



УДК 004.94

В. Н. Гридин

*Центр информационных технологий в проектировании РАН*

Г. Д. Дмитриевич, В. И. Анисимов, С. А. Васильев

*Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина)*

## Методы построения веб-ориентированных схмотехнических САПР на основе БД Oracle

*Рассматриваются подходы построения и работы с БД для хранения архивов проектных решений пользователей и ограничения доступа в сервис-ориентированные системы автоматизации схмотехнического проектирования на основе веб-сервисов с применением протокола полнодуплексной связи WebSocket. Описываются компоненты среды разработки, версии применяемого программного обеспечения и компоненты разрабатываемого ПО, их назначения, протоколы связи. Указываются тонкости установки настройки сервера базы данных, необходимые компоненты для обеспечения функционирования. Рассматриваются алгоритмы построения структуры базы данных, формирования таблиц связей, приводятся соответствующие примеры на языке Transact-SQL. Демонстрируется алгоритм подключения среды разработки Visual Studio 2015 к БД, и настройки приложения для доступа к функционалу базы данных. Описываются элементарные запросы к БД, относящихся к предметной области схмотехнического проектирования, и алгоритмы работы с различными типами ответов базы данных.*

### **Схмотехническая САПР, протокол WebSocket (стандарт RFC 6455), сложноструктурированные данные, Oracle Developer Tools, связывание таблиц, формирование запросов в БД, Transact-SQL**

Создание сервис-ориентированных систем автоматизации схмотехнического проектирования на основе веб-сервисов с применением протокола полнодуплексной связи WebSocket (стандарт RFC 6455) [1] сопряжено с решением ряда проблем – хранения данных пользователей и результатов расчетов, ограничение доступа пользователей к данным, обмен данными между пользователями. Решением этих задач может стать развертывание сервера баз данных или работа с облачными сервисами баз данных. В данное время на рынке имеется достаточное количество предложений по предоставлению SaaS и PaaS баз данных/инфраструктуры для баз данных, ключевым моментом для использования которых становится доверие к хранению и/или администрированию данных пользователей сторонними организациями [2]. Принятие решения о развертывании сервера баз данных также связано с выбором программного обеспечения, предоставляемого различными организациями. Специфика предметной области накладывает обязательства стабильной работы с

большими объемами сложноструктурированных данных и обеспечения быстрой масштабируемости, качественной технической поддержки, простоты развертывания на различных ОС. По данным критериям наилучшим образом подходит продукт Oracle Database 11 g, применение которого позволит разрешить проблемы, связанные с хранением, обменом и доступом к данным [3]. Для применения Oracle Database 11 g в качестве хранилища данных для веб-ориентированной схмотехнической системы автоматизированного проектирования требуется описать настройку данного ПО на сервере, средства управления данным продуктом, методы работы с таблицами и методы взаимодействия со средой разработки.

В настоящей статье рассматривается методика построения сервис-ориентированной системы автоматизации схмотехнического проектирования на основе применения протокола полнодуплексной связи WebSocket. Устанавливаются способы решения ряда проблем, связанных с хранением начальных данных пользователей, хранением ре-

зультатов расчетов, а также проблемы ограничения доступа пользователей к данным и обмена данными между пользователями. Предлагается технология построения таблиц и их связей для решения вышеуказанных проблем. Описывается способ организации взаимодействия инструментальной среды разработки серверного приложения с СУБД Oracle.

**Описание среды разработки и структуры компонентов.** Платформой для разработки/демонстрации сервис-ориентированной системы служит один компьютер под управлением операционной системы Windows 10 x64. Разработка ПО для обработки и распределения запросов клиентов, в дальнейшем именуемого сервером расчетов схем и сервером приема и распределения запросов, производится в среде Visual Studio 2015. На компьютер предустановлен сервер баз данных Oracle Database 11g, для управления им используется СУБД Oracle SQL Developer Version 4.1.3.20. Также для работы Visual Studio 2015 с Oracle Database 11g требуется установить средство управления базой данных Oracle Developer Tools for Visual Studio 2015 12.1.0.2.4.

Основными узлами сервис-ориентированной системы автоматизации схемотехнического проектирования с применением полнодуплексного протокола связи WebSocket (стандарт RFC 6455) являются:

1. Сервер расчета схем – приложение консольного типа, прослушивающее некоторый URL-адрес (например, ws://127.0.0.1:8181) и выполняющее расчет частотных характеристик схемы.

2. Сервер приема и распределения запросов – приложение консольного типа, прослушивающее некоторый URL-адрес (например, ws://127.0.0.1:8282) и выполняющее обработку запросов клиентского приложения, авторизацию пользователей, предоставление информации, сохранение данных пользователей, обмен данными посредством взаимодействия с базой данных, а также предоставления сервиса расчета электронных схем пользователей посредством взаимодействия с сервером расчета схем.

3. База данных ИМЯ\_ПОЛЬЗОВАТЕЛЯ (в дальнейшем БД), располагающаяся по IP-адресу localhost (127.0.0.1), порт 1521, идентификатор службы (SID) XE. БД хранит в себе информацию о пользователях, проектах, включающих в себя описание схем и результаты расчетов, а также о разрешении доступа одного пользователя к проектам другого.

4. Клиентское приложение, предоставляющее конечному пользователю возможность (интерфейс) авторизации, регистрации в сервисе, описания схемы, расчета схемы, сохранения описания и результатов расчета, загрузки готовых проектов, получения отчетов о работе в сервисе.

**Проектирование и создание БД.** База данных должна включать в себя большое количество различной информации, которая касается пользователей, описания схем и прав доступа. Удобно разделить информацию в следующие отдельные таблицы: о пользователях, правах пользователя и описании схемы, которое следует также разделить на несколько связанных частей: таблицу проектов, так как у каждого пользователя может находиться в разработке несколько различных схем расчета, таблицы с элементами в свою очередь разделяются на минимальные функциональные единицы (сопротивление, индуктивность, конденсатор).

Разработка/администрирование базы данных возможно в программном обеспечении Oracle SQL Developer. Для этой цели возможно использование графического интерфейса или исполнения кода на языке Transact-SQL [4] в файле ИМЯ\_ПОЛЬЗОВАТЕЛЯ.sql. Далее для описания методов работы с БД будет приводиться только код на языке Transact-SQL, также стоит учитывать, что те же действия можно выполнить, используя графический интерфейс Oracle SQL Developer.

Для обеспечения безопасности базы данных и удобства проектирования/администрирования следует создать отдельного пользователя («create user ИМЯ\_ПОЛЬЗОВАТЕЛЯ identified by ПАРОЛЬ\_ПОЛЬЗОВАТЕЛЯ»). Требуется наделить пользователя полномочиями для работы (создания, изменения, удаления) с таблицами («grant create table to ИМЯ\_ПОЛЬЗОВАТЕЛЯ; grant alter any table to ИМЯ\_ПОЛЬЗОВАТЕЛЯ; grant delete any table to ИМЯ\_ПОЛЬЗОВАТЕЛЯ;»), по мере необходимости можно также предоставить доступ к триггерам, процедурам, представлениям по одному или нескольким действиям: создание, изменение, удаление [5].

Таблица, обладающая полной информацией о пользователях (AUTHENTIC), должна содержать поля: «USER\_ID», «USER\_NAME», «USER\_LOGIN», «USER\_PASS», «AUTORI\_DATE». Для создания данной таблицы требуется выполнить команду: «create table authentic (user\_id varchar2 (количество символов) Primary key, user\_name varchar2(...), user\_login varchar2(...), user\_pass varchar2(...), autori\_date date)».

Таблица, обладающая полной информацией обо всех проектах, находящихся в работе сервис-

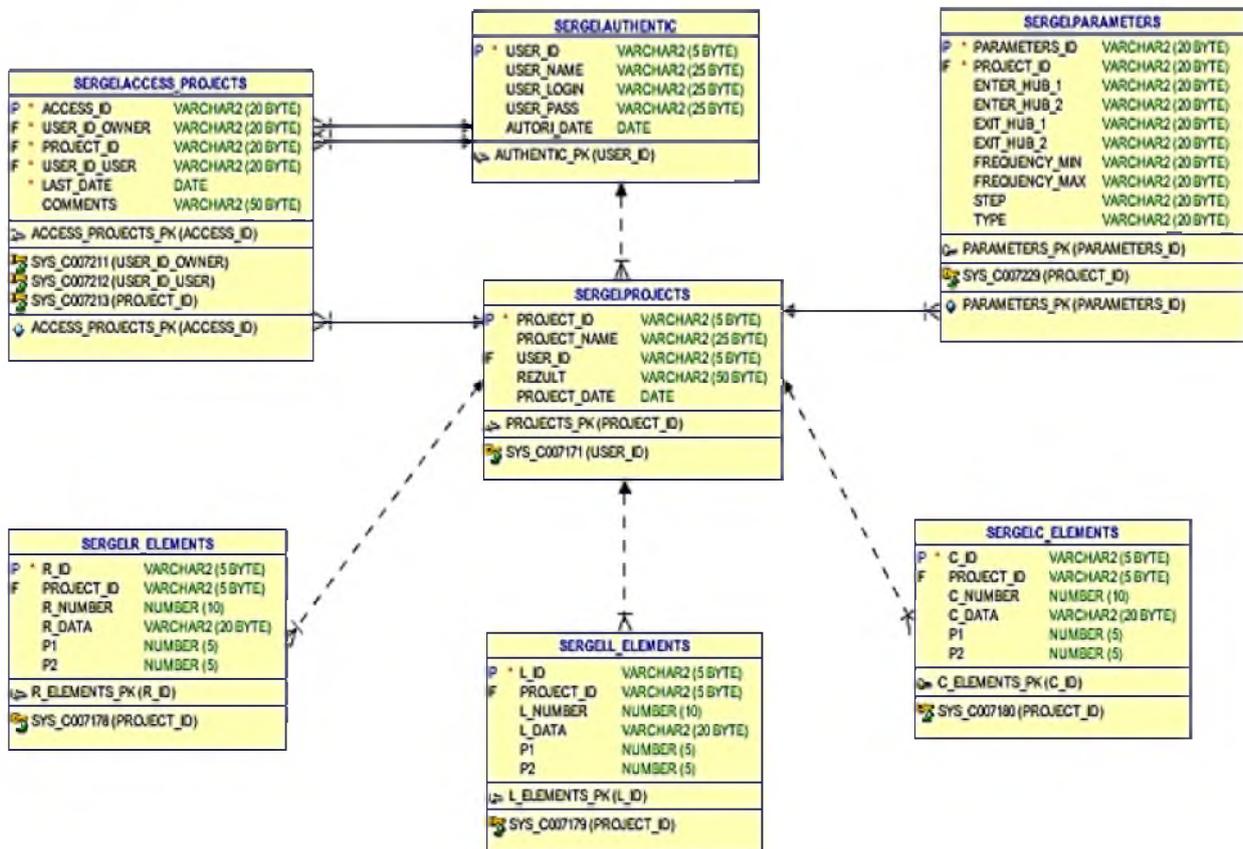
ориентированной системы автоматизации схемотехнического проектирования на основе веб-сервисов (PROJECTS), должна содержать поля: «PROJECT\_ID», «PROJECT\_NAME», «USER\_ID», «REZULT», «PROJECT\_DATE».

Таблица, обладающая полной информацией о всех резисторах, находящихся в проектах пользователей (R\_ELEMENTS) должна содержать следующие поля: «R\_ID», «PROJECT\_ID», «R\_NUMBER», «R\_DATA», «P1», «P2».

Вышеописанная схема на примере трех таблиц позволяет создать остальные таблицы (права доступа, параметры расчета, оставшиеся минимальные функциональные схемотехнические единицы, сложносоставные элементы), необходимые для функционирования системы в соответствии с требованиями, указанными в начале раздела. После создания полного перечня необходимых таблиц требуется их связать, для отсутствия дублирующей информации в таблицах, что, в свою очередь, повлечет уменьшение объема занимаемой памяти, сопоставление информации из одной таблицы относительно другой для корректного автоматизированного (каскадного) удаления записей из базы данных по требованию пользователя или администратора.

Для связи двух таблиц – «AUTHENTIC» и «PROJECTS» – потребуется ключ в виде поля

«USER\_ID», которое соответственно находится в каждой из таблиц. В таком случае сервер приема и распределения запросов при создании проектов в сервис-ориентированной системе автоматизации схемотехнического проектирования на основе веб-сервисов, пользователем USER, имеющим USER\_ID, равный X, из таблицы «AUTHENTIC», будет создавать записи в таблице «PROJECTS», где поле USER\_ID равно X. Таким образом, у одного пользователя может быть большое количество проектов, и все они будут охарактеризованы полем «USER\_ID», равным полю «USER\_ID» из таблицы пользователей. Для создания такой связи требуется выполнить команду «alter table projects add foreign key (user\_id) references authentic (user\_id)». В соответствии со связью таблиц «AUTHENTIC» и «PROJECTS» требуется связать «PROJECTS» и «R\_ELEMENTS» так, чтобы определенное количество резисторов соответствовало определенному проекту конкретного пользователя. Данные таблицы будут соединены с использованием ключевого поля «PROJECT\_ID», имеющегося в обеих таблицах. В соответствии с указанными примерами требуется связать все таблицы, применяя механизмы, подобные описанным в СУБД Oracle Developer Tools, схема связей представлена на рисунке.



**Взаимодействие инструментальной среды Visual Studio с СУБД Oracle Database.** Взаимодействие Visual Studio 2015 с Oracle Database 11 g требуется для обработки различных унифицированных запросов пользователей из клиентского приложения, поступающих по полнодуплексному протоколу связи WebSocket (стандарт RFC 6455). Для работы с запросами пользователей необходим оперативный доступ к информации, находящейся в различных таблицах базы данных, в том числе к статусу аутентификации пользователя и получения доступа к тем или иным сведениям.

Для осуществления взаимодействия между Visual Studio 2015 и Oracle Database 11 g требуется установить Oracle Developer Tools for Visual Studio 2015 (на текущий момент последняя версия 12.1.0.2.4).

Далее необходимо создание консольного приложения, где в пункте меню «проект» требуется «добавить новый источник данных». В открывшемся мастере настройки источников данных в выборе типа источника данных требуется выбрать пункт «база данных», в следующем окне выбора модели базы данных – установить пункт «набор данных», в окне выбора подключений к базе данных нужно выбрать «новое подключение». В открывшемся окне добавления подключений в пункте «источник данных» выбрать: «база данных Oracle (ODP.NET, Managed Driver)», заполнить поля «User name» и «Password» в соответствии с данными, указанными при создании нового пользователя в БД. В пункте «Connection Type» выбрать «EZ Connect», далее заполнить поля в соответствии со следующими указаниями: DataBase host name – localhost; Port number – 1521; DataBase service name – XE. В поле «Source name» должна отобразиться следующая строка: «localhost:1521/XE», после этого требуется запустить проверку подключения. Если произведенная операция имеет положительный результат, то нужно подтвердить источник данных в мастере настройки. В пункте «выбор объектов баз данных» требуется выбрать соответствующие наименования компонентов, предоставляемых подключаемой БД. Данный конкретный случай подразумевает выбор всех таблиц из базы данных, после чего нужно подтвердить выбор компонентов. В меню «обозреватель решения» появится новый пункт «DataSet1», в котором хранится информация о подключении БД и элементах, которые она предоставляет – в данном случае только о таблицах.

Перед описанием запросов из консольного приложения требуется подключить специальную библиотеку с классами для работы с базой данных Oracle, для чего нужно разместить следующую строку в шапке исполняемого файла программы «using Oracle.ManagedDataAccess.Client;». Для создания подключения объявляется новый экземпляр «con1» класса «OracleConnection» с аргументом в виде строки подключения к базе данных – «OracleConnection con1 = new OracleConnection "DATA SOURCE = localhost: 1521/XE; PASSWORD = ПАРОЛЬ\_ПОЛЬЗОВАТЕЛЯ; PERSIST SECURITY INFO = True; USER ID=ИМЯ\_ПОЛЬЗОВАТЕЛЯ");», после чего можно воспользоваться методом для открытия соединения «con1.Open();». Также требуется объявить новый экземпляр «cmd1» класса «OracleCommand» без аргументов – «OracleCommand cmd = new OracleCommand();» и приравнять значение подключения открытому подключению к БД – «cmd.Connection = con1;». Таким образом, подготовка для формирования запросов к БД в виде переменных строкового типа, содержащих текст запроса на языке Transact-SQL выполнена. Основные запросы разделяются на два типа по формату данных возвращаемого в виде ответа на запрос в БД – это может быть одна переменная (какое имя у пользователя с определенным ID, где ID – уникальный идентификатор) либо некоторое количество ответов на запрос (названия проектов у конкретного пользователя). Вследствие указанных особенностей у класса «OracleCommand» имеются особые методы для работы с запросами, подразумевающими односложные (одно значение) ответы БД – «ExecuteScalar()» и ответы, имеющие определенное количество значений – «ExecuteReader()». Примером первого типа запроса может послужить команда «cmd.CommandText = "select distinct USER\_NAME from AUTHENTIC where USER\_ID=1";», где ответ можно присвоить одной переменной таким образом: «var USER\_NAME = cmd.ExecuteScalar();». Ответ на такой запрос будет иметь одно поле из одной записи таблицы «AUTHENTIC» и будет представлять собой имя пользователя. Примером сложного типа запроса может послужить команда «cmd.CommandText = "select PROJECT\_NAME from PROJECTS where user\_id='1'";», ответ на такой запрос можно присвоить специальной переменной типа «OracleDataReader» – «OracleDataReader re = cmd.ExecuteReader();», которую далее можно привести к

удобному виду передачи по протоколу WebSocket в виде строки типа «string». Для этого следует выполнить следующую команду: «foreach (System.Data.Common.DbDataRecord row in re) {str = str + row["PROJECT\_NAME"]. ToString() + "\*";}». Такая строка будет иметь вид последовательности имен проектов определенного пользователя, разделенных символом «\*».

На основании описанной технологии организации взаимодействия инструментальной среды разработки серверного приложения с СУБД имеется возможность выполнять различные сложноструктурированные запросы к базе данных, получая и обрабатывая любые типы ответов от БД, тем самым реализуя наполнение интерфейса пользователя клиентского приложения и предоставляя информацию на его запросы.

Применение протокола полнодуплексной связи WebSocket (стандарт RFC 6455) [1] сопряжено с решением ряда проблем – хранения данных

пользователей и результатов расчетов, ограничения доступа пользователей к данным, обмена данными между пользователями. В статье для решения указанных задач предлагается развертывание сервера баз данных или работа с облачными сервисами баз данных. Специфика предметной области выдвигает требования стабильной работы с большими объемами сложноструктурированных данных и обеспечения быстрой масштабируемости, качественной технической поддержки, простоты развертывания на различных ОС.

В статье рассмотрена методика построения сервис-ориентированной схемотехнической САПР на основе применения протокола связи WebSocket. Предложена технология построения таблиц и их связей для решения ряда проблем, связанных с хранением начальных данных пользователей и результатов расчетов, а также с ограничением доступа пользователей к данным и обмена данными между пользователями.

## СПИСОК ЛИТЕРАТУРЫ

1. Гридин В. Н., Анисимов В. И., Васильев С. А. Методы повышения быстродействия веб-приложений на основе протокола WebSocket // Вестн. комп. и информ. технологий. 2016. № 3 (141). С. 38–43.
2. Коваленко О. С., Курейчик В. М. Обзор проблем и состояний облачных вычислений и сервисов // Изв. Юж. федер. ун-та. Техн. науки. 2012. Т. 132, № 7. С. 146–151.
3. Додохов А. Л., Сабанов А. Г. Исследования применения СУБД Oracle для защиты персональных данных // Докл. Томск. гос. ун-та систем управления и радиоэлектроники. 2011. № 2–2 (24). С. 263–266.
4. Leverage your Oracle 11g skills to learn DB2 9.7 for Linux, UNIX, and Windows. URL: <https://www.ibm.com/developerworks/data/library/techarticle/dm-0401gupta/dm-0401gupta-pdf.pdf> (дата обращения: 27.10.2016).
5. Варкентин В. В., Барбасова Т. А. Разработка программного обеспечения автоматизированных систем управления технологическими процессами с использованием Visual Studio и базы данных ORACLE // Современные наукоемкие технологии. 2013. № 8 (ч. 1). С. 78–80.

---

V. N. Gridin

*The Center of Information Technologies of Projection RAS*

G. D. Dmitrevich, V. I. Anisimov, S. A. Vasilev

*Saint Petersburg Electrotechnical University «LETI»*

## METHODS OF CONSTRUCTING SERVICE-ORIENTED COMPUTER-AIDED CIRCUIT DESIGN SYSTEMS BASED ON ORACLE DATABASE

*Practical implementation of Web technologies in developing computer-aided circuit design (CAD) systems based on the Web Socket protocol is treated. An approach to storing project data by setting up a database server in data management Oracle environment is suggested. Components of the environment as well as those of the developed software, their assignment along with communication protocols are described. Set up configurations of the database server are specified in detail. Algorithms of building the database structure, those of forming the inquiry tables are given. The algorithm to connect Visual Studio 2015 environment to the database and install client application to access the database functionality are demonstrated.*

**Service-oriented system, CAD, web-service, WebSocket Protocol, structures data, Oracle Developer Tools**

---