

## **ИНТЕГРАЦИЯ КОРПОРАТИВНЫХ ИНФОРМАЦИОННЫХ СИСТЕМ НА ПЛАТФОРМЕ 1С С ВНЕШНИМИ БАЗАМИ ДАННЫХ**

В условиях стремительного роста объемов информации использование корпоративных информационных систем и баз данных становится необходимостью для повышения эффективности работы организаций. В странах СНГ одними из наиболее популярных корпоративных решений являются программы платформы 1С. В процессе реинжиниринга возникает потребность в интеграции корпоративных информационных систем, построенных на платформе 1С, с внешними базами данных, такими как реляционные базы данных.

Целью данного исследования является изучение методов и технологий интеграции корпоративных информационных систем на платформе 1С с внешними базами данных, а также анализ их преимуществ, ограничений и практических аспектов реализации.

Платформа 1С обладает встроенными механизмами для работы с внешними источниками данных. Например, возможность использования конструкции ExternalData позволяет подключаться к внешним системам и получать необходимые данные. Кроме того, в 1С реализована поддержка подключения к внешним базам через протоколы ODBC и JDBC, что расширяет возможности обмена информацией.

Также широко применяются интеграционные решения через API, такие как REST и SOAP, позволяющие реализовывать обмен данными в режиме реального времени либо по расписанию. Внутренние компоненты 1С позволяют создавать web-сервисы, обеспечивающие интеграцию с внешними системами.

Первый и один из наиболее популярных методов – доступ к внешним базам данных через драйверы ODBC или JDBC.

Open Database Connectivity (ODBC) – это открытый стандартный API для доступа к системам управления базами данных. ODBC позволяет приложениям получать доступ к данным, хранящимся в различных базах данных, без необходимости знать их специфические особенности [1].

Java Database Connectivity (JDBC) – это стандартный API для языка Java, который обеспечивает механизм выполнения запросов SQL и получения результатов из базы данных [2].

Такой подход позволяет подключаться к различным реляционным базам данных (MySQL, PostgreSQL, MS SQL Server и др.) и вы-

полнять SQL-запросы прямо из 1С. Этот метод обладает простотой в реализации и высокой гибкостью, однако требует внимательного подхода к вопросам безопасности и производительности, так как прямое соединение с внешней базой увеличивает риск утечек или сбоев.

Для реализации подключения к внешней базе данных через ODBC в 1С необходимо настроить драйвер и подключение:

– «Панель управления» → «Администрирование» → «Источники данных ODBC»;

– создать системный DSN для базы данных MS SQL Server.

Для вывода данных в сообщении можно использовать процедуру при начале работы системы в модуле приложения 1С (запуск в режиме «Конфигурация»), пример которой представлен в Листинге 1.

```
Процедура ПриНачалеРаботыСистемы()
    Попытка
        Соединение = Новый СОМОбъект("ADODB.Connection");
        Соединение.Open("Provider=SQLOLEDB;Data Source= DESKTOP-
3JJ5S14;Initial Catalog=PrintingHouse;Integrated Security=SSPI;");
        Результат = Соединение.Execute("SELECT MaterialID, Title FROM
dbo.Materials");
        Вывод = "";
        Пока Не Результат.EOF Цикл
            ID = Результат.Fields("MaterialID").Value;
            Название = Результат.Fields("Title").Value;
            Вывод = Вывод + "ID: " + ID + ", Название: " + Название + Символы.ПС;
            Результат.MoveNext();
        КонецЦикла;
        Сообщить(Вывод);
        Результат.Close(); Соединение.Close();
    Исклучение
        Сообщить("Ошибка подключения к базе данных: " + ОписаниеОшибки());
    КонецПопытки;
КонецПроцедуры
```

**Листинг 1 – Процедура для подключения базы данных на SQL Server к корпоративной информационной системе на базе 1С**

Обработка ошибок и логирование позволяют своевременно обнаруживать и устранять проблемы, связанные с соединением или некорректными данными.

Второй способ – импорт и экспорт данных посредством файлов форматов CSV, XML, JSON. Этот метод предполагает периодическую синхронизацию между системами и подходит для сценариев, когда обмен происходит нечасто или когда необходима архивная фиксация.

Третий способ – использование web-сервисов и API. В этом случае обмен данными осуществляется через HTTP-запросы, что обеспечивает возможность обмена в режиме реального времени. Этот подход особенно актуален для интеграции с облачными системами.

Еще одна используемая технология – брокеры сообщений, таких как AMQP или MQTT. Она позволяет организовать асинхронный обмен данными между системами, что полезно при работе с большими объемами информации или в сценариях с высокой нагрузкой.

Практические сценарии интеграции могут включать синхронизацию данных с ERP-системами, системами складского учета или облачными хранилищами. Например, можно настроить автоматическую выборку данных о материалах из внешней базы и обновлять их в учетной системе 1С, что сокращает ручной труд. Для этого можно использовать модуль с содержанием, приведенным в Листинге 2.

```
Команда = Новый СОМОбъект("ADODB.Command");
Команда.ActiveConnection = *Соединение*;
Команда.CommandText = *ТекстЗапроса*;
Команда.Parameters.Append(*ПараметрДаты*);
НаборДанных = Команда.Execute();
НачатьТранзакцию();
Поиск = Справочники.Материалы.НайтиПоРеквизиту("ВнешнийID", ВнешнийID);
Если Поиск.Пустая() Тогда
    НовыйМатериал = Справочники.Материалы.СоздатьЭлемент();
    НовыйМатериал.Записать();
Иначе
    СуществующийМатериал = Поиск[0].ПолучитьОбъект();
    СуществующийМатериал.Записать();
КонецЕсли;
ЗафиксироватьТранзакцию();
```

**Листинг 2 – Пример кода для обновления данных в системе 1С в соответствии с внешней базой данных**

Обеспечение актуальности данных – важная задача, которая решается за счет настройки регулярных синхронизаций (например, по расписанию, Листинг 3) или внедрения событийных механизмов. Выбор метода определяет ключевые показатели эффективности интеграции: скорость обновления и актуальность данных.

Для синхронизации справочника материалов (~10 000 записей) оценка различных методов показывает:

– ручной файловый обмен (CSV): 5–15 мин, высокая латентность;

- автоматическая синхронизация по расписанию (ODBC): 10–60 с, актуальность в пределах интервала расписания;
- событийная синхронизация (REST): <5 секунд на изменение, актуальность в реальном времени.

```

Процедура ВыполнитьСинхронизациюПоРасписанию() Экспорт
    Настройки = РегистрыСведений.НастройкиСинхронизации. СоздатьНаборЗаписей();
    Настройки.Отбор.Параметр.Установить("ВидСинхронизации", "Материалы");
    Настройки.Прочитать();
    Если Настройки.Количество() > 0 Тогда
        Настройка = Настройки[0];
        Если Настройка.Активность Тогда
            Результат = СинхронизироватьМатериалы(Настройка.
ДатаПоследнегоВыполнения);
            Настройка.ДатаПоследнегоВыполнения = ТекущаяДата();
            Настройка.Записать();
            Если Результат.Ошибок > 0 Тогда
                ОтправитьУведомлениеАдминистратору("Ошибки синхронизации: " + Ре-
зультат.Ошибок);
            КонецЕсли; КонецЕсли; КонецЕсли;
    КонецПроцедуры

```

**Листинг 3 – Пример кода для синхронизации информационной системы на 1С с внешней базой данных**

Для настройки интеграции использованы методы, описанные в официальной документации платформы 1С:Предприятие [3].

Интеграция корпоративных информационных систем на платформе 1С с внешними базами данных является важной задачей, позволяющей объединить разрозненные источники данных и повысить их актуальность. Перспективы исследований связаны с развитием облачных технологий и повышением уровня безопасности данных.

#### ЛИТЕРАТУРА

1. Общие сведения об ODBC / SQL / SQL Server // Microsoft Learn [Электронный ресурс]. – Режим доступа: <https://learn.microsoft.com/ru-ru/sql/odbc/reference/odbc-overview?view=sql-server-ver17> (дата обращения: 18.01.2026).

2. Oracle JDBC Developer's Guide, 19c (Май 2023) // Oracle Help Center [Электронный ресурс]. – Режим доступа: <https://docs.oracle.com/en/database/oracle/oracle-database/19/jjdbc/index.html> (дата обращения: 18.01.2026).

3. 1С:Предприятие 8.3. Руководство разработчика. В 2-х частях / Фирма «1С». – Москва : 1С-Паблишинг, 2023. – 864 с.