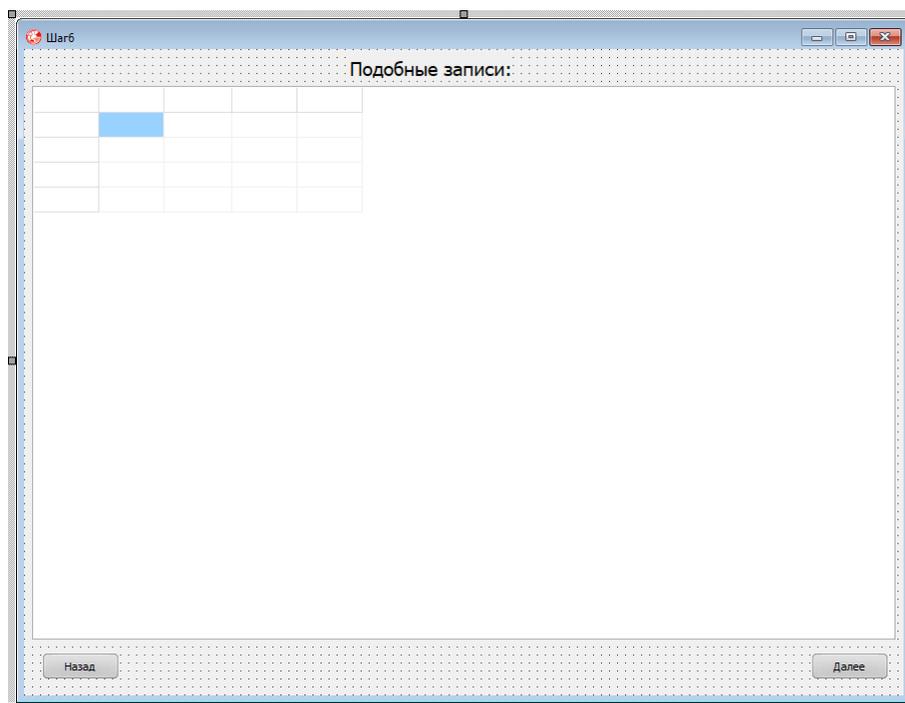


Рисунок 17 – Форма «Tables\_Step\_six»

В компонент StringGrid данной формы выводятся все записи из StringGrid предыдущего шага, которым были присвоены свойства «подобны».

Кнопка «Далее» осуществляет скрытие текущей формы и переход к следующей форме «Tables\_Step\_seven».

На форме «Tables\_Step\_seven» расположены следующие компоненты sLabel, sButton, sRadioGroup, sPageControl, sCheckBoxList, sEdit, sComboBox, MySQLTable, MySQLQuery, DataSource (рис. 18).

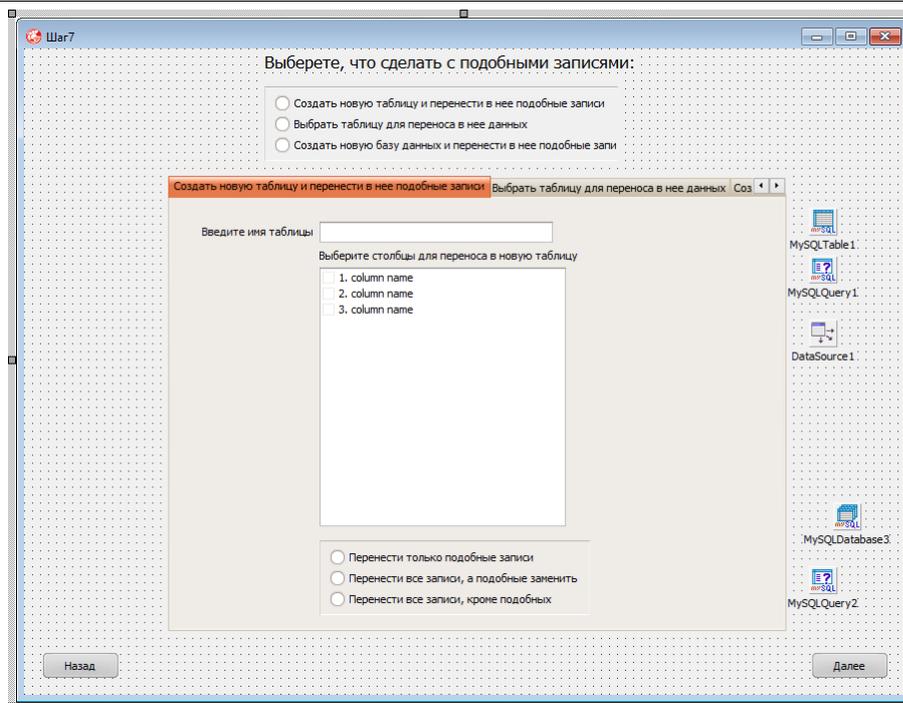


Рисунок 18 – Форма «Tables\_Step\_seven»

Компонент `sRadioGroup1`, расположенный в верхней части формы предназначен для предоставления пользователю выбора дальнейшего действия. При выборе, пользователю отображается соответствующая вкладка компонента `sPageControl1` с расположенными на ней функциями.

Рассмотрим подробнее все возможности функционального выбора.

При выборе пункта «Создать новую таблицу и перенести в нее подобные записи» `sRadioGroup1`, происходит отображение вкладки «Создать новую таблицу и перенести в нее подобные записи» `sPageControl1`. На данной вкладке пользователь может ввести имя новой таблицы в `sEdit`, а также выбрать какие столбцы исходных таблиц задействовать при в переносе информации в `sCheckListBox` (рис. 19).

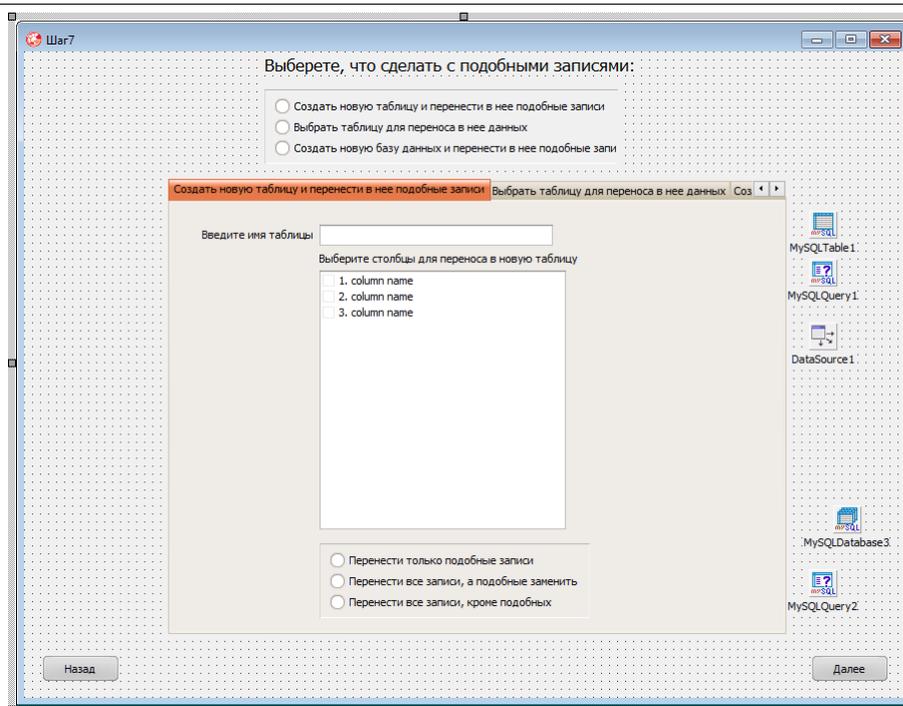


Рисунок 19 – Пункт «Создать новую таблицу и перенести в нее подобные записи» формы «Tables\_Step\_seven»

При выборе пункта «Выбрать таблицу для переноса в нее данных» sRadioGroup1, происходит отображение вкладки «Выбрать таблицу для переноса в нее данных» sPageControl1. На данной вкладке пользователь может выбрать таблицу базы данных в sComboBox, а также выбрать в sCheckBoxListBox какие столбцы исходных таблиц задействовать при в переносе информации в выбранную таблицу (рис. 20).

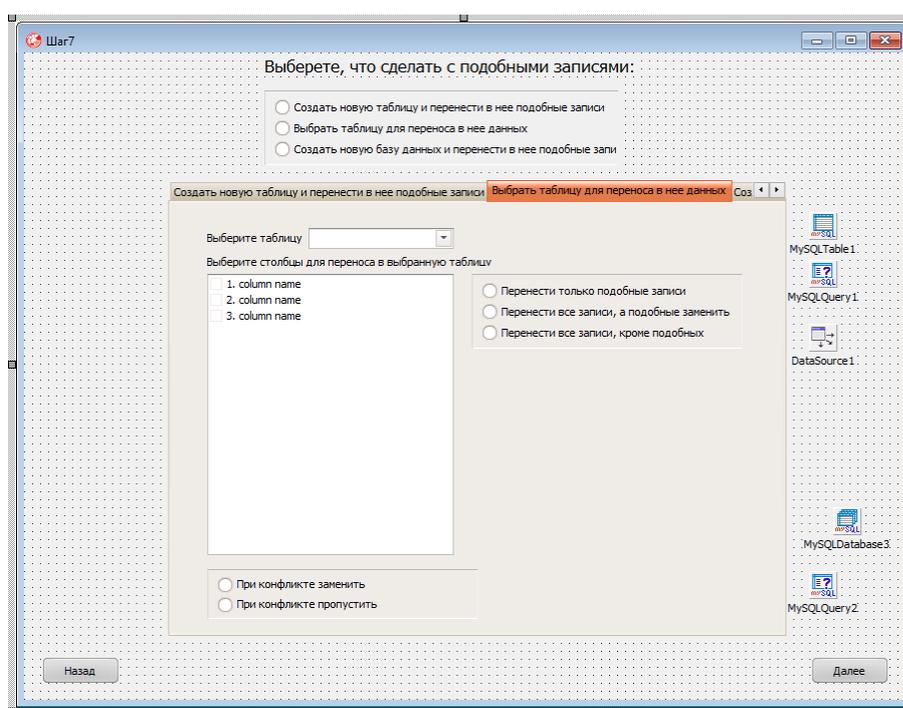


Рисунок 20 – Пункт «Выбрать таблицу для переноса в нее данных» формы «Tables\_Step\_seven»

При выборе пункта «Создать новую базу данных и перенести в нее подобные записи» sRadioGroup1, происходит отображение вкладки «Создать новую базу данных и перенести в нее подобные записи» sPageControll1. На данной вкладке пользователь может задать имя для новой базы данных в sEdit, выбрать в компонентах sCheckListBox какие таблицы и их столбцы задействовать при в переносе информации в новую базу данных, а также указать расположение для новой базы данных.

Кнопка «Проверить подключение к хосту» осуществляет проверку подключения на основе введенной пользователем информации и включает в себя функции отлова ошибок подключения (рис. 21).

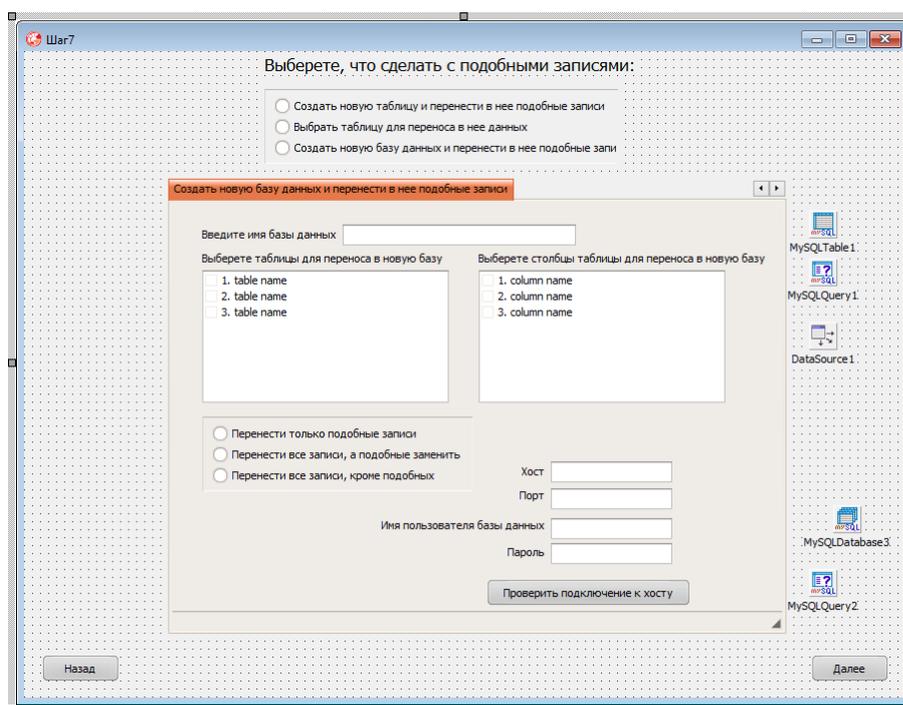


Рисунок 21 – Пункт «Создать новую базу данных и перенести в нее подобные записи» формы «Tables\_Step\_seven»

По завершению работы с формой «Tables\_Step\_seven», необходимо нажать кнопку «Далее». Произойдет скрытие формы и переход к форме «Tables\_Result».

На форме «Tables\_Result» расположены следующие компоненты sGroupBox, sLabel, sButton, sDBGrid, MySQLDatabase, MySQLQuery, DataSource (рис. 22).

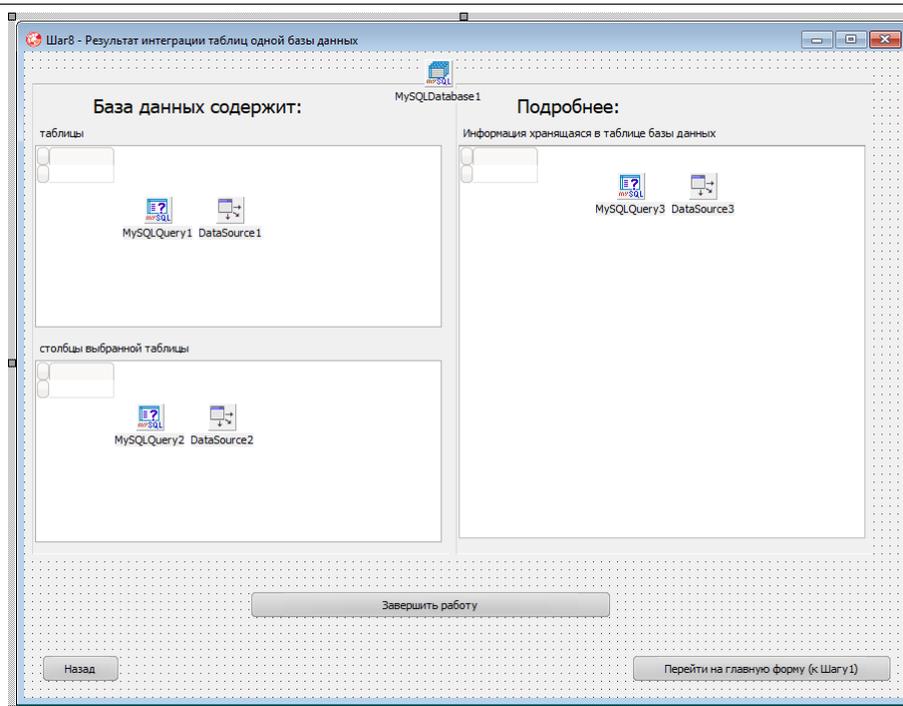


Рисунок 22 – Форма «Tables\_Result»

Компонент `sDBGrid1` предназначен для отображения списка таблиц базы данных. При выборе таблицы в `sDBGrid1`, список, содержащий название столбцов выбранной таблицы, а также полный перечень параметров выводится в `sDBGrid2`. Вся информация, содержащаяся в выбранной в `sDBGrid1` таблицы, выводится пользователю в `sDBGrid3`.

Кнопка «Завершить работу» осуществляет закрытие программы. Кнопка «Перейти на главную форму (к Шагу1)» осуществляет закрытие текущей формы и переход к форме «Step\_one». Кнопка «Назад» осуществляет закрытие текущей формы и освобождение выделенной под нее памяти.

Рассмотрим форму «Database\_Step\_two», предназначенной для подключения к двум базам данных и их последующую интеграцию.

На форме «Database\_Step\_two» расположены следующие компоненты `sLabel`, `sEdit`, предназначенный для ввода параметров подключения к базе, `sButton`, `sCheckBox`, `sGroupBox`, `MySQLDatabase`, `sStatusBar` (рис. 23). Данные компоненты выполняют схожие функции, что и компоненты, расположенные на форме «Tables\_Step\_two», но в отличие от последних, предназначены для подключения сразу двух баз данных.

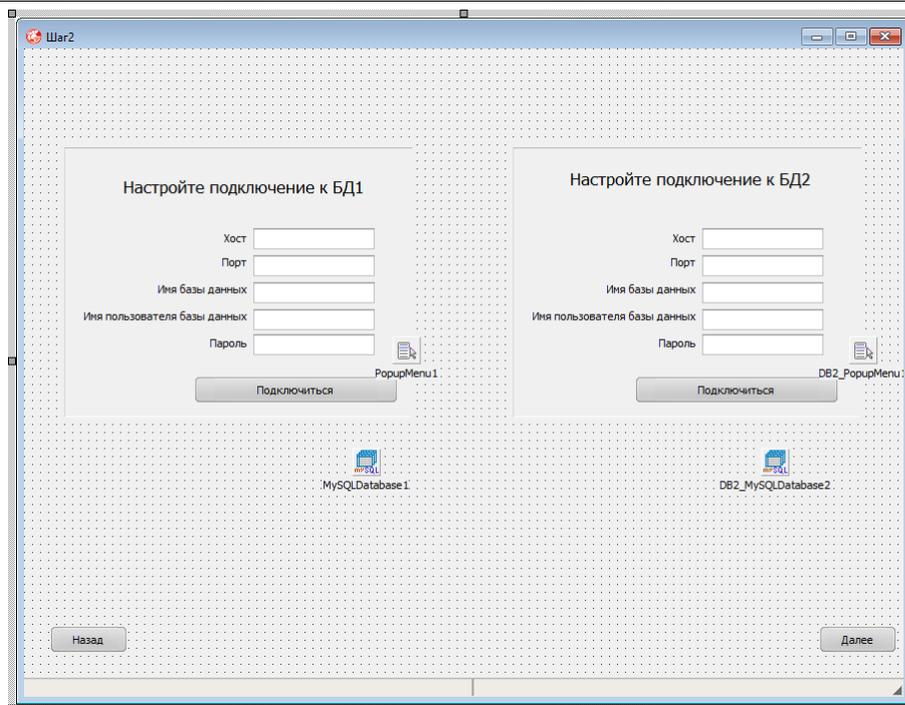


Рисунок 23 – Форма «Database\_Step\_two»

Компоненты, используемые для подключения ко второй базе данных, имеют префикс перед именем «DB2\_». Это сделано с целью упростить ориентирование в коде при его написании.

На форме «Database\_Step\_three» расположены следующие компоненты sLabel, sButton, sGroupBox, sDBGrid, MySQLQuery, DataSource (рис. 24).

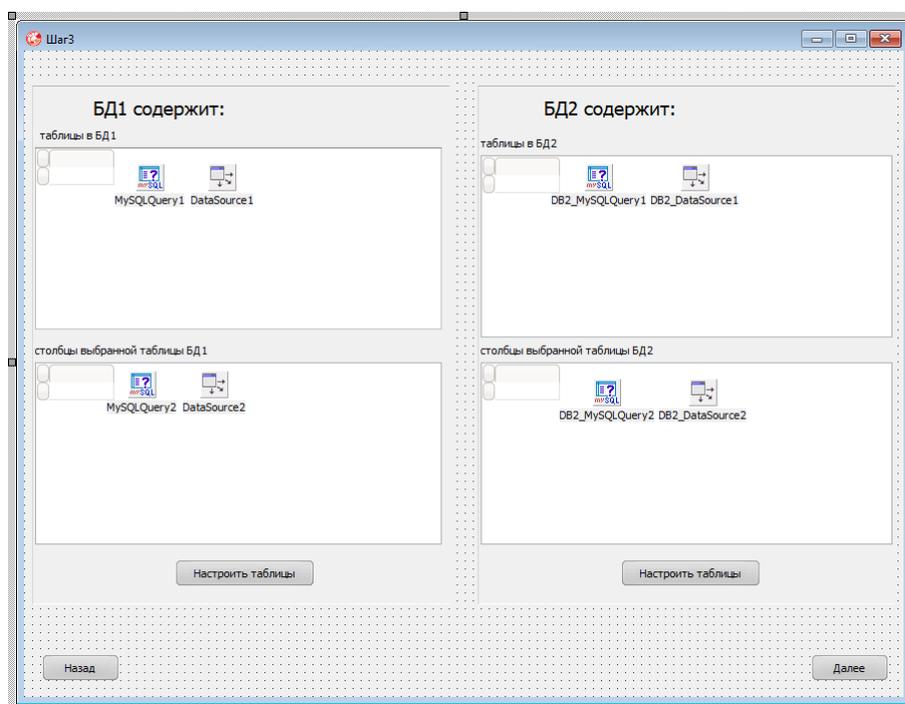


Рисунок 24 – Форма «Database\_Step\_three»

Компонент sDBGrid1 предназначен для отображения списка таблиц базы данных. При выборе таблицы в sDBGrid1, список, содержащий название

столбцов выбранной таблицы, а также полный перечень параметров выводится в sDBGrid2.

Кнопка «Настроить таблицы», производит скрывание текущей формы и переход к специальной форме «Setting\_up\_tables». Кнопка «Далее» осуществляет скрывание текущей формы (формы «Database\_Step\_three») и переход к следующей форме «Database\_Step\_four». Кнопка «Назад» осуществляет закрытие текущей формы и освобождение выделенной под нее памяти.

На форме «Database\_Step\_four» расположены следующие компоненты sLabel, sButton, sGroupBox, sComboBox, StringGrid, MySQLTable, MySQLQuery, DataSource (рис 25).

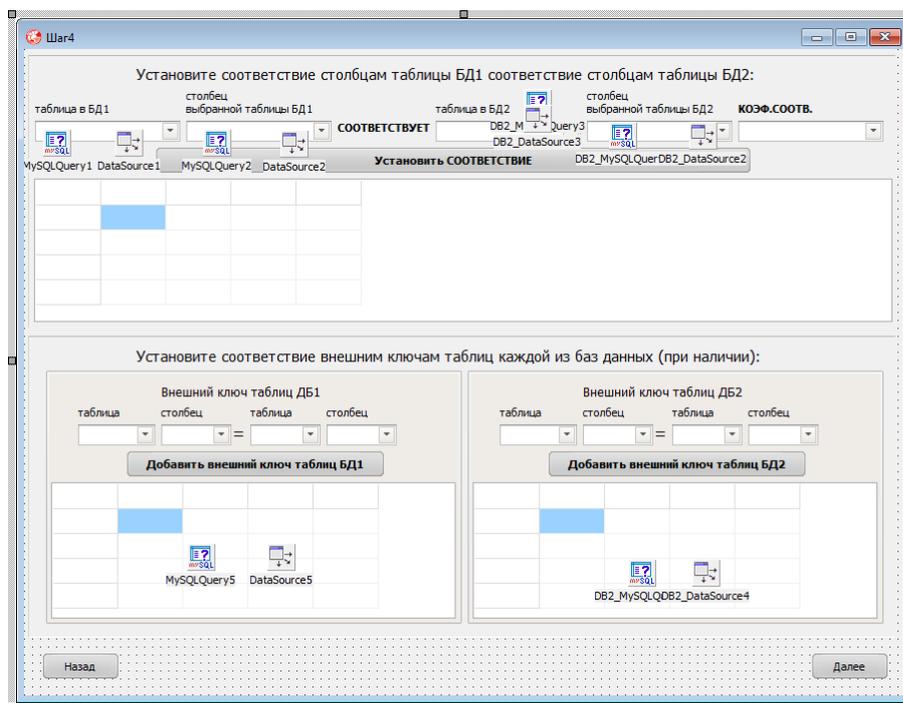


Рисунок 25 – Форма «Database\_Step\_four»

Благодаря компонентам sComboBox, расположенным в верхней части формы, пользователь может выбрать таблицу первой базы данных, ее столбец, а также таблицу второй базы данных и ее столбец, которые содержат схожие данные, а также коэффициент соответствия. Нажатие на кнопку «Установить соответствие» производит вывод выбранных параметров в StringGrid1.

Ниже расположена еще одна группа компонентов sComboBox, отвечающих за добавление внешних ключей. Нажатие на кнопку «Добавить внешний ключ таблиц» производит вывод выбранных параметров в StringGrid2 или StringGrid3 соответственно.

Кнопка «Далее» осуществляет скрывание текущей формы и переход к следующей форме «Database\_Step\_five». Кнопка «Назад» осуществляет закрытие текущей формы и освобождение выделенной под нее памяти.

На форме «Database\_Step\_five» расположены следующие компоненты sLabel, sButton, sDBGrid, DataSource, MySQLQuery, StringGrid (рис. 26).

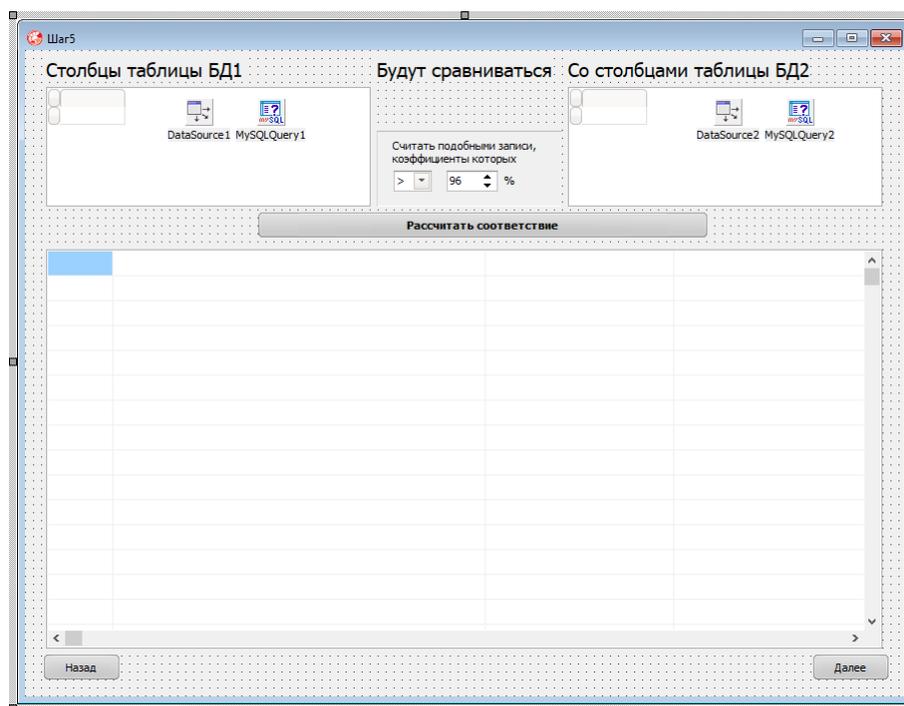


Рисунок 26 – Форма «Database\_Step\_five»

Компонент MySQLQuery содержит SQL запрос, представленный перечислением таблиц, выбранных на предыдущем этапе при расстановке соответствия. Именно в таком порядке информация будет отображена в компонентах sDBGrid, расположенных в верхней части формы.

Кнопка «Расчитать соответствие» производит извлечение записи из  $n$ -го столбца таблицы первой базы данных, и сравнение его со всеми записями  $n$ -го столбца таблицы второй базы данных. Затем берет следующую запись того-же столбца таблицы первой базы данных и сравнивает со всеми записями столбца таблицы второй базы данных. Цикл продолжается до тех пор, пока не закончатся все столбцы таблиц обеих баз данных. По окончании расчета, пользователю будет выведена в StringGrid следующая информация: название сравниваемой записи  $n$ -го столбца таблицы первой базы данных, напротив нее будет выведена название сравниваемой записи  $n$ -го столбца таблицы второй базы данных, рядом с ней коэффициент подобия; а также предусмотрен вывод в последний столбец StringGrid информации, содержащей среднеарифметическое значение коэффициентов подобия и надпись «подобны» на против каждой сравниваемой записи. Данная надпись ставиться на основе значения, выбранного пользователем в пункте «Считать подобными записи, коэффициенты которых». В последнем столбце записям, чье среднее арифметическое значение коэффициентов больше этого значения, автоматически присваивается значение «подобны». Теперь пользователь сам может изменить это значение или присвоить его другим записям, просто кликнув на нужную ячейку последнего столбца StringGrid.

Для удобства был добавлено программно создание компонента ComboBox при клике на ячейку StringGrid.

Кнопка «Далее» осуществляет скрывание текущей формы и переход к следующей форме «Database\_Step\_six». Кнопка «Назад» осуществляет закрытие текущей формы и освобождение выделенной под нее памяти.

На форме «Database\_Step\_six» расположены следующие компоненты sLabel, sButton, StringGrid (рис. 27).

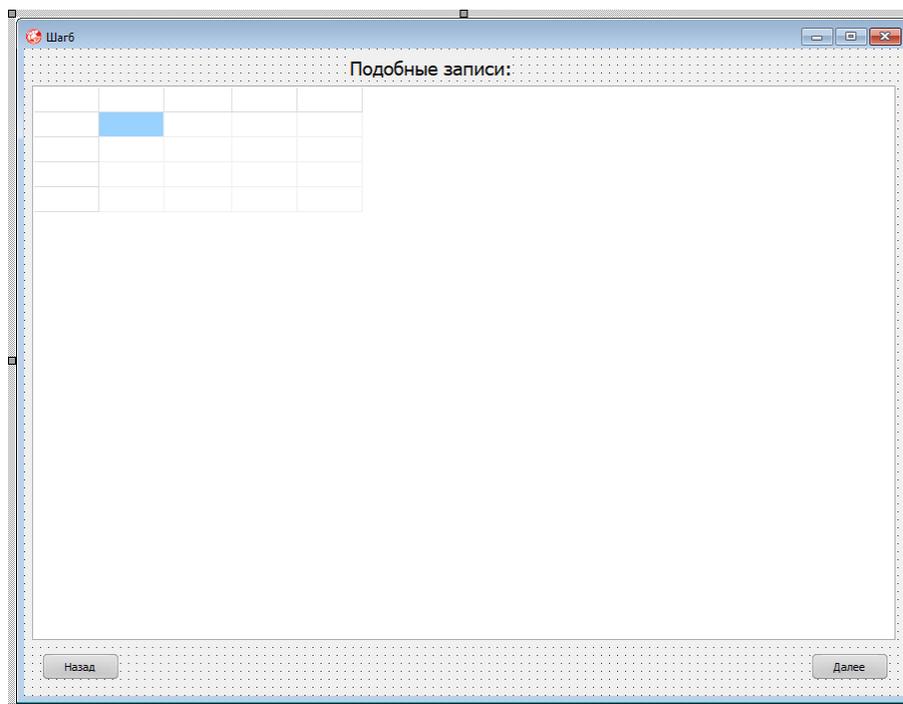


Рисунок 27 – Форма «Database\_Step\_six» программы

В компонент StringGrid данной формы выводятся все записи из StringGrid предыдущего шага, которым было присвоено свойство «подобны».

Кнопка «Далее» осуществляет скрывание текущей формы и переход к следующей форме «Database\_Step\_seven».

На форме «Database\_Step\_seven» расположены следующие компоненты sLabel, sButton, sRadioGroup, sPageControl, sCheckListBox, sEdit, sComboBox, MySQLTable, MySQLQuery, DataSource (рис. 28).

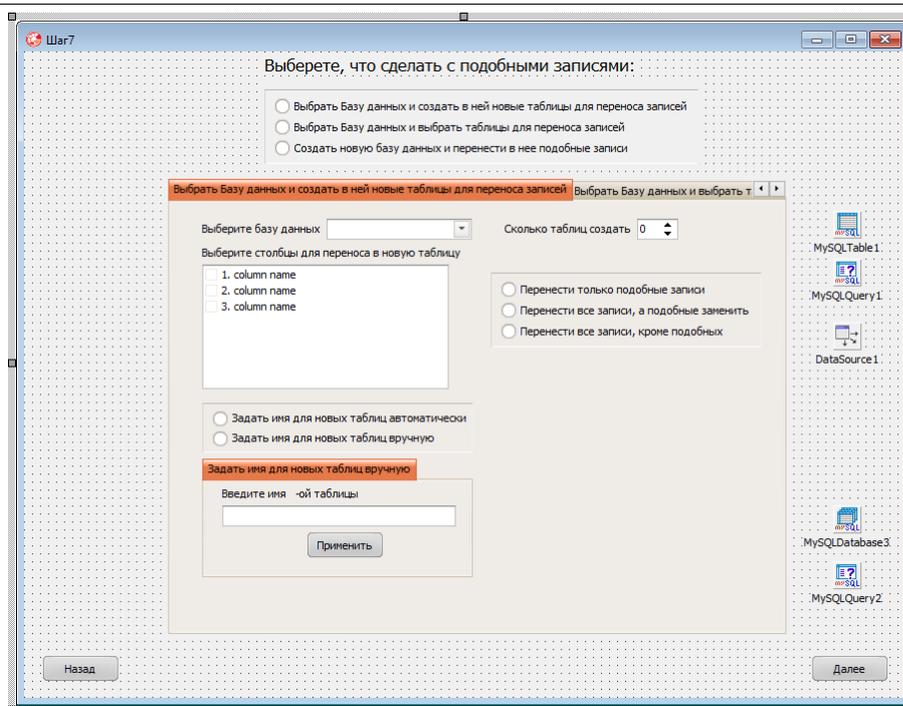


Рисунок 28 – Форма «Database\_Step\_seven»

Компонент `sRadioGroup1`, расположенный в верхней части формы предназначен для предоставления пользователю выбора дальнейшего действия. При выборе, пользователю отображается соответствующая вкладка компонента `sPageControl1` с расположенными на ней функциями.

Рассмотрим подробнее все возможности функционального выбора.

При выборе пункта «Выбрать Базу данных и создать в ней новые таблицы для переноса записей» `sRadioGroup1`, происходит отображение вкладки «Выбрать Базу данных и создать в ней новые таблицы для переноса записей» `sPageControl1`. На данной вкладке пользователь может выбрать название базы данных в `sComboBox`, указать в `sSpinEdit` количество новых таблиц, которые будут созданы в выбранной базе данных, а также выбрать какие именно столбцы таблицы, для которой были рассчитаны коэффициенты, необходимо задействовать.

Далее пользователю необходимо выбрать нужный пункт `sRadioGroup2`. При выборе пункта «Задать имя для новых таблиц автоматически», имя для каждой новой таблицы будет задано программой и будет содержать префикс, а вкладка «Задать имя для новых таблиц вручную» `sPageControl2` будет скрыта. При выборе пункта «Задать имя для новых таблиц вручную», отображается вкладка «Задать имя для новых таблиц вручную» `sPageControl2` на которой пользователю самому необходимо ввести имя для каждой новой таблицы в `sEdit` и при этом каждый раз нажимать кнопку «Применить».

По окончании работы пользователю необходимо нажать на кнопку «Далее» и новые таблицы будут добавлены в выбранную базу при помощи SQL запроса компонента `MySQLQuery` (рис. 29).

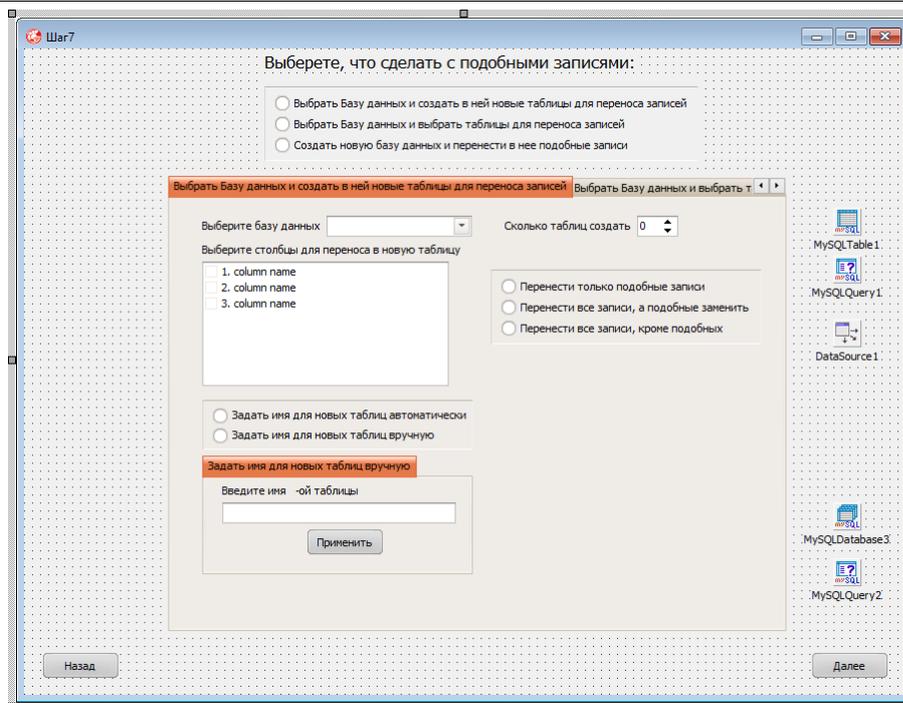


Рисунок 29 – Пункт «Выбрать Базу данных и создать в ней новые таблицы для переноса записей» формы «Database\_Step\_seven»

При выборе пункта «Выбрать Базу данных и выбрать таблицы для переноса записей» sRadioGroup1, происходит отображение вкладки «Выбрать Базу данных и выбрать таблицы для переноса записей» sPageControl1. На данной вкладке пользователь может выбрать название базы данных в sComboBox, выбрать таблицы и столбцы базы данных в sCheckListBox1 и sCheckListBox2, а также в sCheckListBox3 выбрать какие именно столбцы таблицы, для которой были рассчитаны коэффициенты, необходимо задействовать (Рис. 30).

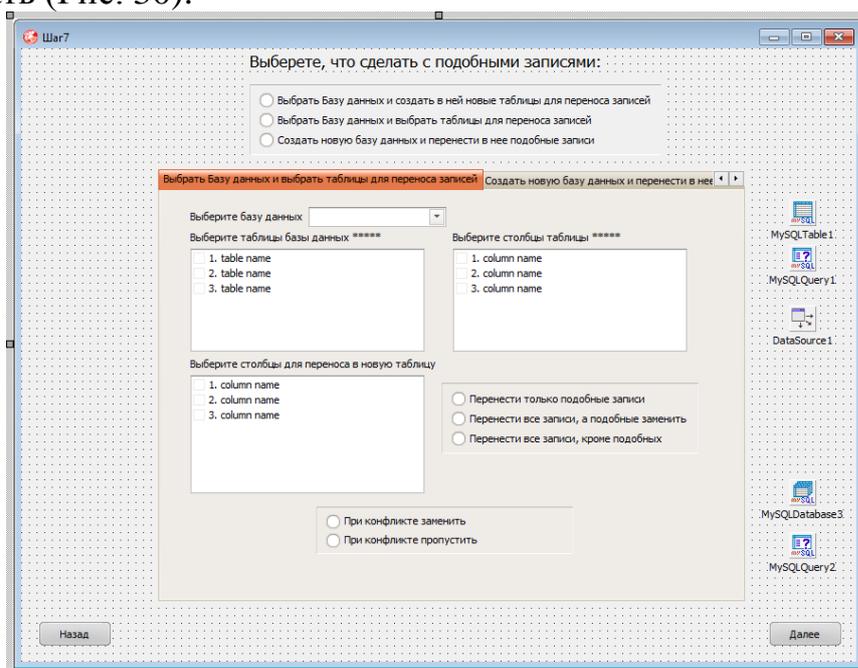


Рисунок 30 – Пункт «Выбрать Базу данных и выбрать таблицы для переноса записей» формы «Database\_Step\_seven»

При выборе пункта «Создать новую базу данных и перенести в нее подобные записи» sRadioGroup1, происходит отображение вкладки «Создать новую базу данных и перенести в нее подобные записи» sPageControll1. На данной вкладке пользователь может задать имя для новой базы данных в sEdit, выбрать в компонентах sCheckListBox, какие таблицы и их столбцы задействовать при в переносе информации в новую базу данных, а также указать расположение для новой базы данных.

Кнопка «Проверить подключение к хосту» осуществляет проверку подключения на основе введенной пользователем информации и включает в себя функции отлова ошибок подключения (рис. 31).

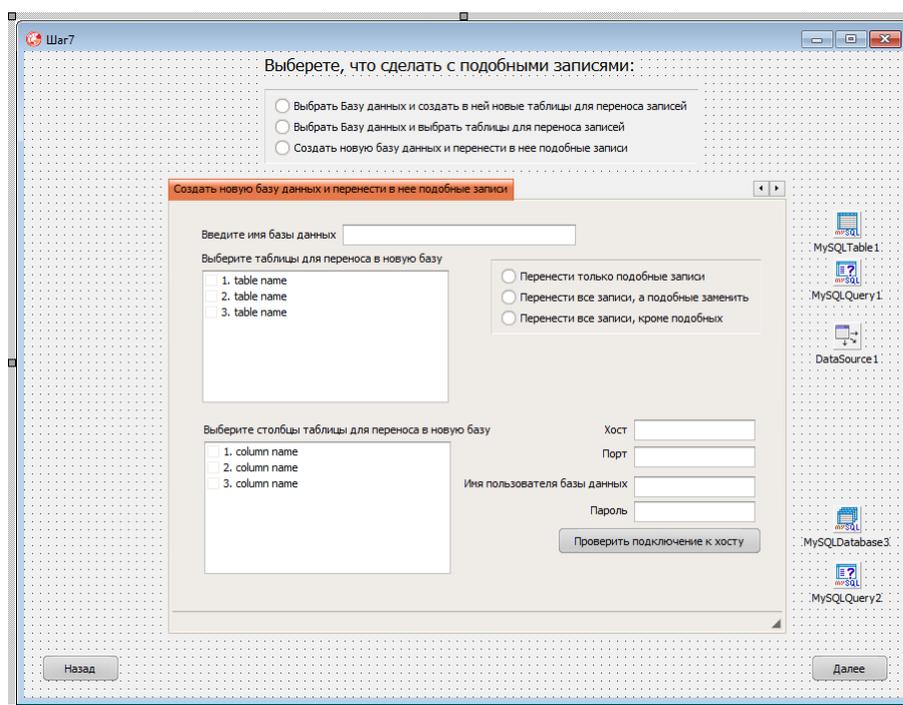


Рисунок 31 – Пункт «Создать новую базу данных и перенести в нее подобные записи» формы «Database\_Step\_seven»

По завершению работы с формой «Database\_Step\_seven», необходимо нажать кнопку «Далее». Произойдет скрытие формы и переход к форме «Database\_Result».

На форме «Database\_Result» расположены следующие компоненты sGroupBox, sLabel, sButton, sDBGrid, MySQLDatabase, MySQLQuery, DataSource (рис. 32).

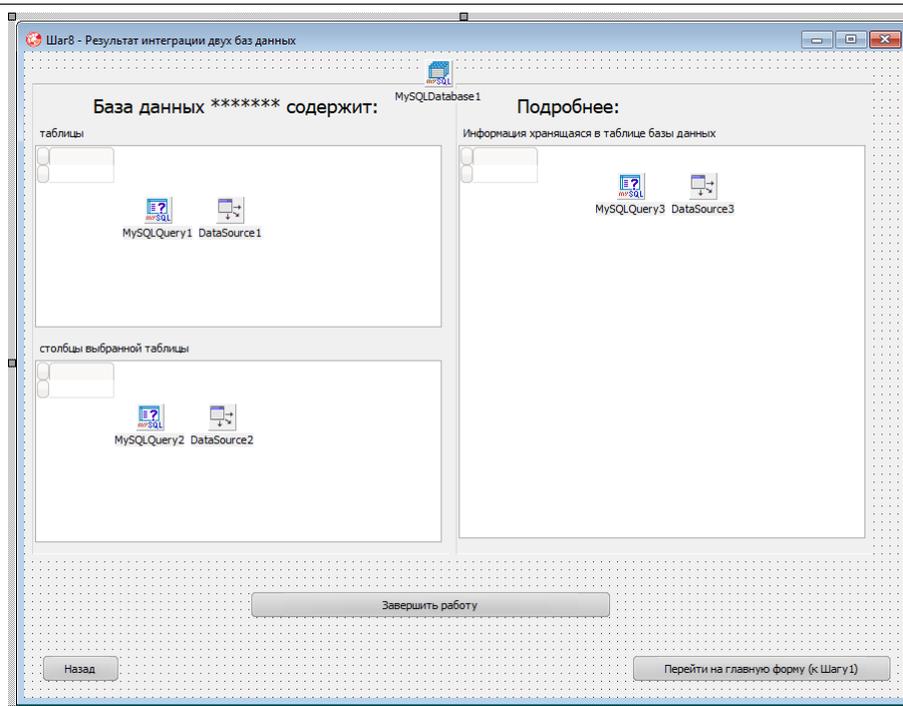


Рисунок 32 – Форма «Database\_Result»

Компонент sDBGrid предназначен для отображения списка таблиц базы, выбранной для интеграции рассчитанной информации на предыдущем этапе. При выборе таблицы в sDBGrid1, список, содержащий название столбцов выбранной таблицы, а также полный перечень параметров выводится в sDBGrid2. Вся информация, содержащаяся в выбранной в sDBGrid1 таблицы, выводится пользователю в sDBGrid3.

Кнопка «Завершить работу» осуществляет закрытие программы. Кнопка «Перейти на главную форму (к Шагу1)» осуществляет закрытие текущей формы и переход к форме «Step\_one». Кнопка «Назад» осуществляет закрытие текущей формы и освобождение выделенной под нее памяти.

Таким образом, можно убедиться, в том, что разработанные нами функциональные формы автоматизированной системы по интеграции частично гомогенных баз данных были созданы на высоком уровне и в дальнейшем должны автоматизировать процесс приведения баз данных к общему знаменателю, процесс интеграции таблиц внутри одной базы данных, а также интеграции двух частично гомогенных баз данных благодаря автоматическому расчету коэффициентов подобия.

Первая рабочая версия автоматизированной системы по интеграции частично гомогенных баз данных представляет собой работоспособное приложение, протестированное на Windows 10, Windows 7, Windows XP, Ubuntu Linux v 18.04.

**Библиографический список**

1. Genesereth M.R., Keller A.M., Duschka O.M. Informaster: An Information Integration System. In Proceedings of ACM SIGMOD International Conference (May 13-15, Tucson, AZ, USA), 1997, pp. 539-542
2. Yan L.L., Otsu M.T., Liu L. Accessing Heterogeneous Data Through Homogenization and Integration Mediators. In Proceedings Second IFCIS International Conference on Cooperative Information Systems (June 24-27, Kiawah Island, SC, USA), 1997, IEEE CS Press, pp.130-139
3. Li C., Yerneni R., Vassalos V., Garcia-molina H., Papakonstantinou Y., Ullman J., Valiveti M. Capability Based Mediation in TSIMMIS. In Proceedings of the 1998 ACM SIGMOD Conference, (June 1-4, Seattle, USA), ACM SIGMOD Record, 27, 2, (June 1998), pp.564-566
4. Klas W., Fankhauser P., Muth P., Rakow T.C., Neuhold E.J. Database Integration using the Open Object-Oriented Database System VODAK. In Object Oriented Multidatabase Systems: A Solution for Advanced Applications, Bukhres O., Elmagarmid A.K. (Eds.), Prentice Hall, 1995
5. Lee J., Madnick S.E., Siegel M.D. Conceptualizing Semantic Interoperability: A Perspective from the Knowledge Level. International Journal of Cooperative Information Systems, 5, 4, (December 1996), pp.367-393
6. Thiran Ph., Hainaut J.-L., Bodart S., Deflorenne A., Hick J.-M. Interoperation of Independent, Heterogeneous and Distributed Databases. Methodology and CASE Support: the InterDB Approach. In Proceedings Third IFCIS International Conference on Cooperative Information Systems (August 20-22, New York, USA), 1998, IEEE CS Press, pp. 54-63
7. Ziegler P., Dittrich K.R. User-Specific Semantic Integration of Heterogeneous Data: The SIRUP Approach. In Mokrane Bouzeghoub, Carole Goble, Vipul Kashyap, and Stefano Spaccapietra, editors, First International IFIP Conference on Semantics of a Networked World (ICSNW 2004). volume 3226 of Lecture Notes in Computer Science, pages 44–64, Paris, France, June 17–19, 2004. Springer.
8. Ziegler P., Sturm C., Dittrich K.R. Unified Querying of Ontology Languages with the SIRUP Ontology Query API. In 11. GI-Fachtagung for Datenbanksysteme in Business, Technologie und Web (BTW 2005), volume P-65 of Lecture Notes in Informatics, pages 325–344, Karlsruhe, Germany, March 2–4, 2005.
9. Ziegler P. Evaluation of SIRUP with the SIRUP Classification of Data Integration Conflicts. Technical Report ifi-2007.07, Department of Informatics, University of Zurich, 2007.
10. Rousset M.-C., Reynaud C. Picisel and Xyleme: two illustrative information integration agents. Book chapter in Intelligent Information Agents Research and Development in Europe, Springer-Verlag, 2003.
11. Reynaud C., Giraldo G. An application of the mediator approach to services over the Web. Concurrent Engineering, 2003.
12. Reynaud C. Building scalable mediator systems. Topical Day in Semantic

- Integration of Heterogeneous Data, IFIP World Computer Congress, 2004.
13. Levy A., Rajaraman A., Ordille J. Querying Heterogeneous Information Sources Using Source Descriptions. Proceedings of the Twenty-second International Conference on VLDB 1996.
  14. Boyd M., Kittivoravikul S., Lazanitis C., McBrien P.J., Rizopoulos N. AutoMed: A BAV Data Integration System for Heterogeneous Data Sources. In Proceedings of CAiSE04, Springer Verlag LNCS Vol 3084, Pages 82–97, 2004.
  15. Boyd M., McBrien P.J. Comparing and Transforming Between Data Models via an Intermediate Hypergraph Data Model. Journal on Data Semantics IV, Pages 69–109, SpringerVerlag, 2005.
  16. Lazarus 1.8.2 – Бесплатная среда по разработке программного обеспечения // Lazarus URL: <https://lazarus-rus.ru> (дата обращения: 17.06.2018).
  17. Download Your Free Delphi Trial Today - Embarcadero // Fast Cross-Platform App Development Software - Embarcadero URL: <https://www.embarcadero.com/ru/products/delphi/start-for-free> (дата обращения: 17.06.2018).
  18. AlphaControls // RAD Studio Delphi and C++ Builder VCL components with themes and skins URL: [http://www.alphaskins.com/index\\_rus.php](http://www.alphaskins.com/index_rus.php) (дата обращения: 17.06.2018).