

**Нижегородский государственный университет
им.Н.И.Лобачевского
Институт информационных технологий,
математики и механики
Центр информатики и интеллектуальных
информационных технологий**

**Система управления архивом
ГИС Терра
Руководство программиста**

**Нижний Новгород
2019**

СОДЕРЖАНИЕ

1. Введение	3
2. Условия использования	3
3. Модель данных	4
3.1. Структура архива	4
3.2. Структура регистрационной карточки	4
3.3. Структура базы данных архива	5
4. Классы	6
4.1. База данных архива	6
4.1.1. Конструктор	6
4.1.2. Подключение к базе данных архива	6
4.1.3. Отключение от базы данных	7
4.1.4. Получение параметров подключения	8
4.1.5. Проверка пользователя	8
4.1.6. Регистрация в журнале работ	8
4.1.7. Блокировка пользователя	8
4.1.8. Управление транзакциями	9
4.1.8.1. Старт транзакции	9
4.1.8.2. Завершение транзакции	9
4.1.8.3. Откат транзакции	9
4.2. Документ архива	9
4.2.1. Конструктор	10
4.2.2. Выборка информации из архива	10
4.2.2.1. Открыть чтение документов архива (читается первый документ по запросу)	10
4.2.2.2. Прочитать следующий документ по запросу	10
4.2.2.3. Принудительное закрытие запроса на чтение	10
4.2.2.4. Чтение документа по номеру	10
4.2.2.5. Чтение заголовка документа по номеру	10
4.2.2.6. Получение списка дополнительных файлов документа	10
4.2.2.7. Получение любого файла из документа	10
4.2.2.8. Установление порядка выдачи документов при чтении	11
4.2.2.9. Установления статуса для выборки документов	11
4.2.3. Установление фильтров для операции чтения Open	11
4.2.4. Запись документов в архив	11
4.2.4.1. Запись документа в архив	11
4.2.4.2. Запись документа старого формата DDC в архив	12
4.2.4.3. Запись документа старого формата DDC в архив с хранилищем на DVD-дисках	12
4.2.5. Удаление документов из архива	12

4.2.5.1. Логическое удаление документов по установленным ранее фильтрам (запросам)	12
4.2.5.2. Физическое удаление документов по установленному ранее статусу	12
4.2.5.3. Логическое удаление документа по его номеру	12
4.2.5.4. Физическое удаление документа по его номеру и статусу	12
4.2.6. Поиск документов	13
4.2.6.1. Поиск документа, ближайшего к заданной точке	13
4.2.6.2. Вычисление расстояние от точки до контура	13
4.2.7. Дополнительные функции	13
4.2.7.1. Получение текста грифа секретности по его номеру	13
4.2.7.2. Получение типа проекции по его номеру	13
4.2.7.3. Формирование регистрационной карточки документа по его номеру	13
5. Пример приложения для работы с архивом	13
Литература	15

1. Введение

В документе описан интерфейс прикладного программиста (API), предназначенный для разработки приложений с использованием средств системы управления архивом ГИС Терра Plus.

Данный API может быть использован для создания VCL-приложений для ОС Windows.

2. Условия использования

Разработка приложений должна производиться в среде системы программирования Embarcadero RAD Studio [1] версии, начиная с XE7.

Функционирование разработанных приложений возможно под управлением русифицированных операционных систем семейства Microsoft Window.

На сервере с базой данных архива должна быть установлена операционная система Microsoft Windows Server версии не ниже 2008. Кроме того, должна быть установлена СУБД PostgreSQL [2] версии не ниже 9.5.1.1 (32-разрядная) с расширением PostGIS [3] версии не ниже 2.5 (32-разрядная).

Технические характеристики сервера будут в значительной степени определять быстродействие работы приложений.

На клиентских местах архива может быть установлена русифицированная операционная система Microsoft Windows версии 7 или 10 (как 32-разрядная, так и 64-разрядная).

На клиенте должны находиться вендорные библиотеки СУБД PostgreSQL: libeay32.dll, libintl.dll, libpg.dll, sslsleay32.dll. Путь к папке с этими библиотеками должен прописываться в конфигурационных файлах создаваемых приложений.

Более никаких дополнительных требований к общесистемному программному обеспечению клиентского места не требуется.

3. Модель данных

3.1. Структура архива

Физически архив представляет собой базу данных архива, содержащую:

- метаданные документов;
- масштабируемое хранилище архива (специальная таблица), содержащее информацию документов;
- картографические объекты, составляющие подложку при визуализации схемы расположения документов на местности.

Единицей хранения в архиве является ДОКУМЕНТ, представляющий собой произвольный набор файлов, объединенных специальным файлом – регистрационной карточкой документа.

3.2. Структура регистрационной карточки

```
[MAIN]
Name=Центральная часть НН (1:2000)
Type=Электронная карта
MainFile=1_2000.ini
Security=0
Scale=2000
DateCreate=10.07.2006
Region=Нижний Новгород
Creator=НИИ ПМК
LatitudeSW=56.323
LongitudeSW=44.003
LatitudeNW=56.327
LongitudeNW=44.003
LatitudeNE=56.327
LongitudeNE=44.014
LatitudeSE=56.323
LongitudeSE=44.014
JsonData={"Разработчик": "Лаб.13 НИИ ПМК", "Руководитель": "Васин Ю.Г."}
[ADDFILE]
File1=1_2000.big
File2=1_2000.idx
File3=1_2000.dpf
```

MAIN – основные данные карточки.

Name – имя документа (любой текст до 256 символов).

Type – тип документа (в базе данных документы структурируются по их типам).

MainFile – имя основного файла документа (без пути).

Security – гриф секретности документа (0 – не секретно, 1 – ДСП, 2 – секретно, 3 – совершенного секретно).

Scale – коэффициент масштаба для картографической информации или 0.

DateCreate – дата создания документа.

Creator – производитель документа.

Region – район местности, в котором позиционируется документ (может отсутствовать).

Latitude и Longitude – долготы и широты минимального охватывающего прямоугольника документа в градусах (могут отсутствовать).

ADDFILE – секция дополнительных файлов:

FileI – имя очередного дополнительного файла документа (без пути). I – меняется от 1 через 1.

3.3. Структура базы данных архива

База данных имеет реляционную структуру и состоит из следующего набора основных таблиц:

Creator – таблица производителей документов. Одно поле name – текст до 256 символов.

If_objects – таблица картографических объектов. Здесь следует отметить, что база данных архива является модификацией базы данных СУБД ГИС Terra Plus [3, 4] (часть таблиц убрана, и ряд таблиц добавлен), что позволило обеспечить работу с картографическими объектами (в усеченном и достаточном для архива варианте).

Scale – таблица масштабов. Одно поле number – целое число (коэффициент масштаба).

Security – таблица грифов секретности:

- number – целое число (0, 1, 2, 3, 4);
- name – текст (наименование соответствующего грифа секретности).

Type – таблица типов документов. Одно поле name – текст до 256 символов.

Users – таблица пользователей, содержащая следующие поля;

- Id – внутренний идентификатор пользователя;
- Login – имя пользователя (текст до 256 символов);
- Password – пароль пользователя (текст до 256 символов);
- Fio – фамилия, имя и отчество пользователя (текст до 256 символов);
- Department – название подразделения пользователя (текст до 256 символов);
- Date_open, date_close – интервал времени функционирования пользователя;
- Operations – целое число, представляющее собой суперпозицию операций, разрешенных пользователю (просмотр, импорт, экспорт, редактирование, удаление, сервис, администрирование).
- Status – статус пользователя: 0 - актуальный, 1 – удаленный, 2 – заблокированный.
- Security – максимальный уровень доступа к документам (0 – не секретно, 1- Для внутреннего пользования, 2 – Для служебного пользования (ДСП), 3 – секретно, 4 – совершенно секретно).

Protokol – таблица протоколирования операций работы с архивом, содержащая следующие поля:

- Id_user – внутренний идентификатор пользователя, выполнявшего операцию;
- Operation – номер операции с архивом;
- Date_start, time_start – начало выполнения операции;
- Date_end, time_end – конец выполнения операции;
- Result – результат выполнения операции (выполнено/не выполнено);
- Comment – комментарий к операции (текст до 256 символов).

Storage – таблица хранилища документов, содержащая следующие поля:

- Name- имя, присвоенное документу (произвольный текст до 256 символов);
- Type – тип документа в соответствии с таблицей Type (произвольный текст до 256 символов);
- Comment – название района местности документа (произвольный текст до 256 символов);
- Region – координаты минимально охватывающего прямоугольника документа (могут отсутствовать) – рамка документа;
- File – имя файла документа (без пути);
- Typefile – тип файла документа (0 – основной, 1 – дополнительный);
- Status – статус документа (0- актуальный, 1 – удаленный, 2 – заблокированный);
- Maker – название производителя (в соответствии с таблицей Creator);
- Datecreate – дата создания документа;

- Scale – масштаб документа (или 0);
- Security – гриф секретности документа;
- Jsem – набор метаданных в формате ключ=значение (в формате json);
- Sem – набор метаданных в формате поля характеристик интегрального файла СУБД ГИС Terra [5, 6];
- Data – содержимое файла документа (имеет тип oid (указатель) СУБД PostgreSQL).

4. Классы

4.1. База данных архива

4.1.1. Конструктор

TArchDB();

4.1.2. Подключение к базе данных архива

4.1.2.1. Подключение обычного пользователя

*int OpenDataBase(String NameBase,String Server,int Port,String MainUser,String PasswordServer,String VendorHome,String Login,String Password,bool Prompt,String Debug,int *UserOper)*

NameBase - Имя базы данных архива

Server - Адрес сервера

Port - Порт сервера

MainUser - Имя встроенного пользователя сервера (обычно 'postgres')

PasswordServer - Пароль доступа к серверу

Vendorhome - Путь к вендорным библиотекам PostgreSQL

Login - Имя пользователя

Password - Пароль пользователя

Prompt - true - включить диалог коннекта

Debug - Полное имя файла для отладочной печати SQL-запросов ("" - нет печати)

UserOper - Возвращает маску (суперпозицию) режимов доступа к операциям

OPER_VIEW – просмотр

OPER_IMPORT - импорт

OPER_EXPORT - экспорт

OPER_EDIT - редактирование

OPER_DELETE - удаление

OPER_SERVICE - сервис

OPER_ADMIN - администрирование

Функция возвращает:

-1 - нет коннекта к базе данных архива

0 - у пользователя (Login) нет прав доступа к базе данных архива

>0 - идентификатор пользователя

4.1.2.2. Подключение суперпользователя

bool open(String NameBase)

NameBase – имя базы данных архива.

Возвращаемое значение:

true – есть подключение;
false – нет подключения.

Перед подключением к базе данных архива в этом режиме необходимо выполнить следующие методы.

4.1.2.3. Задание номера порта

void SetPort(int Port)

Port – номер порта для подключения к базе данных (установлен в PostgreSQL).
По умолчанию используется порт 5432.

4.1.2.4. Задание имени сервера

void SetServer(String Server)

Server – имя сервера с базой данных.
По умолчанию – 'localhost'.

4.1.2.5. Задание имени суперпользователя пользователя

void SetUserName(String Username)

Username – имя суперпользователя базы данных. По умолчанию – 'postgres'.

4.1.2.6. Задание пароля доступа к серверу

void SetPassword(String Password)

Password – пароль. По умолчанию – '123'.

4.1.2.7. Установка режима диалога при коннекте

void SetLoginPrompt(bool Prompt)

Prompt :

true – включить режим диалога при подключении к базе данных (установлен по умолчанию);
false – выключить.

4.1.2.8. Установка режима отладочной печати

void SetDebug(String Filedebug)

Filedebug – полное имя файла, в который будут выводиться все формируемые SQL-запросы.

Если Filedebug="", то режим отладочной печати будет отменен. По умолчанию режим отладочной печати отменен.

4.1.3. Отключение от базы данных

void Close()

4.1.4. Получение параметров подключения

4.1.4.1. Получение номера порта

int GetPort()

Возвращает номер порта подключения.

4.1.4.2. Получение имени сервера

String GetServer()

Возвращает имя сервера.

4.1.4.3. Получение пароля

String GetPassword()

Возвращает пароль подключения.

4.1.4.4. Получение имени пользователя

String GetUserName()

Возвращает имя пользователя.

4.1.4.5. Получение имени базы данных архива

String GetDatabase()

Возвращает имя подключенной базы данных.

4.1.5. Проверка пользователя

*int CheckUser(String Log, String Pass,int *IdUser)*

Log – логин (имя) пользователя

Pass – пароль пользователя

IdUser - возвращает идентификатор пользователя или ноль при его отсутствии.

4.1.6. Регистрация в журнале работ

int PrtWrite(int IdUser,int Oper,TDateTime T1date,TDateTime T1time,TDateTime T2date,TDateTime T2time, int Rezult,String Comment)

IdUser – идентификатор пользователя

Oper – номер операции

T1date, T1time – дата и время начала операции

T2date, T2time – дата и время окончания операции

Rezult – результат выполнения операции

Comment – комментарий к операции (любой текст до 256 символов).

Функция возвращает:

BERR_OK – нормальное завершение

BERR_BASE – ошибка в базе данных архива

4.1.7. Блокировка пользователя

int BlockUser(String Login)

Login – имя блокируемого пользователя

4.1.8. Управление транзакциями

Обычно эти функции должны использоваться при массовых операциях обновления информации в базе данных архива, которые должны быть гарантированно выполнены целиком (например, загрузка в базу данных архива совокупности документов целиком). По умолчанию транзакции объявляются на запись/обновление одного документа).

4.1.8.1. Старт транзакции

bool StartTransaction()

Функция возвращает:

true – транзакция открыта;

false – предыдущая транзакция не завершена.

4.1.8.2. Завершение транзакции

bool EndTransaction()

Функция возвращает:

true – транзакция закрыта (все изменения сделаны в базе данных архива);

false – транзакция не открыта.

4.1.8.3. Откат транзакции

bool RollbackTransaction()

Функция возвращает:

true – произведен откат транзакции;

false – транзакция не была открыта.

4.2. Документ архива

Для работы с документами архива используется структура “Заголовок документа”:

```
typedef struct
{
    int Id;                // Идентификатор (номер)
    String Name;           // Имя документа
    String Region;         // Имя района этого документа
    String Coord;          // Координаты рамки в виде LINESTRING (или "")
    String MainFile;       // Имя основного файла (без пути)
    String Type;           // Тип документа
    int TypeFile;          // Тип файла (0 – основной, 1 – дополнительный)
    int Status;            // Статус (0 – актуальный, 1 – удаленный, 2 – заблокированный)
    int TypePr;            // Тип проекции
    int Scale;             // Коэффициент масштаба (или 0)
    int Security;          // Гриф секретности
    TDateTime DateCreate;  // Время создания документа
    TDateTime DateReg;     // Время регистрации документа
    String Creator;        // Имя производителя
    MET_LONG_2 Met[100];   // Массив координат рамки (или NULL)
    int CountPoints;       // Количество точек рамки (или 0)
    String Disk;           // Метка DVD-диска или ""
    String JsonData;       // Данные в формате Json или ""
    TStringList *List;     // Список дополнительных файлов bkb NULL
} DOC_HEADER;
```

В секции Public данного класса объявлена переменная TabDoc типа DOC_HEADER, с использованием которой происходит чтение и запись информации в архив.

4.2.1. Конструктор

*TArchives(TArchDB *IFPGDB)*

IFPGDB – указатель на открытую базу данных архива.

4.2.2. Выборка информации из архива

4.2.2.1. Открыть чтение документов архива (читается первый документ по запросу)

*DOC_HEADER *Open()*

Функция возвращает указатель на структуру документа или NULL при его отсутствии.

4.2.2.2. Прочитать следующий документ по запросу

*DOC_HEADER *Next()*

Функция возвращает указатель на структуру документа или NULL при его отсутствии.

4.2.2.3. Принудительное закрытие запроса на чтение

void Close()

4.2.2.4. Чтение документа по номеру

*DOC_HEADER *ReadDocDataByNumber(int Id,String DataDir)*

Id - идентификатор (номер_ документа)

DataDir - имя папки для выдачи документа (без \ в конце)

Функция возвращает указатель на структуру документа или NULL при его отсутствии.

4.2.2.5. Чтение заголовка документа по номеру

*DOC_HEADER *ReadDocHeaderByNumber(int Id)*

Id - номер документа

Функция возвращает указатель на структуру документа или NULL при его отсутствии.

4.2.2.6. Получение списка дополнительных файлов документа

*void GetListFiles(int Id, TString *List)*

Id - номер документа

TStrings *List – возвращает список файлов документа или NULL

4.2.2.7. Получение любого файла из документа

bool GetFileFromDoc(int Id, String DirName,String FileName)

Id - номер документа

DirName - имя папки для получения файла (без \ в конце)

FileName - имя файла (без пути)

Функция возвращает:

true – файл получен

false – файл не получен

4.2.2.8. Установка порядка выдачи документов при чтении

void SetOrder(String Order)

Order - имена атрибутов, по возрастанию значений которых выдаются документы:

date - дата регистрации
 datecreate - дата создания
 name - имя документа
 scale - масштаб
 comment - район

4.2.2.9. Установка статуса для выборки документов

void SetStatus(String Status)

Status "0" - актуальные
 "1" - удаленные
 "2" - заблокированные

4.2.3. Установка фильтров для операции чтения Open

void SetFilterName(String Name) По имени документа

void SetFilterType(String Type) По типу

void SetFilterRegion(String Region) По району

void SetFilterCreator(String Creator) По производителю

Для фильтров, указанных выше используется контекстный поиск с начала строки

void SetFilterScale(String Scale) По коэффициенту масштаба

void SetFilterTypePr(String TypePr) По номеру типа проекции

void SetFilterSecr(String Secr) По номеру грифа секретности

void SetFilterDateCreator(String T1, String T2) По интервалу даты создания

void SetFilterDateReg(String T1, String T2) По интервалу даты регистрации

int SetFilterCoord(int Count, MET_LONG_2 *Met) По попаданию внутрь контура или его пересечении. Здесь: Count – количество координат контура (не менее 4-х); Met – Массив координат контура в виде пар координат "широта,долгота" (целочисленные в виде долей градусов, увеличенных в 10000000 раз).

void SetFilterCoordString(String Coord) По попаданию внутри контура или его пересечении. Здесь: Coord массив координат в виде текстовой строки "LINESTRING(X1 Y1,X2 Y2,...Xn Yn)"

При операции выборки документов из архива используется суперпозиция установленных фильтров, объединенных операндом AND.

void ClearAllFilters() Очистка всех фильтров

4.2.4. Запись документов в архив

Перед выполнением функции записи документа необходимо сформировать его информацию в структуре документа AR->TabDOC, где AR – указатель типа TArchives на созданный экземпляр объекта 'документ' архива.

4.2.4.1. Запись документа в архив

bool WriteDocToDB()

Функция возвращает:

true – документ записан

false – документ не записан

4.2.4.2. Запись документа старого формата DDC в архив

bool WriteDDCToDB()

Функция возвращает:

true – документ записан

false – документ не записан

Позволяет записать в архив документ, выгруженный из оперативного архива предыдущей версии архива ГИС Терра [4, 5] (реализован на платформе СУБД Firebird).

4.2.4.3. Запись документа старого формата DDC в архив с хранилищем на DVD-дисках

bool WriteDVD_DDCToDB()

Функция возвращает:

true – документ записан

false – документ не записан

Позволяет записать в архив документ, находящийся на постоянном хранении на DVD-дисках архива предыдущей версии архива ГИС Терра [4, 5] (реализован на платформе СУБД Firebird). В архив не записываются файлы документа, а указывается ссылка на их размещение на DVD-диске.

4.2.5. Удаление документов из архива

4.2.5.1. Логическое удаление документов по установленным ранее фильтрам (запросам)

bool Delete()

Функция возвращает:

true – документ (документы) удалены

false – документ (документы) не удалены

4.2.5.2. Физическое удаление документов по установленному ранее статусу

bool DeleteFull()

Функция возвращает:

true – документ (документы) удалены

false – документ (документы) не удалены

4.2.5.3. Логическое удаление документа по его номеру

bool DeleteByNumber(int Number)

Функция возвращает:

true – документ удален

false – документ не удален

4.2.5.4. Физическое удаление документа по его номеру и статусу

bool DeleteFullByNumber(int Number)

Функция возвращает:

true – документ удален

false – документ не удален

Перед выполнением необходимо установить статус документа функцией SetStatus.

4.2.6. Поиск документов

4.2.6.1. Поиск документа, ближайшего к заданной точке

int FindObject(int X, int Y, int x1, int y1, int x2, int y2, String Attr)

X, Y – координаты точки, относительно которой анализируются рамки документов (в единицах базы данных)

x1, y1, x2, y2 - координаты прямоугольника, внутри которого ищется документ (в единицах базы)

Возвращает номер найденного документа или 0

4.2.6.2. Вычисление расстояние от точки до контура

*double TakeDistanceToObject(int CountPoints, MET_LONG_2 *Met, int X, int Y)*

CountPoints – количество точек контура (не менее 4-х)

Met – точки контура в единицах базы данных

X, Y - координаты точки (в единицах базы данных)

Возвращает расстояние в единицах базы.

4.2.7. Дополнительные функции

4.2.7.1. Получение текста грифа секретности по его номеру

String GetSecrByNumber(int Number)

4.2.7.2. Получение типа проекции по его номеру

String GetTypePrByNumber(int Number)

4.2.7.3. Формирование регистрационной карточки документа по его номеру

bool MakeIRC(int Number, String DirName)

Number - номер документа

DirName - имя папки для файла карточки (без \ в конце)

true – карточка сформирована

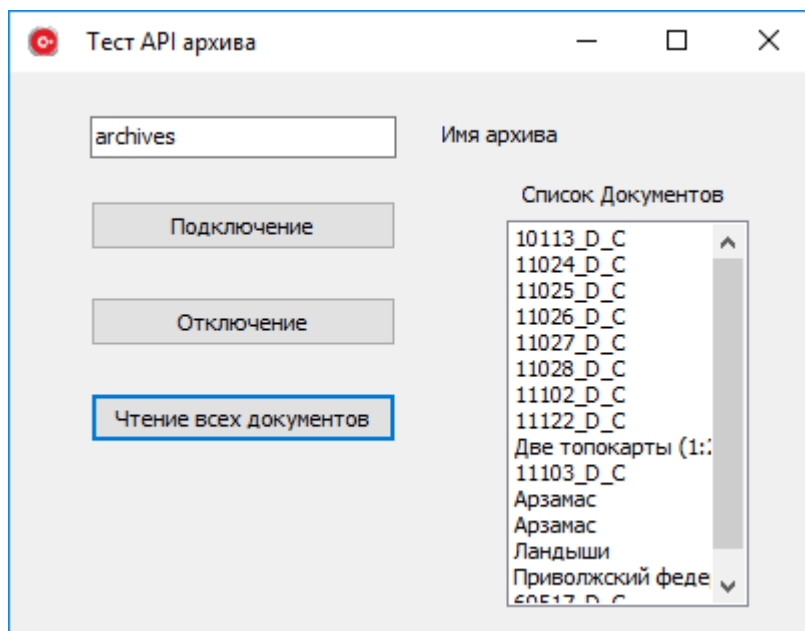
false – карточка не сформирована

5. Пример приложения для работы с архивом

Создание VCL-приложения:

- Создать проект VCL-приложения;
- В опциях проекта установить путь к папке API_TERRA_PLUS/INCLUDE;
- Включить в H-файл главного окна приложения: `#include "if_archives.h"`
- Включить в проект библиотеку ARCHIVES_LIB.LIB

Ниже приведен пример приложения, осуществляющего выбор всех документов из архива. Вид окна приложения:



И-файл приложения:

```
#ifndef Unit1H
#define Unit1H
//-----
#include <System.Classes.hpp>
#include <Vcl.Controls.hpp>
#include <Vcl.StdCtrls.hpp>
#include <Vcl.Forms.hpp>
#include "if_archives.h"
//-----
class TForm1 : public TForm
{
__published: // IDE-managed Components
    TEdit *Edit1;
    TButton *Button1;
    TButton *Button2;
    TButton *Button3;
    TListBox *ListBox1;
    TLabel *Label1;
    TLabel *Label2;
    void __fastcall Button1Click(TObject *Sender);
    void __fastcall Button2Click(TObject *Sender);
    void __fastcall Button3Click(TObject *Sender);
private: // User declarations
    TArchDB *T;
    TArchives *AR;
    DOC_HEADER *D;
public: // User declarations
    __fastcall TForm1(TComponent* Owner);
};
//-----
extern PACKAGE TForm1 *Form1;
//-----
#endif
```

Исходный текст приложения:

```
#include <vcl.h>
#pragma hdrstop
#include "Unit1.h"
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
TForm1 *Form1;
//-----
```

```

__fastcall TForm1::TForm1(TComponent* Owner)
: TForm(Owner)
{
}
//----- Подключение к архиву
void __fastcall TForm1::Button1Click(TObject *Sender)
{
    int err;
    int UserOper;
    T=NULL;
    T=new TArchDB();
    err=T->OpenDataBase(Edit1->Text,"192.168.39.10",
        5432,"postgres","123",
        "d:\\baseutil","yuri","123",false,"",&UserOper);
    if (err <= 0)
    {
        delete T;
        T=NULL;
        ShowMessage("Нет коннекта к архиву");
    }
}
//----- Отключение от архива
void __fastcall TForm1::Button2Click(TObject *Sender)
{
    if (T!= NULL)
    {
        T->Close();
        delete T;
    }
}
//----- Чтение всех документов из архива
void __fastcall TForm1::Button3Click(TObject *Sender)
{
    ListBox1->Clear();
    AR=new TArchives(T);
    D=AR->Open();
    while (D!= NULL)
    {
        ListBox1->Items->Add(D->Name); // Запись имени архива
        D=AR->Next();
    }
    delete AR;
}

```

Литература

- 1.Embarcadero RAD Studio XE7. <https://www.embarcadero.com>
- 2.PostgreSQL 9.3 Documentation [Электронный ресурс]. – <http://www.postgresql.org/docs/9.3/static/index.html>
- 3.PostGIS 2.1 Manual [Электронный ресурс]. –<http://postgis.net/docs/manual-2.13>.
- 4.Васин Ю.Г., Колмогоров С.В., Ясаков Ю.В. Система управления архивом цифровых карт. // Тезисы докладов VIII Всероссийской научной конференции "Методы и средства обработки сложной графической информации". - Н. Новгород: Изд-во ННГУ, 2005. - С. 20-22.
- 5.Васин Ю.Г., Перепелкин А.И., Ясаков Ю.В. Удаленный доступ к электронному архиву морских навигационных карт. // Тезисы докладов VIII Всероссийской научной конференции "Методы и средства обработки сложной графической информации". - Н. Новгород: Изд-во ННГУ, 2005. - С. 31-33.